



FACULTAD DE INGENIERÍA, CIENCIAS Y ADMINISTRACIÓN

TESIS

**SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL
RUBRO HOTELERO DE LA EMPRESA DE SERV. TURÍSTICO
PRINCESS S.A. CHINCHA ALTA –2015**

PRESENTADO PARA OPTAR EL TITULO PROFESIONAL DE:

INGENIERÍA DE SISTEMAS

PRESENTADO POR:

**CASTILLON SIGUAS, LUIS ALFREDO
FERNANDEZ CAHUANA, SMITH RUBEN**

ASESOR:

MG. ARMANDO MORENO HEREDIA

CHINCHA – ICA- PERÚ, 2015

**SISTEMA COMPUTARIZADO PARA LA
ADMINISTRACIÓN DEL RUBRO HOTELERO DE LA
EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.
– CHINCHA ALTA – 2015.**

POR:

**CASTILLON SIGUAS LUIS ALFREDO FERNANDEZ CAHUANA
SMITH RUBEN**

**PRESENTADA A LA FACULTAD DE CIENCIAS, INGENIERÍA Y
ADMINISTRACIÓN DE LA UNIVERSIDAD AUTÓNOMA DE ICA PARA
OPTAR POR EL TITULO PROFESIONAL DE INGENIERIA DE SISTEMAS**

DEDICATORIA

A Dios, por haberme colmado de bendiciones y guiado en el camino para lograr mis objetivos a lo largo de mi formación profesional.

A mis padres Guilda Cahuana Paucar y Francisco Fernandez Aybar, quienes han sido, son y seguirán siendo mi fuente de inspiración, sostén y apoyo en mis esfuerzos de superación a lo largo de mi vida personal y profesional. Por haberme apoyado en todo momento, por sus consejos, por la motivación constante que me permitió hacer frente nuevos retos que me hicieron crecer como persona y profesionalmente, por sus ejemplos de perseverancia y constancia que los caracterizan y que me ha infundado siempre y por su valor mostrado para salir adelante.

Smith Ruben Fernandez Cahuana

Este proyecto está dedicado a mis padres Teofilo Castellón Solano y Ricardina Sigvas Peve, pilares fundamentales de mi vida, que con su tenacidad, apoyo y motivación han hecho de mí una persona Profesional y de bien.

A Dios, por ser nuestro creador, amparo y fortaleza cuando más lo necesitamos, por haber estado conmigo en cada paso que doy, cuidándome y dándome fuerzas para continuar.

Luis Alfredo Castellón Sigvas

AGRADECIMIENTO

Para empezar expresamos nuestro más sincero y profundo agradecimiento a Dios nuestro Señor, por habernos colmado de bendiciones y guiado en el camino para lograr nuestros objetivos a lo largo de nuestra formación profesional.

Al integrante del jurado M.sc. Armando Moreno Heredia, quién fue nuestro docente durante nuestra formación como profesionales y cuyos conocimientos han influido en el proceso de investigación para este proyecto.

Por último, agradecemos a nuestros padres, hermanos, seres queridos y a todas aquellas personas, que de manera directa o indirecta, ayudaron a que sea posible la culminación de este proyecto.

INDICE

RESUMEN.....	15
ABSTRACT.....	16
INTRODUCCIÓN.....	17
CAPITULO I	
ASPECTOS INFORMATIVOS.....	19
1.1 ASPECTOS ORGANIZACIONALES.....	20
1.1.1 DESCRIPCIÓN DE LA ORGANIZACIÓN.....	20
1.1.2 ESTRUCTURA ORGÁNICA.....	22
1.1.3 FACTORES ESTRATÉGICOS.....	22
1.1.3.1 VISIÓN.....	22
1.1.3.2 MISIÓN.....	22
1.1.4 ANÁLISIS FODA.....	23
CAPITULO II	
ASPECTOS DE LA INVESTIGACIÓN	24 2.1
MARCO LÓGICO	25
2.1.1 SITUACIÓN PROBLEMÁTICA.....	25
2.1.2 PROBLEMA.....	26
2.1.3 HIPÓTESIS.....	27
2.1.4 OBJETIVOS.....	27
2.1.4.1 ETIVOS GENERALES.....	27
2.1.4.2 OBJETIVOS ESPECÍFICOS.....	28
2.1.5 JUSTIFICACIÓN E IMPORTANCIA.....	28
2.1.6 DEFINICIÓN DE TERMINOS Y CONCEPTOS.....	29
2.2 MARCO METODOLÓGICO.....	30
2.2.1 TIPO Y NIVEL DE INVESTIGACIÓN.....	30
2.2.2 CONTRASTACIÓN DE LA HIPÓTESIS.....	31
2.2.2.1 MODELO LÓGICO DE CONTRASTACIÓN.....	31
2.2.3 POBLACIÓN Y MUESTRA.....	32
2.2.4 MATERIALES DE LABORATORIO.....	33
2.2.5 TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS.....	35

2.2.5.1 TÉCNICAS DE GABINETE.....	35
2.2.5.2 TÉCNICAS DE CAMPO.....	35
2.2.6 MÉTODOS Y PROCEDIMIENTOS PARA LA RECOLECCIÓN DE DATOS.....	35
2.2.7 ANÁLISIS ESTADÍSTICOS DE DATOS.....	36
2.2.8 EVALUACIÓN DE IMPACTO.....	36
2.2.9 COMUNICACIÓN DE RESULTADOS.....	37
2.2.10 METODOLOGÍA.....	37
2.3 MARCO TEÓRICO.....	38
2.3.1 ANTECEDENTES.....	38
2.3.2 BASE TEÓRICA.....	42
2.3.2.1 LA ADMINISTRACIÓN.....	42
2.3.2.2 LA EMPRESA.....	50
2.3.2.3 LA INDUSTRIA HOTELERA.....	53
2.3.2.4 EL CLIENTE.....	61
2.3.2.5 EL SERVICIO.....	65
2.3.2.6 METODOLOGÍA UML.....	67
2.3.2.7 HERRAMIENTAS.....	72
2.3.2.8 REDES.....	79
2.3.2.9 INTERNET.....	84
2.3.2.10 LAS METODOLOGÍAS ÁGILIES.....	85
CAPITULO III	
PLANIFICACIÓN DEL PROYECTO.....	99 3.1
ALCANCE DEL SISTEMA.....	100
3.2 CRONOGRAMA DE ACTIVIDADES DEL PROYECTO.....	102
3.3 FACTORES CRÍTICOS DE ÉXITO.....	102
3.4 SUPUESTOS.....	102
3.5 RESTRICCIONES.....	102
3.6 ANÁLISIS DE FACTIBILIDAD.....	103
3.6.1 FACTIBILIDAD TÉCNICA.....	103
3.6.2 FACTIBILIDAD OPERATIVA.....	103
3.6.3 CTIBILIDAD ECONÓMICA.....	103

3.6.3.1	COSTO DE INVERSIÓN.....	104
	COSTO DE HARDWARE.....	104 COSTO
	DE SOFTWARE.....	104
3.6.3.2	COSTO DE DESARROLLO.....	104
	RECURSOS HUMANOS	105 RECURSOS
	MATERIALES.....	105
	SERVICIOS.....	105
3.6.3.3	COSTO DE OPERACIÓN.....	107
	MANTENIMIENTO DE HARDWARE.....	108
	MANTENIMIENTO DE SOFTWARE.....	108
	RECURSOS HUMANOS	108
3.6.4	ÁLISIS DE COSTO/BENEFICIO.....	109
CAPITULO IV		
ANÁLISIS Y DESARROLLO DEL SISTEMA.....		
		111
4.1	ANÁLISIS DE LA LÓGICA DEL NEGOCIO.....	112
4.1.1	DIAGRAMA DE CASO DE USO DEL NEGOCIO.....	112
4.1.1.1	DIAGRAMA DE ACTIVIDADES DEL DCUN RESERVACIÓN.....	113
4.1.1.2	DIAGRAMA DE ACTIVIDADES DEL DCUN ALQUILER DE UNA HABITACIÓN.....	114
4.1.1.3	DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO AL CUARTO.....	115
4.1.1.4	DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO DE PLATILLOS.....	116
4.1.1.5	DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO DE LAVANDERIA.....	117
4.1.1.6	DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO COCHERA.....	118
4.1.1.7	DIAGRAMA DE ACTIVIDADES DEL DCUN CUADRE CAJA.....	119
4.1.2	DIAGRAMA DE CASO DE USO DEL SISTEMA.....	120
4.1.3	DIAGRAMA DE SECUENCIA DE LOS DCU DEL	

SISTEMA.....		121
4.1.3.1	DIAGRAMA DE SECUENCIA ASIGNACIÓN DE OPERACIONES DE PERFIL.....	121
4.1.3.2	DIAGRAMA DE SECUENCIA ASIGNACIÓN PERFIL A USUARIO	122
4.1.3.3	DIAGRAMA DE SECUENCIA REGISTRAR OPERACIONES.....	123
4.1.3.4	DIAGRAMA DE SECUENCIA REGISTRAR MENUS.....	123
4.1.3.5	DIAGRAMA DE SECUENCIA REGISTRAR PERFIL USUARIO.....	124
4.1.3.6	DIAGRAMA DE SECUENCIA REGISTRAR NUEVO USUARIO AL SISTEMA.....	124
4.1.3.7	DIAGRAMA DE SECUENCIA MANTENIMIENTO PISOS.....	125
4.1.3.8	DIAGRAMA DE SECUENCIA MANTENIMIENTO TIPO HABITACIÓN.....	125
4.1.3.9	DIAGRAMA DE SECUENCIA MANTENIMIENTO MARCA AUTO.....	126
4.1.3.10	DIAGRAMA DE SECUENCIA MANTENIMIENTO DE COLOR AUTO.....	126
4.1.3.11	DIAGRAMA DE SECUENCIA MANTENIMIENTO VEHÍCULO.....	127
4.1.3.12	DIAGRAMA DE SECUENCIA MANTENIMIENTO PROVEEDOR.....	127
4.1.3.13	DIAGRAMA DE SECUENCIA MANTENIMIENTO PRODUCTO.....	128
4.1.3.14	DIAGRAMA DE SECUENCIA MANTENIMIENTO HABITACIÓN.....	128
4.1.3.15	DIAGRAMA DE SECUENCIA MANTENIMIENTO TALLAS.....	129
4.1.3.16	DIAGRAMA DE SECUENCIA MANTENIMIENTO TIPO	

VEHICULO.....		129
4.1.3.17	DIAGRAMA DE SECUENCIA MANTENIMIENTO TIPO PARQUEO.....	130
4.1.3.18	DIAGRAMA DE SECUENCIA MANTENIMIENTO UBICACIÓN VEHÍCULO.....	130
4.1.3.19	DIAGRAMA DE SECUENCIA MANTENIMIENTO TIPO BANCO.....	131
4.1.3.20	DIAGRAMA DE SECUENCIA MANTENIMIENTO TIPO PAGO.....	131
4.1.3.21	DIAGRAMA DE SECUENCIA MANTENIMIENTO ÁREA.....	132
4.1.3.22	DIAGRAMA DE SECUENCIA MANTENIMIENTO EMPLEADO.....	132
4.1.3.23	DIAGRAMA DE SECUENCIA CAMBIAR IGV.....	133
4.1.3.24	DIAGRAMA DE SECUENCIA REGISTRAR CLIENTE.....	133
4.1.3.25	DIAGRAMA DE SECUENCIA RESERVACION/ ALQUILER.....	134
4.1.3.26	DIAGRAMA DE SECUENCIA MOSTRAR CIERRE CAJA POR USUARIO.....	135
4.1.3.27	DIAGRAMA DE SECUENCIA MOSTRAR SERVICIOS ANULADOS.....	135
4.1.3.28	DIAGRAMA DE SECUENCIA DETALLE DE CAJA.....	136
4.1.3.29	DIAGRAMA DE SECUENCIA MOSTRAR COMPROBANTE DE PAGO GENERADO.....	136
4.1.3.30	DIAGRAMA DE SECUENCIA ANULAR SERVICIOS.....	137
4.1.3.31	DIAGRAMA DE SECUENCIA CAMBIAR ESTADO DE HABITACIONES.....	138
4.1.3.32	DIAGRAMA DE SECUENCIA SERVICIO DE COCHERA.....	139

4.1.3.33	DIAGRAMA DE SECUENCIA	DE	SECUENCIA	APERTURA	
	CAJA.....				140
4.1.3.34	DIAGRAMA DE SECUENCIA	DE	SECUENCIA	CIERRE	
	CAJA.....				140
4.1.3.35	DIAGRAMA DE SECUENCIA	DE	SECUENCIA	INGRESO	
	CAJA.....				141
4.1.3.36	DIAGRAMA DE SECUENCIA	DE	SECUENCIA	EGRESOS	
	CAJA.....				141
4.1.3.37	DIAGRAMA DE SECUENCIA MOSTRAR RESUMEN				
	DIA.....				142
4.1.3.38	DIAGRAMA DE SECUENCIA			REGISTRO	
	CONECEPTO CAJA INGRESO Y SALIDA.....				142
4.2	ANÁLISIS DEL SISTEMA				143
4.2.1	DEFINICIÓN DE ESTÁNDARES.....				143
4.2.1.1	ESTANDARES PARA LAS HISTORIAS			DEL	
	USUARIO.....				143
4.2.1.2	ESTANDARES PARA LAS TAREAS			DEL	
	USUARIO.....				144
4.2.2	STORIAS DE USUARIO.....				145
4.2.2.1	ITERACIÓN I.....				145
4.2.2.2	ITERACIÓN II				149
4.2.2.3	ITERACION III				151
4.2.3	PLAN DE HISTORIAS DE USUARIO.....				155
4.3	DISEÑO DEL SISTEMA.....				156
4.3.1	DEFINICIÓN DE ESTANDARES.....				156
4.3.1.1	ESTANDAR DE NOMENCLATURAS PARA LA BASE				
	DE DATOS.....				156
	TABLAS.....				157
	COLUMNAS.....				157
4.3.2	DISEÑO DE LA BASE DE DATOS.....				157
4.3.2.1	DISEÑO LÓGICO.....				158
4.3.2.2	DISEÑO FÍSICO.....				160

4.3.2.3	DICCIONARIO DE DATOS.....	162
4.3.3	DIAGRAMA DE COMPONENTES.....	191
4.4	CONSTRUCCIÓN DEL SISTEMA.....	192
4.4.1	DEFINICIÓN DE ITERACIONES.....	192
4.4.1.1	ITERACIÓN I.....	192 SPIKES PARA LAS HISTORIAS DEL USUARIO.....192 TAREAS DE LAS HISTORIAS DEL USUARIO.....201 INTERFAZ DE LAS HISTORIAS DEL USUARIO.....207
	CODIFICACIÓN.....	215
4.4.1.2	ITERACIÓN II.....	243 SPIKES PARA LAS HISTORIAS DEL USUARIO.....243 TAREAS DE LAS HISTORIAS DEL USUARIO.....247 INTERFAZ DE LAS HISTORIAS DEL USUARIO.....249
	CODIFICACIÓN.....	252
4.4.1.3	ITERACIÓN III.....	262 SPIKES PARA LAS HISTORIAS DEL USUARIO.....262 TAREAS DE LAS HISTORIAS DEL USUARIO.....262 INTERFAZ DE LAS HISTORIAS DEL USUARIO.....267
	CODIFICACIÓN.....	274
CAPITULO V		
	INTEGRACIÓN.....	300
5.1	CONSIDERACIONES DE SEGURIDAD.....	301
5.1.1	CONSIDERACIONES DE SERGURIDAD EN LA BASE DE DATOS.....	301
5.1.1.1	ENCRIPCIÓN.....	301
5.1.1.2	INYECCIÓN SQL.....	301
5.1.2	CONSIDERACIONES DE SERGURIDAD EN LA APLICACIÓN	
5.1.2.1	AUTENTICACIÓN DE USUARIOS.....	302
5.1.2.2	OPCIONES DE MENÚ.....	302
5.2	PUESTA EN MARCHA.....	302
5.2.1	ESPECIFICACIONES TÉCNICAS.....	302
5.2.1.1	CLIENTE.....	302

CAPITULO VI

DISCUSIÓN	303
6.1 RESULTADOS DE LA PRIMERA ENCUESTA.....	304
6.1.1 PREGUNTA 1: TIEMPO PROMEDIO EN MINUTOS PARA REGISTRAR UN CLIENTE.....	304
6.1.1.1 PREGUNTA ASOCIADA.....	304
6.1.1.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS.....	304
6.1.1.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	305
6.1.1.4 RESULTADOS.....	305
6.1.2 PREGUNTA 2: CUAL ES EL TIEMPO PROMEDIO PARA REALIZAR UN COMPROBANTE DE PAGO.....	305
6.1.2.1 PREGUNTA ASOCIADA.....	305
6.1.2.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS	305
6.1.2.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	306
6.1.2.4 RESULTADOS.....	306
6.1.3 PREGUNTA 3: CUAL ES EL TIEMPO PROMEDIO EN MINUTOS PARA LA BÚSQUEDA DE INFORMACIÓN PARA LOS REPORTES QUE SE ENVÍAN A MINCETUR.....	306
6.1.3.1 PREGUNTA ASOCIADA.....	306
6.1.3.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS.....	306
6.1.3.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	307
6.1.3.4 RESULTADOS.....	307
6.1.4 PREGUNTA 4: CUALES SON LOS PROBLEMAS COMUNES QUE SE PRESENTAN EN LA EMPRESA.....	307
6.1.4.1 PREGUNTA ASOCIADA.....	308
6.1.4.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS.....	308
6.2 RESULTADOS DE LA SEGUNDA ENCUESTA.....	308
6.2.1 PREGUNTA 1: TIEMPO PROMEDIO EN MINUTOS PARA	

REGISTRAR UN CLIENTE.....	308
6.2.1.1 PREGUNTA ASOCIADA.....	308
6.2.1.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS.....	308
6.2.1.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	309
6.2.1.4 RESULTADOS.....	309
6.2.2 PREGUNTA 2: CUAL ES EL TIEMPO PROMEDIO PARA REALIZAR UN COMPROBANTE DE PAGO.....	309
6.2.2.1 PREGUNTA ASOCIADA.....	309
6.2.2.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS	310
6.2.2.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	310
6.2.2.4 RESULTADOS.....	310
6.2.3 PREGUNTA 3: CUAL ES EL TIEMPO PROMEDIO EN MINUTOS PARA LA BÚSQUEDA DE INFORMACIÓN PARA LOS REPORTES QUE SE ENVÍAN A MINCETUR	311
6.2.3.1 PREGUNTA ASOCIADA.....	311
6.2.3.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS.....	311
6.2.3.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	311
6.2.3.4 RESULTADOS.....	312
6.2.4 GUNTA 4: CUALES SON LOS PROBLEMAS COMUNES QUE SE PRESENTAN EN LA EMPRESA	312
6.2.4.1 PREGUNTA ASOCIADA.....	312
6.2.4.2 RESPUESTA PROPORCIONADA POR LOS USUARIOS.....	312
6.2.4.3 ANÁLISIS DE LOS DATOS OBTENIDOS.....	312
6.2.4.4 RESULTADOS.....	313
CAPITULO VII	
CONCLUSIONES Y RECOMENDACIONES.....	314
7.1 CONCLUSIÓN.....	315
7.2 RECOMENDACIÓN.....	315

CAPITULO VIII

GLOSARIO DE TÉRMINOS.....317 8.1

GLOSARIO.....318 **CAPITULO**

IX

REFERENCIAS BIBLIOGRÁFICAS.....319

9.1 REFERENCIAS BIBLIOGRÁFICAS.....320

ANEXOS..... 323

RESUMEN

Como sabemos hoy en día existen muchos tipos de empresas, empresas que se desenvuelven un rubro específico, empresas Industriales, textiles, Distribuidoras, Empresas Hoteleras, que es el tipo de empresa en el cual se enfoca este proyecto, y muchas más. Muchas de las empresas tienen un sistema manual al menos en nuestra Provincia, o porque no decirlo las empresas más grandes que son conocidas a nivel mundial iniciaron sus operaciones con un sistema manual, con el cual realizan o realizaban todas sus operaciones, que les resultaba muy útil hasta cierto límite ya que como sabemos una empresa busca crecer, y por ende crece también la cantidad de información que maneja, información como la de sus proveedores, clientes, información sobre ventas diarias, mensuales, anuales, etc.

Es por esta razón que las empresas se ven obligadas a tomar una importante decisión el cual implica automatizar sus procesos que puede ser mediante un sistema computarizado, si bien el cambio de lo manual a lo automatizado es un cambio muy drástico pero se debe tener en cuenta que trae muchos beneficios para la empresa.

Estos beneficios son la razón por la cual se pretende mediante este proyecto implementar un sistema computarizado para la Empresa de Serv. Turístico Princess S.A. en el rubro hotelero como se mencionó anteriormente.

Durante los siguientes capítulos se verán detalladamente los problemas que presenta la empresa, la situación actual de la empresa (como la empresa realiza sus operaciones), las metodologías que se utilizarán para la solución, los objetivos generales y específicos del Proyecto, los instrumentos de recolección de información que se utilizaron para el proyecto, los materiales que serán requeridos para el proyecto, los antecedentes de algún tipo de proyecto, también se realizará un cronograma con el cual se tendrá un mejor control de la duración del proyecto, el costo de cuanto demandara realizar el proyecto.

ABSTRACT

As we know today there are many types of businesses, companies that a specific category, industrial, textile, Distributors, Hotel Business, which is the type of company in which the project is focused, operate and many more. Many companies have a manual system at least in our province, or because they say the biggest companies that are known worldwide began operations with a manual system, which carried out or performed all its operations, it was for them too useful to some limit as we know a company looking to grow, and therefore so does the amount of information handled, information such as your suppliers, customers, information on daily sales, monthly, yearly, etc. It is for this reason that companies are forced to make an important decision which involves automating processes that may be using a computerized system, although the return for the automated manual is a drastic change but keep in mind which brings many benefits to the company. These benefits are the reason for which it is intended by this project to implement a computerized system for Enterprise Serv. Tourist Princess S.A. in the hotel business as mentioned above. For the next chapters will detail the problems of the company, the current situation of the company (as the company conducts its operations), the methodologies to be used for the solution, the general and specific objectives of the project, data collection instruments information that were used for the project, the materials that will be required for the project, a history of some type of project, also a schedule which better control over the duration of the project will be held, as the cost of sue carry out the project

INTRODUCCIÓN

El presente proyecto de investigación lleva por título: “Sistema Comutarizado para la Administración del rubro hotelero de la empresa de Serv. Turístico Princess S.A, el cual tiene por objetivo implementar un sistema de información, para lograr un mejor manejo del Area del hotel de la empresa.

Este proyecto ha sido dividido en los siguientes siete capítulos: Aspectos informativos, Aspectos de la investigación, Planificación del proyecto, Análisis y desarrollo del sistema, Integración, Discusión, conclusiones y recomendaciones. En el primer capítulo, Aspectos Informativos, se describen aspectos informativos de la empresa.

En el siguiente capítulo, aspectos de la investigación, se incluyen el marco lógico, el cual presenta la situación problemática, el problema de la investigación, la hipótesis de la misma, la definición de los objetivos, la justificación e importancia y definición de conceptos utilizados en el desarrollo del proyecto; el marco metodológico, en el cual se plantea el modo y tipo de investigación, la contrastación de la hipótesis, la población y muestra con la que se trabajará, los materiales de laboratorio, técnicas e instrumentos de recolección de datos a utilizar, así como los métodos y procedimientos para los mismos, análisis estadístico de datos, evaluación de impacto, comunicación de resultados y metodología; y por último el marco teórico, el cual incluye los antecedentes y base teórica de la investigación.

En el siguiente capítulo, planificación del proyecto, se presenta una descripción general y el alcance del sistema, la definición de los procesos de negocio, los usuarios del sistema, el cronograma de actividades, los factores críticos de éxito de la investigación, supuestos y restricciones y por último un análisis de viabilidad del mismo.

El cuarto capítulo, Análisis y desarrollo del sistema, incluye las fases consideradas para el desarrollo del mismo: análisis, diseño, construcción,

pruebas, puesta en marcha y especificaciones de seguridad del sistema. En el quinto capítulo, Integración del proyecto, se describe las consideraciones de seguridad, puesta en marcha del proyecto.

A continuación, en el sexto capítulo, discusión, se presenta un análisis estadístico de los resultados obtenidos en las dos encuestas realizadas al inicio y al final del proyecto con la finalidad de determinar si se logró el objetivo de la investigación.

Finalmente, en el último capítulo se incluyen las conclusiones y recomendaciones de la investigación.

CAPITULO I ASPECTOS INFORMATIVOS

1.1 ASPECTOS ORGANIZACIONALES

1.1.1 DESCRIPCIÓN DE LA ORGANIZACIÓN

La Empresa de Serv. Turísticos Princess S.A., cuya denominación en la sociedades mercantiles es como Sociedad Anónima (S.A.); según los dueños actuales. La empresa inicia siendo una panadería, y que los dueños que inicián el negocio eran de desendencia oriental, mas Adelante la empresa inicia los servicios de restaurante y hospedaje como hoy día la conocemos, actualmente el representante legal es el señor Chia Kam Fermin identificado con Dni. 21879070.

La empresa cuenta con 17 empleados capacitados y con buena atención a los clientes; esta empresa a nivel nacional además brinda a sus clientes lo mejor en calidad y en servicios de tal manera que les permita gozar de ventajas competitivas; obteniendo de esta manera ser líder en el rubro de la industria hotelera, reconocimiento y prestigio de la población peruana.

DATOS DE LA EMPRESA

- **Número de RUC:** 20279522835 - EMP DE SERV TURISTICOS PRINCESS S.A.
- **Tipo Contribuyente:** Sociedad Anónima.
- **Fecha de Inscripción:** 29/08/1995
- **Fecha de Inicio de Actividades:** 29/08/1995
- **Estado del Contribuyente:** Activo
- **Condición del Contribuyente:** Habido
- **Dirección:** JR. LIMA NRO. 109 ICA - CHINCHA - CHINCHA ALTA
- **Sistema de emission de comprobante:** Manual
- **Sistema de Contabilidad:** Manual

Ubicación:



FUENTE: GOOGLE MAPS

1.1.2 ESTRUCTURA ORGÁNICA



Fuente: Elaboración Propia

1.1.2 FACTORES ESTRATÉGICOS

1.1.2.1 VISIÓN

La empresa de servicios turísticos Princess s.a. tiene la visión de lograr un liderazgo dentro del grupo de hoteles del mismo nivel. Tenemos la visión de ser el mejor hotel en nuestro medio, y como tal ubicarnos como el preferido de nuestros clientes. La calidad de servicio esta basada en los años de experiencia adquiridos por los propietarios, socios y trabajadores.

1.1.2.2 MISIÓN

Tenemos como misión hacer que el cliente se sienta cómodo, tranquilo y engréido, mantener una filosofía de excelencia en servicio al cliente y hacer uso en todo momento de nuestras incomparables ventajas competitivas para proporcionar a nuestros clientes el más elevado valor percibido.

1.1.3 ANÁLISIS FODA

El Análisis DAFO, también conocido como Matriz o Análisis 'FODA'.

	FORTALEZAS	DEBILIDADES
	Amplia experiencia por parte de los accionistas y del personal. Trabajo en equipo con eficiencia.	Falta de un Sistema Informático que automatice las actividades comerciales. Carencia de un Plan Estratégico Empresarial.
OPORTUNIDADES	Estrategias Agresivas	Estrategias Oportunistas
Los días festivos en nuestra localidad. Las nuevas tecnologías que salen al mercado.	1.- Aprovechar los días festivos y nuestra amplia experiencia para liderar con nuevos clientes siempre el mercado. 2.- Desempeñar un buen trabajo en equipo para captar la mayor cantidad de clientes en días festivos,	1.- Automatizar el proceso de comercialización a través de un sistema informático. 2.- Diseñar un plan estratégico empresarial que ayude a liderar la empresa.
AMENAZAS	Estrategias Diversificación	Estrategias Defensivas
Incremento de la competencia. Alza de precios según	1.- Aprovechar nuestra amplia experiencia para contrarrestar el incremento de la competencia. 2.- Aprovechar el trabajo en equipo para contrarrestar las alzas de los precios en lo que involucra a los insumos del raturant.	1.- Implementar un sistema Informático que automatice el los procesos que involucran al hotel. 2.- Diseñar plan estratégico empresarial para poder competir con el incremento de la competencia.

CAPITULO II ASPECTOS DE LA INVESTIGACIÓN

2.1 MARCO LÓGICO

2.1.1 SITUACIÓN DE LA PROBLEMÁTICA

La empresa Princess S.A., está ubicada en la provincial de Chíncha del departamento de Ica, se dedica actualmente a brindar servicios de hospedaje y restaurant (Turístico), la cual cuenta con un período de 22 años de funcionamiento, brindando un servicio de calidad en el mercado a nivel nacional, esta empresa cuenta con empleados capacitados y con buena atención a los clientes.

La empresa de Serv. Turístico Princess S.A., como mencionamos actualmente brinda los servicios de hotel y restaurant, en lo que respecta al restaurant podemos mencionar que la empresa brinda un servicio basándose en la buena atención al cliente, ya que como sabemos una buena atención al cliente les garantiza un crecimiento en el mercado con respecto a sus competidores, de parte del hotel se podría decir que la idea de servicio van acorde con los del restaurant centrándose en hacer sentir al cliente como en casa.

Actualmente los problemas que presenta la empresa, dentro de esta área los podemos encontrar en el proceso de venta, ya que no cuenta con un sistema de comercialización y todo lo realiza de forma manual esto provoca pérdida de tiempo, también se presentan problemas con la malinterpretación de la información al momento de la atención al cliente, ya que las áreas del restaurant no se encuentran comunicadas y el envío de información entre dichas áreas es muy lento, la empresa no tiene control de almacén, presenta pérdidas insumos para la preparación de platillos. También encontramos problemas con el control del personal, ya que algunos empleados no respetan la hora de entrada, y cometen faltas, faltas que generan pérdida para la empresa. En el control del hotel la empresa todo lo realiza al igual que el restaurant de forma manual en el cobro por el servicio, el registro de habitaciones, la verificación de cuartos disponibles, verificación de la cantidad de huéspedes durante el mes, cabe mencionar que la empresa envía

mensualmente un reporte al MINCETUR (Ministerio de Comercio Exterior y Turismo) en el cual se indican la cantidad de huéspedes que tuvieron, nacionalidad de los mismos, todas estas actividades dificulta el control, y conlleva a un mayor esfuerzo para la administradora y demás empleados del Área del hotel.

Todos estos problemas encontrados en las diferentes áreas, si bien cumplen con su objetivo que es la atención del cliente demandan mucho tiempo.

La empresa de Serv. Turístico Princess S.A., teniendo un conocimiento acerca de la creciente de las tecnologías, y con el fin de solucionar los problemas decidió implementar un sistema que abarque el área del hotel, en lo que respecta al proceso de reservación, registro de clientes del hotel, cálculo por los servicios, etc. para así, tener un mejor control del área del hotel, logrando ser más competitivo y brindar un servicio de óptima calidad para su clientes.

2.1.2 PROBLEMA

PROBLEMA GENERAL O PRINCIPAL

El presente proyecto investigara el área de administración del rubro hotelero de la empresa Princess S.A. y sus diferentes problemas que presenta para luego darle solución mediante la implantación un sistema computarizado, seguidamente se realizara un seguimiento para analizar los resultados obtenidos con el sistema.

PROBLEMAS ESPECÍFICOS

PE1 ¿Cómo recolectar información de la empresa para el desarrollo del sistema computarizado?

PE2 ¿como hacer que sistema muestre información confiable y veraz?

PE3 ¿Cómo agilizar el flujo de información que maneja la empresa hacia otras entidades?

PE4 ¿cómo reducir el tiempo que se toma el atender un cliente u otro proceso?

PE5 ¿cómo ayudará el sistema con los objetivos de la empresa?

PE6 ¿Cómo adiestrar a los empleados para el uso del nuevo sistema?

2.1.3 HIPÓTESIS HIPÓTESIS GENERAL

El sistema ha implementar a satisfecerá las necesidades de la empresa Princess SA en lo que respecta a la administración del hotel.

HIPÓTESIS ESPECÍFICA

HPE1 ¿se tendrá fácil acceso a la información de la empresa?

HPE2 ¿el sistema brindará información confiable?

HPE3 ¿el sistema agilizará el flujo de información que maneja la empresa?

HPE4 ¿el sistema reducirá el tiempo que se toma en realizarse cada proceso de hotel?

HPE5 ¿él sistema ayudará con los objetivos de la empresa?

HPE6 ¿se logrará adiestrar a los empleados para el del sistema?

2.1.4 OBEJETIVOS

2.1.4.1 ETIVO GENERAL

Implementar un sistema computarizado para lograr la mayor eficiencia en la administración del rubro hotel de la Empresa de Serv. Turistico Princess S.A. - Chincha Alta.

2.1.4.2 ETIVOS ESPECÍFICOS

- Recopilar información acerca de la administración para su respectivo análisis.
- Brindar información integra, veraz, confiable y disponible a la organización.
- Permitir que los documentos que se entregan a MINSTUR y a al PNP se procesen más rápido.
- Minimizar el tiempo de atención a los clientes que solicitan un servicio de la empresa.
- Contribuir a no solo el cumplimiento de los objetivos organizacionales, sino también superarlo, gracias a la satisfacción de los clientes, lo cual se verá reflejado en la afluencia de los clientes.
- Orientar al personal sobre el sistema una vez probado e instalado.

2.1.5 JUSTIFICACIÓN E IMPORTANCIA

Este proyecto de investigación se debe de realizar porque así se podrán solucionar los problemas encontrados en el área específica de la empresa como lo es el área del hotel y para poder medir los resultados obtenidos una vez implantado el sistema computarizado. Los resultados obtenidos mediante el sistema beneficiarán a la empresa de modo cuantitativo (generando rentabilidad), y nosotros los investigadores de forma cualitativo mediante la obtención de nuevos conocimiento y experiencia.

JUSTIFICACIÓN METÓDICA

El trabajo de investigación se desarrollará siguiendo los procedimientos del método científico, del método descriptivo y el diseño descriptivo correlacional, con la rigurosidad que requiere la investigación científica, para tener resultados científicos que

expliquen la relación entre las dos variables de estudio.

JUSTIFICACIÓN SOCIAL

El presente proyecto permitirá crecer a la empresa cumpliendo sus objetivos generando rentabilidad, para así ser más competitiva, este proyecto también puede ser implantado en otras instituciones similares en la región y el país. Ya que será de utilidad para el ejercicio de administración de cualquier empresa hotelera.

Nuestro reto es crear un sistema computarizado no solo en esta empresa, sino también en todas las empresas de nuestra localidad que se dediquen a un determinado rubro y que las personas vean que el uso de la tecnología aplicada a un determinado problema puede beneficiarlos, maximizando sus ganancias.

2.1.6 DEFINICIÓN DE TERMINOS Y CONCEPTOS COMENSAL HOTEL DE TRES ESTRELLAS

Estos hoteles tienen un costo medio. Cuentan con amplios espacios en cada habitación y un mobiliario completo con sillas, mesas, armarios, televisor, teléfono privado y baños confortables. Algunos incluso poseen una pequeña heladera que ya viene con bebidas que se pagan al final de la estadía en caso de que las consumas. Siempre están bien ubicados, sea porque están en el casco céntrico de la ciudad o por encontrarse en lugares turísticos cerca de grandes atracciones.

ITERACIÓN

En el contexto de un proyecto se refieren a la técnica de desarrollar y entregar componentes incrementales de funcionalidades de un negocio.

PROCESO

Es un conjunto de actividades recurrentes mediante las cuales se transforma un grupo de entradas (insumos) en un grupo de salidas o resultados valiosos para un cliente (interno o externo).

PLATAFORMA

Una plataforma es un sistema que sirve como base para hacer funcionar determinados módulos de hardware o de software con los que es compatible.

PROTOCOLO

Conjunto de reglas usadas por computadoras para comunicarse unas con otras a través de una red por medio de intercambio de mensajes.

2.2 MARCO MÉTODOLÓGICO

2.2.1 TIPO Y NIVEL DE INVESTIGACIÓN

Por el tipo de investigación, el presente estudio reúne las condiciones metodológicas de una investigación aplicada, en razón que se utilizaron conocimientos de las ciencias administrativas, y también porque se busca resolver un problema específico.

Por su temporalidad en el tiempo es de forma longitudinal ambispectiva o también llamada estudio mixto (retrospectivo - prospectivo), ya que para realizar la investigación y dar solución al problema se toman datos del pasado empleando para ello información consignada en registros (fuentes secundarias), pero a la vez inicia el estudio, indagando sus datos en una fuente primaria, es decir de las personas que hacen parte de la investigación.

Por su nivel de complejidad es correlacional ya que con esta investigación se pretende determinar el grado de relación que

pueden tener las variables de la investigación.

2.2.2 CONTRASTACIÓN DE LA HIPÓTESIS

2.2.2.1 MODELO LÓGICO DE CONTRASTACIÓN

CONSECUENCIAS LÓGICAS

Si se implementará el sistema computarizado, entonces se logrará una mejor administración en el área del hotel haciendo el trabajo más eficiente.

OPERACIONALIZACIÓN CON VARIABLES VARIABLE INDEPENDIENTE

VARIABLES	NOMBRES	DIMENSIONES	INDICADORES
V1	Sistema computarizado	Recopilación de información	Formularios Entrevistas
		Agilizar el proceso de envío de documentos a entidades públicas	Diagramas de flujos
		Brindar información integra veraz y confiable.	Formulario de comparación de datos. Reportes del sistema.
		Agilizar la atención al cliente.	Afluencia de los clientes. Encuestas a los clientes.
		- Inducir al Personal sobre el uso del sistema	- El uso del corrector del sistema

VARIABLE DEPENDIENTE

VARIABLES	NOMBRES	DIMENSIONES	INDICADORES
V1	La buena administración	Control de recursos de la empresa.	- Reducción de desperdicios y pérdidas de la

	del rubro hotelero		empresa.
		Control de los ingresos y egresos.	- Reportes mensuales o anuales
		Generar rentabilidad.	Maximizar las ganancias de la empresa.
		Realización de un plan estratégico.	Obtención de los resultados.

2.2.3 POBLACIÓN Y MUESTRA

La población esta dirigida al hotel Princess S.A. y la muestra son los trabajadores del área de administración del hotel.

DISEÑO DE LA INVESTIGACIÓN

El diseño de la investigación que se utilizará es el descriptivo – correlacional, que se formula de la siguiente manera.

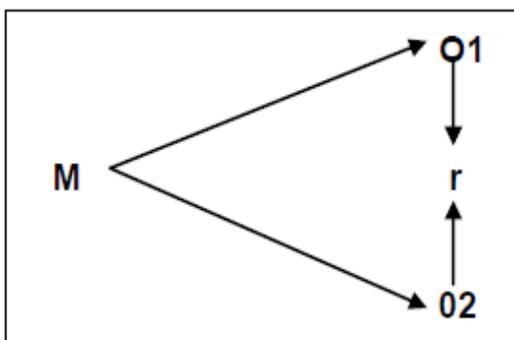
Donde:

M = Muestra

O1 = VARIABLE 1

O2 = VARIABLE 2

r = Relación entre variable1 y variable2



TÉCNICA DE MUESTREO

Para la presente investigación se escogió la técnica de la observación libre, la encuesta y la entrevista no estructurada. Los instrumentos que se aplicaron fue el libro de anotaciones diarias,

donde se lleva un registro semanal de las novedades más resaltantes observadas en la administración del hotel y la formulación de preguntas sin poseer un formato prediseñado. Las preguntas que se realizaron se formularon de manera espontánea, se dirigían a conocer sobre el manejo actual de la administración en el hotel, la importancia de los instrumentos de recolección de información, es el seguimiento del libro de sugerencias, reclamos y planificación para recoger y registrar los datos: hojas de control, test, fichas o guías de observación (Check out), etc.

Propósito: Es obtener datos (información), válidos y confiables, para su procesamiento y posterior análisis.

TECNICAS DE ANÁLISIS E INTERPRETACIÓN DE DATOS

Se utilizan los siguientes estadígrafos:

- La estadística descriptiva: Media, Moda y media aritmética
- La estadística inferencial : para la pruebas de la hipótesis se utilizarán las medidas de dispersión

2.2.4 MATERIALES DE LABORATORIO

a) HARDWARE

Descripción del Hardware a Utilizar

HARDWARE	CANTIDAD
Computadoras	2
Impresoras	1
Escáner	1

Fuente: Elaboración Propia

b) SOFTWARE

Descripción de Software a Utilizar

SOFTWARE	DESCRIPCIÓN

Microsoft Visual Studio 2010	Entorno de desarrollo integrado.
SQL Server 2008 R2 Epress Edition	Sistema gestor de base de datos.
CA Erwin Data r9.2 Modeler Community Edition	Entorno de modelado del Negocio.
IBM- Rational Rose Enterprise	Herramientas de modelado de datos.
PS - Adobe Photoshop CS5	Editor de gráficos rasterizados.

Fuente: Elaboración Propia

c) SISTEMA OPERATIVO

Descripción del Sistema Operativo a Utilizar

SITEMA OPERATIVO		TIPO DE LICENCIA	
Windows 7 Professional (32 bit)		Microsoft (EULA)	CLUF

Fuente: Elaboración Propia

d) MATERIALES DE ESCRITORIO

Descripción de Los Materiales de Escritorio a Utilizar

MATERIALES DE ESCRITORIO
Cd
DVD
Papel Bond A4
Lapiceros
Folders
Sobre Manila A4

Fuente: Elaboración Propia

e) SERVICIOS

Descripción de los servicios a utilizar

SERVICIOS
Movilidad
Alimentos
Telecomunicaciones
Fotocopias
Impresiones
Espiralados
Encuadernación
Energía Eléctrica
Internet

Fuente: Elaboración Propia

2.2.5 TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

2.2.5.1 TÉCNICAS DE GABINETE

Utilizaremos libros de programación y revistas Científicas- tecnológicas e informes científicos referidos al problema de investigación, del mismo modo se hará uso de materiales de escritorio para el desarrollo del trabajo de investigación.

2.2.5.2 TÉCNICAS DE CAMPO

Se hará uso de encuestas y entrevista para recojo de la información pertinente relacionado al trabajo de investigación.

2.2.6 MÉTODOS Y PROCEDIMIENTOS PARA LA RECOLECCIÓN DE DATOS

Para la ejecución de la tarea número uno haremos uso del método histórico tendencial, el mismo que está vinculado al conocimiento de las distintas etapas del surgimiento desarrollo y evolución del sistema computarizado del rubro hotelero para la empresa Princess

S.A., tema de investigación.

Para desarrollar la segunda tarea, haremos uso del método empírico que permitirá recoger toda la información empírica respecto a la opinión de los usuarios, pre y post a la implantación del sistema.

La tercera tarea, está referida a la Implementación del sistema computarizado, esta se pasara a sistematizar; para lo cual utilizaremos la metodología agil Extreme Programming (XP) junto al marco de trabajo SCRUM para la gestión y desarrollo del sistema. Para la tarea cuatro se alojará la aplicación y la base de datos en una terminal de la empresa, esto permitirá posteriormente realizar las pruebas necesarias para poder determinar si se logro alcanzar el objetivo del proyecto.

2.2.7 ANÁLISIS ESTADÍSTICOS DE DATOS

El procesamiento y análisis estadístico de datos serán vertidos en una base creada en la herramienta Microsoft Excel.

A partir de los datos recolectados en las dos encuestas; las cuales se realizarán al inicio del proyecto para determinar la situación actual de la empresa Princess sac y los problema que la afectan, y al finalizar el proyecto para poder obtener los resultados de la puesta en marcha del sistema computarizado; se presentarán tablas y gráficos las cuales permitirán realizar una comparación entre los datos obtenidos al inicio del proyecto con los obtenidos durante la puesta en marcha del mercado virtual.

2.2.8 EVALUACIÓN DEL IMPACTO

El impacto de la investigación se evaluará mediante la realización de dos encuestas. La primera se realizara al inicio del proyecto para determinar la situación actual de las empresa y el principal problema que las afecta; la segunda se realizará al finalizar el proyecto, para poder obtener los resultados de la puesta en marcha del sistema computarizado para la administración del rubro hotelero. Posterior a

estas dos etapas se pasará a realizar una comparación entre los datos obtenidos al inicio del proyecto con los obtenidos durante la puesta en marcha del sistema computarizado para la administración del rubro hotelero para determinar si se logró satisfacer la hipótesis y alcanzar el objetivo planteado en la investigación.

2.2.9 COMUNICACIÓN DE RESULTADOS

Los resultados que se obtengan como producto de la investigación se darán a conocer a los beneficiarios mediante entregables, que serán presentadas en reuniones a los dueños de la empresa.

2.2.10 METODOLOGÍA

La metodología que se utilizará para el desarrollo del trabajo de investigación será la programación extrema o XP junto al marco de trabajo SCRUM. Esta consta de las siguientes etapas:

1. IDENTIFICACIÓN DE REQUISITOS

En esta etapa se aclaran y detallan las funcionalidades que deberá tener el sistema, además se realizarán las siguientes tareas:

- a) Revisión del alcance propuesto para la aplicación
- b) Identificación de los requerimientos de la aplicación
- c) Priorización de los requerimientos

2. DEFINICIÓN DE ITERACIONES

En esta etapa se estimarán las iteraciones o ciclos de desarrollo necesarios para culminar el proyecto con éxito, esto estará en función a la lista de requerimientos establecida en la etapa anterior. Además, cada iteración abarcará de 3 a 4 semanas y a su término debe generar un entregable funcionando y listo para ser probado.

3. DESARROLLO DE ITERACIONES

En esta etapa se desarrollan las iteraciones estimadas en la etapa cada iteración incluye las siguientes tareas:

- a) Identificación y captura de información necesaria

- b) Diseño de base de datos
- c) Diseño de Formularios
- d) Implementación de base de datos
- e) Carga de datos mínimos necesarios
- f) Construcción.
- g) Pruebas Unitarias.
- h) Aprobación de resultados de iteración

Además, se debe tener en cuenta que al finalizar cada iteración se realizara la integración respectiva y se realizaran las pruebas de integración del sistema.

4. CARGA DE DATOS PUESTA EN MARCHA

En esta etapa se registrarán los datos iniciales necesarios para la aplicación.

5. PUESTA EN MARCHA

En esta etapa se realizara el alojamiento del sistema y de la base de datos, además incluirá las siguientes tareas:

- a) Pruebas del Sistema
- b) Ajuste
- c) Ejecución

2.3 MARCO TEÓRICO

2.3.1 ANTECEDENTES

a) INTERNACIONALES ANTECEDENTE 1

AUTOR: Adrián Esteban Cartagenova Zúñiga.

AÑO: Diciembre 2010

TEMA: Diseño de un sistema de gestión de calidad en el servicio para hoteles todo incluido caso hotel royal Acme Pórtate Beach Resort.

LUGAR: Ecuador, Quito

OBJETIVO GENERAL:

Diseñar un sistema de Gestión de la Calidad en el Servicio para Hoteles Todo Incluido, que permita mejorar la satisfacción del cliente externo mediante la gestión de procesos a través de indicadores. **CONCLUSIONES:**

Este trabajo finaliza con su investigación afirmando que el sistema de gestión de calidad propuesto deberá de permitir conocer a profundidad el funcionamiento del hotel, sus procesos y actividades, Así como sus interrelaciones, factores que generan valor agregado al servicio proveedores y procesos de apoyo. El sistema deberá ser medible y cuantificable a través de indicadores de gestión claramente definidos e implementados los cuales permitirán la toma de decisiones objetivas y el seguimiento de las mismas, transformada así al hotel en una organización dinámica, abierta al cambio con capacidad de adaptación y que mejora continuamente, reflejándose todo esto en satisfacción al cliente y participación en el mercado.

Por otra parte, una administración en el servicio basada en el compromiso gerencial, las personas y la gestión de la calidad centrada en los procesos de la empresa, no serán útiles si no se hallan mecanismos que aseguren la continuidad de los esfuerzos realizados y la adaptabilidad de la empresa a nuevos escenarios. Para esto se requiere de un cambio en la manera de ver, pensar, hacer y analizar las cosas, es decir un cambio en la cultura y filosofía organizacional, se necesita de un enfoque holístico el cual se fundamente en la fuerza laboral como uno de sus pilares fundamentales así como una constante mejora de los procesos y prácticas, todo esto orientado hacia el cliente y el servicio prestado a él.

ANTECEDENTE 2

AUTOR: Alina Concepción Criollo Reyes

AÑO: Junio 2011

TEMA: Administración del área de Recursos Humanos del Hotel Chachalacas

LUGAR: Veracruz, México

OBJETIVO GENERAL:

Elaborar un plan de trabajo como propuesta en la administración de recursos humanos del hotel Chachalacas.

CONCLUSIÓN:

En la administración de empresas, se denomina recursos humanos al trabajo que aporta el conjunto de los empleados o colaboradores de esa organización, se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización. El objetivo básico que persigue la función de Recursos Humanos con estas tareas es alinear las políticas de la empresa.

Hoy en día se hace necesario que entendamos la importancia de cada uno de los procesos y registros establecidos en el área de recursos humanos, encontrándonos ante un entorno de constantes cambios y de alta competitividad.

La Administración de recursos humanos busca compenetrar el Recurso Humano con el proceso productivo de la empresa, haciendo que éste último sea más eficaz como resultado de la selección y contratación de los mejores talentos disponibles en función del ejercicio de una excelente labor de estos. Así como también la maximización de la calidad del proceso productivo depende de igual modo de la capacitación de los elementos de los recursos humanos para hacer más valerosos sus conocimientos. El departamento de recursos humanos de una empresa busca que las estrategias y políticas que usa cada departamento sean las más adecuadas, y en todo caso funge como asesoría y consultoría de cada departamento. Este es un departamento que une de manera más eficiente los Recursos Humanos.

b) NACIONALES ANTECEDENTE I AUTOR:

- Claudia María del Rosario, Chiclayo Del Carpio
- Naddia Grethel Mía, Segura Salazar

AÑO: 06 de Enero del 2009

TEMA: herramientas informáticas que dan soporte a la gestión estratégica en el sector hotelero de categoría tres estrellas. Caso Garza hotel– Chiclayo – Lambayeque - Perú

LUGAR: Chiclayo – Perú

OBJETIVO GENERAL:

Elegir una herramienta informática gerencial basada en Balanced Scorecard que mejor se adapte al soporte de la gestión estratégica en empresas hoteleras de categoría tres estrellas según los lineamientos para la selección de software.

CONCLUSIÓN:

Se eligió a Delphos Manager como la Herramienta Informática Gerencial que mejor se adapta al soporte de la gestión estratégica basada en el BSC, como se demostró con el caso aplicativo de la empresa Garza Hotel de categoría tres estrellas previo análisis según los lineamientos para la selección de software.

Se dio a conocer a la gerencia y personal en general sobre la importancia del Balanced Scorecard a través de las reuniones concertadas para el levantamiento de la información de la empresa.

Se realizó un análisis de toda la empresa obteniendo las características resaltantes para la elaboración del BSC. Se analizó a la empresa Garza Hotel desde la perspectiva financiera, clientes, aprendizaje y crecimiento, procesos internos y de

Responsabilidad Social.

Se realizó el análisis comparativo de tres herramientas informáticas gerenciales: Microsoft Office Performance Point Server, Delphos Manager y Tablero de comando, evaluadas según los lineamientos funcionales, técnicos, sobre el proveedor, estratégicos y económicos para la adecuada selección de software.

Se realizó un análisis de la empresa desde la perspectiva de responsabilidad social concluyendo que esta goza de un buen clima laboral de tal manera que los trabajadores se encuentren contentos y con motivación para brindar un mejor servicio; esto contribuye a una participación en la empresa y con la región.

2.3.2 BASE TEÓRICA

2.3.2.1 LA ADMINISTRACIÓN

La administración se define como el proceso de diseñar y mantener un ambiente en el que las personas trabajando en grupo alcance con eficiencia metas seleccionadas. Esta se aplica a todo tipo de organizaciones bien sean pequeñas o grandes empresas lucrativas y no lucrativas, a las industrias manufactureras y a las de servicio.

Otros conceptos:

- **Koontz & O'Donnell**

Es la dirección de un organismo social, y su efectividad en alcanzar sus objetivos, fundada en la habilidad de conducir a sus integrantes.

- **George R. Terry**

La Administración es un proceso muy particular consistente en las actividades de planeación, organización, ejecución y control, que son desempeñadas para determinar y alcanzar los objetivos señalados con el uso de seres humanos y otros recursos.

- **Wilburg Jiménez Castro**

Se define como el proceso de prever, planear, organizar, integrar, dirigir y controlar el trabajo de los elementos de la organización y de utilizar los recursos disponibles para alcanzar los objetivos preestablecidos.

- **Agustín Reyes Ponce**

La Administración es el conjunto sistemático de reglas para lograr la máxima eficiencia en las formas de estructurar y manejar un organismo social.

La administración consiste en darle forma, de manera consistente y constante a las organizaciones. Todas las organizaciones cuentan con personas que tienen el encargo de servirle para alcanzar sus metas, llamados gerentes, administradores, etc.

a) GERENTE

Un Gerente es una persona responsable de dirigir las actividades que ayudan a las organizaciones para alcanzar sus metas. La medida de la eficiencia y la eficacia de un gerente, es el grado en que determine y alcance lo objetivos apropiados.

- Los gerentes actúan mediante relaciones que son vías de dos sentidos; una parte está sujeta a la otra
- Los gerentes actúan mediante relaciones que tienen repercusiones que involucran a otras personas, para bien o para mal.

b) ADMINISTRADORES

Son individuos en una organización que dirigen las actividades de otros. Estos también podrán tener algunas responsabilidades operativas. Se pueden dividir en dos grupos:

- Los operativos son personas que trabajan directamente un

un puesto o actividad y no tienen responsabilidad de supervisor el trabajo de otros empleados.

- Los administradores dirigen las actividades de otras gentes.

Los administradores convierten un conjunto de recursos humanos, materiales, técnicos, monetarios, de tiempo y espacio en una empresa útil y efectiva. Estos tienen la responsabilidad de realizar acciones que permitan que las personas hagan sus mejores aportaciones a los objetivos del grupo.

Los administradores se clasifican en:

- Los administradores de primera línea por lo general lo llamamos supervisores.
- Los de mandos medios pueden ostentar títulos como de jefe de departamentos o de oficina, líder del proyecto, jefe de unidad, gerente de distrito, decano, obispo o gerente divisional.

Los administradores por lo común ostentan títulos de vicepresidente, presidente, canciller, director administrativo, director general etc.

c) OBJETIVOS DE LA ADMINISTRACIÓN

- Alcanzar en forma eficiente y eficaz los objetivos de un organismo social.
- Eficacia, cuando la empresa alcanza sus metas y eficiencia cuando logra sus objetivos con el mínimo de sus recursos.
- Es permitirle a la empresa tener una perspectiva más amplia del medio en el cual se desarrolla.
- Asegurar que la empresa produzca o preste sus servicios.

d) CARACTERÍSTICAS DE LA ADMINISTRACIÓN

- SU UNIVERSABILIDAD

El fenómeno administrativo se da donde quiera que existe un organismo social, porque en él tiene siempre que existir coordinación sistemática de medios. La Administración se da por lo mismo en el Estado, en el ejército, en la empresa, en una sociedad religiosa, etc.

Además, los elementos esenciales en todas esas clases serán los mismos, aunque, lógicamente, existan variantes accidentales.

- SU ESPECIFICIDAD

Aunque la administración va siempre acompañada de otros fenómenos de distinta índole (contables, sociológicos, psicológicos, jurídicos, etc.), el fenómeno administrativo es específico y distinto a los que acompaña.

Se puede ser un magnífico ingeniero de producción (como técnico en esta especialidad) y un pésimo administrador.

- SU UNIDAD TEMPORAL

Aunque se distingan etapas, fases y elementos del fenómeno administrativo, éste es único y, por lo mismo, en todo momento de la vida de una empresa se están dando, en mayor o menor grado, todos o la mayor parte de los elementos administrativos. Así, al hacer los planes, no por eso se deja de mandar, de controlar, de organizar, etc.

- SU UNIDAD JERÁRQUICA

Todos cuantos tienen carácter de jefes en un organismo social participan, en distintos grados y modalidades, de la misma Administración. Así, en una empresa, forman "un solo cuerpo administrativo: desde el Gerente General hasta el último mayordomo.



e) **IMPORTANCIA DE LA ADMINISTRACIÓN**

- La Administración se da donde quiera que exista una organización.
- El éxito de una empresa u organismo social se debe a la buena administración que poséa.
- Para las empresas pequeñas y medianas, la manera más indicada de competir con otras es el mejoramiento de su administración, dicho en otras palabras, tener una mejor coordinación de sus recursos incluyendo al humano.
- Para las organizaciones que están en vías de desarrollo, el principal elemento para desarrollar su productividad y su competitividad con otras es mejorar la calidad en su administración.

f) **NIVELES ADMINISTRATIVOS**

- **NIVEL ESTRATÉGICO**

Es la administración del nivel superior que tiene el mayor poder y lleva la responsabilidad total de una empresa.

- **NIVEL TÁCTICO**

Es la administración que reporta a la administración del nivel más alto el funcionamiento detallado de la empresa, además de desarrollar planes para implementar las metas generales establecidas por la alta dirección.

- **NIVEL OPERATIVO**

Es la administración que supervisa a los trabajadores y las operaciones que realizan.



g) EVOLUCIÓN DE LA ADMINISTRACIÓN

Existen dificultades al remontarse al origen de la historia de la administración. Algunos escritores remontan su desarrollo a los comerciantes sumerios y a los egipcios, antiguos constructores de las pirámides o a los métodos organizativos de la iglesia y las milicias antiguas.

- Egipto (4000 a.c.)

Las pirámides de Egipto son evidencia actual de que se intentaron proyectos de enorme alcance, que emplearon decenas de miles de personas, con bastante anticipación a los tiempos modernos.

- Siglo XIX

Algunos piensan en la Administración moderna como una disciplina que Comenzó como un vástago de la economía en el siglo XIX. Los Economistas clásicos, tales como Adam Smith y John Stuart Mill Proporcionaron un fondo teórico a la asignación de los recursos, a la Producción y a la fijación de precios. Al mismo tiempo, innovadores Como Eli White, James Wat y Matthew Boulton, desarrollaron herramientas técnicas de producción, tales como la estandarización, procedimientos de control de calidad, contabilidad analítica, y planeamiento del trabajo. Para fines del siglo XIX, León Walras, Alfred Marshall y otros economistas introdujeron una nueva capa de complejidad a los principios teóricos de la Administración. Además, Joseph Wharton ofreció el primer curso de nivel terciario sobre Administración en 1881.

- **Siglo X**

Durante el siglo XX, la Administración fue evolucionando en la medida en que las organizaciones fueron haciéndose más complejas y las ciencias, como la Ingeniería, la Sociología, la Psicología y la Teoría de sistemas fueron desarrollándose.

- **Primeras Prácticas Administrativas**

La Administración se ha desarrollado desde hace siglos, pero inicia a partir de la revolución industrial, dada la necesidad de dirigir en forma más eficiente las fábricas, de las que surgen diversos individuos.

h) EL PROCESO ADMINISTRATIVO

Es el proceso de lograr que las actividades se realicen por medio de la planeación, organización, coordinación o integración, dirección y control de otras personas, creando y manteniendo un ambiente, en el cual la persona se pueda desempeñar, entusiastamente, en conjunto con otras, sacando a relucir su potencial, eficacia y eficiencia, y lograr así fines determinados. El proceso se da al mismo tiempo. Es decir, el administrador realiza estas funciones simultáneamente.

Las funciones o procesos detallados no son independientes, sino que están totalmente interrelacionados. Cuando una organización elabora un plan, debe ordenar su estructura para hacer posible la ejecución del mismo. Luego de la ejecución (o tal vez en forma simultánea), se controla que la realidad de la empresa no se aleje de la planificación o, en caso de hacerlo, se busca comprender las causas de dicha situación. Finalmente, del control realizado puede surgir una corrección en la planificación lo que realimenta el proceso.

El proceso administrativo consiste en las siguientes funciones:

- **PLANIFICAR**

Implica definir metas, establecer estrategias y elaborar planes para coordinar actividades. Decidir qué hacer y cómo hacerlo. Implica priorizar y comprometerse. Históricamente era un proceso de “arriba hacia abajo” (la alta dirección formulaba planes y estrategias y daba las instrucciones a los niveles inferiores). Actualmente, no se concibe como una función exclusiva de alta dirección sino que, en mayor y menor medida, todos los niveles (y personas) planifican.

- **ORGANIZAR**

Asignar tareas, recursos y responsabilidades. Establecer una estructura de relaciones de modo que los empleados puedan interactuar y cooperar para alcanzar las metas organizativas. Se agrupa a las personas en departamentos y, después, se coordinan todas las partes.

El resultado de organizar es la creación de una estructura organizativa, siendo el organigrama su representación gráfica.

- **DIRIGIR**

Orientar, guiar y motivar la actuación de cada individuo de la organización con el fin de que ayuden a la consecución de las metas organizacionales. La comunicación es muy importante. En el pasado se identificaba esta función como mandar y supervisar (uso de poder y aplicación de permisos y/o sanciones en función del desempeño).

- **CONTROLAR**

Vigilar que el desempeño de las actividades se ajuste a lo planificado. En otras palabras, evaluar el desempeño y adoptar, si fuera necesario, medidas correctivas.



2.3.2.2 LA EMPRESA

La empresa es una unidad económica que combina los diferentes factores productivos, que se encuentran ordenados, según determinada estructura organizativa, localizados en unas o más unidades técnicas y físico-espaciales y dirigidos sobre la base de cierta relación de propiedad y control con el ánimo de alcanzar unos objetivos, entre los que destaca el beneficio empresarial.

TIPOS DE EMPRESA

a) SEGÚN EL SECTOR DE ACTIVIDAD

- EMPRESAS DE SECTOR PRIMARIO

También, son denominados extractivos, ya que el elemento básico de la actividad se obtiene, directamente, de la naturaleza: agricultura, ganadería, caza, pesca, extracción de áridos, agua, minerales, petróleo, energía eólica, etc.

- EMPRESAS DE SECTOR SECUNDARIO INDUSTRIAL

Se refieren a aquellas que realizan algún proceso de transformación de la materia prima. Abarca actividades tan diversas como la construcción, la óptica, la maderera, la textil, etc.

- EMPRESAS DEL SECTOR TERCIARIO O DE SERVICIOS

Incluyen a las empresas cuyo principal elemento es la capacidad humana para realizar trabajos físicos o intelectuales. Comprenden, también, una gran variedad de empresas, como las de transporte, bancos, comercio, seguros, hotelería, asesorías, educación, restaurantes, etc.

b) SEGÚN SU TAMAÑO

- GRANDES EMPRESAS

Son empresas que manejan grandes capitales. Por lo general, tienen instalaciones propias. Sus ventas son de varios millones de dólares, cuentan con un sistema de administración y operación muy avanzado, y pueden obtener líneas de crédito y préstamos importantes.

- MEDIANAS EMPRESAS

En este tipo de empresas, intervienen varios cientos de personas. Además, hay áreas bien definidas con responsabilidades y funciones. Asimismo, tienen sistemas y procedimientos automatizados.

- PEQUEÑAS EMPRESAS

En términos generales, las pequeñas empresas son entidades independientes. Además, son creadas para ser rentables, que no predominan en la industria a la que pertenecen, cuya venta anual en valores no excede un determinado tope y el número de personas que las conforman no excede un determinado límite.

- MICROEMPRESAS

Por lo general, la empresa y la propiedad son de propiedad individual, los sistemas de fabricación son, prácticamente, artesanales, la maquinaria y el equipo son elementales y reducidos, los asuntos relacionados con la Administración,

producción, ventas y finanzas son elementales y reducidas, y el director o propietario puede atenderlos personalmente.

c) SEGÚN AL PROPIEDAD DEL CAPITAL

- **EMPRESA PRIVADA**

La propiedad del capital está en manos privadas.

- **EMPRESA PÚBLICA**

Es el tipo de empresa en la que el capital le pertenece al estado, que puede ser Nacional, Provincial o Municipal.

- **EMPRESA MIXTA**

Es el tipo de empresa en la que la propiedad del capital es compartida entre el Estado y los particulares.

d) SEGÚN LA FORMA JURÍDICA

- **UNIPERSONAL**

El empresario o propietario, persona con capacidad legal para ejercer el comercio, responde de forma ilimitada con todo su patrimonio ante las personas que pudieran verse afectadas por el accionar de la empresa.

- **SOCIEDAD COLECTIVA**

En este tipo de empresas de propiedad de más de una persona, los socios responden, también, de forma ilimitada con su patrimonio y existe participación en la dirección o gestión de la empresa.

- **COOPERATIVAS**

No poseen ánimo de lucro y son constituidas para satisfacer las necesidades o intereses socioeconómicos de los cooperativistas, quienes, además, son a la vez trabajadores y, en algunos casos, también proveedores y clientes de la empresa.

- **COMANDITARIAS**

Poseen dos tipos de socios: los colectivos, con la característica de la responsabilidad ilimitada; y los comanditarios, cuya responsabilidad se limita a la aportación de capital efectuado.

- **SOCIEDAD DE RESPONSABILIDAD LIMITADA**

Los socios propietarios de estas empresas tienen la característica de asumir una responsabilidad de carácter limitada respondiendo solo por capital o patrimonio que aportan a la empresa.

- **SOCIEDAD ANÓNIMA**

Tiene el carácter de la responsabilidad limitada al capital que aporta, pero posee la alternativa de tener las puertas abiertas a cualquier persona que desee adquirir acciones de la empresa. Por este camino, estas empresas pueden realizar ampliaciones de capital, dentro de las normas que las regulan.

2.3.2.3 LA INDUSTRIA HOTELERA

El movimiento de millones de personas que se desplazan abandonando sus lugares de residencia habitual, motivados por diversas causas, provoca a su vez la creación de establecimientos que oferten tanto el alojamiento como la manutención, bien sea por separado o conjuntamente. Este da lugar a gran número de empresas de alojamientos turísticos de distintos tipos.

No obstante, para definir de un modo generalizado el concepto de alojamiento hay que tipificar a casi todos ellos y admitir las siguientes bases generales:

- Facilitan el alojamiento.
- Tienen capacidad de ofertar o no el servicio de

manutención según la demanda del cliente o turista.

- Se debe tener una clasificación y calificación que identifique al establecimiento, bien sea de carácter jurídico, bien a nivel comercial.
- Se establecerá un contrato entre ambas partes, establecimiento y cliente a priori, en el que se determinen una serie de factores de importancia para ambos, como son, entre otros:
 - Tipo de alojamiento
 - Cantidad de personas a alojar.
 - Dias de estancia.
 - Servicios contratados de manutención.
 - Precios, etc.
- Deben publicarse los precios a percibir por el establecimiento según los servicios ofertados.
- Se percibirá, en contraprestación de los servicios, ciertas cantidades de dinero.

MODALIDADES DE ALOJAMIENTO

Atendiendo a su denominación existen los siguientes tipos de alojamiento:

a) HOTELES

Son establecimientos que facilitan alojamiento con o sin servicios complementarios, distintos de los correspondientes a cualquiera de las otras dos modalidades” (hoteles- apartamentos y moteles).

Para que un establecimiento sea considerado en el grupo de hoteles deberá reunir además de las condiciones para la categoría que le corresponda, las siguientes:

- Ocupar la totalidad de un edificio o parte del mismo completamente independizada, constituyendo sus dependencias un todo homogéneo, con entradas, ascensores y escaleras de uso exclusivo.

- Facilitar al público tanto el servicio de alojamiento como el de comidas, con sujeción o no al régimen de pensión completa, a elección del cliente, y con excepción de los hoteles-residencias.
- Disponer al menos de un 10 por ciento de habitaciones individuales.

b) HOTELES – APARTAMENTOS

Son aquellos establecimientos que por su estructura y servicios disponen de las instalaciones adecuadas para la conservación, elaboración y consumo de alimentos dentro de la unidad del alojamiento.

c) MOTELES

Se define los Moteles como aquellos establecimientos situados en las proximidades de carreteras que facilitan alojamiento en departamentos con garaje y entrada independiente para estancias de corta duración. Son Moteles los establecimientos situados fuera de los núcleos urbanos y en las proximidades de las carreteras, en los que se facilita alojamiento, para estancias normalmente no superiores a veinticuatro horas, en departamentos, con entradas independientes desde el exterior, compuestos de dormitorio y cuarto de aseo y con garajes o cobertizos para automóviles, contiguos o próximos a aquellos”.

d) HOSTALES

Son Hostales aquellos establecimientos que, con las instalaciones y servicios exigidos como mínimo en esta sección a sus respectivas categorías, faciliten al público tanto el servicio de alojamiento como el de comidas, con sujeción o no al régimen de pensión completa, a la elección del cliente, y con excepción de los hostales-residencias.

e) PENSIONES

Son Pensiones aquellos establecimientos comprendidos por sus instalaciones y servicios es esta Sección que, no disponiendo de más de doce habitaciones, faciliten hospedaje, habitualmente, en régimen de pensión completa. Estos establecimientos quedan autorizados para exigir a sus clientes que se sometan a dicho régimen.

CARACTERÍSTICAS DE LA EMPRESA HOTELERA

En principio no existe ninguna diferencia conceptual entre una empresa hotelera y el resto de las empresas, sea cual sea el sector en que operen éstas.

Encontrada en el sector productivo de los servicios, no puede destacarse ningún elemento diferenciador, con respecto a otro tipo de empresas. La empresa hotelera, como cualquier otra, es un sistema abierto, con una organización habitualmente lineal, en la que pueden diferenciarse sin dificultad los subsistemas de aprovisionamiento, de producción, comercial, administrativo y financiero.

Las características son las siguientes:

- a) La primera característica a destacar, común a todas las empresas del sector, es la intangibilidad del producto principal que generan: el alquiler de habitaciones. Las habitaciones de un hotel que un día no son ocupadas por clientes, podrán serlo al día siguiente, pero no pueden almacenarse para ser alquiladas con posterioridad, ya que todo servicio solo puede consumirse cuando se genera. No toda la producción de un hotel es intangible; la producción de restauración no lo es, sin embargo también altamente precedera, en muchos hoteles esta última, puede tener incluso más importancia, valorándola en términos

monetarios, pero si hemos denominado producción principal al alquiler de habitaciones es por ser la razón primera que justifica la existencia de un hotel.

- b) La segunda característica, es la fuerte inversión de capital, que hay que realizar para acometer la creación de una industria hotelera, unida a una producción totalmente rígida, que hace muy complicada la conversión en otro tipo de industria, cuando por las razones que sean no se alcanzan los objetivos planeados, ni se satisfacen las expectativas que inspiraron el proyecto.
- c) La última característica – común a la mayoría de las empresas de servicios –, cabe destacar el gran peso del factor mano de obra, plasmado en que éste supone entre un 30 y un 45 por 100 de los costes totales de explotación; esto hace que sea con diferencia el de mayor peso específico de todos los factores de producción sacrificados en el proceso productivo.

TIPOLOGÍAS DE LAS EMPRESAS HOTELERAS

No son excesivas las clasificaciones tipológicas que pueden realizarse de las empresas hoteleras. Vemos las más significativas:

a) CLASIFICACIÓN POR EL TIPO DE ESTABLECIMIENTO

- Hoteles (hoteles, hoteles-apartamentos, moteles).
- Pensiones.

b) CLASIFICACIÓN POR SU LOCALIZACIÓN

- De ciudad.
- De costa.
- De montaña.
- De carretera.

c) CLASIFICACIÓN POR LA CATEGORÍA DE SERVICIOS PRESTADOS:

- De gran lujo.
- De lujo.
- De tipo intermedio.
- De tipo popular.

d) CLASIFICACIÓN POR LA DIMENSIÓN

- De gran tamaño: mas de 300 habitaciones
- De tamaño intermedio: mas de 100 y menos de 300 habitaciones.
- De tamaño pequeño: menos de 100 habitaciones.

e) CLASIFICACIÓN POR LA FORMA DE OPERACIÓN Y PROPIEDAD

- Explotadas directamente por la propiedad
- Explotadas de forma mixta
- Explotadas en régimen de franquicia, arrendamiento y gestión o management.

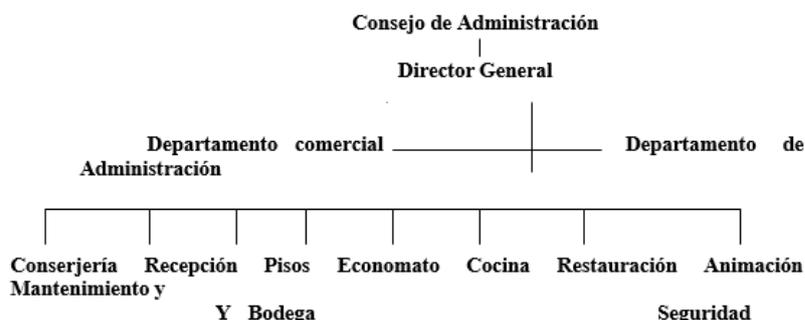
f) CLASIFICACIÓN POR AL FORMA DE PRODUCCIÓN

- Producción continua, o establecimientos que permanecen abiertos ininterrumpidamente
- Producción discontinua. Establecimientos de temporada.

LA ORGANIZACIÓN EN LA EMPRESA HOTELERA

La función de organización en un HOTEL se encarga de la identificación de cada una de las actividades a realizar, de su agrupamiento por departamentos, de la asignación a cada grupo de un administrador o supervisor y de establecer las medidas necesarias para la coordinación horizontal y vertical en la compañía.

En función de las características del entorno, de las estrategias y objetivos genéricos de la empresa, de la tecnología utilizada, del tamaño de la compañía, de su expansión geográfica, etc., la empresa hotelera presentará una estructura organizativa determinada. Con carácter general el organigrama que con más frecuencia presentan los hoteles independientes o individuales, es el siguiente:



En cada uno de estos departamentos trabajarán los siguientes empleados:

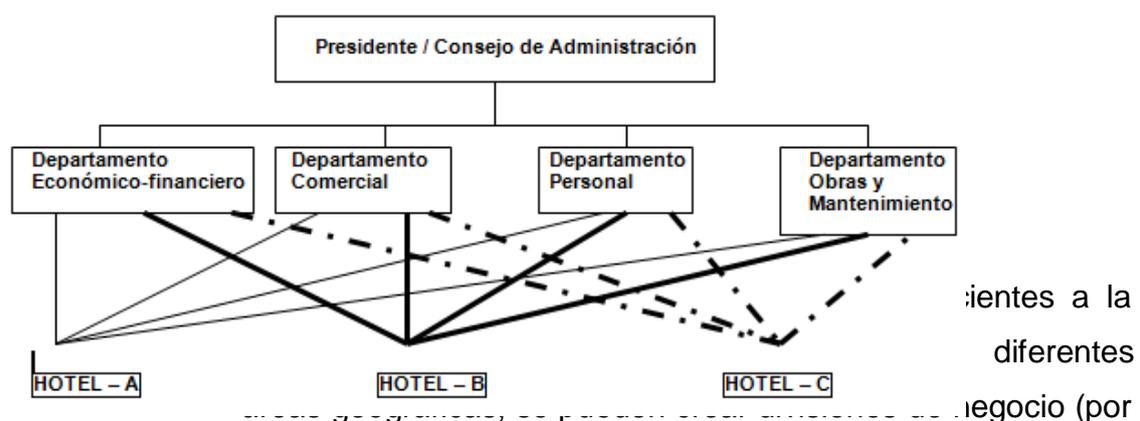
- **Departamento comercial:** director comercial, personal administrativo y en ocasiones técnicos o expertos en publicidad y relaciones públicas, además del director.
- **Departamento de administración:** jefe de administración, que en ocasiones realiza también las funciones propias del jefe de personal, contables y personal administrativo.
- **Conserjería:** jefe de conserjería, conserjes, botones, porteros de accesos y de aparcamiento.
- **Recepción:** jefe de recepción, recepcionistas, telefonistas, cajero.
- **Pisos:** gobernanta, su gobernanta, camareras de pisos.
- **Economato y bodega:** jefe de compras, ayudantes de economato y bodega.
- **Cocina:** jefe de cocina, cocineros, ayudantes de cocina,

pinches de cocina, reposteros, cafeteros, marmitones (se encargan de la limpieza y funcionamiento de la maquinaria de la cocina).

- **Restaurante:** maître o jefe de sala, camareros, ayudantes de camarero, somelier, barman.
- **Animación:** jefe de animation, monitores y personal técnico especializado en las distintas materias.
- **Mantenimiento y seguridad:** jefe de mantenimiento, electricistas, fontaneros, jardineros, etc.

En el caso de las cadenas hoteleras, suele existir una dirección comercial, de obras y mantenimiento, económico- financiera y de personal común para todos los establecimientos que la integran, siendo el director del hotel más bien un “director de operaciones”, que se limita a cumplir los objetivos fijados por estas direcciones generales para toda la cadena y dirige y controla los distintos departamentos operacionales del hotel (recepción, conserjería, pisos, cocina, restaurante, animación, mantenimiento y seguridad).

En este caso el organigrama más común sería:



ejemplo, división Europa, división América, división Asia, etc.), estableciendo las direcciones funcionales una serie de objetivos para cada división y controlando y dirigiendo éstas cada uno de los hoteles pertenecientes a la división.

Además de esta estructura intencionada y organización formal, también se suele encontrar una organización informal, bien sea por lazos de amistad, por relaciones de compañeros dentro de cada departamento, etc.

2.3.2.4 EI CLIENTE

- Es la persona más importante de nuestro negocio.
- No depende de nosotros, nosotros dependemos de él.
- Nos está comprando un producto o servicio y no haciéndonos un favor.
- Es el propósito de nuestro trabajo, no una interrupción al mismo.
- Es un ser humano de carne y hueso con sentimientos y emociones (como uno), y no una fría estadística.
- Es la parte más importante de nuestro negocio y no alguien ajeno al mismo.

Es una persona que nos trae sus necesidades y deseos y es nuestra misión satisfacerlo.

Es merecedor del trato más cordial y atento que le podemos brindar.

Es alguien a quien debemos complacer y no alguien con quien discutir o confrontar.

Es la fuente de vida de este negocio y de cualquier otro.

El cliente es nuestro jefe y nuestra razón de ser como empresarios.

a) Conocer al cliente

Cada empresario debe responder preguntas tales como:

¿Para qué mejorar la atención a mis clientes?, ¿Cómo hacerlo?,

¿Con quién?, ¿Me traerán ventajas o desventajas esos cambios?

Estos interrogantes encuentran respuesta a diario en el comportamiento de los consumidores, que reciben influencia de los medios de comunicación para modificar sus hábitos de compra con mucha rapidez. (Uso de distintos medios de pago, entrega a domicilio, compras por Internet, etc.)

Por otra parte, al consumidor ya "no le venden" nada sino que él decide y exige libremente dónde, qué, cómo y cuánto comprar. No Habrá llegado para el pequeño y mediano empresario la hora de preguntarse cosas como: ¿Soy consciente de esto? ¿Qué hago para que los consumidores "me elijan a mí"?

Si uno abre simplemente la puerta del local y espera a que vengan los clientes, ¿Vendrán a comprarme? ¿Seguirán viniendo? ¿Por qué deberían hacerlo? ¿Por qué no a la competencia? ¿Me alcanzará con lo que hoy hago para crecer?

¿Y para subsistir?

Alguien le puede preguntar: ¿Ud. vende o le compran?, ¿Cómo construye sus ingresos diarios?, ¿Conoce "a fondo" a sus clientes?,

¿Cómo se entera de sus necesidades y que hace para satisfacerlas? Estas y otras muchas reflexiones son las que nos permiten conocer a nuestros clientes, crear valor, mejorar la calidad del servicio, lograr fidelidad, crecer, etc.

b) ¿Por qué se pierden los clientes?

En primer lugar debemos conocer que requiere un cliente de nuestra empresa. El siguiente listado enumera una serie de valores apreciados por los consumidores a la hora de realizar una compra. Pensemos por un momento en nosotros mismos, ocupando el rol de consumidores, para verificar si estamos o no de acuerdo con los mismos.

Estos son los valores más importantes que un pequeño empresario debe privilegiar a la hora de enfrentar las épocas de crisis y superarlas con éxito. De nada sirven las "vivezas" o "picardías" del tipo: "el cliente no sabe nada y no se va a dar cuenta", o la falta de control en ciertas actitudes de empleados y propietario con alguna manifestación de "soberbia" hacia el cliente.

Un cliente se siente insatisfecho cuando no recibe la atención adecuada y además experimenta un comprensible estado de incomodidad. Esto lo lleva a preguntarse: ¿me quejo? ¿O no vuelvo?

Algunos estudios que se realizaron sobre particular señalan:

- Solo un 3 % de las operaciones de un negocio terminan en una queja.
- El 15 % adicional se queja por vías informales, por ejemplo el boca a boca, que puede ser fatal para un negocio.
- Un 30 % adicional no se quejará nunca, pero está seguro de tener motivos suficientes para protestar.

Lo seguro, es que ninguno de los tres grupos volverá a comprar en esa firma y además inducirá a muchos para que compren en otro lado.

Como se puede apreciar el problema central no está solamente en la deuda que tenemos con los bancos, los impuestos

atrasados, dificultades con los proveedores u otras cuestiones que nos preocupan a diario: "el fondo de la cuestión pasa siempre por la atención y satisfacción del cliente".

Si se hace demasiado hincapié en los servicios al cliente, sus costos en general tienden a incrementarse notablemente y pierden competitividad.

Desde la perspectiva del cliente, comprar en estos centros comerciales tiene ventajas y desventajas.

c) Ventajas

- Precios bajos en el servicio
- Amplia variedad de servicio en el mismo espacio físico
- Menores gastos de compra
- Playa de estacionamiento Propia
- Servicio/crédito.

d) Desventajas

- Importante limitación en servicios al cliente
- Despersonalización de quién compra
- El consumidor se enfrenta a una góndola donde se exhiben productos y se encuentra en la alternativa "lo toma o lo deja", sin consulta ni asesoramiento alguno.
- En estos grandes centros de ventas, generalmente en muchos rubros hay falta de marcas líderes.
- El cliente debe hacer colas para pagar, perdiendo un tiempo valioso.

Por ello, algunos de los valores diferenciales más importantes que tienen las micro y pequeñas empresas, frente a las grandes corporaciones, supermercados, hipermercados, etc.;

son los servicios y la atención personalizada.

Hay que utilizar las variables del marketing y trabajar sobre:

- El diseño del producto/servicio.
- La exhibición y sus mil variantes.
- El precio y sus alternativas.
- La comunicación, sus medios y posibilidades. Hacer un uso intensivo de la base de datos como una herramienta habitual.
- La confección creativa y dinámica de vidrieras.
- Utilizar folletería para llegar al cliente con promociones, premios, bonos, etc.
- Muy buena presencia y alta imagen personal.
- Orden, prolijidad y organización en las funciones del personal.
- Mucha acción creativa en los puntos de venta.
- Hay que consultar a los clientes sobre sus deseos y necesidades.
- Participación y organización habitual de eventos.

2.3.2.5 EL SERVICIO

El servicio es el conjunto de prestaciones que el cliente espera, además del producto o servicio básico.

El comprador de una computadora espera cierto número de prestaciones, antes durante y después de la compra propiamente dicha: demostraciones, prueba de la máquina con sus componentes, soluciones financieras, reparaciones rápidas, garantía post-venta, etc.

El servicio es algo que va más allá de la amabilidad y de la gentileza. El servicio es "un valor agregado para el cliente", y en ese campo el cliente es cada vez más exigente.

En la percepción de la calidad del servicio influyen también los símbolos que rodean al producto:

- El pescado parece más fresco cuando se lo presenta sobre una cama de hielo.
- Se suele juzgar la competencia de un médico por su sala de espera.
- El precio. La exigencia de calidad aumenta en relación directa con el precio.
- Cuando el cliente recibe poca información sobre lo que va a comprar, percibe que el producto o servicio es de menor calidad.

Por lo general las pequeñas empresas no están en condiciones de ganar una batalla competitiva basada en los precios. En este juego siempre hay un competidor más poderoso que puede hacerlo mejor. El precio pocas veces constituye una ventaja competitiva. Incluso desde la perspectiva del desarrollo tecnológico, cuando alguien logra algo novedoso, al poco tiempo lo alcanzan los demás.

En las posibilidades de diferenciación, la orientación al servicio, es decir el "saber hacer" es la que se lleva la "parte del león". La comunicación desempeña un papel importantísimo en el éxito de una estrategia de servicio. Es el vehículo indispensable para ampliar la clientela, conseguir lealtad, motivar a los empleados y darles a conocer las normas de calidad que deben poner en práctica.

La comunicación es la única forma de hacer conocer la ventaja de la empresa en relación a sus competidores. Es la que

permite ocupar un lugar en la mente de los consumidores (posicionamiento).

La compañía de aviación alemana Lufthansa para afirmar su "saber hacer" en cuanto a organización, puntualidad y eficacia; mediante la publicidad muestra a dos técnicos delante de un avión con el texto "En Lufthansa el servicio comienza mucho antes de su primer cóctel".

La comunicación debe ajustarse a la magnitud del servicio que se brinda. Una promesa que aumente desmesuradamente las expectativas del cliente, provoca decepción y con frecuencia la pérdida del mismo cuando no se cumple acabadamente.

CARACTERÍSTICAS DEL SERVICIO

- Intangible: no se puede tocar, sentir, escuchar y oler antes de la compra.
- Inseparable: se fabrica y se consume al mismo tiempo.
- Variable: depende de quién, cuándo, cómo y dónde se ofrece.
- Perecedero: No se puede almacenar.

2.3.2.6 METODOLGÍA UML

UML es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

a) DIAGRAMAS DE CASO DE USO

El Caso de Uso, es una técnica para capturar información de cómo un sistema o negocio trabaja, o de cómo se desea que trabaje. No pertenece estrictamente al enfoque orientado a objetos, es una técnica cuyo objetivo es capturar requisitos. **Actores**

Principales: Personas que no usan el sistema.

Secundarios: Personas que mantienen o administran el sistema.

Material Externo: Dispositivos, materiales imprescindibles que forman parte del ámbito de la aplicación y deben ser utilizados. **Otros Sistemas:** Sistemas con los que el sistema interactúa. La misma persona física puede interpretar varios papeles como actores distintos, el nombre del actor describe el papel desempeñado.

b) DIAGRAMA DE CLASES

Es el diagrama principal para el análisis y diseño. Un diagrama de Clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clases incluye definiciones para atributos y operaciones. El modelo de Caso de Uso aporta información para establecer las clases, objetos, atributos y operaciones. El mundo real puede ser visto desde abstracciones diferentes (subjetividad).

Cada clase se representa en un rectángulo con tres comportamientos:

- Nombre de la Clase.
- Atributos de la Clase.
- Operaciones de la Clase.

Los atributos de una clase no deberían ser manipuladas

directamente por el resto de objetos. Por esta razón se crearon niveles de visibilidad para los elementos que son:

- 1) **(-) Privado:** Es el más fuerte, esta parte es totalmente invisible (excepto para clases Friends en terminología C++)
- 2) **(#) Protected:** Los atributos/operaciones protegido están visibles para las clases Friends y para las clases derivadas de la original.
- 3) **(+) Publica:** Los atributos/operaciones públicos son visibles a otras clases (cuando se trata de atributos se está transgrediendo el principio de encapsulación).

Relación entre clases

Son los enlaces entre objetos que se podrán representar entre las respectivas clases y sus formas de relación son: Asociación y Agregación (vista como un caso particular de Asociación).

Generalización / Especialización

Las relaciones de Agregación y Generalización forman jerarquías de clases.

Asociación

La Asociación expresa una conexión bidireccional entre objetos. Una asociación es una abstracción de la relación existente en los enlaces entre los objetos. Puede determinarse por la especificación de multiplicidad (mínima y máxima).

- Uno y sólo uno.
- 0...1 Cero a uno.
- M...N Desde M hasta N (enteros naturales).
- 0...* Cero a Muchos.
- 1...* Uno o Muchos.

Agregación

En UML se proporciona una cascada, caracterización de la

agregación. Esta relación puede ser caracterizada con precisión determinando las relaciones de comportamiento y estructura que existen entre el objeto agregado y cada uno de sus objetos componentes.

c) Diagrama de Secuencia

Es Diagrama de Secuencia representa la forma en cómo un Cliente (Actor) u Objetos (clases) se comunican entre sí en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.

Dicho Diagrama puede ser obtenido de dos partes, desde el Diagrama Estático de Clases o el Caso de Uso (son diferentes). Los Componentes de un Diagrama de Secuencia son:

- Un Objeto o Actor.
- Mensaje de un Objeto a otro Objeto.
- Mensaje de un objeto a sí mismo.

- Elementos

Objeto/Actor

El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

Mensaje a otro Objeto

Se presenta por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular. Mensaje al Mismo Objeto

No solo llamadas a métodos de objetos externos pueden realizarse, también es posible visualizar llamadas a métodos desde el mismo objeto de estudio.

d) Diagrama de Colaboración

Son útiles en la fase exploratoria para identificar objetos.

La distribución de los objetos en el diagrama permite observar adecuadamente la interacción de un objeto con respecto a los demás. La estructura estática viene dada por los enlaces, la dinámica por el envío de mensajes por los mensajes.

¿Qué es una colaboración?

Es una descripción de una colección de objetos que interactúan para implementar un cierto comportamiento dentro de un contexto. Describe una sociedad de objetos cooperantes unidos para realizar un cierto propósito. Una Colaboración contiene ranuras que son rellenadas por los objetos enlace en tiempo de ejecución. Una ranura de Colaboración se llama Rol porque describe el propósito de un objeto o un enlace dentro de la Colaboración

e) Diagrama de Actividades

El Diagrama de Actividades es una especialización del Diagrama de Estado, organizado respecto de las acciones y usado para especificar.

- Un Método
- Un Caso de Uso
- Un Proceso de Negocio

Un Diagrama de Actividades es provechoso para entender el comportamiento de alto nivel de la ejecución de un sistema, sin profundizar en los detalles internos de los mensajes. Los parámetros de entrada y salida de una acción se pueden mostrar usando las relaciones de flujo que conectan la acción y un estado de flujo de objeto.

f) Diagrama de Estado

Muestra el conjunto de estados por los cuales pasa un objeto durante su vida en una aplicación, junto con los cambios que permiten pasar de un estado a otro. Los Diagramas de Estado

representan autómatas de estados finitos, desde el proceso de vida de los estados y las transiciones. Son útiles solo para los objetos con un comportamiento significativo, cada objeto está en un estado en cierto instante el cual está caracterizado parcialmente por los valores de alguno de los atributos del objeto.

Los Diagramas de Estado son autómatas, jerárquicos que permiten expresar concurrencia, sincronización y jerarquías de los objetos. Son gastos dirigidos y deterministas, esta transición es instantánea y se debe a la concurrencia de un objeto.

2.3.2.7 HERRAMIENTAS

- IBM RATIONAL ROSE

El Rational Unified Process o Proceso Unificado de Racional. Es un proceso de ingeniería de software que suministra un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga la necesidad del usuario final dentro de un tiempo y presupuesto previsible. Es una metodología de desarrollo iterativo enfocada hacia “los casos de uso, manejo de riesgos y el manejo de la arquitectura”.

El RUP mejora la productividad del equipo ya que permite que cada miembro del grupo sin importar su responsabilidad específica acceda a la misma base de datos de conocimiento.

a) CICLO DE VIDA

En el ciclo de vida RUP veremos una implementación del desarrollo en espiral. Con el ciclo de vida se establecen tareas

en fases e iteraciones. El RUP maneja el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una base de inicio.

b) FASES

- Fase de Inicio

Durante esta fase de inicio las iteraciones se centran con mayor énfasis en las actividades de modelamiento de la empresa y en sus requerimientos.

- Fase de Elaboración

Durante esta fase de Elaboración, las iteraciones se centran al desarrollo de la base de la diseño, encierran más los flujos de trabajo de requerimientos, modelo de la organización, análisis, diseño y una parte de implementación orientada a la base de la construcción.

- Fase de construcción

Durante esta fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones las cuales se seleccionan algunos Casos de Uso, se redefine su análisis y diseño y se procede a su implantación y pruebas. En esta fase se realiza una pequeña cascada para cada ciclo, se realizan tantas iteraciones hasta que se termine la nueva implementación del producto.

- Fase de Transición

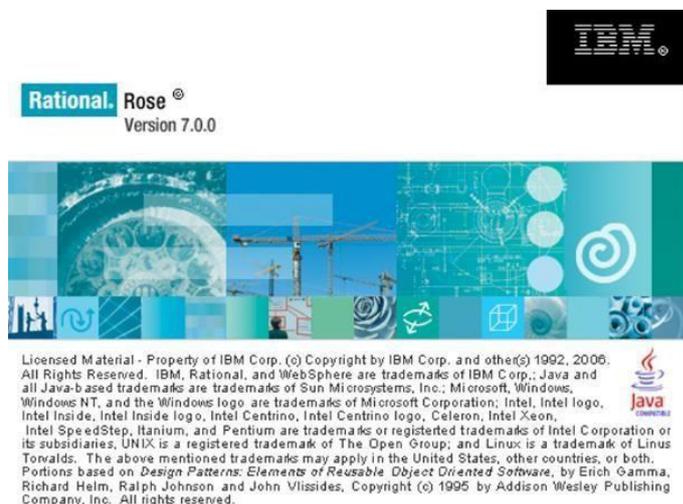
Durante esta fase de transición busca garantizar que se tiene

un producto preparado para su entrega al usuario.

c) Principales Características

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en Ingeniería de Software.
- Desarrollo iterativo.
- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.

El RUP es un producto de Rational (IBM). Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).



- **ADOBE PHOTOSHOP CS5**

Es el software estándar de edición de imágenes profesional y el líder de la gama de productos de edición de imágenes digitales que aporta más de lo que nosotros esperamos. Las innovadoras herramientas creativas nos ayudarán a conseguir resultados excepcionales. Además, gracias a unos procesos de edición, tratamiento y gestión de archivos más eficaces podrá trabajar con mayor rapidez.

- **ERWIN DATA MODELER**

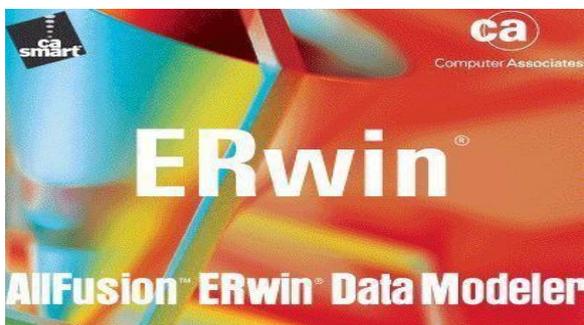
Es una Herramienta de modelado de datos (datos de análisis de requerimientos, diseño de base de datos etc.) personalizados desarrollados, sistemas de información, incluyendo bases de datos de sistemas transaccionales y datamart.

CARACTERÍSTICAS DE ERWIN DATA MODELER

- **Modelado de datos lógico:** Modelos puramente lógicos pueden crearse, de la cual se pueden derivar modelos físicos. También se admiten combinaciones de modelos lógicos y físicos. Apoya el tipo de entidad y lógica nombres de atributos y descripciones, lógicos dominios y tipos de datos, así como relación de nombres.

- **Modelado de datos físico:** Se pueden crear modelos puramente físicos así como combinaciones de modelos lógicos y físicos. Apoya la denominación y descripción de tablas y columnas, tipos de datos definidos por el usuario, claves primarias, llaves foráneas, claves alternativas y el nombramiento y la definición de restricciones. También se incluye soporte para índices, vistas, procedimientos almacenados y desencadenadores.

- **Transformación de lógico a físico:** Incluye un diccionario de abreviatura llamado “Nombres de Editor de normas” y una asignación de tipo de datos lógico-a-RDBMS llamado “Tipo de datos estándares Editor”.



- GESTORES DE BASE DE DATOS

a) MICROSOFT SQL SERVER

Microsoft SQL Server es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional. Sus lenguajes para consultas son T- SQL y ANSI SQL. Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle, PostgreSQL o MySql.

Características de Microsoft SQL Server

- Soporte de transacciones.
- Soporta procedimientos almacenados.
- Incluye también un entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.
- Además permite administrar información de

otros servidores de datos.

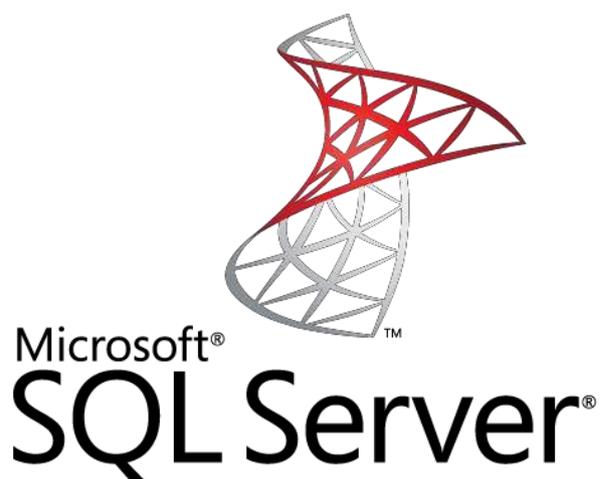
Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños, que en sus versiones 2005 y 2008 pasa a ser el SQL Express Edition, que se distribuye en forma *gratuita*.

Es común desarrollar completos proyectos complementando *Microsoft SQL Server* y *Microsoft Access* a través de los llamados **ADP** (Access Data Project). De esta forma se completa la base de datos (*Microsoft SQL Server*), con el entorno de desarrollo (VBA Access), a través de la implementación de aplicaciones de dos capas mediante el uso de formularios Windows.

En el manejo de SQL mediante líneas de comando se utiliza el SQLCMD, osql, o PowerShell.

Para el desarrollo de aplicaciones más complejas (tres o más capas), *Microsoft SQL Server* incluye interfaces de acceso para varias plataformas de desarrollo, entre ellas .NET, pero el servidor sólo está disponible para sistemas operativos.

El tipo NUMERIC fue mejorado para ser usado como identificador de columna a partir de la versión 2008 R2.



SISTEMA OPERATIVO

En esta ocasión describiré el S.O en la cual se desenvolverá el proyecto a implantar:

WINDOWS 7

Es la versión más reciente de Microsoft Windows, línea de sistemas operativos producida por Microsoft Corporation. Windows 7 incluye varias características nuevas, como mejoras en el reconocimiento de escritura a mano, soporte para discos duros virtuales, rendimiento mejorado en procesadores multinúcleo, mejor rendimiento de arranque, DirectAccess, y mejoras en el núcleo. Windows 7 añade soporte para sistemas que utilizan múltiples tarjetas gráficas de proveedores distintos (heterogéneos multi-adapter o Multi-GPU), una nueva versión de Windows Media Center y un gadget para él, y aplicaciones como Paint, Wordpad y la Calculadora rediseñadas. Se añadieron varios elementos al Panel de control, como un asistente para calibrar el color de la pantalla, un calibrador de texto ClearType, Solución de problemas, Ubicación y otros sensores, Administrador de credenciales, Iconos en el área de notificación, entre otros.

El Centro de seguridad de Windows se renombró a Centro de actividades, y se integraron las categorías de seguridad y el mantenimiento del equipo en él.

La barra de tareas fue rediseñada, haciéndola más ancha, y los botones de las ventanas ya no traen texto, sino únicamente el icono de la aplicación. Estos cambios se hacen para mejorar el desempeño en sistemas de pantalla táctil.



- **VISUAL STUDIO 2010 – VISUAL BASIC**

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros. Ya que posee la plataforma .NET Framework (Net 4.0) y es diseñado para aprovechar las nuevas ventajas que ofrece el nuevo sistema operativo Windows Seven. A través de sus subsistemas "Windows Communication Foundation" (WCF) y "Windows Presentation Foundation" (WPF). El primero tiene como objetivo la construcción de aplicaciones orientadas a servicios mientras que el último apunta a la creación de interfaces de usuario más dinámicas que las conocidas hasta el momento.

2.3.2.8 REDES

Una red, es aquella conexión entre dos o más computadoras, las cuales comparten algunos elementos del hardware, como las impresoras o el CD-ROM, al igual que información, como diversos archivos de la organización. Al igual que ciertos servicios comunes, como salas de chat y mensajería instantánea, correos electrónicos, etc.

Existen diversas tecnologías de red, siendo las más conocidas las de tipo PPP, la HDLC, ETHERNET, TOKEN RING, entre otras.

Con respecto a los estándares de red, tenemos el IEEE 802.3, IEEE 802.5, IEEE 802.11, IEEE 802.15. El primero se utiliza en el ETHERNET, el segundo en el TOKEN RING, el tercero en el sistema Wi-Fi y el último en el BLUETOOTH.

Ahora, una red, puede llegar a abarcar distintas distancias, por lo que cada una tiene una categorización independiente. Tenemos la PAN, que es una red de área personal que abarca hasta 10 metros cuadrados. Por otra parte tenemos la LAN, que es la más habitual de todas. Ya que se utiliza muchísimo en las empresas. Se le llama área de red local y esta abarca hasta un kilómetro cuadrado. Asimismo, existe la MAN o área de red metropolitana, que llega a abarcar hasta diez kilómetros cuadrados. Por último, existe la red WAN o red de área amplia, la cual se utiliza en todo un país, continente, etc.

Por último, podemos categorizar a una red, según las direcciones que tiene, para compartir la información o transmitirla. Esta aquella en la cual, una terminal transmite y la otra recibe. Por otra parte, esta aquella red que permite que una computadora transmita información y las otras reciban y por último, esta aquella red que permite que varias envíen y reciban información de manera simultánea.

El uso más común de una red lo podemos ver en las oficinas, en donde varios usuarios comparten recursos como una impresora desde sus computadoras. En la actualidad es frecuente incluso que las personas configuren redes internas en el hogar para compartir los recursos informáticos con la familia, como por ejemplo el acceso a Internet y el uso de impresoras y fax.

a) Características de una Red

Compartición de archivos: Fue la razón principal para tener una red. Para que se cumpla se requiere de un directorio compartido que pueda ser accesado por muchos usuarios de la red, junto a toda la lógica asociada para que más de una

persona no realice cambios conflictivos a un archivo al mismo tiempo.

Compartición de impresoras

Con esto reducimos el número de impresoras en la organización. Se hace necesario el uso de colas de impresión para que las impresiones se lleven a cabo y de forma automática enviar los trabajos en espera en dicha cola.

Servicios de aplicación

Así como se pueden compartir archivos o carpetas en una red, se pueden compartir aplicaciones, las más comunes son aplicativos de contabilidad. Si se requiere por ejemplo de instalar algún programa en diversas computadoras de la red, en lugar de ir colocando el CD-ROM en cada una, se puede tener una carpeta con el contenido del mismo y ejecutar el instalador desde cada equipo.

Correo Electrónico

Es un recurso bastante valioso y que incluso muchas organizaciones no lo aprovechan al máximo. No solamente es útil para las comunicaciones internas sino también para las externas.

Acceso Remoto

Se usa principalmente para acceder desde el exterior a los recursos de la red interna. Los usuarios la utilizan para ver sus archivos, correo electrónico ya sea que se encuentren de viaje, desde su hogar, etc.

b) TOPOLOGÍAS

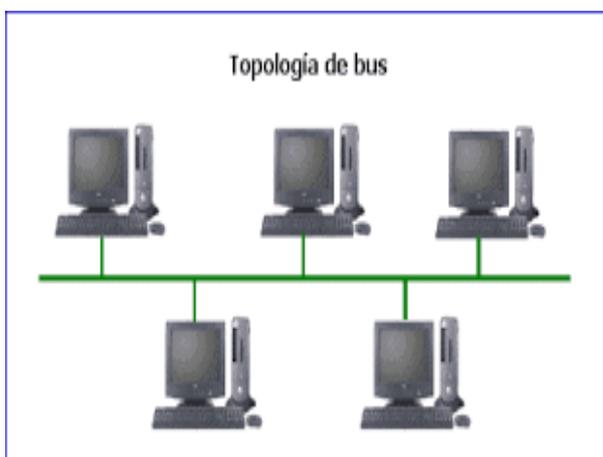
Se denomina topología a la forma geométrica en la que se encuentran distribuidos las estaciones de trabajo y los cables que las conectan.

Las estaciones de trabajo de una red se comunican entre sí

mediante una conexión física y el objeto de la topología es el de buscar la forma más económica y eficaz de conectarlas para, al mismo tiempo, facilitar la fiabilidad del sistema, evitar los tiempos de espera en la transmisión de los datos, permitir un mejor control de la red y permitir de forma eficiente el aumento de las estaciones de trabajo. Las formas más utilizadas son:

- **CONFIGURACIÓN EN BUS**

En ella todas las estaciones comparten el mismo canal de comunicaciones, toda la información circula por ese canal y cada una de ellas recoge la información que le corresponde. Esta configuración es fácil de instalar, la cantidad de cable a utilizar es mínima, tiene una gran flexibilidad a la hora de aumentar o disminuir el número de estaciones y el fallo de una estación no repercute en la red, debido a que los datos pasan de largo por las estaciones de trabajo, en vez de pasar a través de ellos, aunque la ruptura de un cable dejará la red totalmente inutilizada. Las estaciones de trabajo, a su vez actúan de concentradores o hubs. La mayor desventaja de esta configuración es que debido a los pocos puntos de concentración, son difíciles el diagnóstico y el aislamiento de fallos.



- CONFIGURACIÓN EN ANILLO

En ella todas las estaciones están conectadas entre sí formando un anillo, de forma que cada una sólo tiene contacto directo con otras dos.

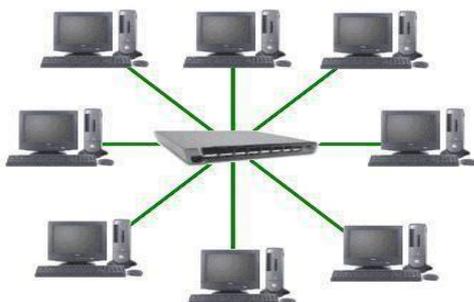
Este tipo de redes permite aumentar o disminuir el número de estaciones sin dificultad, pero, a medida que aumenta el flujo de información, será menor la velocidad de respuesta de la red.

Un fallo en una estación o en un canal de comunicaciones dejará bloqueada la red en su totalidad y, además, será bastante difícil localizar un fallo y repararlo de forma inmediata.



- CONFIGURACIÓN ESTRELLA

Esta forma es una de las más antiguas y, en ella, todas las estaciones están conectadas directamente al servidor o a un ordenador central y todas las comunicaciones se han de hacer necesariamente a través de él. Permite incrementar y disminuir fácilmente el número de estaciones. Si se produce un fallo en una de ellas no repercutirá en el funcionamiento general de la red, pero, si se produce un fallo en el servidor o en el ordenador central, la red completa se vendrá abajo. Tiene un tiempo de respuesta rápido en las comunicaciones de las estaciones con el servidor o con el ordenador central, y lenta en las comunicaciones entre las distintas estaciones de trabajo.



- **TIPOS DE CABLE DE CONEXIÓN**

En el siguiente esquema (aún con riesgo de realizar una excesiva simplificación) se muestran las características comparadas de los cuatro tipos de cables utilizados para transmisión de datos:

	Par trenzado	Coaxial de banda base	Coaxial de banda ancha	Fibra óptica
Ancho de Banda	Baja	Moderada	Alta	Muy alta
Instalación	Sencilla	Fácil	Fácil	Difícil
Longitud	Baja	Moderada	Alta	Muy alta
Costo	Barato	Moderado	Caro	Muy caro
abilidad de la transmisión	Baja	Alta	Alta	Muy alta
Interferencias	Alta	Moderada	Baja	Ninguna
Seguridad	Baja	Baja	Moderada	Alta
Topología	Estrella Anillo	Bus	Bus Estrella	Estrella Anillo

2.3.2.9 ERNET

Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas

heterogéneas que la componen funcionen como una red lógica única, de alcance mundial.

Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos.

- **WWW**

En informática, la World Wide Web es un sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet. Con un navegador web, un usuario visualiza sitios web compuestos de páginas web que pueden contener texto, imágenes, videos u otros contenidos multimedia, y navega a través de ellas usando hiperenlaces.

2.3.2.10 LAS METODOLOGIAS ÁGILES

LOS PRINCIPALES VALORES DE LAS METODOLOGÍAS ÁGILES

- a) **Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas:** La gente es el principal factor de éxito de un proyecto de software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- b) **Desarrollar software que funciona más que conseguir una buena documentación:** La regla a seguir es “no producir documentos a menos que sean necesarios de

forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.

- c) **La Colaboración con el cliente mas que la negociación de un contrato:** se propone que exista un interacción constante entre el cliente y el equipo de desarrollo. Esta Colaboración entre ambos sera la que marque la marcha del proyecto y asegure su éxito.
- d) **Responder los cambios mas que seguir estrictamente un plan:** La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

METODOLOGÍAS ÁGILES VERSUS METODOLOGÍAS TRADICIONALES

Tener metodologías diferentes para aplicar de acuerdo con el proyecto que se desarrolle resulta una idea interesante. Estas metodologías pueden involucrar prácticas tanto de metodologías ágiles como de metodologías tradicionales. De esta manera podríamos tener una metodología para cada proyecto, la problemática sería definir cada una de las prácticas, y en el momento preciso definir parámetros para saber cuál usar.

Es importante tener en cuenta que el uso de un método ágil no es para todos. Sin embargo, una de las principales ventajas de los métodos ágiles es su peso inicialmente ligero y por eso las personas que no estén acostumbradas a seguir procesos encuentran estas metodologías bastante agradables.

METODOLOGÍAS ÁGILES	METODOLOGÍAS TRADICIONALES
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en la normas provenientes de estándares seguidos por entorno de desarrollo.
Especialmente preparadas para cambios durante el proyecto.	Cierta resistencia a los cambios.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas/ normas.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos roles.	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.

¿POR QUE USAR METODOLOGÍAS ÁGILES?

Las metodologías tradicionales presentan los siguientes problemas a la hora de abordar proyectos:

- ✓ Existen unas costosas fases previas de especificación de requisitos, análisis y diseño. La corrección durante el desarrollo de errores introducidos en estas fases será costosa, es decir, se pierde flexibilidad ante los cambios.
- ✓ El proceso de desarrollo está encorsetado por documentos firmados.

- ✓ El desarrollo es más lento. Es difícil para los desarrolladores entender un sistema complejo en su globalidad.

Las metodologías ágiles de desarrollo están especialmente indicadas en proyectos con requisitos poco definidos o cambiantes. Estas metodologías se aplican bien en equipos pequeños que resuelven problemas concretos, lo que no está reñido con su aplicación en el desarrollo de grandes sistemas, ya que una correcta modularización de los mismos es fundamental para su exitosa implantación. Dividir el trabajo en módulos abordables minimiza los fallos y el coste.

METODOLOGÍAS ÁGILES DE DESARROLLO DE SOFTWARE

A) XP – EXTREME PROGRAMMING

Básicamente se podría decir que la programación extrema es una “metodología ligera o ágil” para el desarrollo de software eficiente y altamente efectivo.

Como metodología, la programación extrema, presenta muchos puntos comunes con el desarrollo incremental, comenzando por el hecho de que el software desarrollado con XP se realiza de forma incremental.

- LAS HISTORIAS DEL USUARIO

Es la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Contenidos de la ficha:

- ✓ Fecha.

- ✓ Tipo de actividad (nueva, corrección, mejora).
- ✓ Prueba funcional.
- ✓ Número de historia.
- ✓ Prioridad técnica y del cliente.
- ✓ Referencia a otra historia previa.
- ✓ Riesgo.
- ✓ Estimación técnica.
- ✓ Descripción.
- ✓ Notas.

Lista de seguimiento con la fecha, estado, cosas por terminar y comentarios.

A efectos de planificación, las historias pueden ser de una a tres semanas de tiempo de programación (para no superar el tamaño de una iteración). Las historias de usuario son descompuestas en tareas de programación (task card) y asignadas a los programadores para ser implementadas durante una iteración.

PROCESO XP

El ciclo de desarrollo consiste en los siguientes pasos:

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito

de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.

El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega (Release), Iteraciones, Producción, Mantenimiento y Muerte del Proyecto.

PRACTICAS XP

- ✓ **Planificación:** se utilizan las “user-stories” (“historias del usuario”), para realizar el análisis, estas “historias”, se dividirán en tareas (unidades pequeñas, de 1 a 5 días de trabajo en pareja). Además, se priorizarán las tareas, y cada una de ellas tendrá un desarrollo incremental.
- ✓ **Versiones pequeñas:** La primera versión contendrá el conjunto mínimo de requisitos más útiles/necesarios para el sistema global.
- ✓ **Sistema metafórico:** Cada proyecto debe tener una metáfora asociada que nos ofrezca unos criterios para nombrar lo que vayamos haciendo de forma fácil.
- ✓ **Diseño simple:** Como los requerimientos cambian, o pueden hacerlo, diariamente, hay que utilizar los diseños más simples posibles para cumplir los requerimientos que tenemos en la actualidad.
- ✓ **Testeo continuo:** Antes de que se implemente cualquier característica de un sistema, se debe escribir un test para ella.
- ✓ **Refactoring:** Cuando tenemos que introducir una nueva característica del sistema, si esta tiene mucho en común con otra previa, lo mejor es eliminar el código duplicado, sin miedo a que falle, debido a que el test probará el correcto funcionamiento.

- ✓ **Pair programming (“programación en parejas”)**: Se trabaja en parejas, cada una utilizando un único ordenador. Así, el código se revisa mientras se desarrolla.
- ✓ **Propiedad colectiva del código**: Cualquiera puede modificar cualquier módulo en cualquier momento, nadie tiene la propiedad de ningún módulo.
- ✓ **Integración continua**: Todos los cambios se introducen en el sistema, al menos, una vez al día.
- ✓ **Semanas de 40 horas de trabajo**: Los programadores se deben ir a casa a su hora.
- ✓ **Cliente en su sitio**: Siempre hay un usuario del sistema que es accesible por los miembros del equipo de trabajo.

Estándares de codificación: Todos deben usar los mismos criterios a la hora de programar. De esta forma, no sería posible determinar quién ha realizado una determinada parte de la implementación.

B) SCRUM

Scrum es una metodología de desarrollo muy simple, que requiere trabajo duro porque no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto.

Se comienza con la visión general del producto, especificando y dando detalle a las funcionalidades o partes que tienen mayor prioridad de desarrollo y que pueden llevarse a cabo en un periodo de tiempo breve (normalmente de 30 días). Cada uno de estos periodos de desarrollo es una iteración que finaliza con la producción de un incremento operativo del producto.

Estas iteraciones son la base del desarrollo ágil, y Scrum gestiona su evolución a través de reuniones breves diarias en

las que todo el equipo revisa el trabajo realizado el día anterior y el previsto para el día siguiente.

CONTROL DEL EVOLUCIÓN DEL PROYECTO

Scrum controla de forma empírica y adaptable la evolución del proyecto, empleando las siguientes prácticas de la gestión ágil:

✓ **Revisión de las Iteraciones**

Al finalizar cada iteración (normalmente 30 días) se lleva a cabo una revisión con todas las personas implicadas en el proyecto. Este es el periodo máximo que se tarda en reconducir una desviación en el proyecto o en las circunstancias del producto.

✓ **Desarrollo incremental**

Durante el proyecto, las personas implicadas no trabajan con diseños o abstracciones. El desarrollo incremental implica que al final de cada iteración se dispone de una parte del producto operativa que se puede inspeccionar y evaluar.

✓ **Desarrollo evolutivo**

Los modelos de gestión ágil se emplean para trabajar en entornos de incertidumbre e inestabilidad de requisitos. Intentar predecir en las fases iniciales cómo será el producto final, y sobre dicha predicción desarrollar el diseño y la arquitectura del producto no es realista, porque las circunstancias obligarán a remodelarlo muchas veces.

Para qué predecir los estados finales de la arquitectura o del diseño si van a estar cambiando. En Scrum se toma a la inestabilidad como una premisa, y se adoptan técnicas de trabajo para permitir esa evolución sin degradar la calidad de la

arquitectura que se irá generando durante el desarrollo.

El desarrollo Scrum va generando el diseño y la arquitectura final de forma evolutiva durante todo el proyecto. No los considera como productos que deban realizarse en la primera “fase” del proyecto.

✓ **Auto-organización**

En Scrum los equipos son auto-organizados (no auto-dirigidos), con margen de decisión suficiente para tomar las decisiones que consideren oportunas.

✓ **Colaboración**

Las prácticas y el entorno de trabajo ágiles facilitan la colaboración del equipo. Ésta es necesaria, porque para que funcione la auto-organización como un control eficaz cada miembro del equipo debe colaborar de forma abierta con los demás, según sus capacidades y no según su rol o su puesto.

VISIÓN GENERAL DEL PROCESO

Scrum denomina “sprint” a cada iteración de desarrollo y recomienda realizarlas con duraciones de 30 días.

El sprint es por tanto el núcleo central que proporciona la base de desarrollo iterativo e incremental.

Los elementos que conforman el desarrollo Scrum son:

✓ **Las reuniones**

- Planificación de sprint: Jornada de trabajo previa al inicio de cada sprint en la que se determina cuál va a ser el trabajo y los objetivos que se deben cumplir en esa iteración.
- Reunión diaria: Breve revisión del equipo del trabajo realizado hasta la fecha y la previsión para el día siguiente.

- Revisión de sprint: Análisis y revisión del incremento generado.
- ✓ **Los elementos**
 - Pila del producto: lista de requisitos de usuario que se origina con la visión inicial del producto y va creciendo y evolucionando durante el desarrollo.
 - Pila del sprint: Lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.
 - Incremento: Resultado de cada Sprint.

✓ **Los roles**

Scrum clasifica a todas las personas que intervienen o tienen interés en el desarrollo del proyecto en: propietario del producto, equipo, gestor de Scrum (también Scrum Manager o Scrum Master) y “otros interesados”. Los tres primeros grupos (propietario, equipo y gestor) son los responsables del proyecto, los que según la comparación siguiente (y sin connotaciones peyorativas) serían los “cerdos”; mientras que el resto de interesados serían las gallinas.

Cerdos y gallinas.

Esta metáfora ilustra de forma muy gráfica la diferencia de implicación en el proyecto entre ambos grupos:

Una gallina y un cerdo paseaban por la carretera.

La gallina dijo al cerdo: “Quieres abrir un restaurante conmigo”.

El cerdo consideró la propuesta y respondió: “Sí, me gustaría. ¿Y cómo lo llamaríamos?”.

La gallina respondió: “Huevos con beicon”.

El cerdo se detuvo, hizo una pausa y contestó:

“Pensándolo mejor, creo que no voy a abrir un restaurante contigo. Yo estaría realmente comprometido, mientras que tu estarías sólo implicada”.

COMPROMETIDOS (cerdos)	IMPLICADOS (gallinas)
Propietario del producto	Otros interesados (Dirección General,
Equipo	Dirección Comercial,
Scrum Master	Marketing, Usuarios, etc.)

- Propietario del producto: El responsable de obtener el mayor valor de producto para los clientes, usuarios y resto de implicados.
- Equipo de desarrollo: grupo o grupos de trabajo que desarrollan el producto.
- Scrum Manager: gestor de los equipos que es responsable del funcionamiento de la metodología Scrum y de la productividad del equipo de desarrollo.

✓ **Valores**

Scrum es una “carrocería” para dar forma a los principios ágiles. Es una ayuda para organizar a las personas y el flujo de trabajo.

La carrocería sin motor, sin los valores que dan sentido al desarrollo ágil, no funciona.

- Delegación de atribuciones (empowerment) al equipo para que pueda auto-organizarse y tomar las decisiones sobre el desarrollo.

- Respeto entre las personas. Los miembros del equipo deben confiar entre ellos y respetar sus conocimientos y capacidades.
- Responsabilidad y auto-disciplina (no disciplina impuesta).
- Trabajo centrado en el desarrollo de lo comprometido.
- Información, transparencia y visibilidad del desarrollo del proyecto.

C) RAD

La metodología de desarrollo conocida como diseño rápido de aplicaciones RAD (por sus siglas en inglés) ha tomado gran auge debido a la necesidad que tienen las instituciones de crear aplicaciones funcionales en un plazo de tiempo corto. Esta modalidad de desarrollo consiste de diferentes etapas que suceden de forma paralela y exigen la colaboración de los usuarios en todos los niveles. Por el contrario, en la metodología de diseño tradicional, las etapas suceden de forma lineal y el usuario es excluido del proceso, lo que hace que esta modalidad sea más lenta y poco eficiente.

LA METODOLOGÍA DE DISEÑO RAPIDO DE APLICACIONES

Esta metodología propone un proceso de desarrollo de "**software**" que permite que se creen sistemas de computadoras utilizables en un periodo de tiempo entre 60 a 90 días. RAD es un ciclo de desarrollo diseñado para crear aplicaciones de computadoras de alta calidad de las que acontecen en corporaciones grandes. El desarrollo de aplicaciones enfrenta una transformación fundamental. Hace cinco años un proyecto para desarrollar una

aplicación tomaba un periodo de entre 18 a 24 meses; actualmente, con la práctica del modelo RAD toma entre 1 a 3 meses.

LAS CUATRO ETAPAS DEL CICLO RAD

✓ **Etapa de planificación de los requisitos**

Esta etapa requiere que usuarios con un vasto conocimiento de los procesos de la compañía determinen cuáles serán las funciones del sistema. Debe darse una discusión estructurada sobre los problemas de la compañía que necesitan solución.

✓ **Etapa de diseño**

Esta consiste de un análisis detallado de las actividades de la compañía en relación al sistema propuesto. En ellos descomponen funciones y definen entidades asociadas con el sistema. Una vez se completa el análisis se crean los diagramas que definen las alteraciones entre los procesos y la data. Al finalizar el análisis se traza el diseño del sistema. Se desarrollan los procedimientos y los esquemas de pantallas.

✓ **Construcción**

En la etapa de construcción el equipo de desarrolladores trabajando de cerca con los usuarios finaliza el diseño y la construcción del sistema. La construcción de la aplicación consiste de una serie de pasos donde los usuarios tienen la oportunidad de afirmar los requisitos y repasar los resultados. Las pruebas al sistema se llevan a cabo durante esta etapa. También se crea la documentación y las instrucciones necesarias para manejar la nueva aplicación, rutinas y procedimientos para operar el sistema.

✓ **Implementación**

Esta etapa envuelve la implementación del nuevo producto y el manejo del cambio del viejo al nuevo sistema. Se hacen pruebas comprensivas y se adiestran los usuarios. Los cambios organizacionales y la operación del nuevo sistema se hacen en paralelo con el viejo sistema hasta que el nuevo se establezca completamente.

CAPITULO III ASPECTOS DE LA INVESTIGACIÓN

3.1 ALCANCE DEL SISTEMA

El objetivo principal del sistema computarizado para la administración del rubro hotelero, es ayudar con el control de flujo de información que maneja el área del hotel, para así hacer Ágil las transacciones y la emisión de reportes que se hace a otras instituciones.

✓ **REGISTROS**

- Registro de Pisos.
- Registro de Tipo de Habitación.
- Registro Marca.
- Registro Color.
- Registro vehículo.
- Registro proveedores.
- Registro clientes (Persona, Empresa).
- Registro Productos.
- Registro Habitación.
- Registro de Prendas.
- Registro tallas.
- Registro tipo de vehículo.
- Registro tipo de Parqueo.
- Registro ubicación de vehículo.
- Registro tipo de banco.
- Registro tipo Pago.
- Registro área.
- Registro empleado.

✓ **PROCESOS**

- Cambio Igv actual.
- Reservación
- Cambiar estado de habitaciones.

- Servicio de cochera.

- ✓ **REPORTES**
 - Deudas de clientes.
 - Hospedajes Eliminados.
 - Deudas de Servicio de Cochera.
 - Servicio de cochera eliminados.
 - Hospedajes realizados por usuarios
 - Listado actual de las habitaciones
 - Listado actual de los tipos de parqueo

- ✓ **SEGURIDAD**
 - Registro de Usuario.
 - Registro de Perfil.
 - Registro de Menus.
 - Registro de operaciones.
 - Registro de Perfil a usuario.
 - Permiso de operaciones a Perfil.

- ✓ **CONTROL DE CAJA**
 - Apertura de Caja.
 - Cierre de Caja.
 - Concepto I/S (ingresos y salidas).
 - Egresos de caja.
 - Ingresos de caja.
 - Listado detallado de caja
 - Listado de cierre caja realizados
 - Resumen del día.

- ✓ **SERVICIO GENERADO**
 - Anulación de servicios
 - Listado de productos anulados
 - Listado de comprobante generados

3.2 CRONOGRAMA DE ACTIVIDADES DEL PROYECTO.

El cronograma del proyecto se presenta como anexo del presente documento.

3.3 FACTORES CRÍTICOS DE ÉXITO

- Investigar sobre los sistemas implantados existentes dentro de la empresa.
- Definir las iteraciones y las funcionalidades que formarán parte del alcance del sistema a implementar.
- Definir los estándares de desarrollo y documentación necesaria para el inicio de la ejecución del proyecto.
- Interés de parte de la alta dirección de la empresa y de parte del equipo de desarrollo.

3.4 SUPUESTOS

- El equipo de desarrollo deberá de capacitar a los usuarios finales del sistema.
- La empresa deberá adquirir el equipo el hardware necesario para el sistema.
- La empresa proporcionará la información sobre los procesos que se dan dentro del negocio poniendo más énfasis en los procesos del hotel.

3.5 RESTRICCIONES

- El sistema se enfocará en los procesos del Área del hotel.
- Los usuarios que se autentiquen en el sistema solo tendrán acceso a ciertas operaciones, según los roles y permisos asignados.
- Las contraseñas de los usuarios serán almacenadas de forma encriptada dentro de la base de datos.

3.6 ANÁLISIS DE LA FACTIBILIDAD

3.6.1 FACTIBILIDAD TÉCNICA

- El sistema contará con interfaces amigables, la cual permitirá al usuario interactuar con facilidad y de forma natural.
- Actualmente la empresa cuenta con la arquitectura tecnológica necesaria para la implementación del proyecto.
- El equipo de desarrollo se encuentra capacitados en cada una de las herramientas utilizadas para el desarrollo del sistema.

3.6.2 FACTIBILIDAD OPERATIVA

- Se cuenta con el apoyo del personal correspondiente de la empresa Princess, para el desarrollo del proyecto proporcionando la información necesaria.
- Se capacitará a los usuarios finales.
- Los usuarios finales del sistema cuentan con conocimientos básicos de informática.
- El sistema agilizará los procesos involucrados con el envío de documentos hacia otras entidades.

3.6.3 FACTIBILIDAD ECONÓMICA

El estudio de factibilidad económica involucra las etapas de desarrollo e implantación del proyecto. Para estas etapas será necesario contar con recursos humanos, hardware, software, materiales y servicios.

Estos recursos significan un costo para el proyecto el cual se ha dividido en:

- ✓ Costo de inversión
- ✓ Costo de desarrollo
- ✓ Costo de operación

3.6.3.1 COSTO DE INVERSIÓN

Los costos de inversión son aquellos gastos que se realizan al inicio del proyecto, se refieren a la adquisición del equipo de cómputo y programas tecnológicos necesarios para el desarrollo del sistema. Estos se han clasificado en costo de hardware y costo de software.

- COSTOS DE HARDWARE

A continuación se muestra la siguiente tabla:

Hardware	Cantidad	Precio unitario	Subtotal(S/.)
PC SERVIDOR	2	1700.00	3 400.00
TOTAL			3 400.00

Fuente: Elaboración Propia

- COSTO DE SOFTWARE

SOFTWARE DE DESARROLLO	Subtotal(S/.)
Visual Studio 2010(Licencia 1 año 3 mese)	1 876.47
SQL Server 2008 r2(Licencia 1 año 3 meses)	3 880.00
Rational Rose	0.00
CA Erwin Data Modeler Community Edition	0.00
Adobe Photoshop CS5	0.00
TOTAL	5 756.47

Fuente: Elaboración Propia

El resumen de todos los costos de inversion se detallan en la siguiente tabla.

COSTO DE INVERSIÓN	Subtotal(S/.)
Costo Hardware	3 400.00
Costo Software	5 756.47
TOTAL	9 156.47

Fuente: Elaboración Propia

3.6.3.2 COSTO DE DESARROLLO

- Estos costos son los que se generan durante la implementación del proyecto. Estos costos se han

clasificado en costos de recursos humanos, recursos materiales, servicios.

- **RECURSOS HUMANOS**

Para el desarrollo del proyecto se contará con analistas, programador. Estos integrantes participarán voluntariamente en el proyecto lo cual hace que los costos por recursos humano sea de cero como se vera a continuación.

CANTIDAD	FUNCIÓN	SUELDO MENSAL	TIEMPO	SUBTOTAL
1	Analista	0.00	*	0.00
1	Diseñador	0.00	*	0.00
Total				0.00

Fuente: Elaboración Propia

- **RECURSOS MATERIALES**

Estos son los materiales de escritorio utilizados para la documentación del proyecto.

ITEM	CANTIDAD	SUB TOTAL(S/.)
CD	5 unidades	5.00
DVD	5 unidades	6.00
PAPEL BOND A4	03 millares	76.50
LAPICEROS	05 unidades	5.00
FOLDERS	24 unidades	16.80
SOBRE MANILA A4	10 unidades	5.00
Total		114.30

Fuente: Elaboración Propia

- **SERVICIOS**

Dentro de los costos generados por servicios se han considerado los costos por energía eléctrica, los costos por el servicio de internet y otros costos adicionales.

Energía Eléctrica

El consumo de energía eléctrica será calculado para un periodo de tiempo de un año dos meses. A continuación se describe a detalle el consumo de energía que se generó en el

el desarrollo del proyecto.

DESCRIPCIÓN	CONSUMO(Watts. por hora)	CONSUMO (Kw. por hora)
Laptop	65	1.00
Pc escritorio	330	1.50
Servicio Luz	75	0.50
TOTAL	470	0.47

Fuente: Elaboración Propia

Cálculo del consumo de energía mensual: $CEM = EH \times NHT \times NDT \times CEH$

$$CEM = 0.47 \times 6 \times 28 \times 0.3710 \text{ CEM} = S/. 21.80$$

Donde:

CEM: consumo de energía mensual.

EH: consumo de energía por hora en Kw/h. NHT: Número de horas trabajados por día. NDT: Número de días trabajados por mes.

CEH: Costo de Energía por hora. Tiene un valor de S/. 0.3710 Por Kwh, según la empresa ElectroDunas S.A.

Como e proyecto se ha planificado para un periodo de 14 meses, entonces el costo total de energía al año seria el siguiente:

$$CEA = CEM \times NMT \text{ CEM} = 21.80 \times 14 \text{ CEM} = S/. 305.20$$

Donde:

CEA: Costo de energía anual.

CEM: Consumo de energía mensual. NMT: Número de meses trabajados al año

Internet

Se hará un pago mensual de S/. 69.00 por el servicio de

internet, por lo que el costo es de 1 035.00 soles.

Otros Servicios

Como costos extras se han considerado otros gastos adicionales como son los gastos que se detallan en la siguiente tabla:

ITEM	SUBTOTAL(S/.)
Movilidad	500.00
Telecomunicaciones	300.00
Fotocopias	90.00
Impresiones	200.00
Encuadernación	20.00
TOTAL	1 110.00

Fuente: Elaboración Propia

En resumen los gastos generados por los servicios.

DESCRIPCIÓN	SUBTOTAL(S/.)
Energía eléctrica	305.20
Internet	1 035.00
Otros servicios	1 110.00
TOTAL	2 450.20

Fuente: Elaboración Propia

Resumen de los costos de desarrollo se detallan a continuación:

DESCRIPCIÓN	SUBTOTAL(S/.)
Recursos humanos	00.00
Recursos Materiales	114.30
Servicios	2 450.20
TOTAL	2 564.50

Fuente: Elaboración Propia

3.6.3.3 COSTO DE OPERACIÓN

Los costos de operación involucran los costos de mantenimiento de hardware, mantenimiento de software, recursos humanos.

- **MANTENIMIENTO DE HARDWARE**

Está definido por los gastos generados por la operación mantenimiento y soporte de los equipos de desarrollo. Para esto se ha estimado un monto anual de S/. 500.00.

- **MANTENIMIENTO DE SOFTWARE**

Está definido por los gastos generados para el mantenimiento y actualización del software a utilizar durante el desarrollo y puesta en marcha del sistema.

Para el mantenimiento y actualización del software se tendrá un costo de S/. 00.00, ya que el equipo de desarrollo se encargará de dicha función.

- **RECURSOS HUMANOS**

Está definido por el personal encargado de dar soporte al sistema, además del software y hardware utilizado para el desarrollo e implantación del proyecto. Los costos generados por recursos humanos será de S/ 00.00 nuevos soles ya que los responsables del desarrollo del proyecto serán los encargados de cubrir el mantenimiento y soporte del sistema.

Los costos de operación quedan resumidos en la siguiente tabla:

DESCRIPCIÓN	SUBTOTAL(S/.)
Mantenimiento de Hardware	1 500.00
Mantenimiento de Software	00.00
Recursos Humanos	00.00
TOTAL	1 500.00

Fuente: Elaboración Propia

El resumen de todos los costos se muestran en la siguiente tabla

DESCRIPCIÓN	SUBTOTAL(S/.)
Costo de Inversión	9 156.47
Costo de desarrollo	2 564.50
Costo de Operación	1 500.00
TOTAL	13 220.97

Fuente : Elaboración Propia

3.6.4 ANÁLISIS COSTO / BENEFICIO

Para realizar el análisis de los costos y beneficios tenemos como datos lo siguiente:

INVERSIÓN

El monto que desembolsa la empresa al inicio del proyecto es de S/ 9 156.47

COSTOS

Dentro de los costos que genera el proyecto anualmente podemos mencionar el costo de mantenimiento anual que seria de S/ 1 500.00 y las variaciones por el uso de energía eléctrica, de parte del sistema.

BENEFICIOS

Dentro de los beneficios que tenemos dentro de la empresa podemos mencionar que el sistema puede reducir el número de un empleado, si bien a un empleado le pago el mínimo vital que es de S/ 750.00 mensuales, la empresa esta ahorrando S/ 9 000.00 anuales, también podemos mencionar como beneficios los gastos que se ahorra la empresa al momento de enviar los reportes a MINCETUR, también podemos mencionar reducción de tiempo para la realización de transacciones y búsqueda de información.

Con los datos mencionados tenemos que :

INVERSIÓN: S/ 9 156.47

FLUJO	AÑO	COSTOS	BENEFICIOS	VALOR NETO
F1	1	5 565.00	9 350.00	3 785.00
F2	2	5 700.00	9 450.00	3 750.00
F3	3	5 650.00	9 500.00	3 850.00
F4	4	5 650.00	9 550.00	3 900.00
F5	5	5 780.00	9 450.00	3 670.00

El interes aplicado es de 10% (0.10).

a) **CÁLCULO PARA OBTENER EL VAN (VALOR ACTUAL NETO):**

FÓRMULA:

$$VAN = \frac{f_1}{(1+i)^{n1}} + \frac{f_2}{(1+i)^{n2}} + \frac{f_3}{(1+i)^{n3}} + \frac{f_4}{(1+i)^{n4}} + \frac{f_5}{(1+i)^{n5}} - I_0$$

Donde:

F1 = flujo de efectivo 1 F2 = flujo de efectivo 2 F3 = flujo de efectivo 3 F4 = flujo de efectivo 4 F5 = flujo de efectivo 5 n = tasa de Interes IO = Inversión

COMO RESULTADO TENEMOS: VAN: S/ 908.53

Por ende obtenemos una **TIR** (Tasa Interna de Retorno) del 30% por ciento

PRI (PERÍODO DE RECUPERACIÓN DE LA INVERSIÓN) FORMULA:

(INVERSIÓN TOTAL / TOTAL BENEFICIO) * 12 Meses

$(S/ 9156.47 - S/ 18 955.00) * 12 = 5.796$

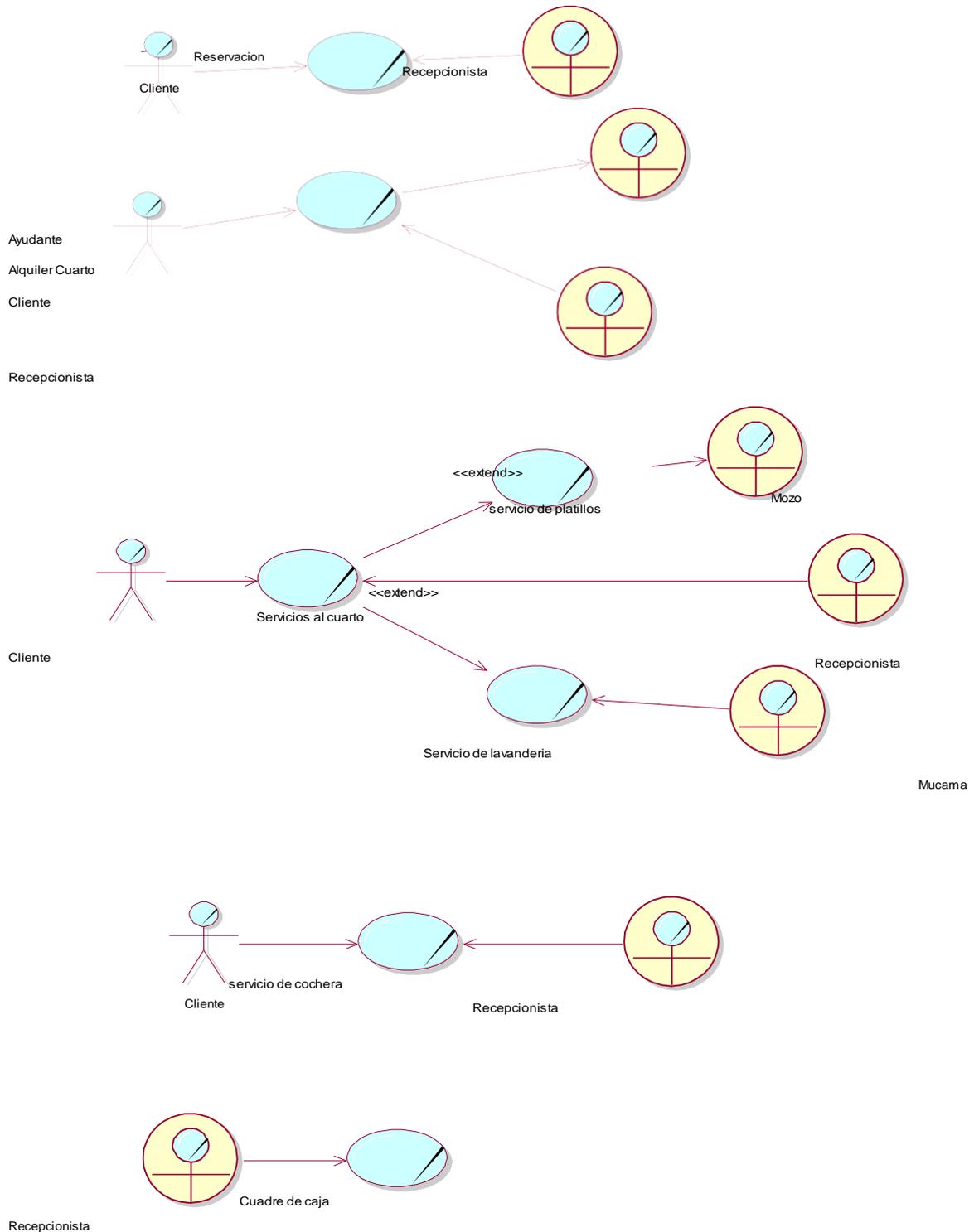
Se deduce que para recuperar la inversion se tomaria en 5 años y 6 mese.

CAPITULO IV ANÁLISIS Y DESARROLLO DEL SISTEMA

4.1 ANÁLISIS DE LA LÓGICA DEL NEGOCIO

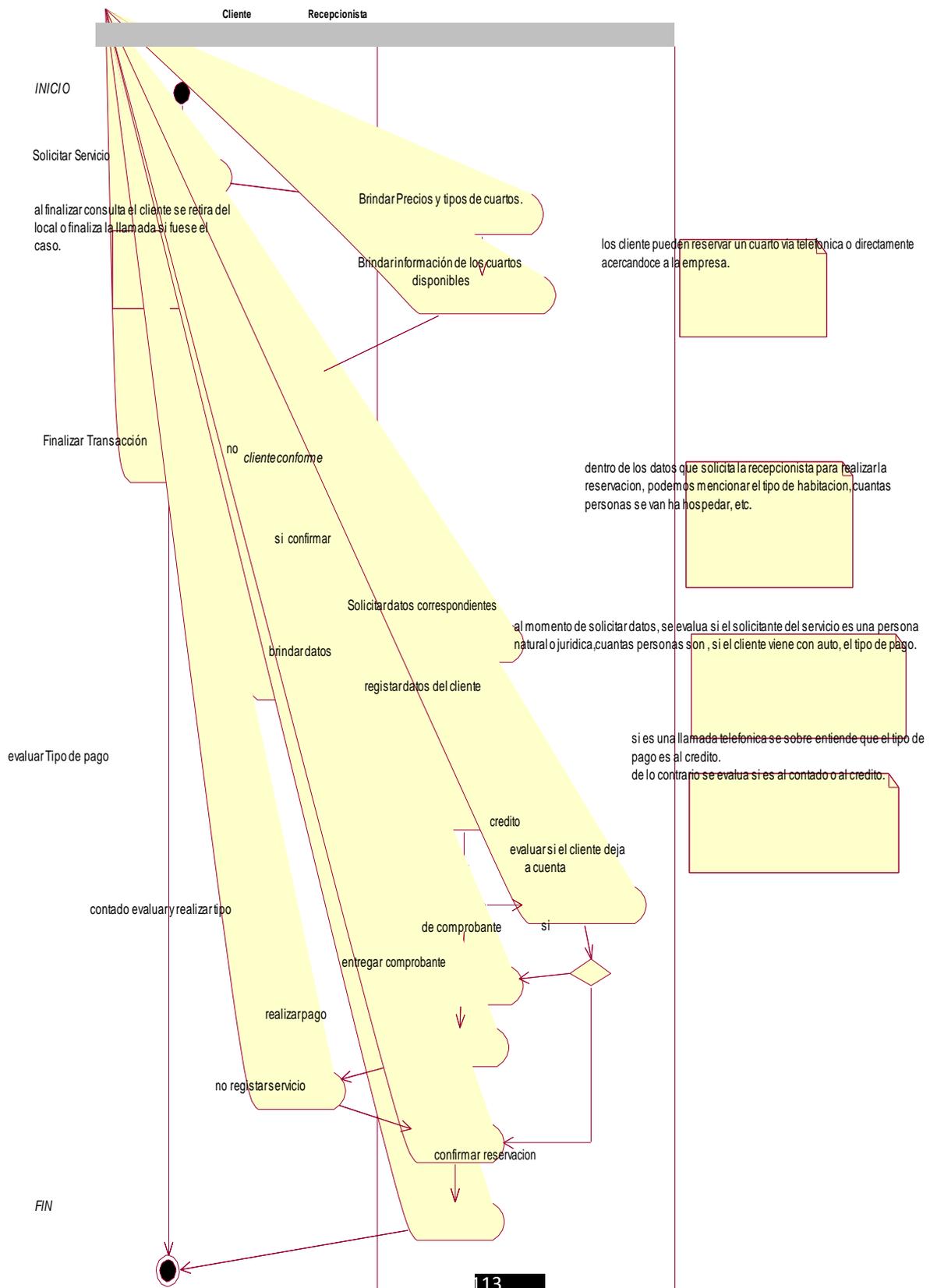
4.1.1 DIAGRAMA DE CASOS DE USO DEL NEGOCIO

A continuación se presenta los procesos encontrados, mediante diagramas:

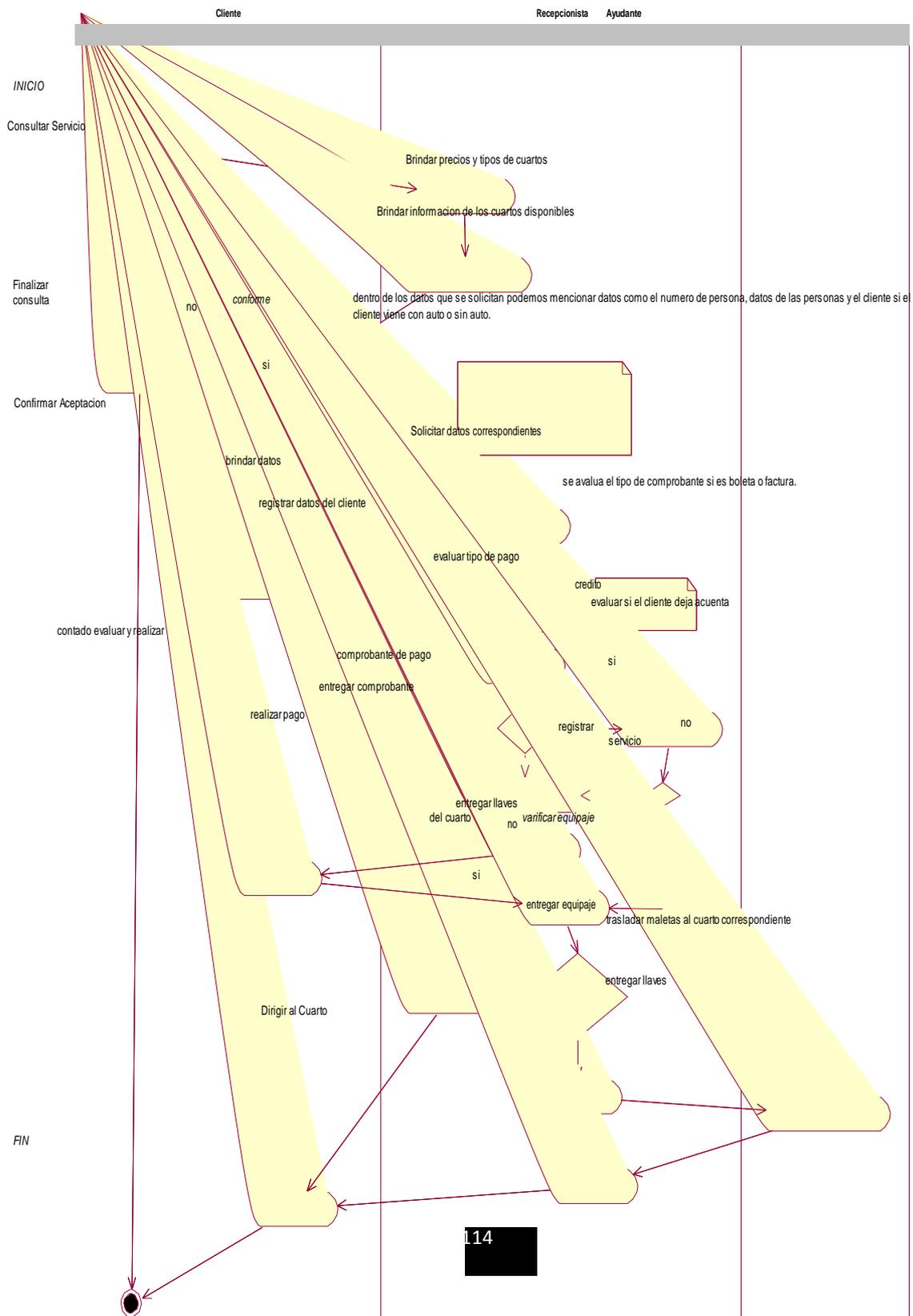


4.1.1.1 RESERVACIÓN

DIAGRAMA DE ACTIVIDADES DEL DCUN



4.1.1.2 DIAGRAMA DE ACTIVIDADES DEL DCUN ALQUILER DE UNA HABITACIÓN



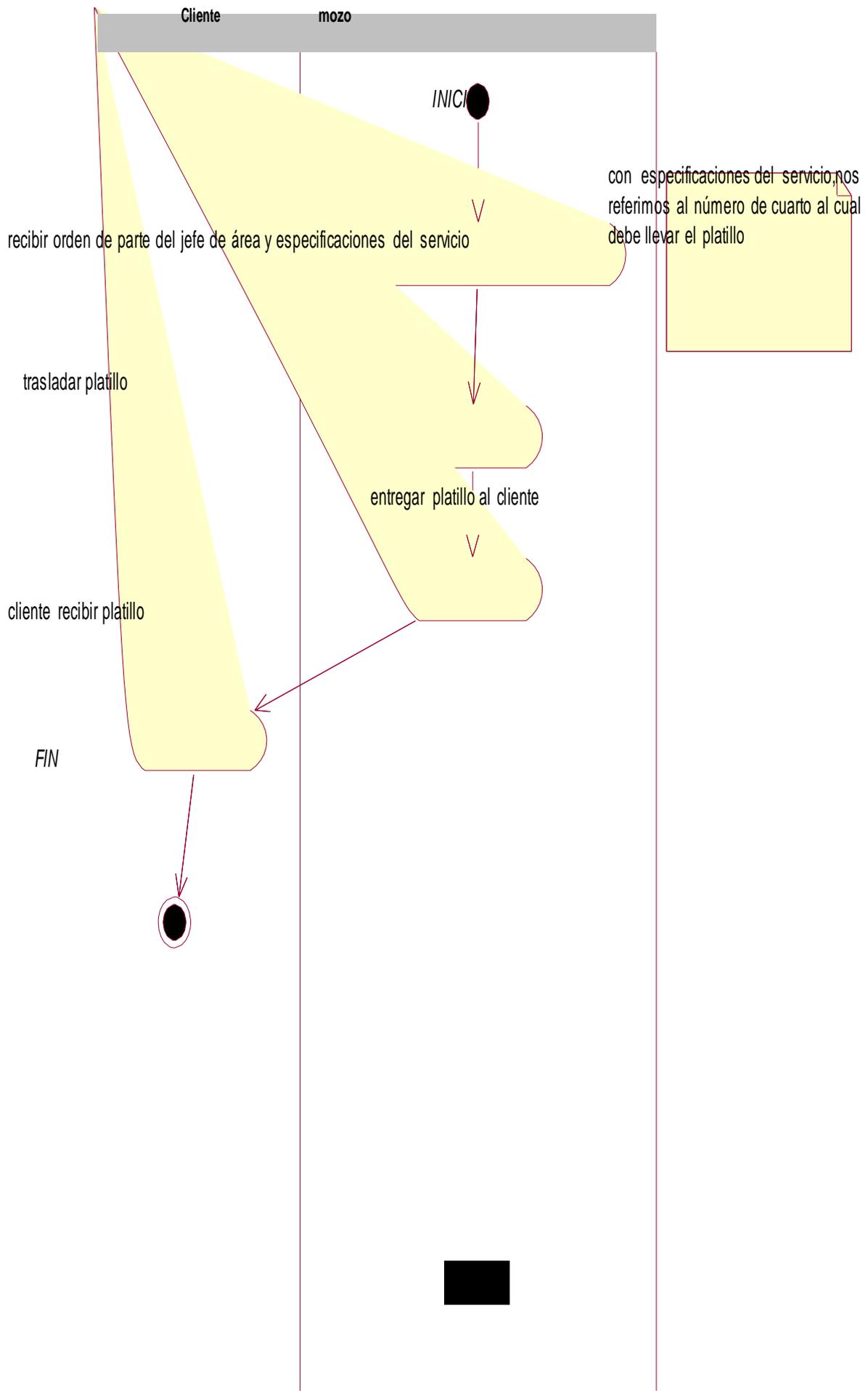
4.1.1.3
AL CUARTO

DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO



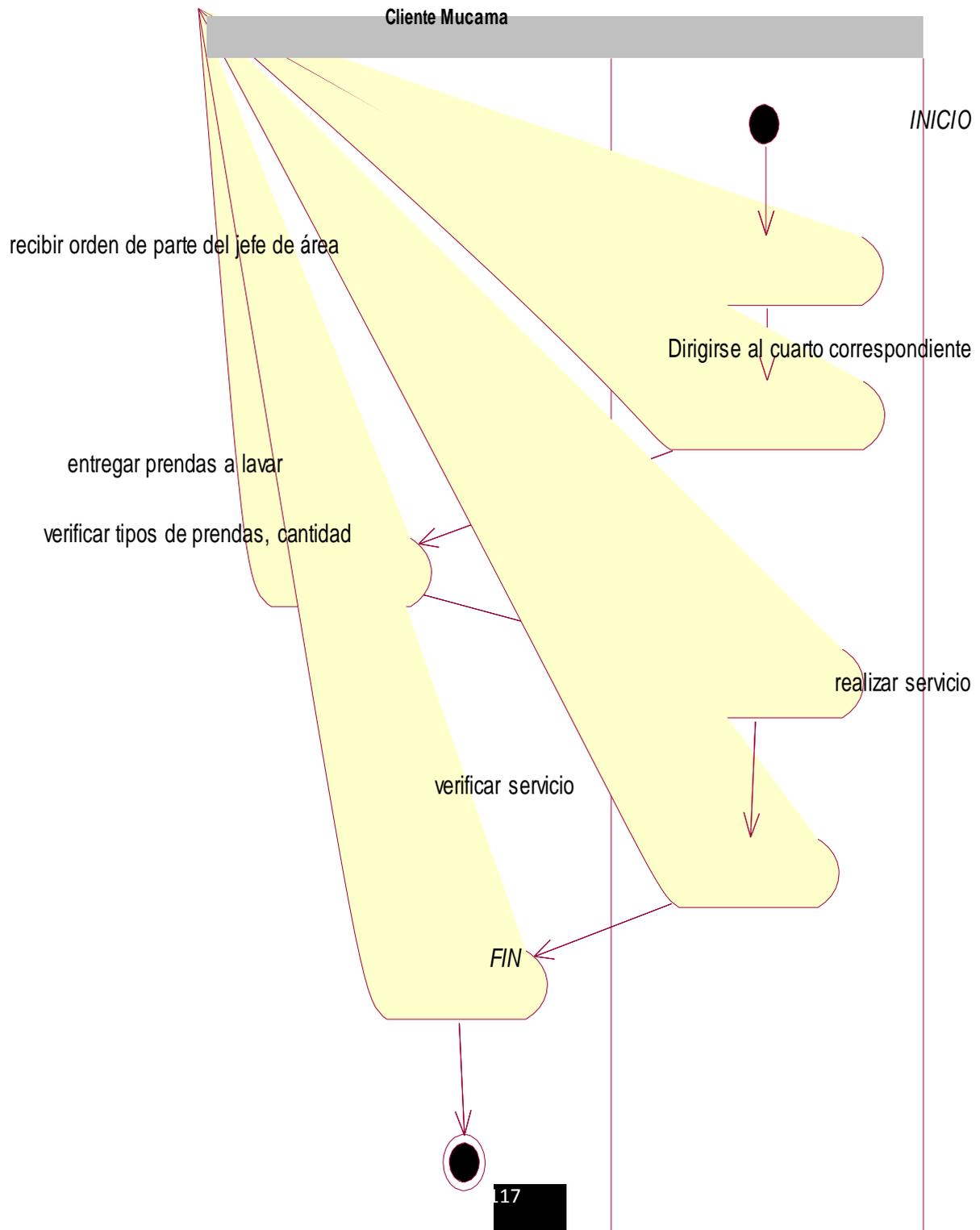
4.1.1.4
DE PLATILLOS

DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO



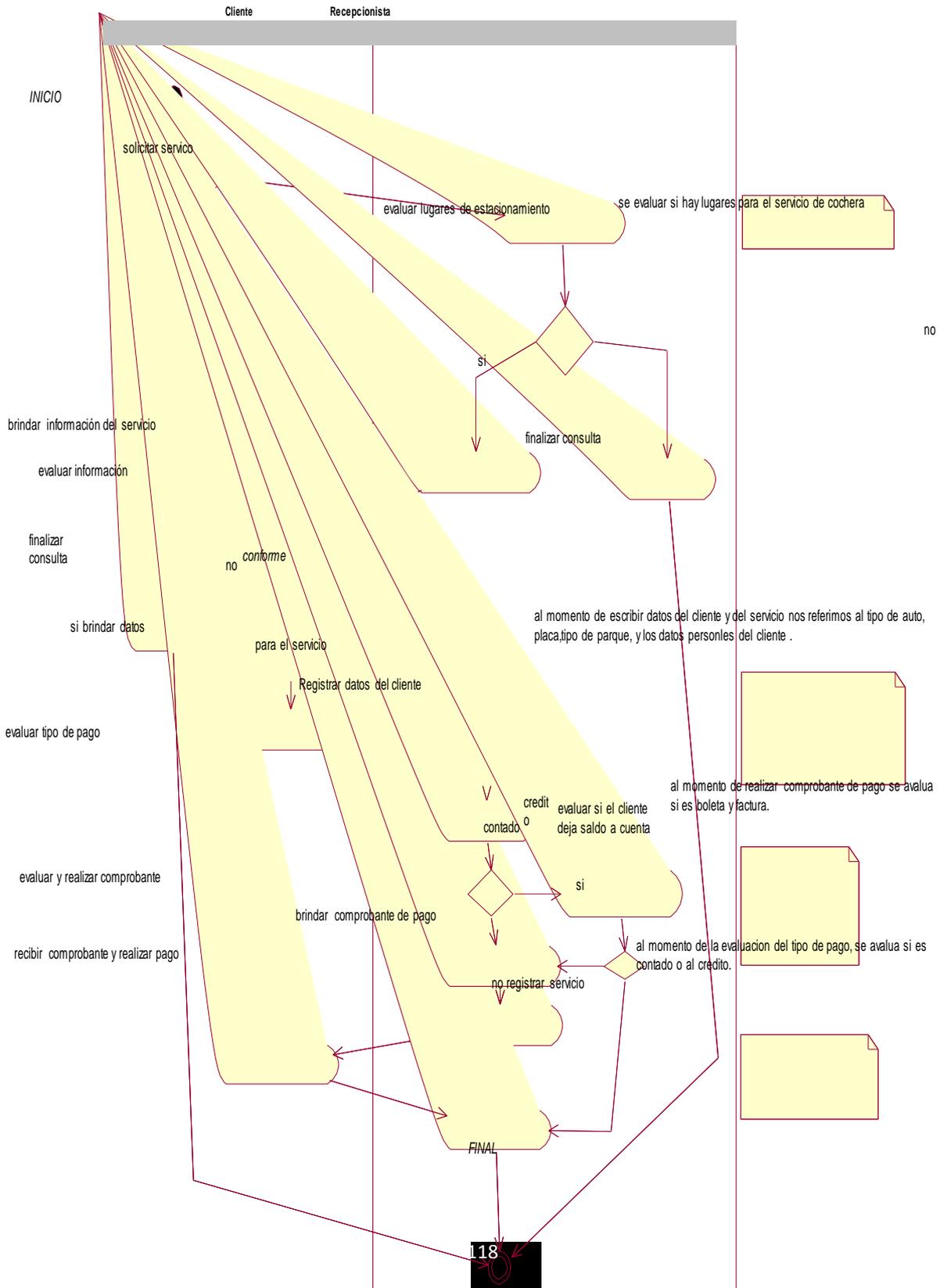
4.1.1.5
DE LAVANDERIA

DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO



4.1.1.6
DE COCHERA

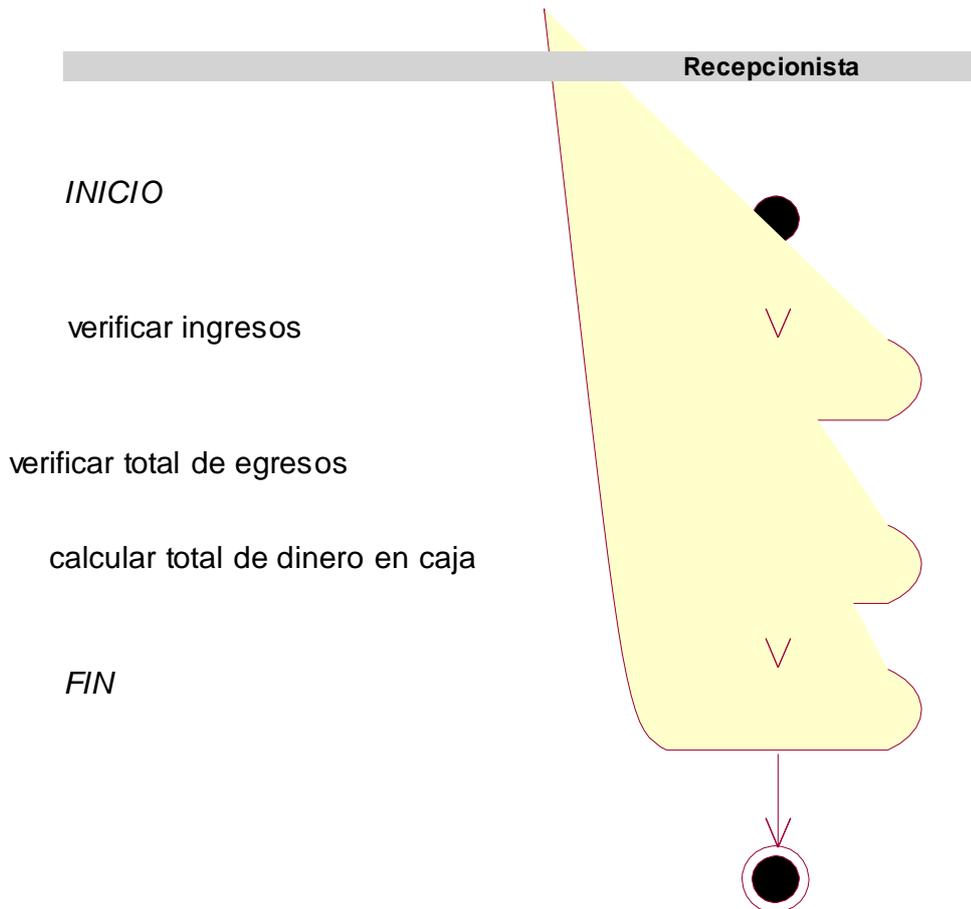
DIAGRAMA DE ACTIVIDADES DEL DCUN SERVICIO



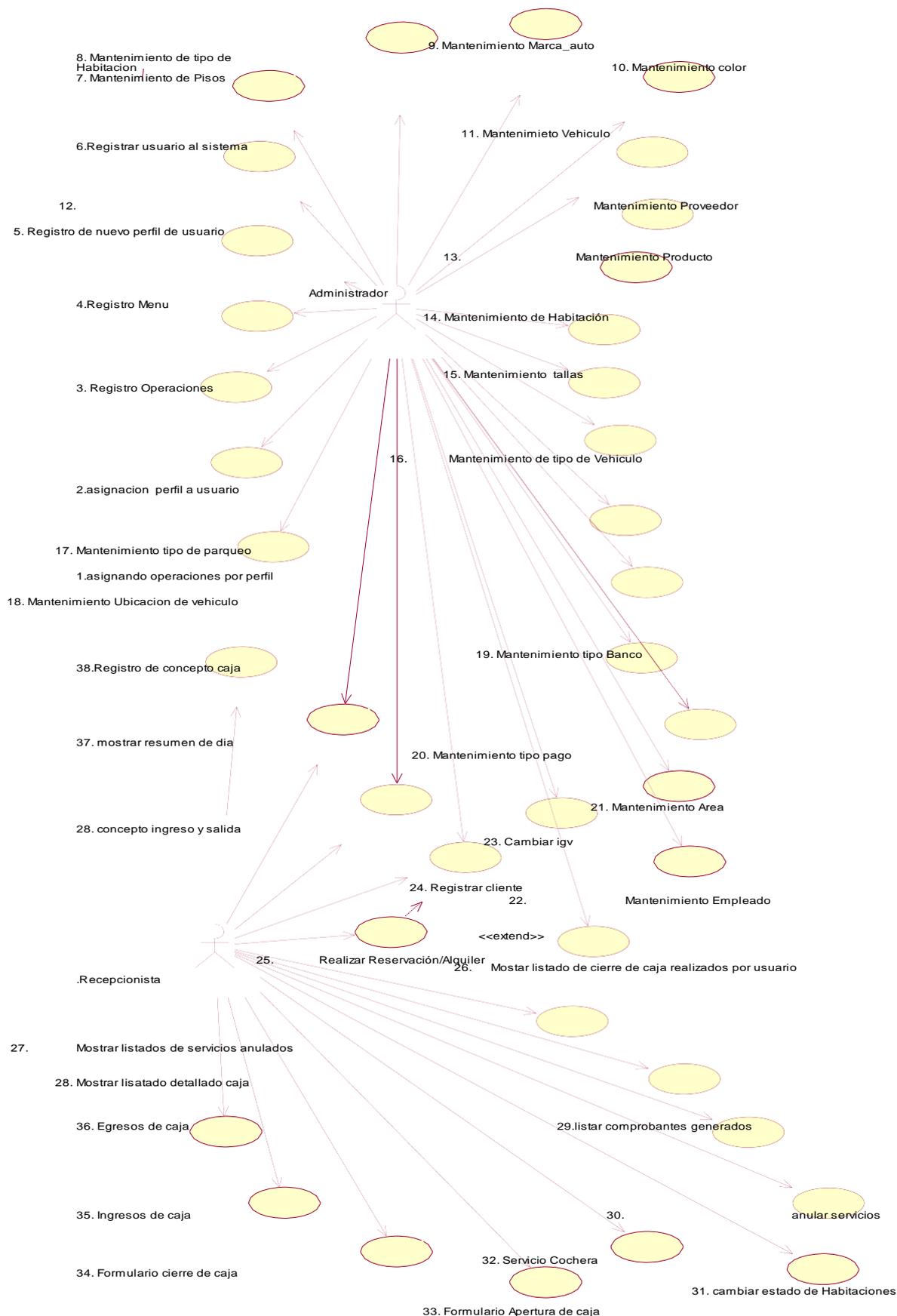
**4.1.1.7
DE CAJA**

DIAGRAMA DE ACTIVIDADES DEL DCUN CUADRE

Cabe mencionar que el proceso de cuadro de caja se realiza por cada turno, actualmente la empresa cuenta con 3 turnos para el puesto de recepcionistas, las cuales son el de Mañana, Tarde y Noche.

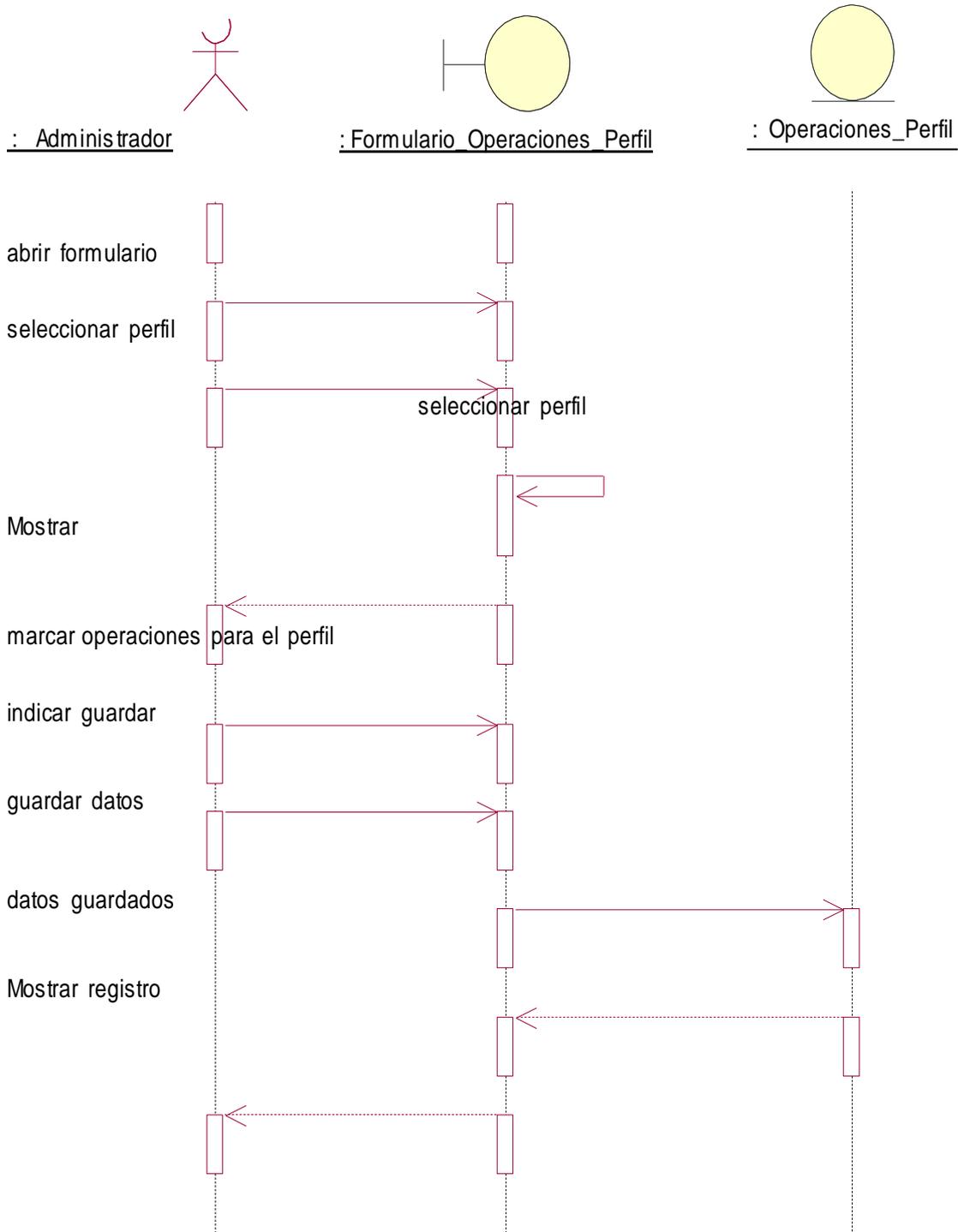


4.1.2 IAGRAMAS DE CASO DE USO DEL SISTEMA

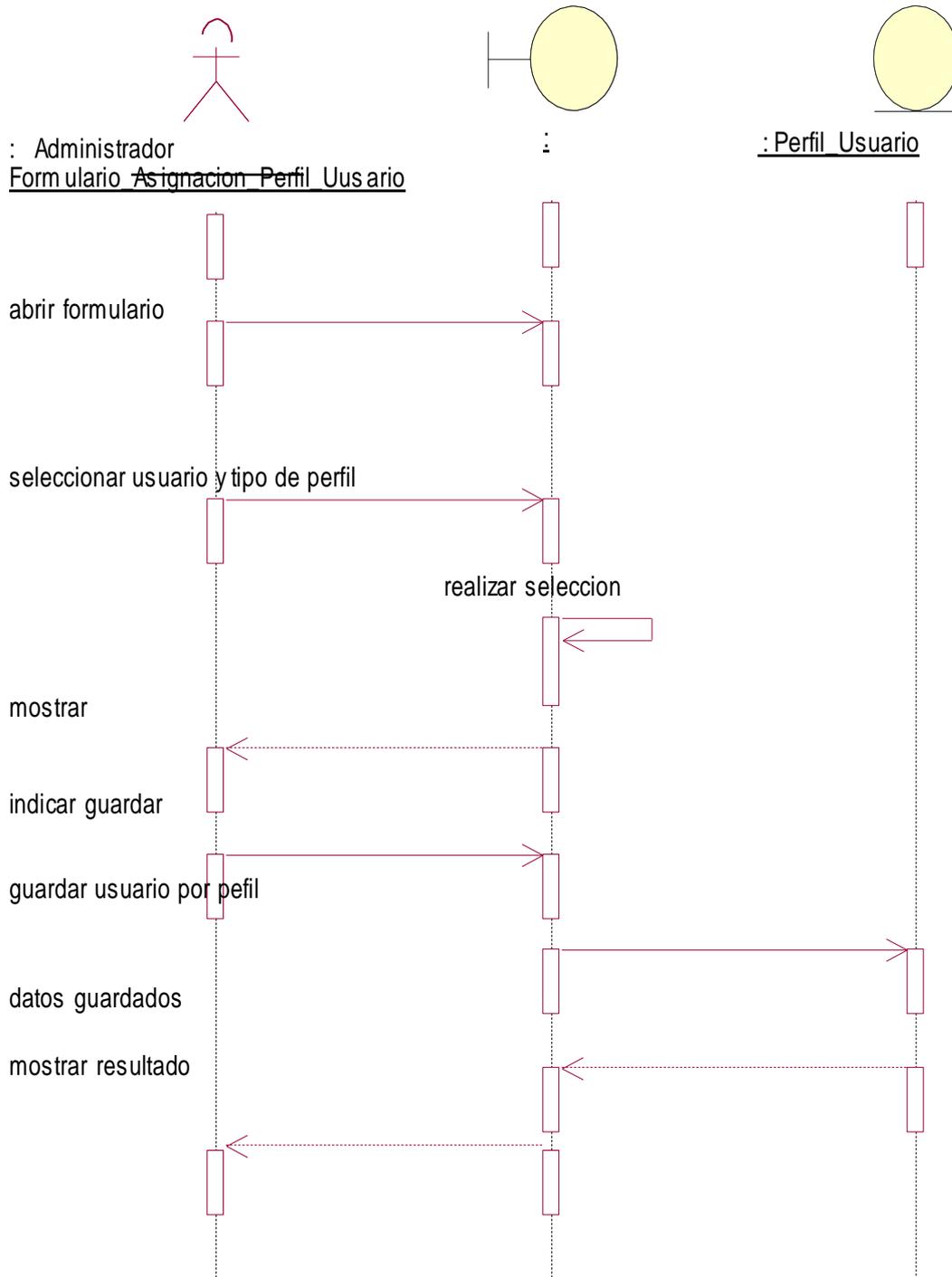


4.1.3 IAGRAMAS DE SECUENCIA DE LOS DCU DEL SISTEMA

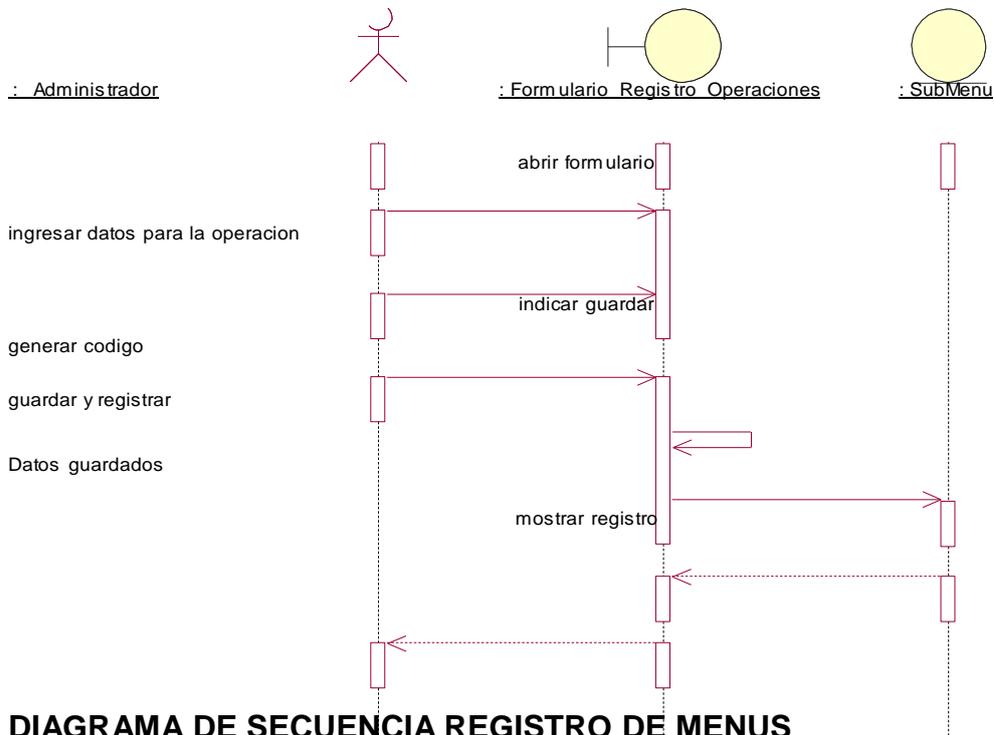
4.1.3.1 DIAGRAMA DE SECUENCIA ASIGNACIÓN DE OPERACIONES POR PERFIL



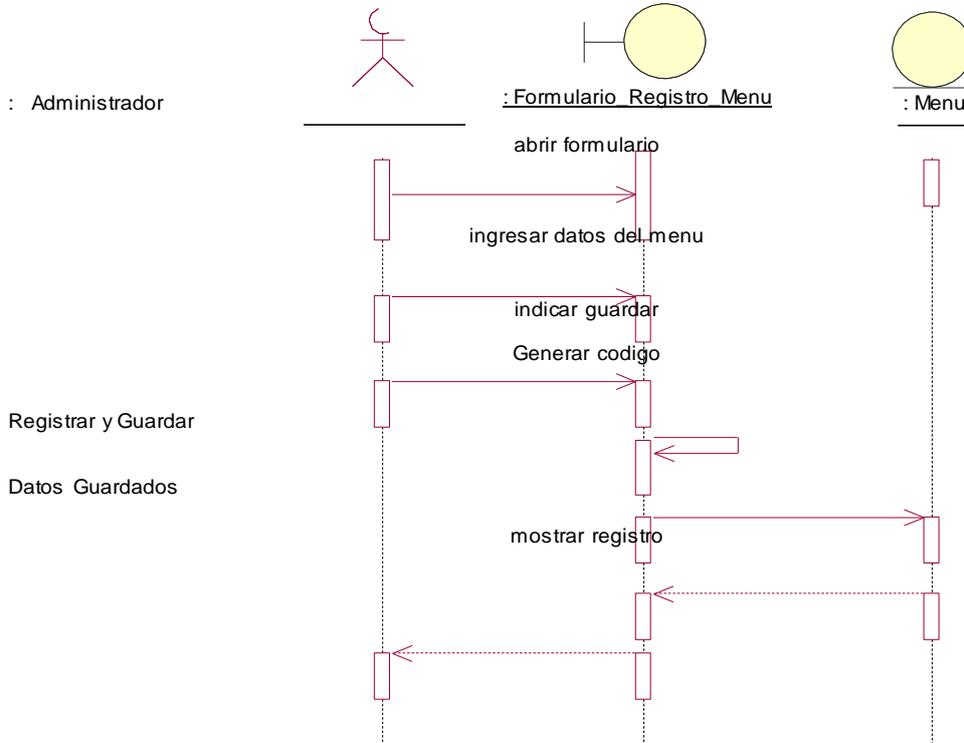
4.1.3.2 DIAGRAMA DE SECUENCIA ASIGNACIÓN DE PERFIL A USUARIO



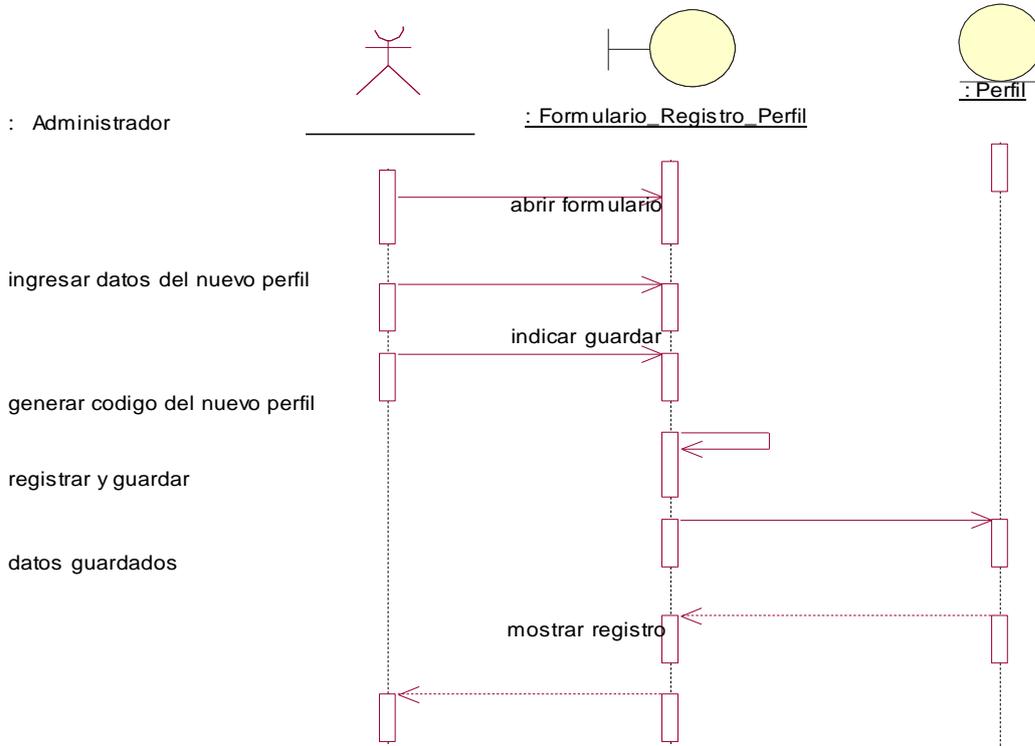
4.1.3.3 DIAGRAMA DE SECUENCIA REGISTRAR OPERACIONES



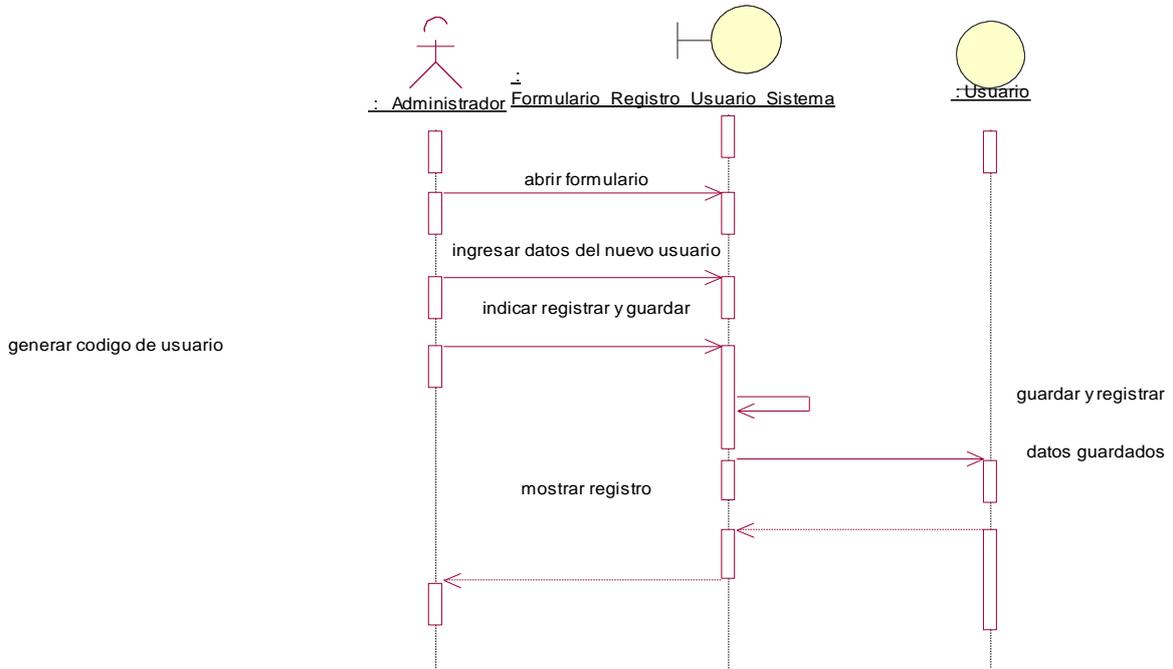
4.1.3.4 DIAGRAMA DE SECUENCIA REGISTRO DE MENUS



4.1.3.5 DIAGRAMA DE SECUENCIA REGISTRO DE PERFIL DE USUARIO

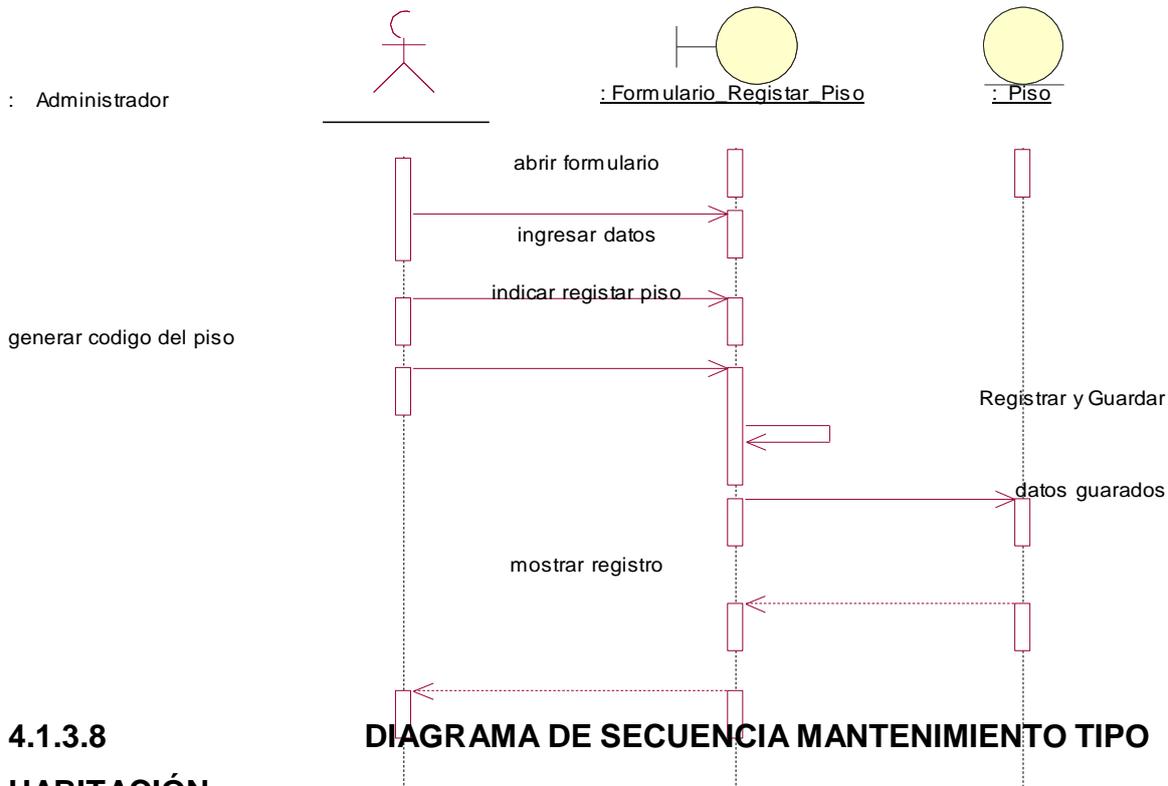


4.1.3.6 DIAGRAMA DE SECUENCIA REGISTRO DE NUEVO USUARIO AL SISTEMA



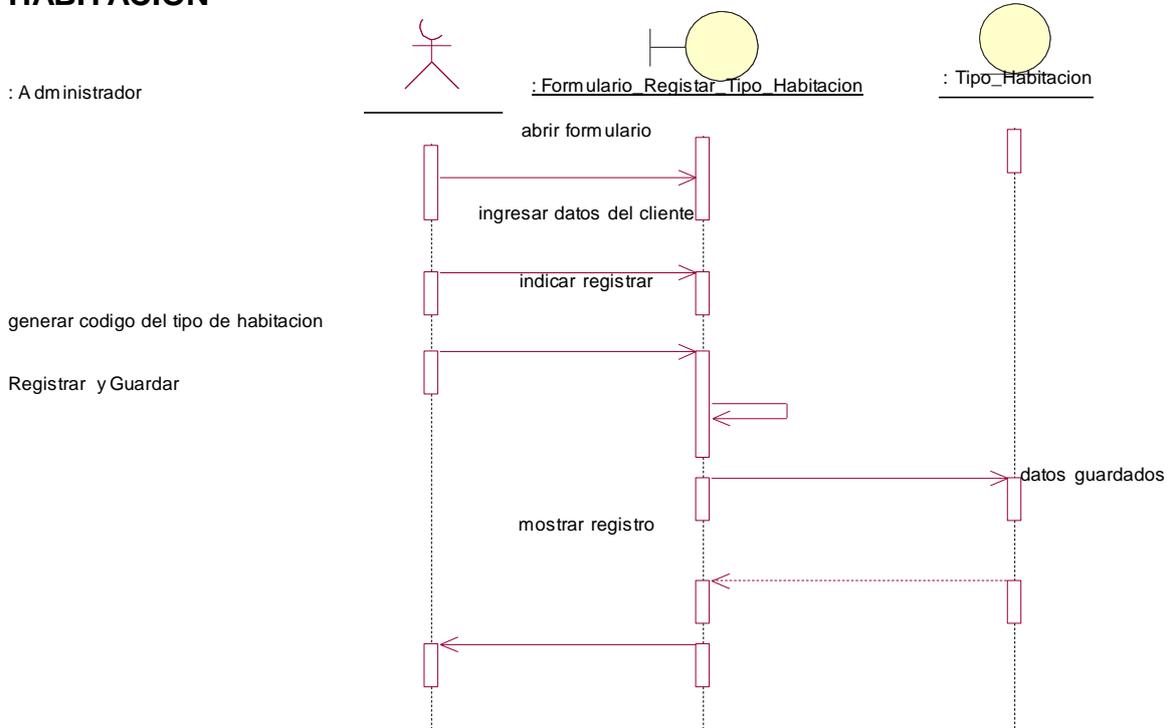
4.1.3.7
PISOS

DIAGRAMA DE SECUENCIA MANTENIMIENTO DE



4.1.3.8
HABITACIÓN

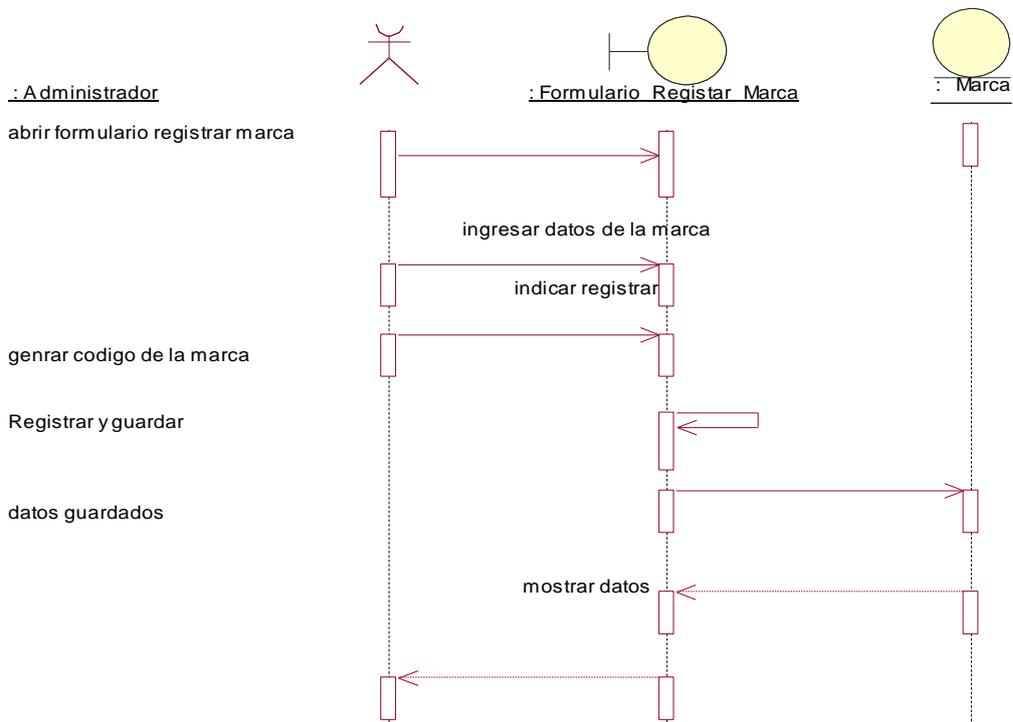
DIAGRAMA DE SECUENCIA MANTENIMIENTO TIPO



4.1.3.9

DIAGRAMA DE SECUENCIA MANTENIMIENTO

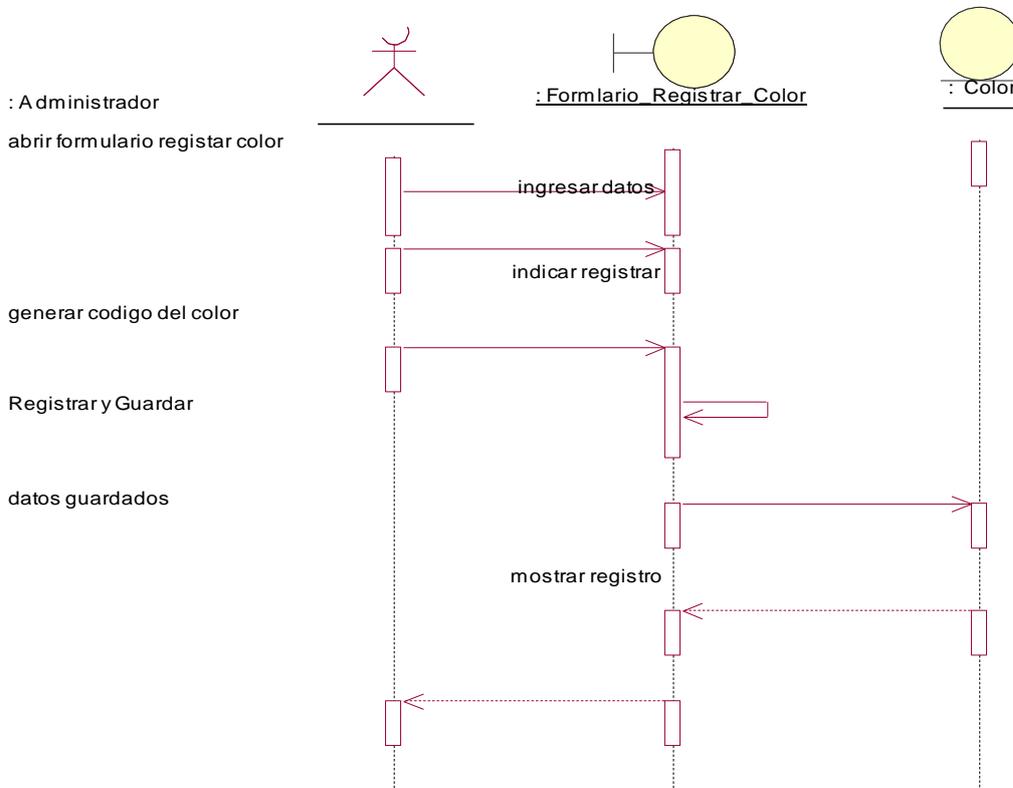
MARCA AUTO



4.1.3.10

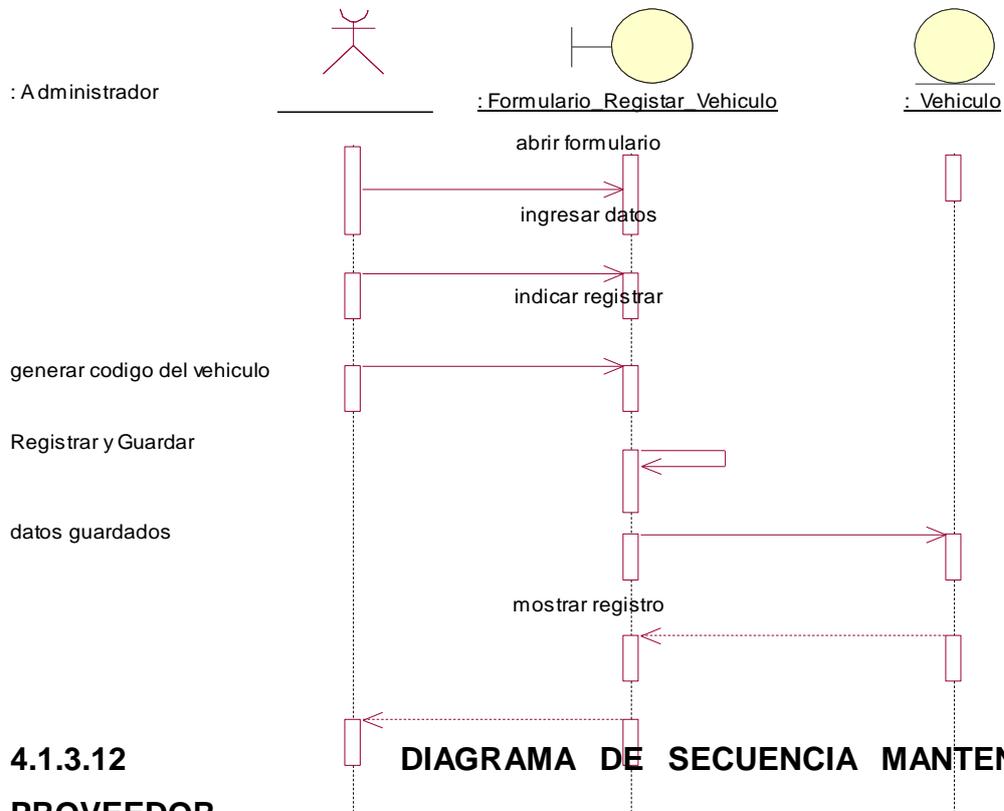
DIAGRAMA DE SECUENCIA MANTENIMIENTO DE

COLOR AUTO



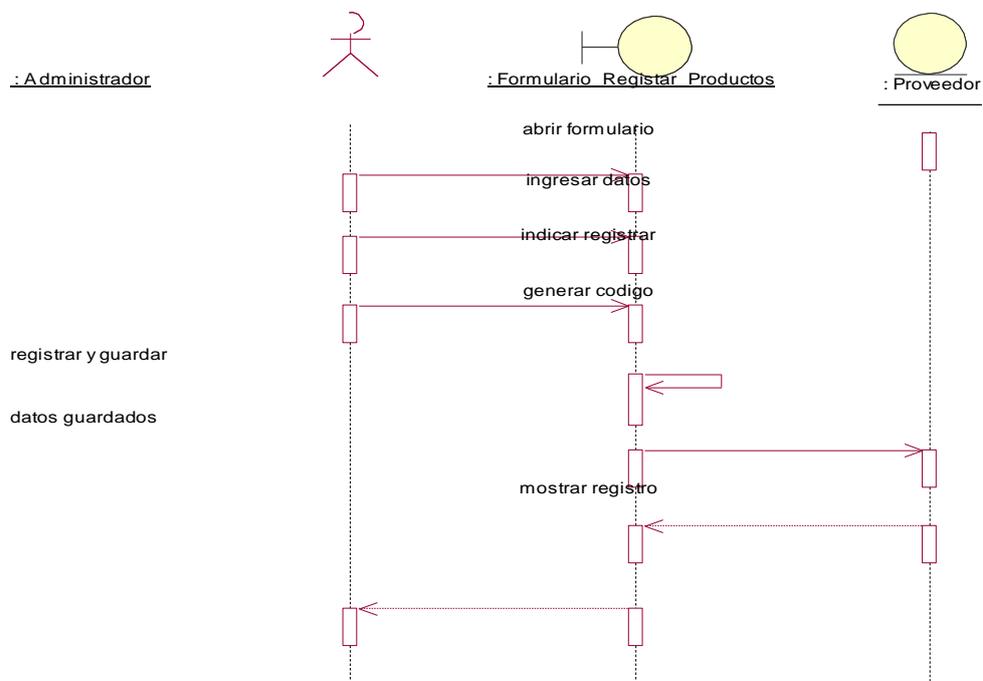
4.1.3.11
VEHÍCULO

DIAGRAMA DE SECUENCIA MANTENIMIENTO



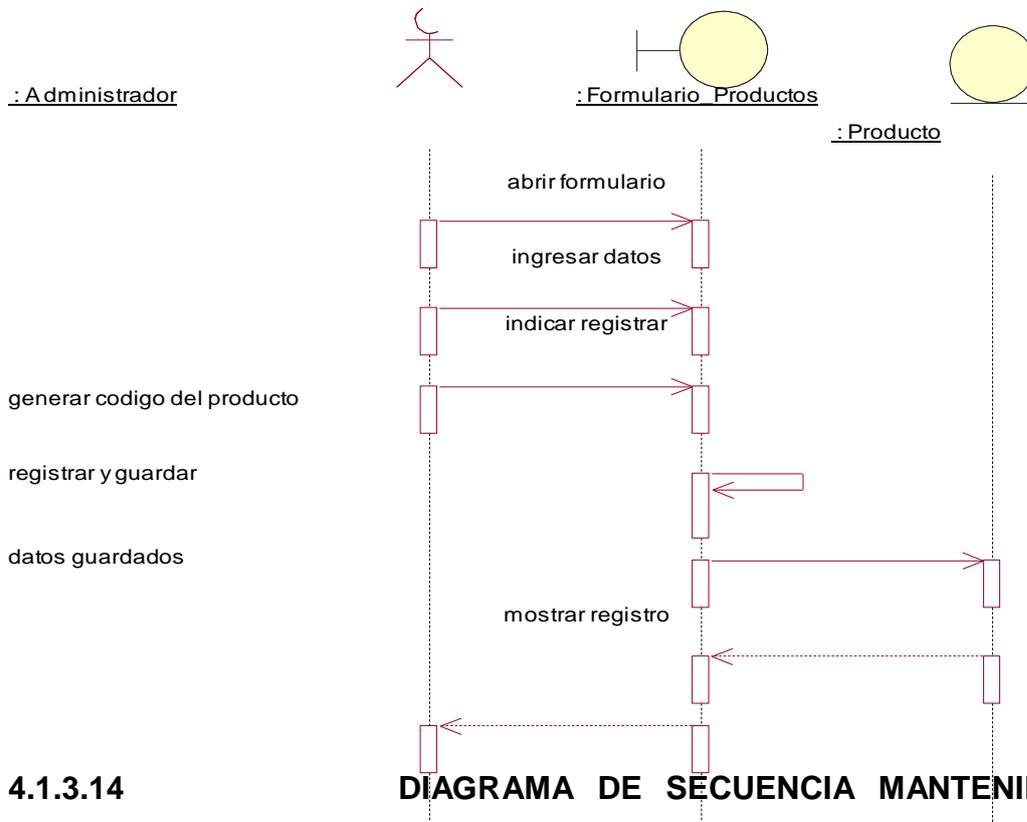
4.1.3.12
PROVEEDOR

DIAGRAMA DE SECUENCIA MANTENIMIENTO



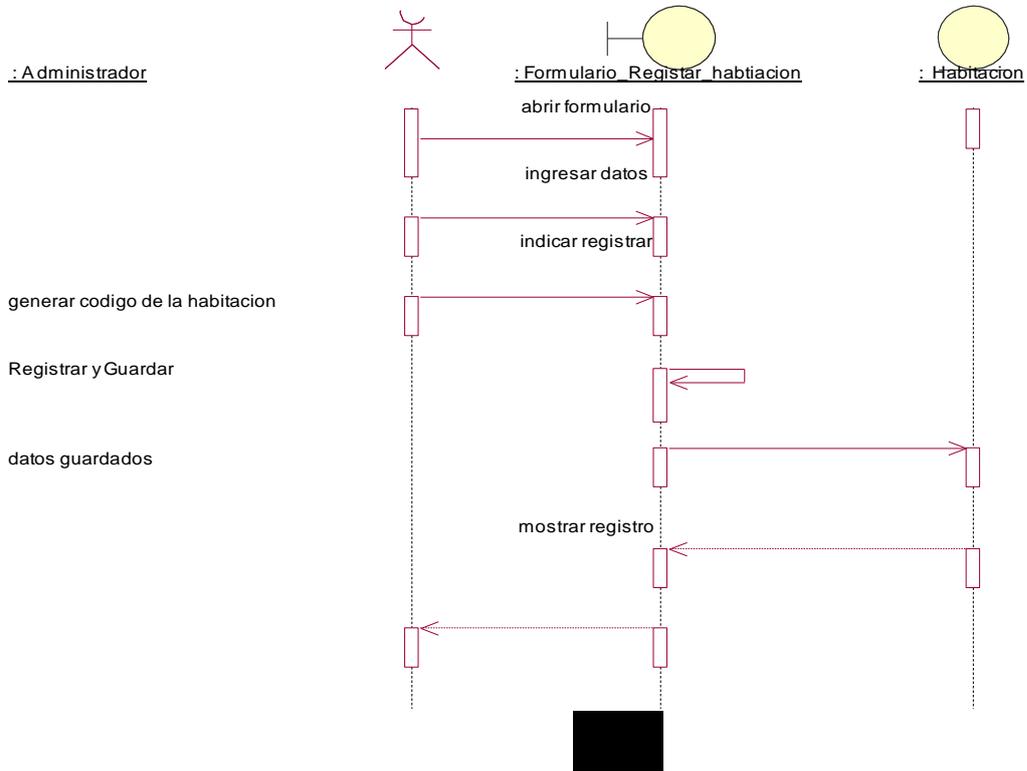
4.1.3.13
PRODUCTO

DIAGRAMA DE SECUENCIA MANTENIMIENTO



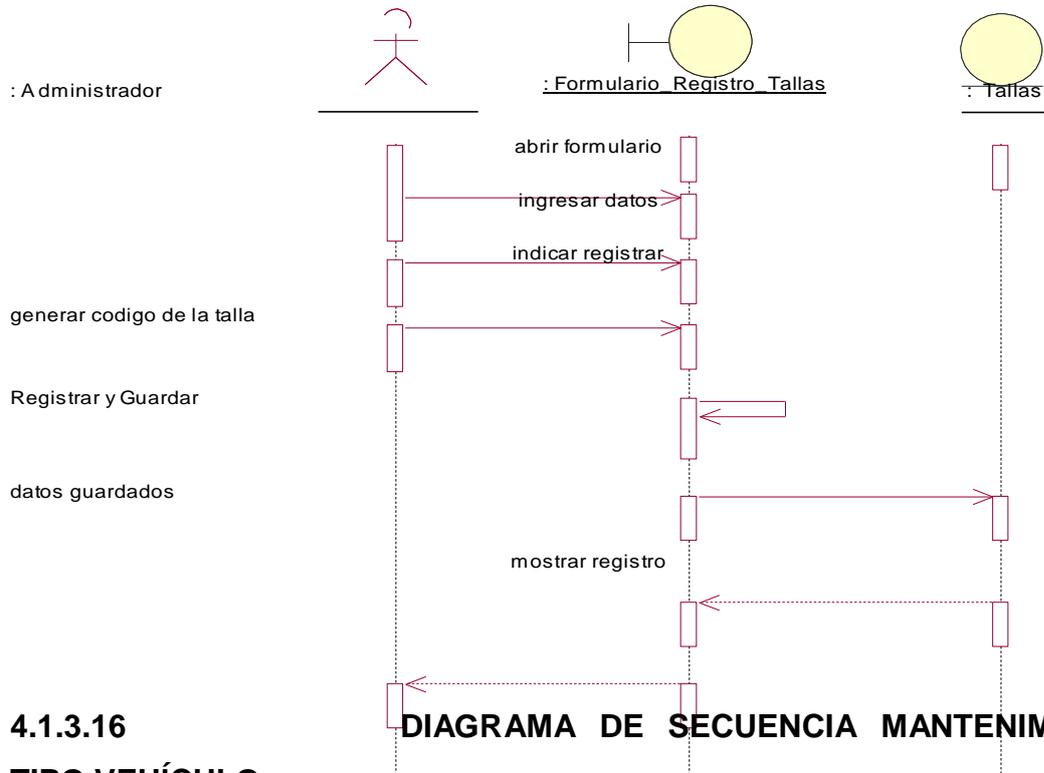
4.1.3.14
HABITACIÓN

DIAGRAMA DE SECUENCIA MANTENIMIENTO



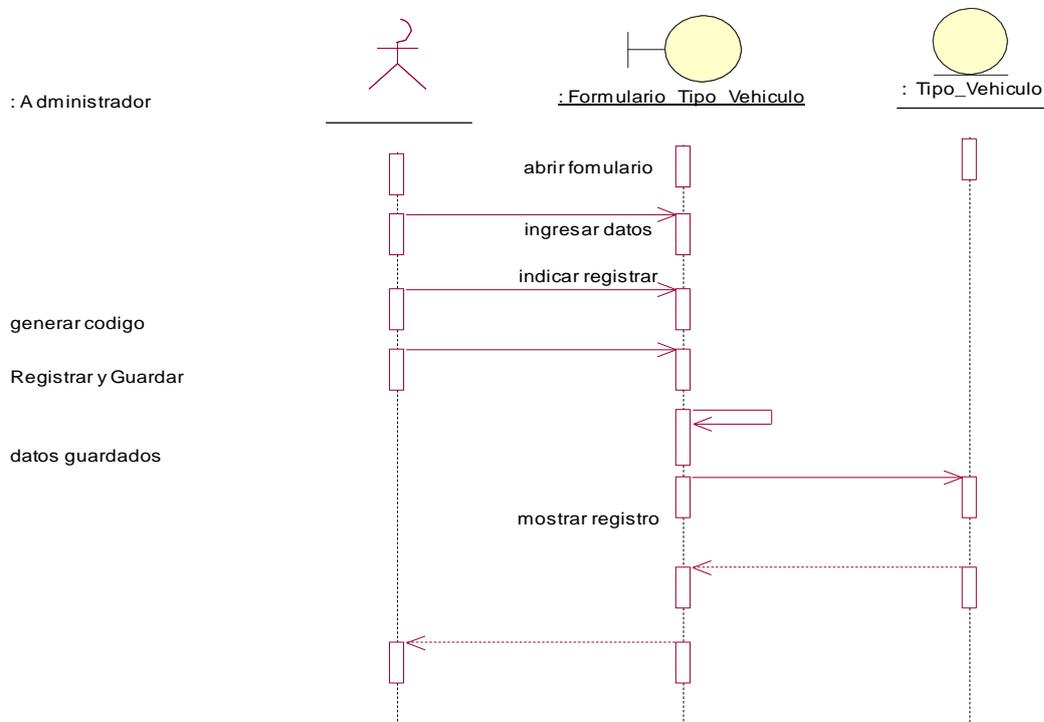
4.1.3.15
TALLAS

DIAGRAMA DE SECUENCIA MANTENIMIENTO



4.1.3.16
TIPO VEHÍCULO

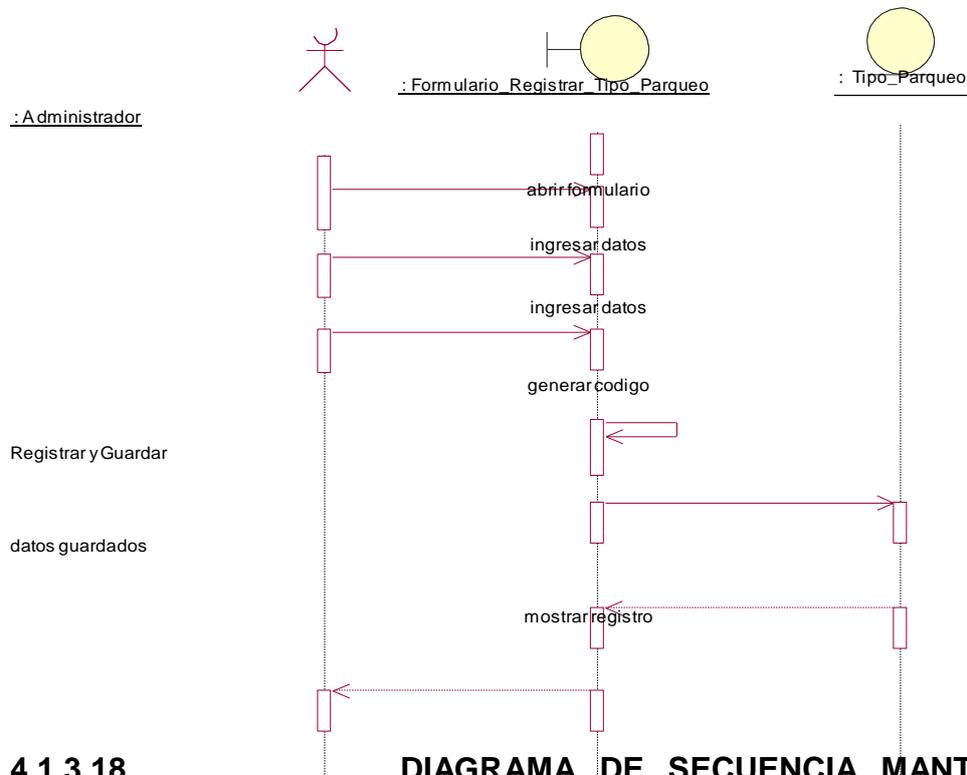
DIAGRAMA DE SECUENCIA MANTENIMIENTO



4.1.3.17

DIAGRAMA DE SECUENCIA MANTENIMIENTO

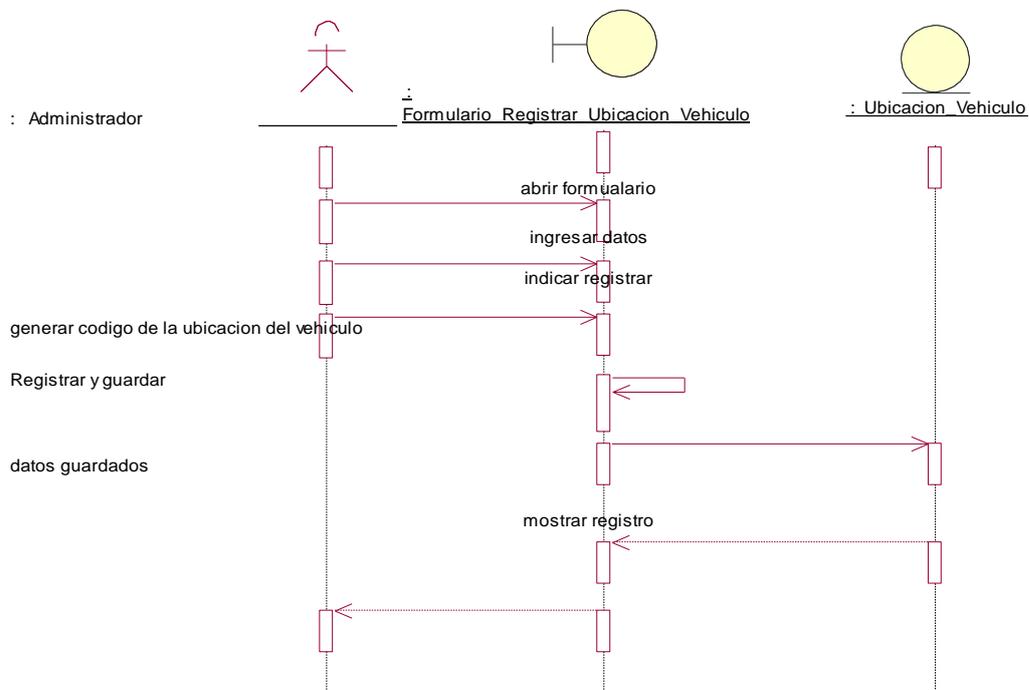
TIPO PARQUEO



4.1.3.18

DIAGRAMA DE SECUENCIA MANTENIMIENTO

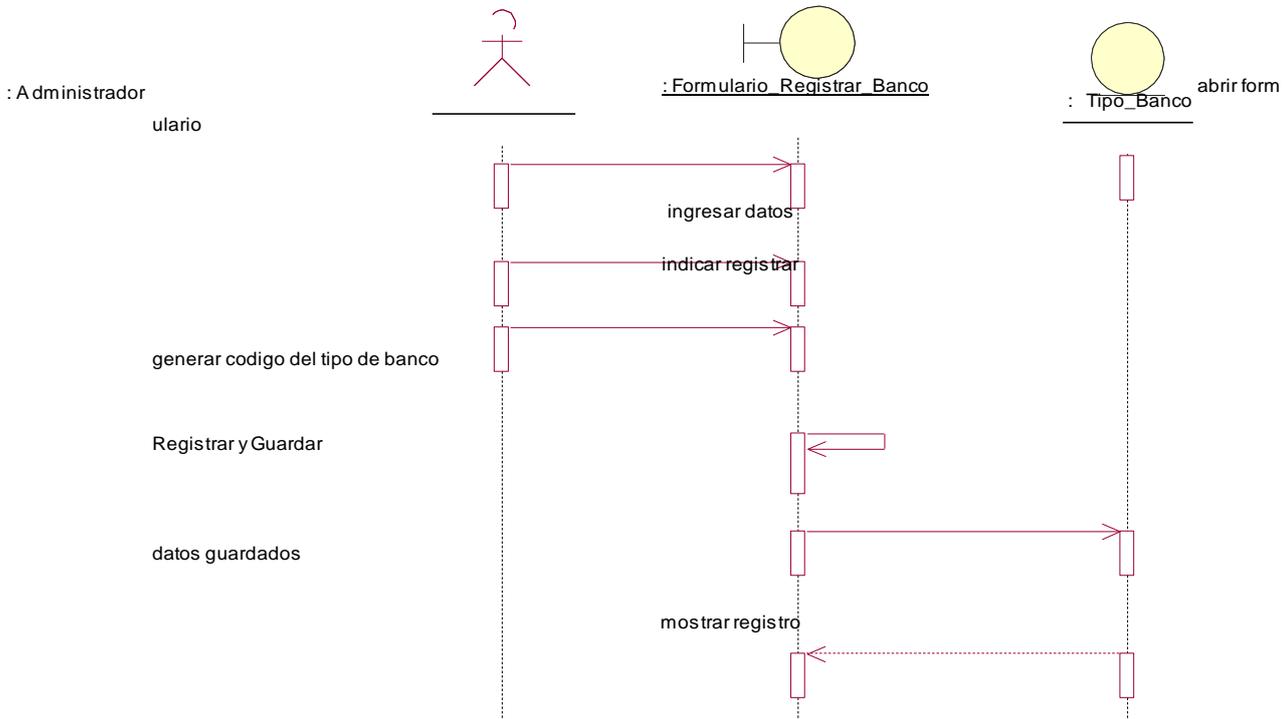
UBICACIÓN DE VEHÍCULO



4.1.3.19

DIAGRAMA DE SECUENCIA MANTENIMIENTO

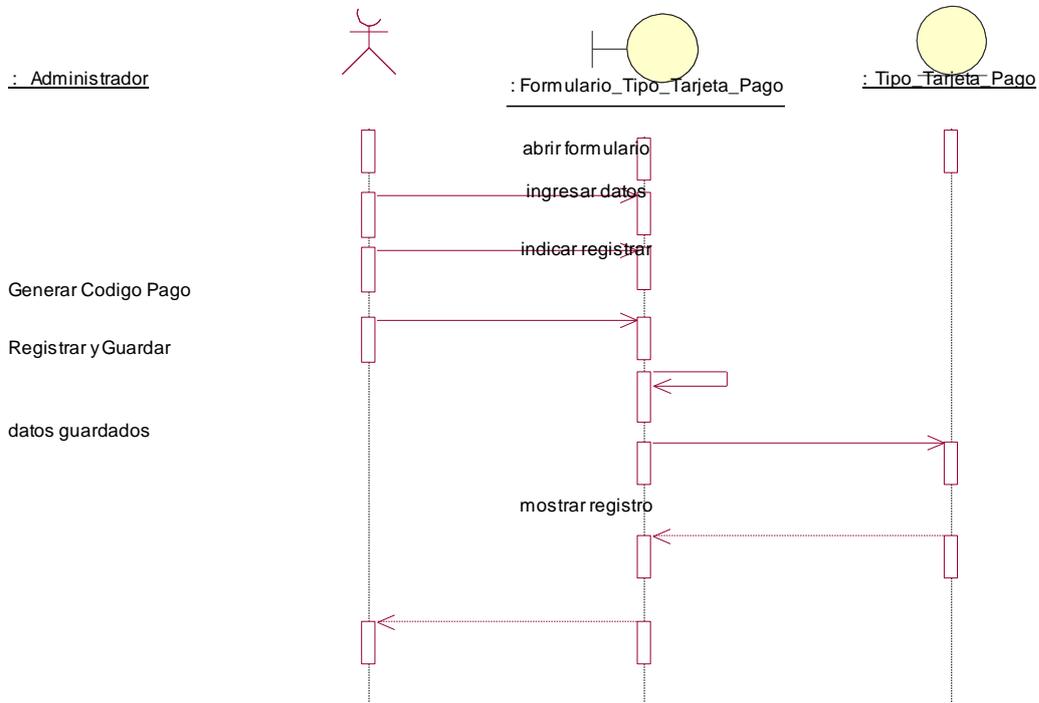
TIPO BANCO



4.1.3.20

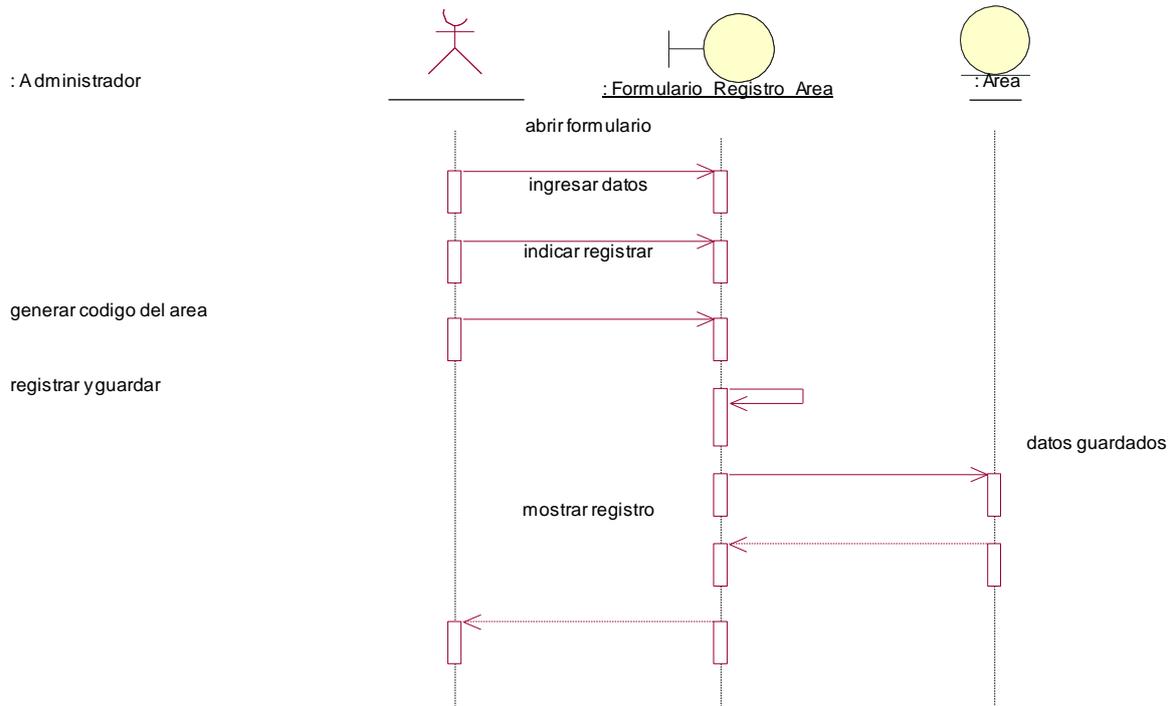
DIAGRAMA DE SECUENCIA MANTENIMIENTO

TIPO PAGO



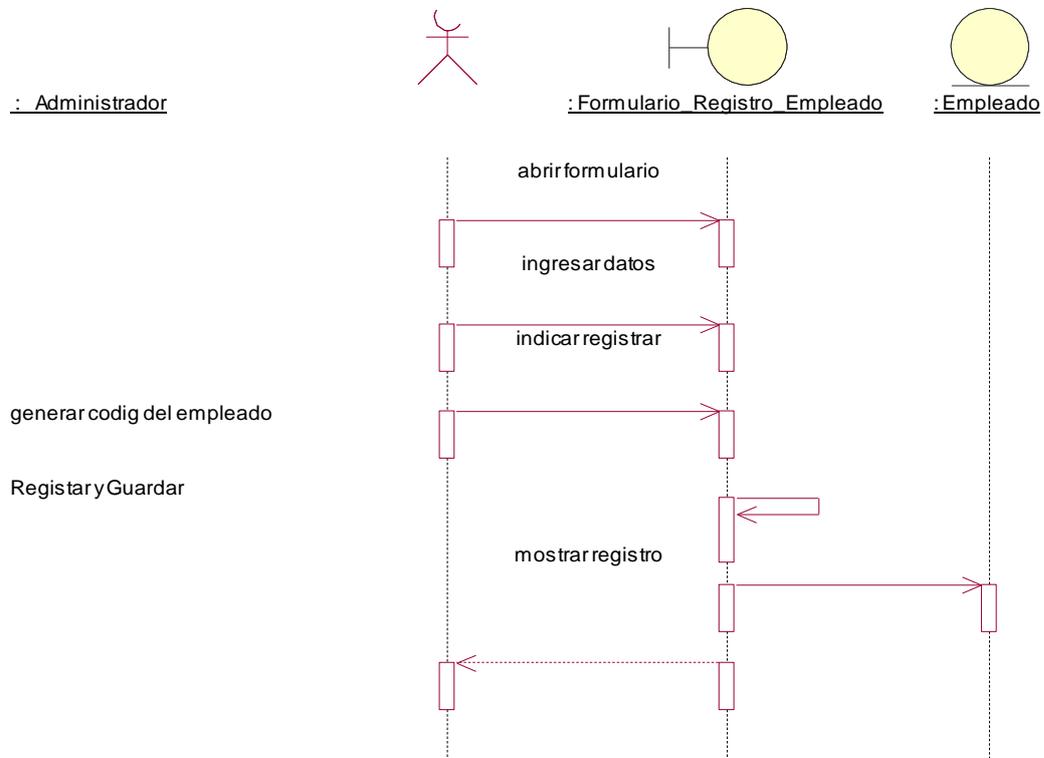
4.1.3.21
ÁREA

DIAGRAMA DE SECUENCIA MANTENIMIENTO

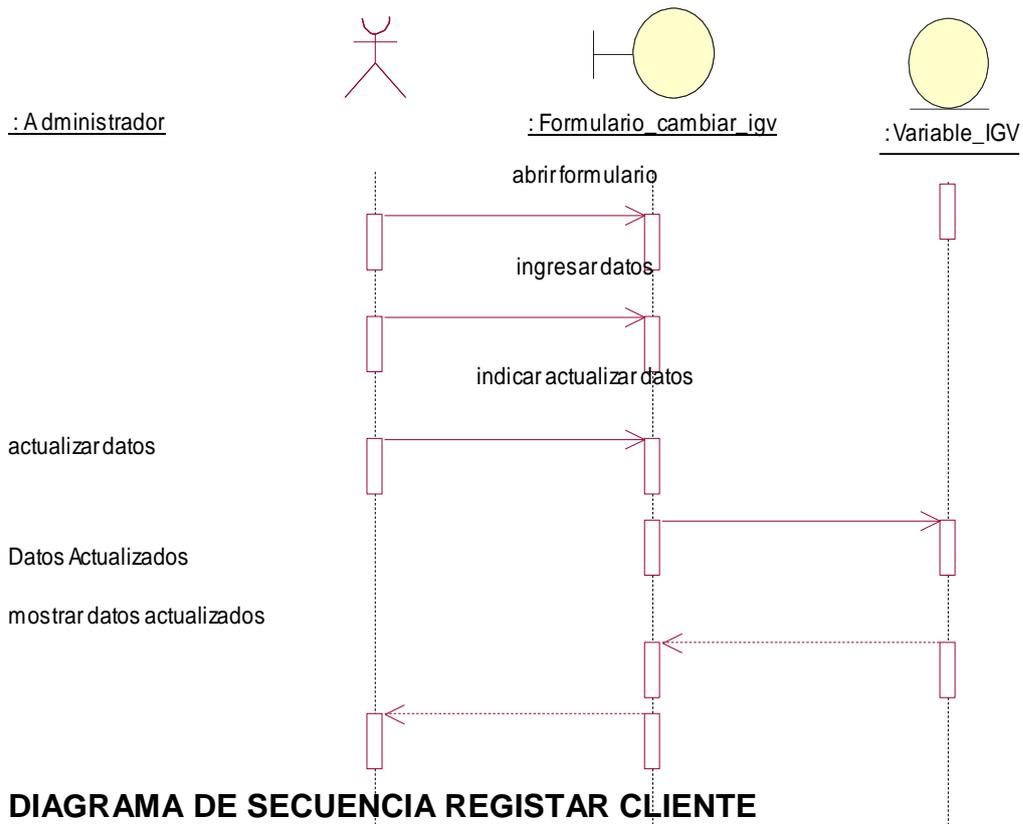


4.1.3.22
EMPLEADO

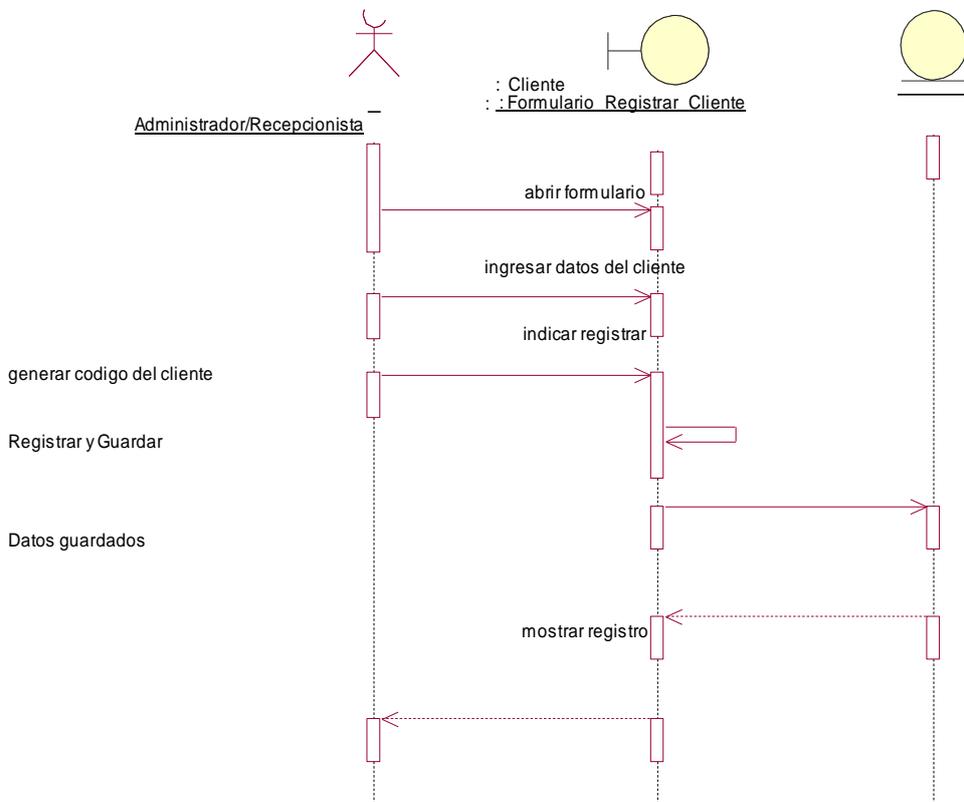
DIAGRAMA DE SECUENCIA MANTENIMIENTO



4.1.3.23 DIAGRAMA DE SECUENCIA CAMBIO IGV



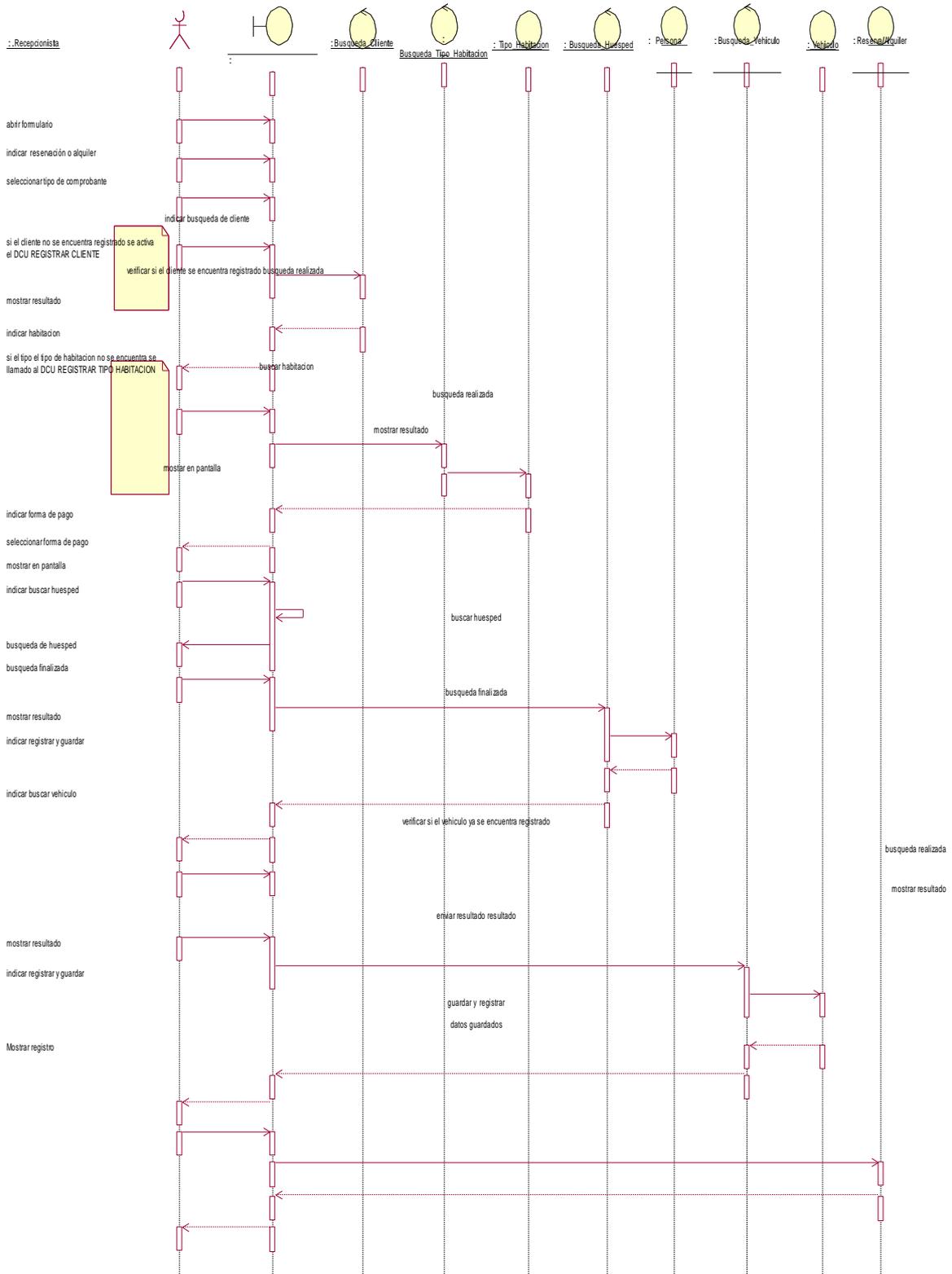
4.1.3.24 DIAGRAMA DE SECUENCIA REGISTRAR CLIENTE



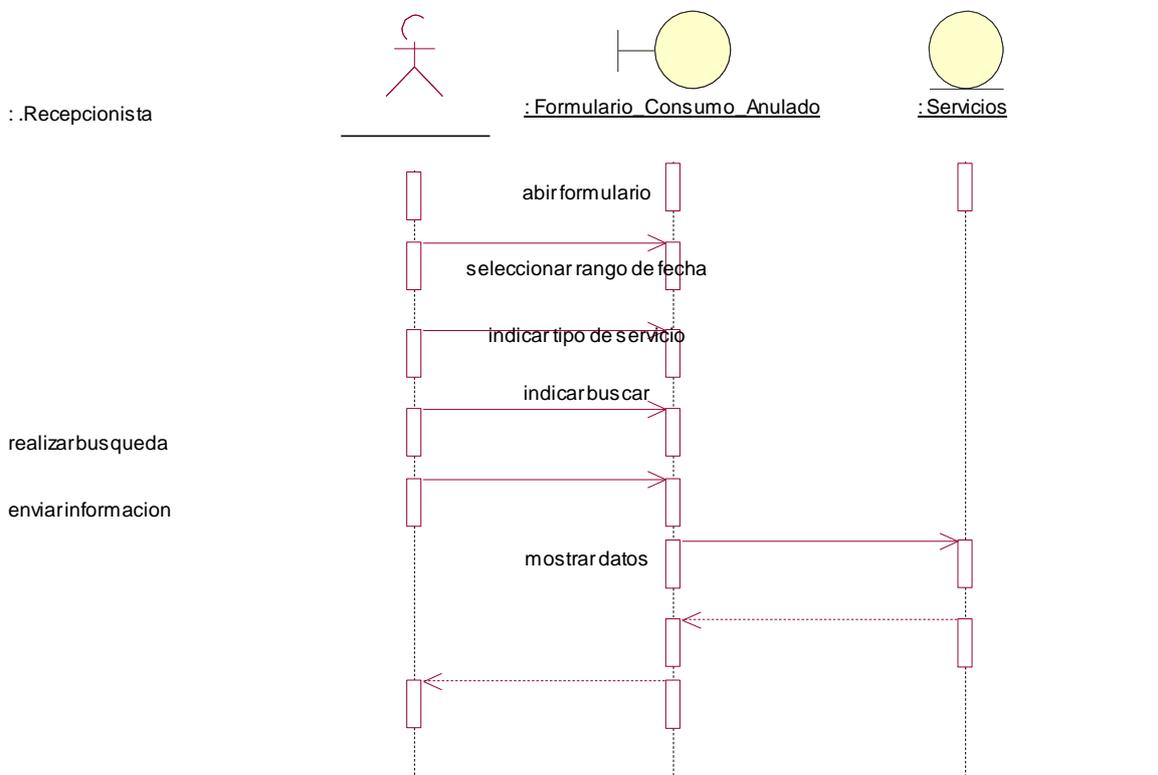
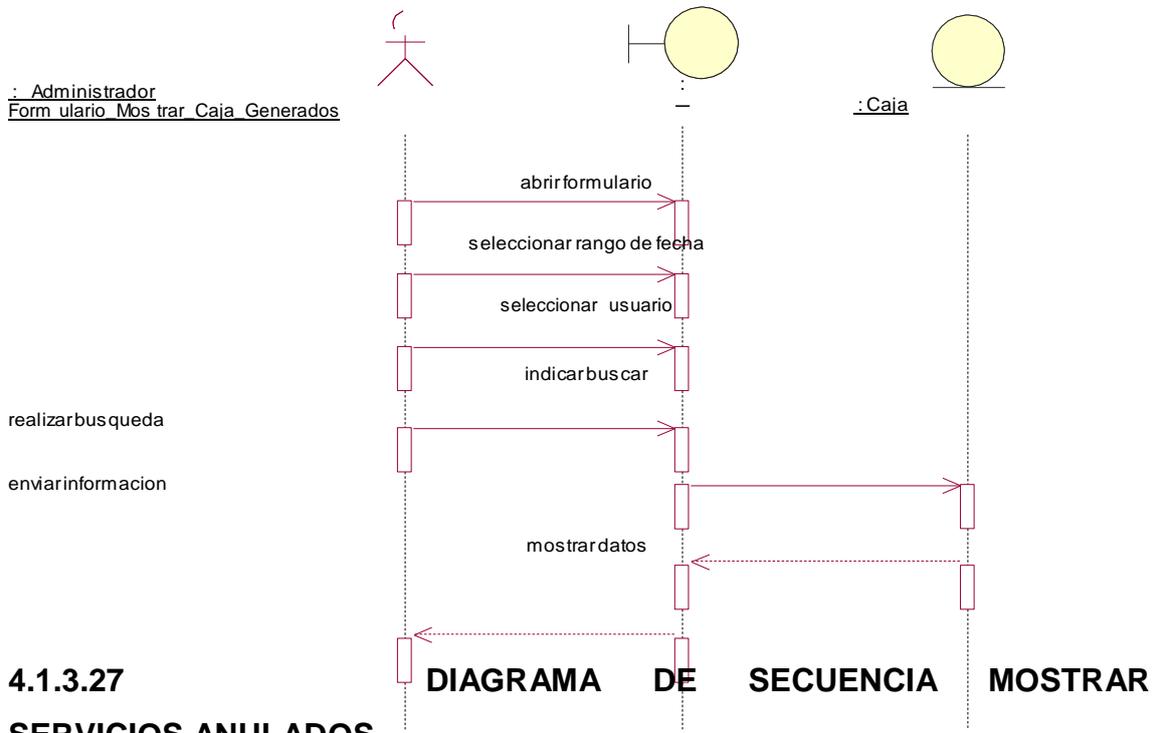
4.1.3.25

DIAGRAMA DE SECUENCIA

RESERVACIÓN/ALQUILER

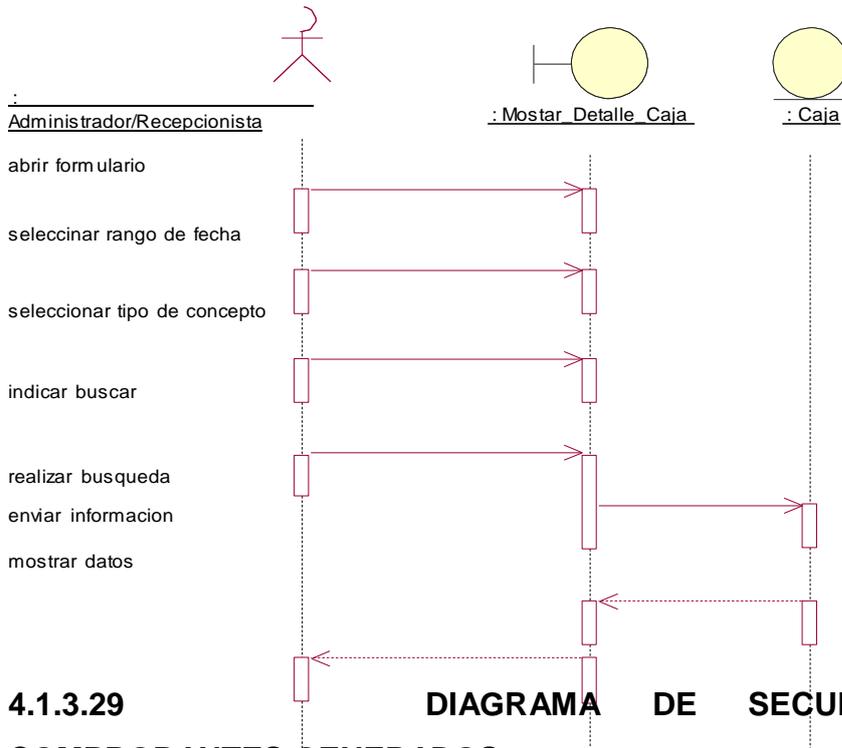


4.1.3.26 DIAGRAMA DE SECUENCIA MOSTRAR CIERRE DE CAJA POR USUARIO



4.1.3.28
DE CAJA

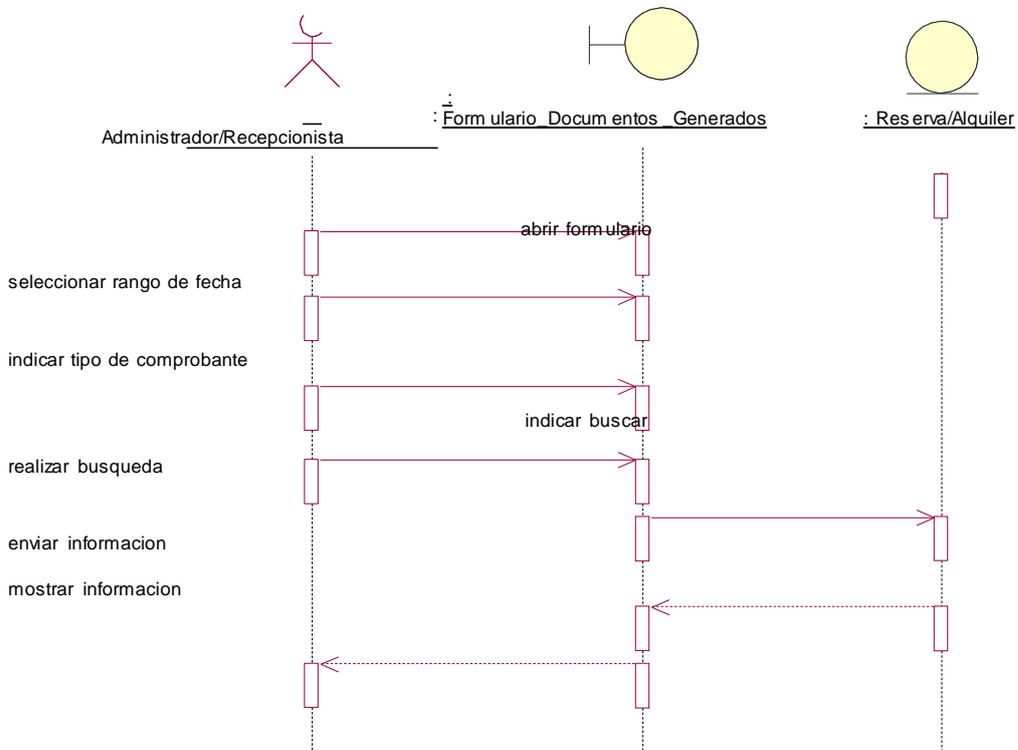
DIAGRAMA DE SECUENCIA MOSTRAR DETALLE



4.1.3.29

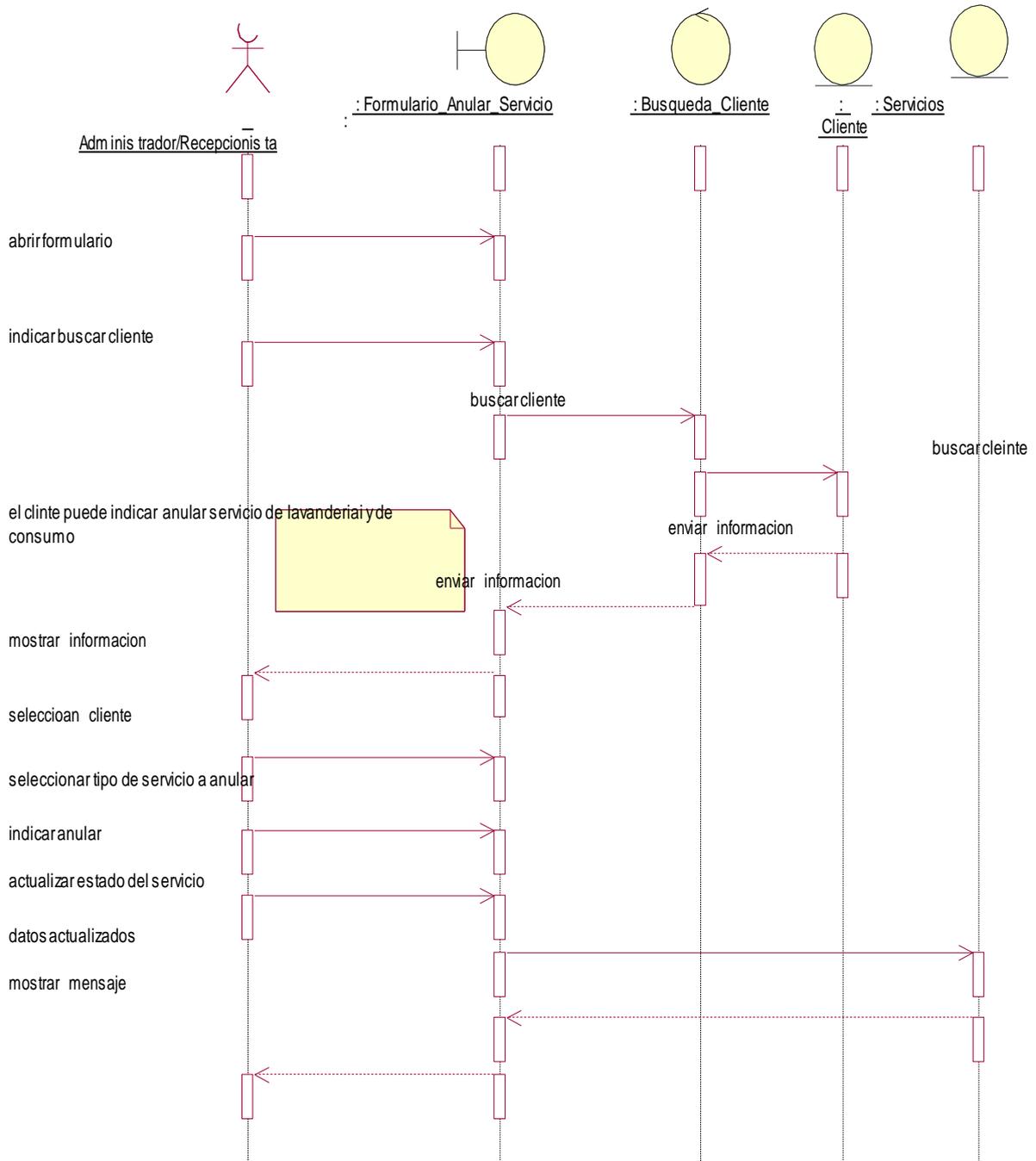
DIAGRAMA DE SECUENCIA MOSTRAR

COMPROBANTES GENERADOS

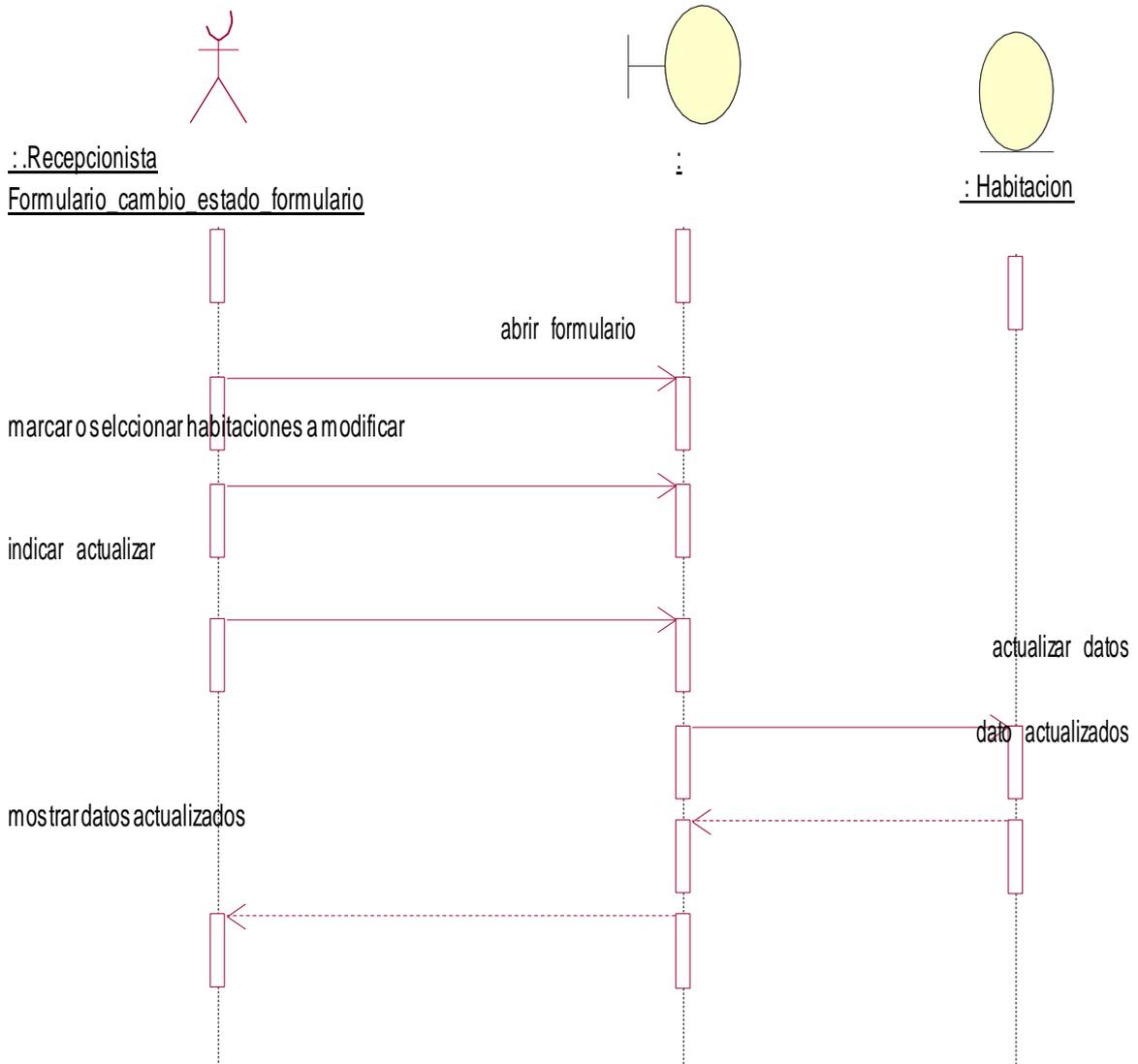


4.1.3.30
SERVICIO

DIAGRAMA DE SECUENCIA ANULACIÓN DE

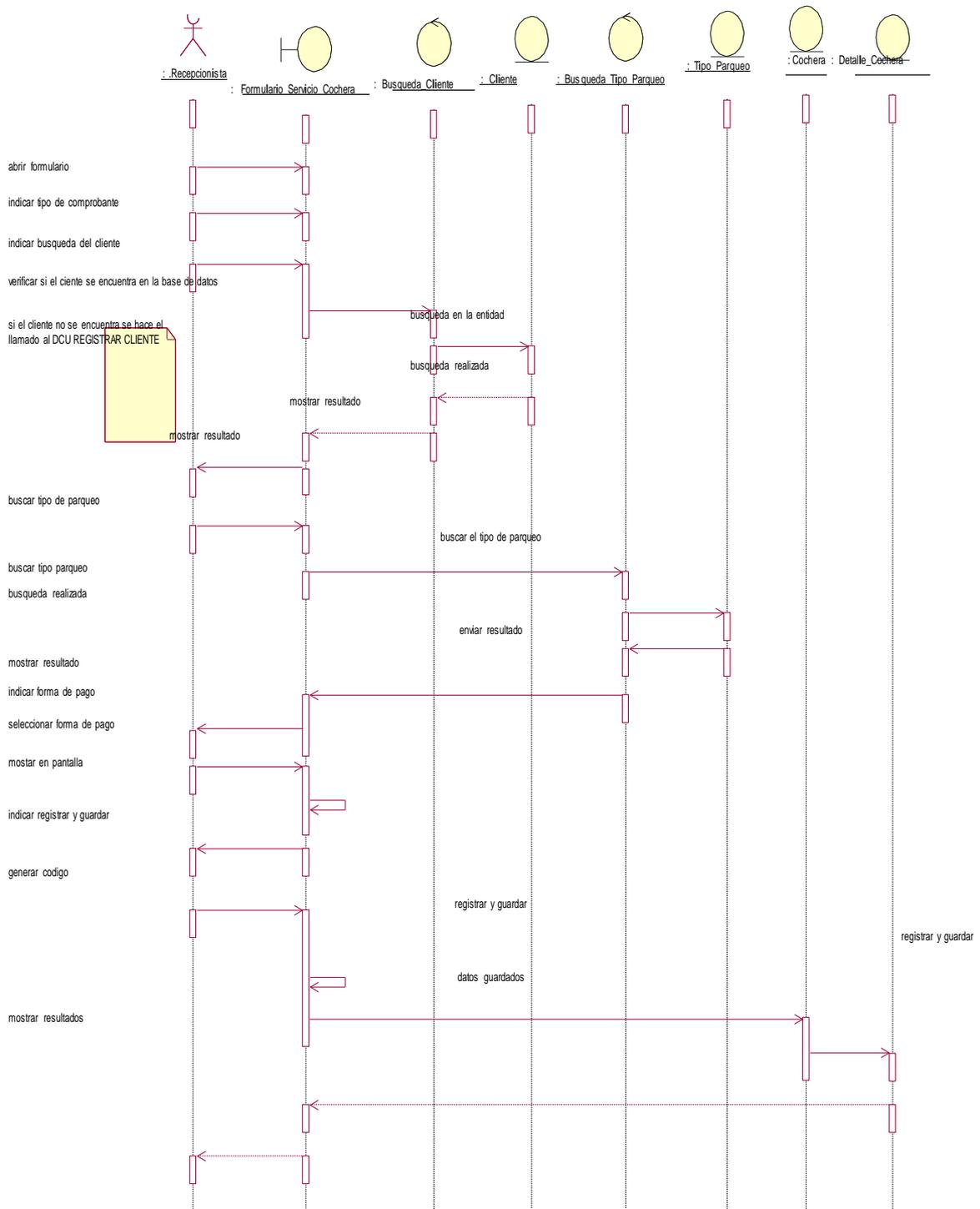


4.1.3.31 DIAGRAMA DE SECUENCIA CAMBIAR ESTADO DE HABITACIONES

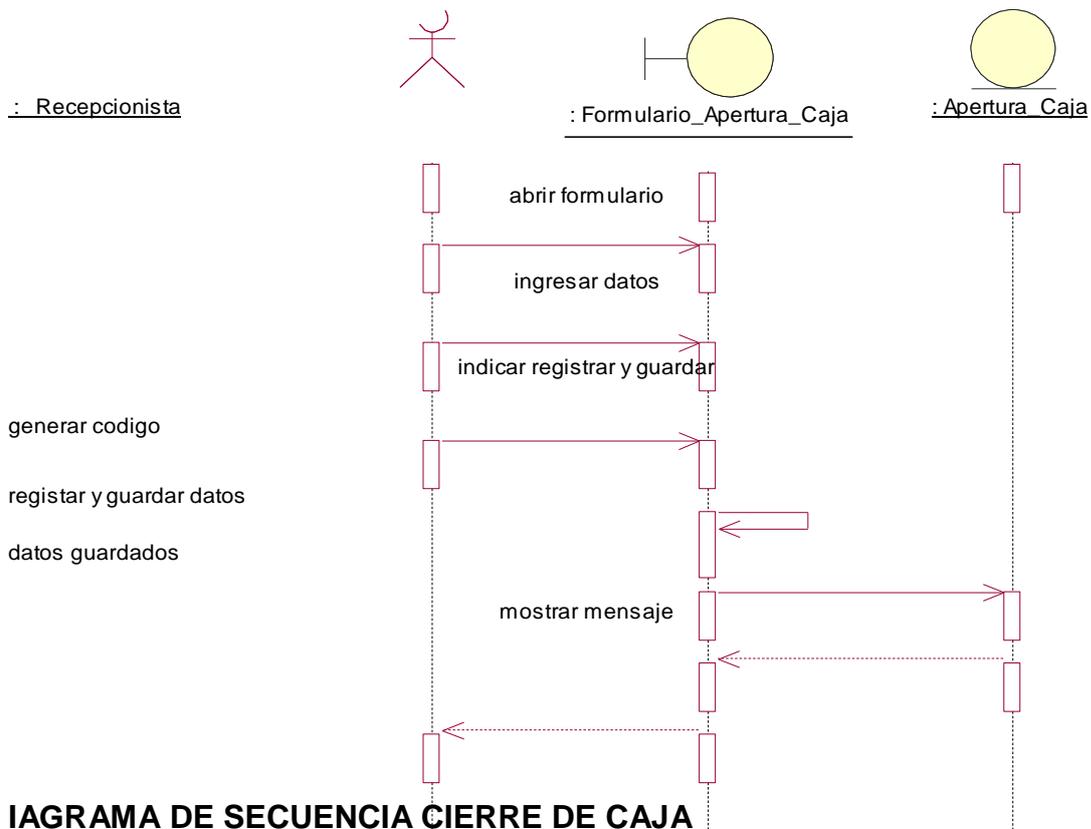


4.1.3.32 COCHERA

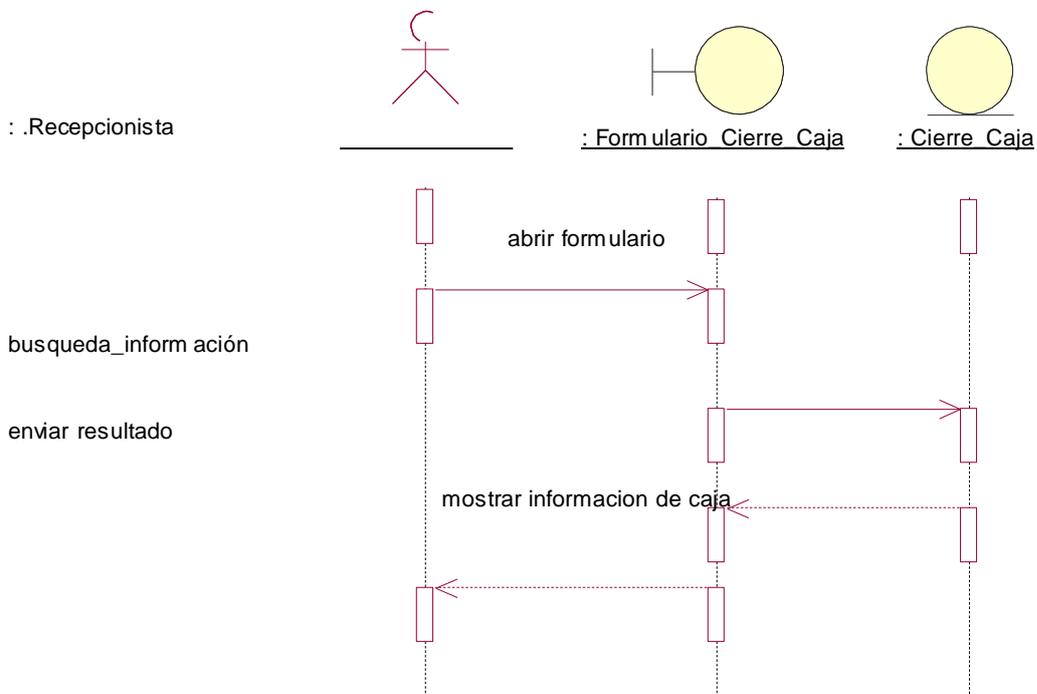
DIAGRAMA DE SECUENCIA SERVICIO DE COCHERA



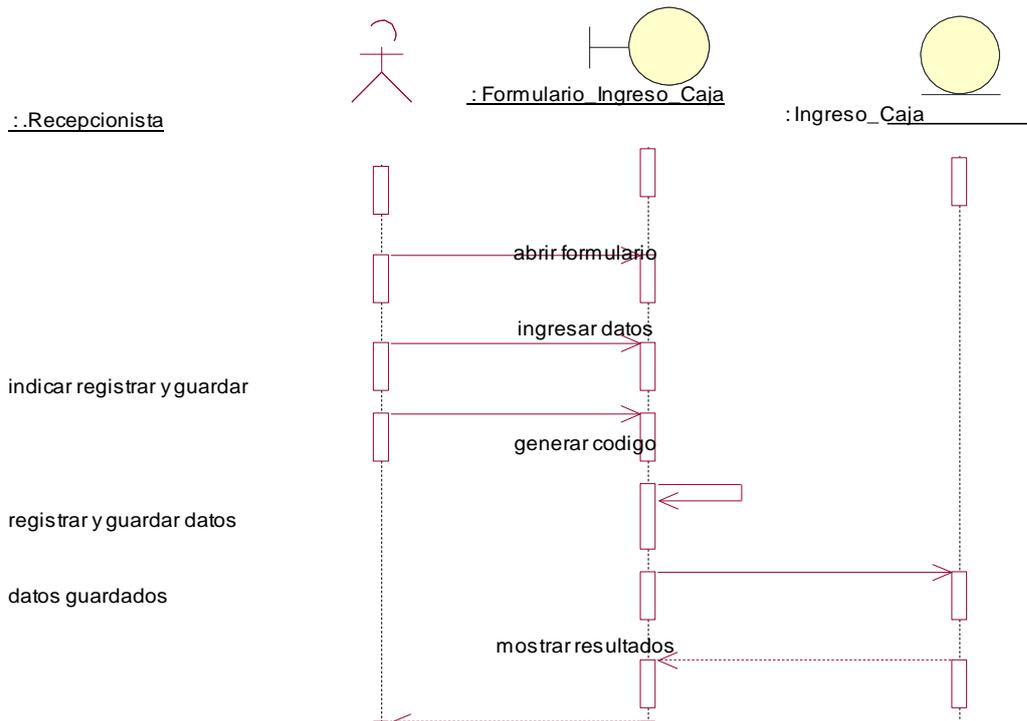
4.1.3.33 IAGRAMA DE SECUENCIA APERTURA DE CAJA



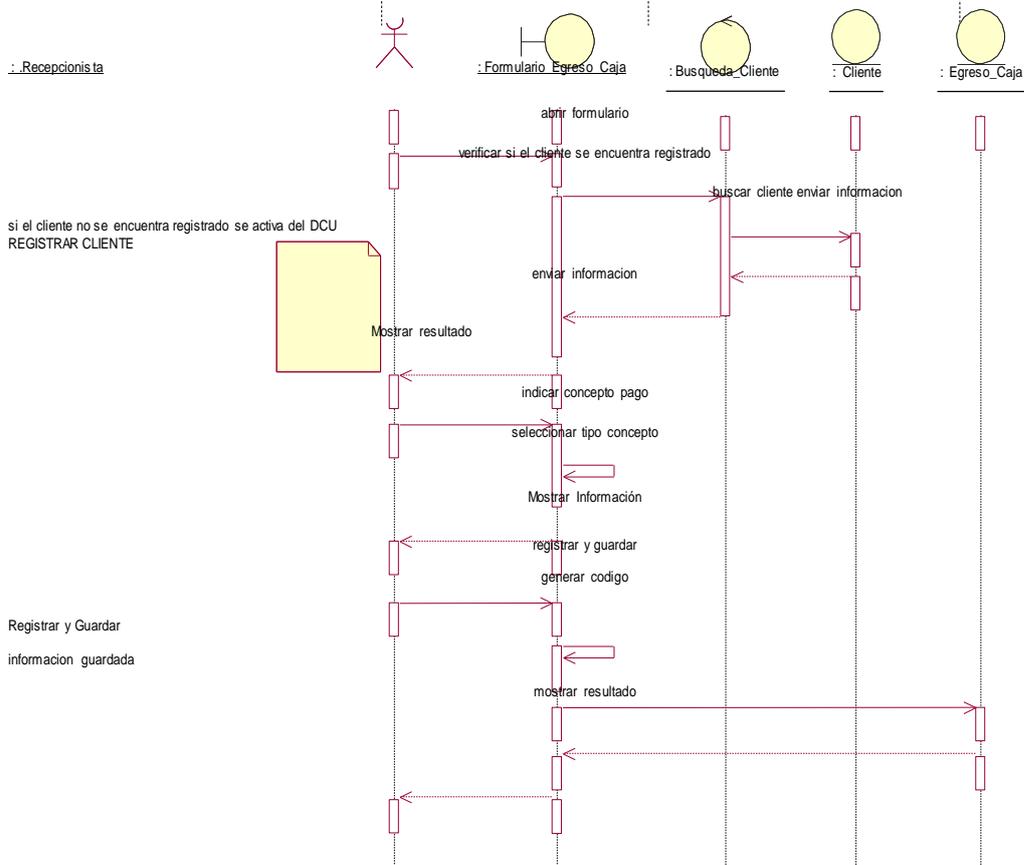
4.1.3.34 IAGRAMA DE SECUENCIA CIERRE DE CAJA



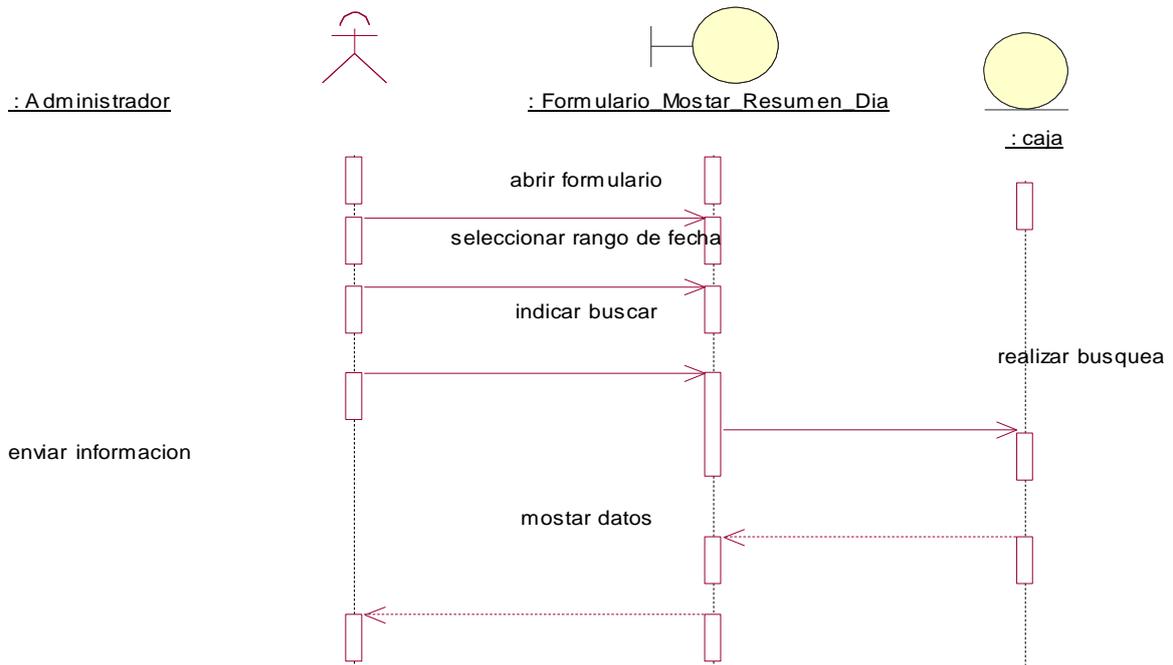
4.1.3.35 IAGRAMA DE SECUENCIA INGRESO CAJA



4.1.3.36 IAGRAMA DE SECUENCIA EGRESO CAJA

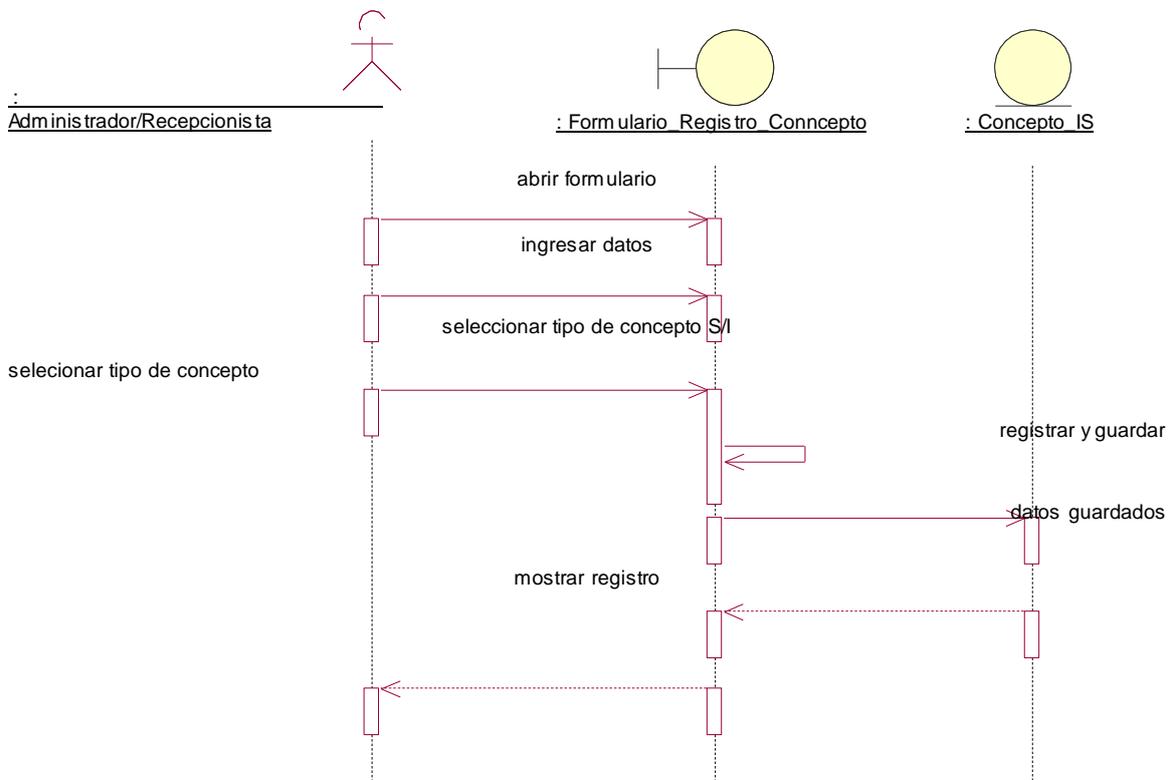


4.1.3.37 IAGRAMA DE SECUENCIA RESUMEN DIA



4.1.3.38 DIAGRAMA DE SECUENCIA REGISTRO

CONCEPTO DE CAJA I/S



4.2 ANÁLISIS DEL SISTEMA

4.2.1 INICIACIÓN DE ESTANDARES

4.2.1.1 TANDARES PARA LAS HISTORIAS DE LOS USUARIOS

Los estándares descritos a continuación están referidos a las fichas donde se describirán las historias de usuario a implementar en cada iteración:

- ✓ Los nombres de las historias de usuario se escriben en mayúscula.
- ✓ Los nombres de las historias de usuario deben ser claros y entendibles.
- ✓ Elaborar un prototipo de interfaz para cada historia de usuario.
- ✓ Se describirá cada historia de usuario usando un determinado formato que contiene los siguientes puntos:

- **Número:** Indica el número de historia de usuario.
- **Usuario:** Persona que solicita la historia de usuario.
- **Nombre:** Nombre de la historia de usuario.
- **Iteración asignada:** Número de la iteración que pertenece dicha historia de usuario.
- **Prioridad en negocio:** Importancia de la historia de usuario, categorizada en: Alta, media y baja.
- **Puntos estimados:** Estimación en número de la historia de usuario, se utiliza el siguiente intervalo: 1-10.
- **Descripción:** Breve descripción del requerimiento del usuario.
- **Observaciones:** Comentario adicional acerca de la historia de usuario.

HISTORIA DE USUARIO	
Número:	Nombre:
Usuario:	Iteración Asignada:
Prioridad del Negocio: (Alta/Media/Baja)	Puntos Estimados:
Riesgo de Desarrollo: (Alto/Medio/Bajo)	
Descripción:	
Observaciones:	

FUENTE: ELABORACIÓN PROPIA

4.2.1.2 ESTANDARES PARA LAS TAREAS DEL USUARIO

- Los nombres de las tareas de usuario se escriben en mayúscula la primera letra y las demás en minúsculas.
- Los nombres de la tarea de usuario deben ser claro y entendibles.
- Se describirá cada historia de usuario usando un determinado formato que contiene los siguientes puntos:
 - **Número de tarea:** indica el número de tarea.
 - **Historia de usuario (Número y nombre):** indica el número de historia de usuario a la que pertenece.
 - **Nombre de tarea:** nombre de la tarea de usuario.
 - **Tipo de tarea:** identificar la tarea categorizada en desarrollo, corrección, mejora, otra (especificar).
 - **Fecha de inicio:** fecha de inicio de desarrollo de la tarea de usuario.
 - **Fecha de fin:** fecha de fin de desarrollo de la tarea de usuario.
 - **Programador responsable:** persona encargada del desarrollo de la tarea.
 - **Descripción:** indica la funcionalidad de la tarea que va a ser desarrollada por el programador.

TAREA DE USUARIO	
Número de Tarea:	Historia de Usuario (Numero y Nombre):
Nombre de Tarea:	
Tipo de Tarea: Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio:	Fecha de Fin:
Programador Responsable:	
Descripción:	

FUENTE: ELABORACIÓN PROPIA

4.2.2 TORIAS DE USUARIO

4.2.2.1 ITERACIÓN I

HISTORIA DE USUARIO	
Número:01	Nombre: REGISTRO DE PISOS
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar los pisos dentro del area del hotel.	

HISTORIA DE USUARIO	
Número:02	Nombre: REGISTRO TIPO DE HABITACIÓN
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar el tipo de habitaciones existentes dentro del area del hotel.	

HISTORIA DE USUARIO	
Número:03	Nombre: REGISTRO DE MARCA
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción:	

El sistema permitirá registrar y modificar las marcas de auto existentes dentro del área del hotel.

HISTORIA DE USUARIO	
Número:04	Nombre: REGISTRO DE COLOR
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar colores, datos que servirán para el registro de autos.	

HISTORIA DE USUARIO	
Número:05	Nombre: REGISTRO DE VEHÍCULO
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar los autos de los huéspedes.	

HISTORIA DE USUARIO	
Número:06	Nombre: REGISTRO DE PROVEEDORES
Usuario: Administrador/Recepcionista	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar los datos de los proveedores.	

HISTORIA DE USUARIO	
Número:07	Nombre: REGISTRO DE CLIENTE
Usuario: Administrador/Recepcionista	Iteración Asignada: 2
Prioridad del Negocio: Alta	Puntos Estimados: 6/10

(Alta/Media/Baja)	
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar datos de los clientes del hotel.	

HISTORIA DE USUARIO	
Número:08	Nombre: REGISTRO DE PRODUCTOS
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar los datos de los productos de la empresa	

HISTORIA DE USUARIO	
Número:09	Nombre: REGISTRO DE HABITACIONES
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar habitaciones.	

HISTORIA DE USUARIO	
Número:10	Nombre: REGISTRO TALLAS
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar tallas para las prendas, para el servicio de lavandería.	

HISTORIA DE USUARIO	
Número:11	Nombre: REGISTRO TIPO DE VEHÍCULO

Usuario: Administrador	Iteración Asignada: 2	
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10	
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)		
Descripción: El sistema permitirá registrar y modificar los tipo de vehículo para el servicio del hotel.		

HISTORIA DE USUARIO		
Número:12	Nombre: REGISTRO TIPO DE PARQUEO	
Usuario: Administrador	Iteración Asignada: 2	
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10	
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)		
Descripción: El sistema permitirá registrar y modificar el tipo de Parqueo para el servicio de cochera.		

HISTORIA DE USUARIO		
Número:13	Nombre: REGISTRO DE UBICACIÓN DE VEHÍCULO	
Usuario: Administrador	Iteración Asignada: 2	
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10	
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)		
Descripción: El sistema permitirá registrar y modificar las ubicaciones de los vehículos.		

HISTORIA DE USUARIO		
Número:14	Nombre: REGISTRO TIPO DE BANCOS	
Usuario: Administrador	Iteración Asignada: 2	
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10	
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)		
Descripción: El sistema permitirá registrar y modificar el tipo de bancos, datos que se utilizarán para el registro de egresos e ingresos.		

HISTORIA DE USUARIO	
Número:15	Nombre: REGISTRO TIPO PAGO
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar los tipos de pagos, datos que servirán para el pagos de servicios.	

HISTORIA DE USUARIO	
Número:16	Nombre: REGISTRO DE AREAS
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar las areas de la empresa	

HISTORIA DE USUARIO	
Número:17	Nombre: REGISTRO DE EMPLEADOS
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar nuevos empleados y modificar sus datos.	

4.2.2.2 RACIÓN II

HISTORIA DE USUARIO	
Número:18	Nombre: REGISTRO DE USUARIOS AL SISTEMA
Usuario: Administrador	Iteración Asignada: 2

Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar datos de los nuevos usuarios.	

HISTORIA DE USUARIO	
Número: 19	Nombre: REGISTRO DE NUEVO PERFIL DEL USUARIO
Usuario: Administrador	Iteración Asignada:
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar perfiles de usuario del sistema.	

HISTORIA DE USUARIO	
Número: 20	Nombre: REGISTRO DE MENÚ
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar menús para el acceso a ciertos formularios dentro del hotel.	

HISTORIA DE USUARIO	
Número: 21	Nombre: REGISTRO DE OPERACIONES
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar y modificar operaciones que puede que los usuarios pueden realizan, Dependiendo del perfil que tengan.	

HISTORIA DE USUARIO	
Número:22	Nombre: ASIGNACIÓN DE PERFIL A USUARIO
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá asignar perfiles a los usuarios.	

HISTORIA DE USUARIO	
Número:23	Nombre: ASIGNACIÓN DE OPERACIONES POR PERFIL
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá asignar operaciones un determinado perfil de usuario.	

HISTORIA DE USUARIO	
Número:24	Nombre: CAMBIAR IGV
Usuario: Administrador	Iteración Asignada: 2
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá actualizar IGV para el cálculo de los comprobantes	

4.2.2.3 ITERACIÓN III

HISTORIA DE USUARIO	
Número:25	Nombre: RESERVACIÓN/ALQUILER
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10

Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá realizar los procesos de alquiler y reservación	

HISTORIA DE USUARIO	
Número:26	Nombre: SERVICIO COCHERA
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá realizar el registro de los servicios de cochera dentro del hotel.	

HISTORIA DE USUARIO	
Número:27	Nombre: CAMBIAR ESTADO DE HABTACIONES
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá realizar los cambios de estado de las habitaciones (ocupadas, desocupadas, por limpiar, etc.)	

HISTORIA DE USUARIO	
Número:28	Nombre: APERTURA CAJA
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá realizar la apertura de caja por usuario.	

HISTORIA DE USUARIO	
Número:29	Nombre: CIERRE CAJA
Usuario: Administrador	Iteración Asignada: 3

Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá realizar la apertura de caja por usuario.	

HISTORIA DE USUARIO	
Número: 30	Nombre: INGRESO CAJA
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar los ingresos de durante el turno de un empleado determinado.	

HISTORIA DE USUARIO	
Número: 31	Nombre: EGRESOS CAJA
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar los egresos durante un turno de un empleado determinado.	

HISTORIA DE USUARIO	
Número: 32	Nombre: CONCEPTO INGRESO Y SALIDA
Usuario: Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá registrar los tipos de conceptos tanto de ingresos como salidas.	

HISTORIA DE USUARIO	
Número: 33	Nombre: LISTADO

	DETALLADO CAJA
Usuario: Administrador/RECEPCIONISTA	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá Visualizar el listado de caja detallado	

HISTORIA DE USUARIO	
Número: 34	Nombre: LISTADO DE CIERRE DE CAJA REALIZADOS
Usuario: Recepcionista	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá Visualizar el listado de caja detallado	

HISTORIA DE USUARIO	
Número: 35	Nombre: RESUMEN DEL DÍA
Usuario: Recepcionista/Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá Visualizar el listado de caja detallado	

HISTORIA DE USUARIO	
Número: 36	Nombre: ANULAR SERVICIOS
Usuario: Recepcionista/Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá Visualizar los servicios generados	

HISTORIA DE USUARIO	
Número:37	Nombre: LISTADO DE SERVICIO ANULADO
Usuario: Recepcionista/Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá Visualizar los servicios servicios de consumos anulados.	

HISTORIA DE USUARIO	
Número:38	Nombre: LISTADO DE DOCUMENTOS GENERADOS
Usuario: Recepcionista/Administrador	Iteración Asignada: 3
Prioridad del Negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 6/10
Riesgo de Desarrollo: Medio (Alto/Medio/Bajo)	
Descripción: El sistema permitirá Visualizar los comprobantes emitidos con su detalle.	

4.2.3 LAN DE HISTORIAS DE USUARIO

IT.	NRO	NOMBRE DE HISTORIAS	INICIO	FIN
1	01	REGISTRO /MANTENIMIENTO PISOS	27/03/2015	31/03/2015
	02	REGISTRO / MANTENIMIENTO TIPO HABITACIÓN	01/04/2015	03/04/2015
	03	REGISTRO / MANTENIMIENTO MARCA	06/04/2015	08/04/2015
	04	REGISTRO / MANTENIMIENTO COLOR	09/04/2015	13/04/2015
	05	REGISTRO / MANTENIMIENTO VEHÍCULO	14/04/2015	16/04/2015
	06	REGISTRO / MATENIMIENTO PROVEEDOR	17/04/2015	21/04/2015
	07	REGISTRO / MANTENIMIENTO CLIENTE	22/04/2015	24/04/2015
	08	REGISTRO / MANTENIMIENTO PRODUCTOS	30/04/2015	01/05/2013
	09	REGISTRO / MANTENIMIENTO HABITACIÓN	04/05/2015	06/05/2015

	10	REGISTRO / MANTENIMIENTO TALLA	07/05/2015	11/05/2015
	11	REGISTRO / MANTENIMIENTO TIPO VEHÍCULO	12/05/2015	14/05/2015
	12	REGISTRO / MANTENIMIENTO TIPO PARQUEO	15/05/2015	15/05/2015
	13	REGISTRO / MANTENIMIENTO UBICACIÓN DE VEHÍCULO	18/05/2015	18/05/2015
	14	REGISTRO / MANTENIMIENTO DE BANCOS	19/05/2015	20/05/2015
	15	REGISTRO / MANTENIMIENTO TIPO DE PAGO	21/05/2015	21/05/2015
	16	REGISTRO / MANTENIMIENTO AREA	22/05/2015	22/05/2015
2	17	REGISTRO / MANTENIMIENTO EMPLEADO	25/05/2015	26/05/2015
	18	REGISTRO DE USUARIO AL SISTEMA	27/05/2015	01/06/2015
	19	REGISTRO DE NUEVO PERFIL DE USUARIO	02/06/2015	05/06/2015
	20	REGISTRO DE MENU	08/06/2015	12/06/2015
	21	REGISTRO DE OPERACIONES	15/06/2015	22/06/2015
	22	ASIGNACIÓN DE PERFIL A USUARIO	23/06/2015	30/06/2015
	23	ASIGNACIÓN DE OPERACIONES A PERFIL	01/07/2015	17/07/2015
3	24	CAMBIO DE IGV	20/07/2015	10/08/2015
	25	RESERVACIÓN / ALQUILER	11/08/2015	13/08/2015
	26	SERVICIO DE COCHERA	15/08/2015	17/08/2015
	27	CAMBIAR ESTADO DE HABITACIÓN	18/08/2015	21/08/2015
	28	APERTURA CAJA	24/08/2015	26/08/2015
	29	CIERRE CAJA	27/08/2015	28/08/2015
	30	INGRESO CAJA	31/08/2015	02/09/2015
	31	EGRESOS CAJA	03/09/2015	04/09/2015
	32	CONCEPTO INGRESO Y SALIDA	07/09/2015	08/09/2015
	33	LISTADO DETALLADO DE CAJA	09/09/2015	10/09/2015
	34	LISTADO DE CIERRE DE CAJA REALIZADOS	11/09/2015	15/09/2015
	35	RESUMEN DEL DIA	16/09/2015	18/09/2015
	36	ANULACIÓN DE SERVICIO	21/09/2015	24/09/2015
	37	LISTADO DE SERVICIOS ANULADOS	25/09/2015	30/09/2015
	38	LISTADO DE DOCUMENTOS GENERADO	01/10/2015	05/10/2015

4.3 DISEÑO DEL SISTEMA

4.3.1 INICIÓN DE ESTANDARES

4.3.1.1

ESTANDAR DE NOMENCLATURA PARA LA BASE

DE DATOS

TABLAS

- ✓ Los nombres de las tablas deben ser lo suficientemente descriptivos para identificar los datos que contiene, deben estar en singular e incluir caracteres de la A-Z, además de empezar con letra mayúscula. Ejemplo: Usuario, Piso, etc.

COLUMNAS

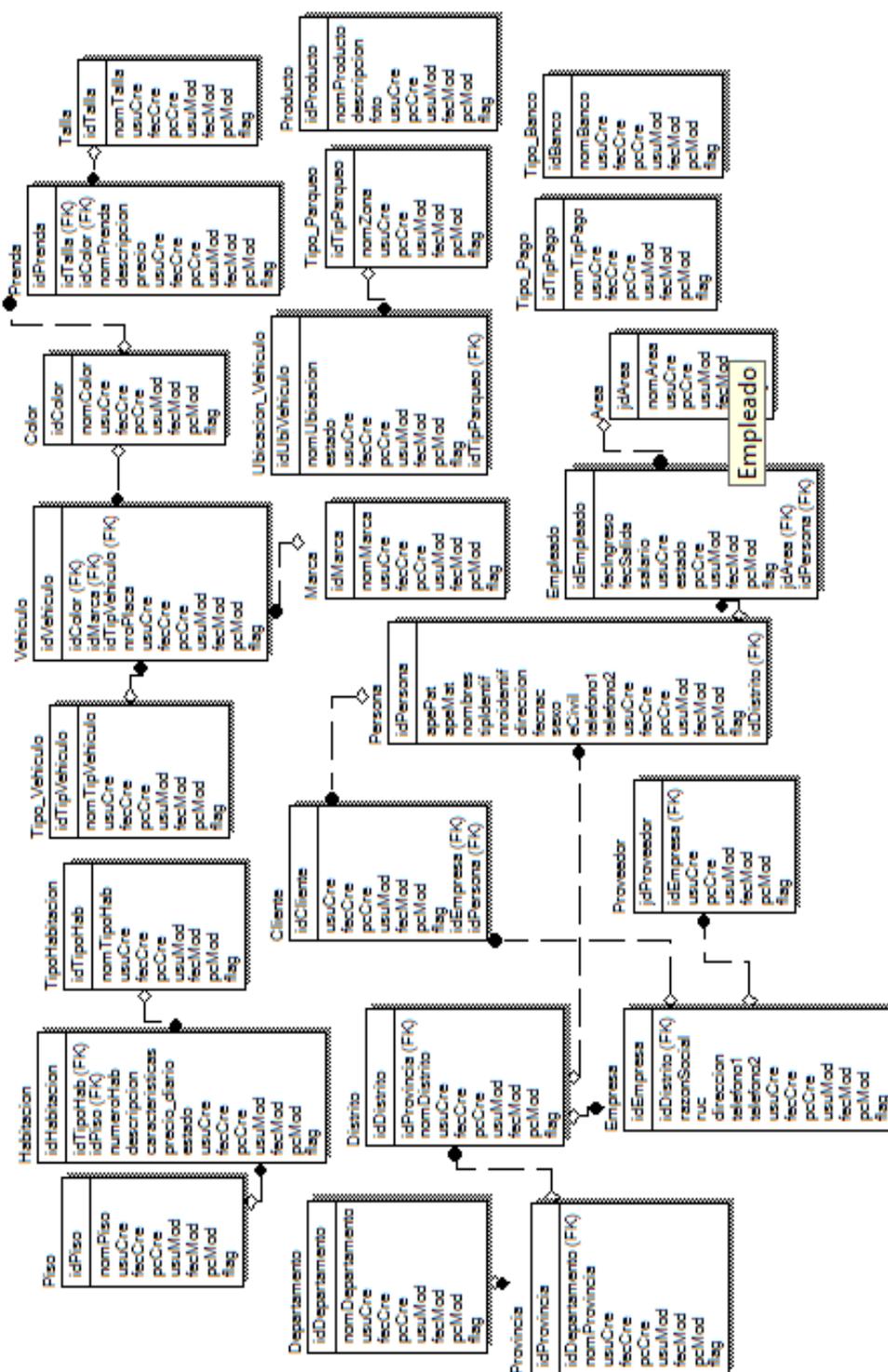
- ✓ No usar palabras reservadas de Transact-SQL.
- ✓ Los nombres de las columnas deben ser lo suficientemente descriptivos, pueden ser en singular o plural. Debe incluir caracteres de la A-Z.
- ✓ Deben tener un prefijo, separado con un subguión, que estará formado por los primeros tres caracteres del nombre de la tabla a la que pertenece. Ejemplo: exp_fechaRegistro, usu_nombre, etc.
- ✓ Si la columna es clave primaria se usará la siguiente estructura: id<Nombre de la tabla>. Ejemplo: idUsuario.

4.3.2 EÑO DE LA BASE DE DATOS

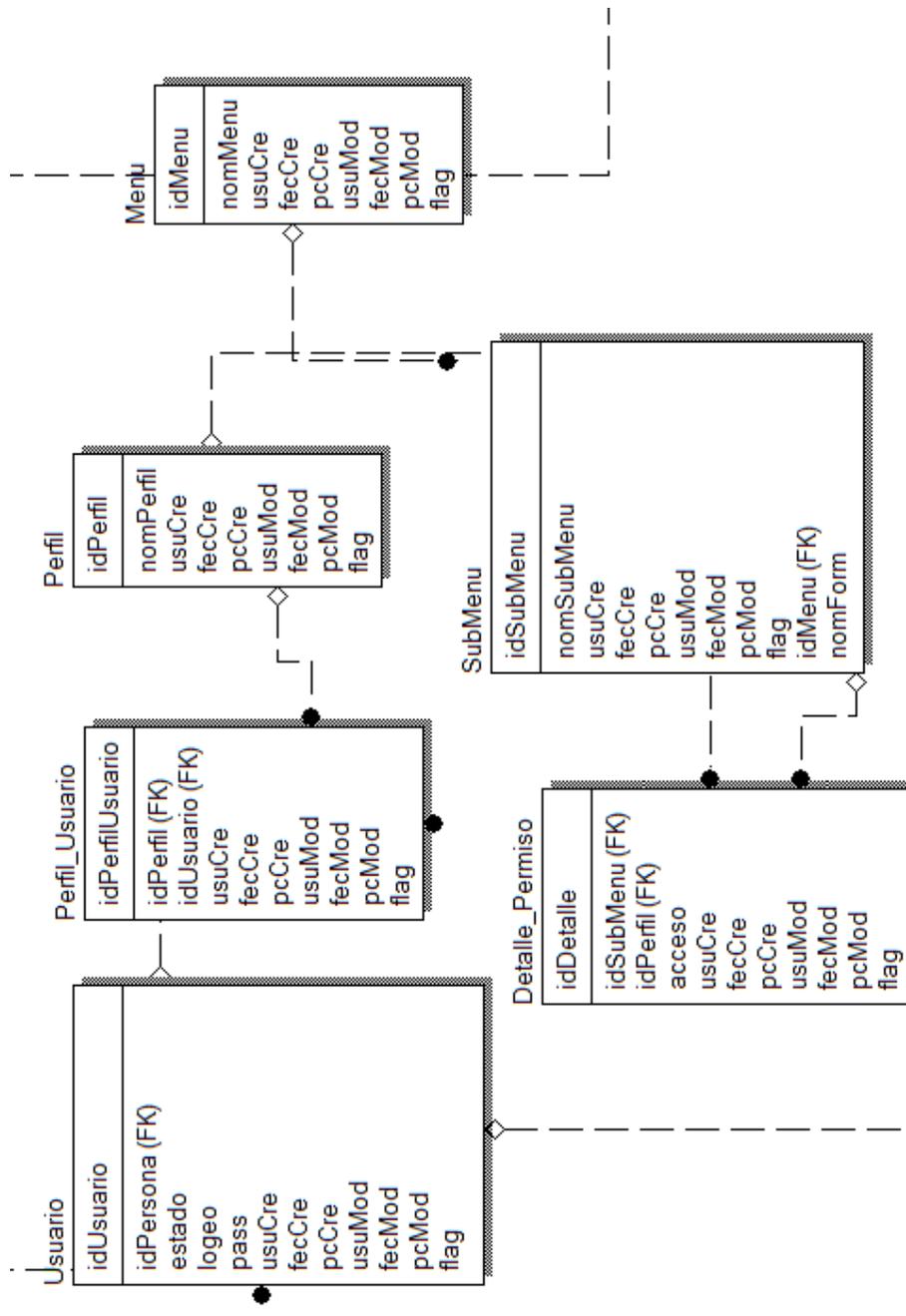
A continuación se muestran el diseño de las base de datos de las iteraciones.

4.3.2.1

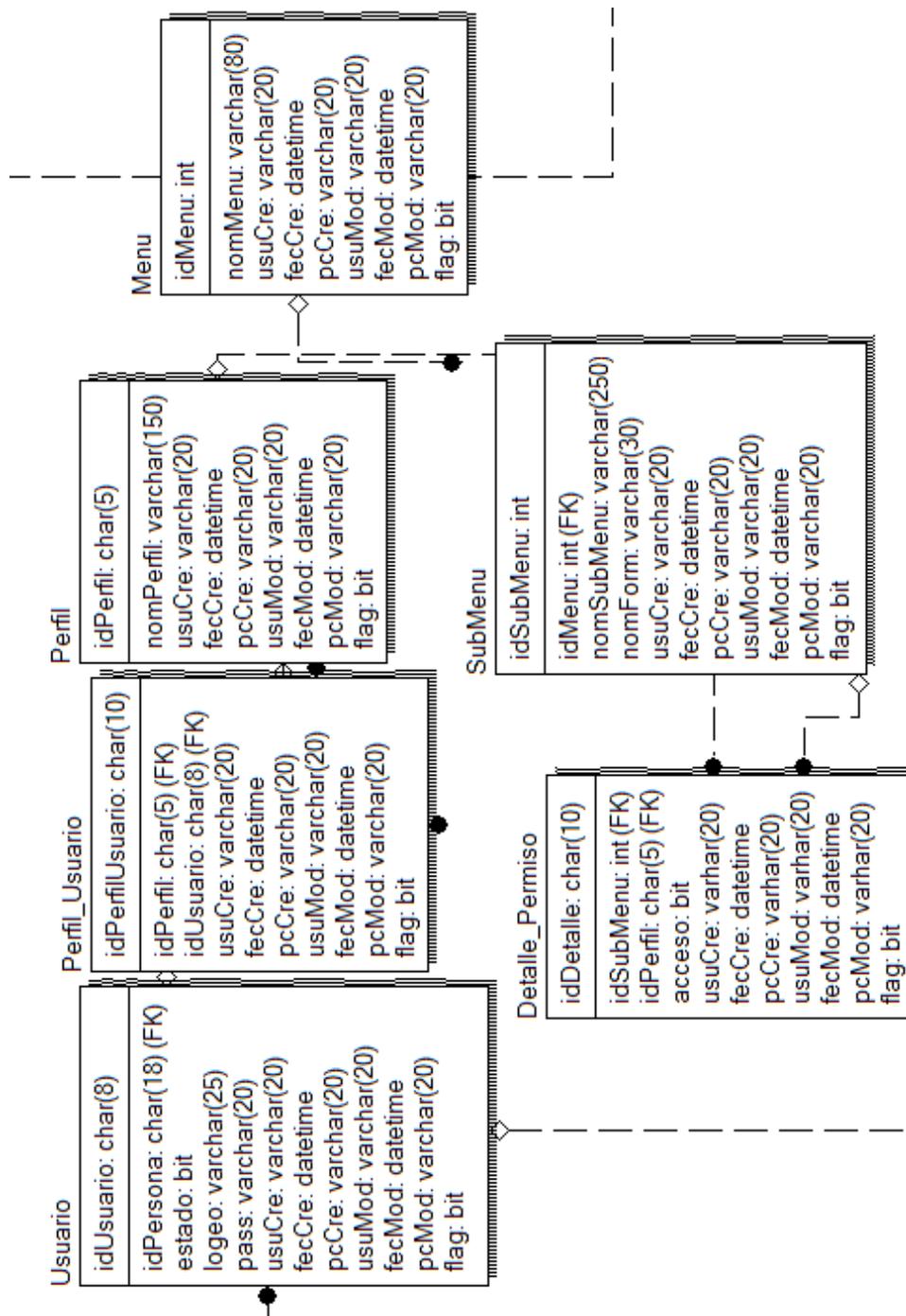
DISEÑO LÓGICO ITERACIÓN I



ITERACIÓN II



ITERACIÓN II



ITERACIÓN III

El diseño físico de la iteración III se muestra como anexo del presente documento.

4.3.2.3 DICCIONARIO DE DATOS

TABLAS DEL SISTEMA	
NOMBRE	DESCRIPCIÓN
Piso	Contiene los nombres de los números los pisos.
Habitación	Contiene los datos de los cuartos del hotel.
TipoHabitación	Contiene el nombre de los tipos de cuarto.
TipoVehículo	Contiene el nombre de los tipos de los vehículos.
Vehículo	Contiene los datos de los vehículos registrados.
Color	Nombre de los colores.
Marca	Contiene el nombre de los diferentes Marca de Vehículos.
Prenda	Contiene los datos de las prendas para el servicio de lavandería.
Talla	Contiene los nombres de las tallas para las prendas.
Nacionalidad	Contiene los nombre de los paises.
Departamento	Contiene los nombres de los departamentos del pais.
Provincia	Contiene los nombres de las Provincia.
Distrito	Contiene los nombres de los Distritos.
Persona	Contiene los datos de todos los clientes naturales.
Empresa	Contiene el registro de todos los clientes jurídicos.
Cliente	Contiene los códigos de los clientes naturales y jurídicos.
Tipo_Pago	Contiene los nombres de los tipos de pago.
Tipo_Banco	Contiene los nombres de las entidades financieras al momento de pagos al hotel.
UbicacionVehiculo	Contiene el nombres de los tipos de ubicación registro.
TipoParqueo	Contiene los nombres de los tipos de parqueo.
Usuario	Contiene los datos de los usuarios del sistema.

Perfil_Usuario	Contiene el registro de los usuarios con sus respectivos perfiles.
Perfil	Contiene el nombre de los perfiles de los usuarios.
Menu	Contiene los nombres de los menús del formulario principal.
Submenu	Contiene los nombres de las operaciones de los usuarios
Detalle_Permission	Contiene la lista de operaciones por perfil de usuario.
Reserva_Alquiler	Contiene el registro de los servicios de alquiler como el de reservación.
Cochera	Contiene el registro de los servicios prestados por cochera.
Detalle_Cochera	Contiene el detalle de los servicios de cochera.
Consumo	Contiene el registro de los consumos realizados por un alquiler.
Deudas	Contiene el registro de la Deuda de un determinado alquiler de habitación.
Detalle_Deudas	Contiene el detalle de las Deudas de un alquiler
Apertura_Cierre_Caja	Contiene los datos de apertura y Cierre de caja.
Egresos_caja	Contiene el registro de pagos se realizan durante un turno por empleado
Ingreso_caja	Contiene el registro de pagos que se hacen a la empresa por Deudas que tiene los clientes.
Concepto_IS	Contiene nombres de concepto que se hacen tanto en egresos y salidas.
Pago_Consumo	Contiene registro de los pagos al finalizar el alquiler de una habitación, incluyendo todos los servicios de lavandería y servicios de platillos.
Lavandería	Contiene el registro de los servicios por lavandería.
Huésped	Contiene el registro de las personas catalogadas como huéspedes.
Deudas_Cochera	Contiene el Registro de la Deudas por los servicios de cochera.
CambiosPrecios	Contiene registro de los cambios de precio de las habitaciones.
Proveedor	Contiene Registro de los proveedores de la empresa.
Producto	Contiene Registro de los productos.
Empleado	Contiene registro de los empleados de la empresa.

Contiene los registros de las áreas de la empresa.

COLUMNAS DE LA TABLA PISO

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idPiso	Char(6)	No null	Si	Si	Código de los pisos del hotel
nomPiso	tinyint	No null	No	No	Número de piso
usuCre	Varchar(20)	No null	No	No	Usuario que registro el Piso
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el piso
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA HABITACIÓN

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idHabitación	Char(6)	No null	Si	Si	Código de los pisos del hotel
idTipoHab	Char(5)	No null	Si	Si	Codigo del tipo de habitación
idPiso	Char(6)	No null	Si	Si	Codigo del número de piso
numHab	tinyint	No null	No	No	Número de habitación
descripción	Varchar(80)	No null	no	No	Descripción de la habitación
Características	Varchar(80)	No null	No	No	Características de la habitación
Precio_diario	Decimal (9,2)	No null	No	No	Precio de la habitación
Estado	Char(1)	No null	No	No	Estado de la habitación(M,L, etc)
usuCre	Varchar(20)	No null	No	No	Usuario que registro la habitación

fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro la habitación
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA TIPO HABITACIÓN					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idTipHab	Char(5)	No null	Si	Si	Código del tipo de habitación
nomTipoHab	Varchar (25)	No null	No	No	Nombre del tipo de habitación
usuCre	Varchar(20)	No null	No	No	Usuario que registro el Tipo de habitación
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el tipo de habitación
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA TIPO VEHÍCULO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idTipVehiculo	Char(10)	No null	Si	Si	Código del tipo de Vehículo
nomTipoHab	Varchar (30)	No null	No	No	Nombre del tipo de vehículo
usuCre	Varchar(20)	No null	No	No	Usuario que registro el Tipo de vehículo
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro

pcCre	Varchar(20)	No null	No	No	Estación que registro el tipo de vehículo
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA VEHÍCULO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idVehículo	Char(10)	No null	Si	Si	Código del vehículo
idMarca	Char(6)	No null	Si	Si	Codigo de la marca del vehículo
idColor	Char (6)	No null	Si	Si	Codigo del color de vehículo
idTipVehiculo	Char (10)	No null	Si	Si	Codigo del tipo de vehículo
nroPlaca	Varchar(10)	No null	No	No	Número de placa del vehículo
usuCre	Varchar(20)	No null	No	No	Usuario que registro el vehículo
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el vehículo
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA COLOR					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idColor	Char(10)	No null	Si	Si	Código del color
nomColor	Varchar(30)	No null	No	No	Nombre del color
usuCre	Varchar(20)	No null	No	No	Usuario que registro el color

fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el color
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA MARCA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idMarca	Char(6)	No null	Si	Si	Codigo de la Marca
nomMarca	Varchar (20)	No null	No	No	Nombre de la marca
usuCre	Varchar(20)	No null	No	No	Usuario que registro la marca
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro la marca
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PRENDA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idPrenda	Char(10)	No null	Si	Si	Código de la prenda
idTalla	Char (6)	No null	Si	Si	Codigo de la talla
idColor	Char (6)	No null	Si	Si	Codigo del color
nomPrenda	Varchar(50)	No null	No	No	Nombre de la prenda
Descripción	Varchar (50)	No null	No	No	Descripción de la prenda
Precio	Decimal(9,2)	No null	No	No	Precio de la prenda

usuCre	Varchar(20)	No null	No	No	Usuario que registro de la prenda
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro la prenda
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA TALLA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idTalla	Char(6)	No null	Si	Si	Codigo de la talla
nomTalla	Varchar (20)	No null	No	No	Nombre de la tala
usuCre	Varchar(20)	No null	No	No	Usuario que registro el talla
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro la talla
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA NACIONALIDAD					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idNacionalidad	Char(6)	No null	Si	Si	Codigo de la Nacionalidad
nomNacionalidad	Varchar (20)	No null	No	No	Nombre de la nacionalidad
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA DEPARTAMENTO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDepartamento	Char(10)	No null	Si	Si	Codigo del departamento
idNacionalidad	Char (6)	No null	Si	Si	Codigo de la nacionalidad
nomDepartamento	Varchar (20)	No null	No	No	Nombre del departamento
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PROVINCIA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idProvincia	Char(10)	No null	Si	Si	Codigo de la provincia
idDepartamento	Char (10)	No null	Si	Si	Codigo del departamento
nomProvincia	Varchar (20)	No null	No	No	Nombre de la provincia
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA DISTRITO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDistrito	Char(10)	No null	Si	Si	Codigo del distrito
idProvincia	Char (10)	No null	Si	Si	Codigo del distrito
nomDistrito	Varchar (20)	No null	No	No	Nombre del distrito
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PERSONA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idPersona	Char(10)	No null	Si	Si	Codigo de la persona
idDistrito	Char(10)	No null	Si	Si	Codigo del distrito
apePat	Varchar (30)	No null	No	No	Apellido paterno
apeMat	Varchar (30)	No null	No	No	Apellido materno
Nombres	Varchar(50)	No null	No	No	Nombres de la persona
TiplIdentif	Char(1)	No null	No	No	Tipo de identificación
nroIdentif	Varchar(15)	No null	No	No	Número de identificación

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

Dirección	Varchar (50)	No null	No	No	Dirección de la persona
fecNac	Date	No null	No	No	Fecha de nacimiento
Sexo	Char(1)	No null	No	No	Sexo de la persona
eCivil	Char(1)	No null	No	No	Estado civil
Telefono1	Varchar(11)	No null	No	No	Teléfono 1
Telefono2	Varchar(11)	No null	No	No	Telefono2
usuCre	Varchar(20)	No null	No	No	Usuario que registro la persona
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro la persona
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA EMPRESA

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idEmpresa	Char(10)	No null	Si	Si	Codigo de la empresa
idDistrito	Char (10)	No null	Si	Si	Codigo del distrito
razonSocial	Varchar(80)	No null	No	No	Razón social de la empresa
Ruc	Char(11)	No null	No	No	Identificación de la empresa
Dirección	Varchar(80)	No null	No	No	Dirección de la empresa
Telefono1	Varchar(11)	No null	No	No	Teléfono 1
Telefono2	Varchar(11)	No null	No	No	Telefono2
usuCre	Varchar(20)	No null	No	No	Usuario que registro la empresa
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro la empresa
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación

pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA CLIENTE					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idCliente	Char(10)	No null	Si	Si	Codigo del cliente
idPersona	Char (10)	No null	Si	Si	Codigo de la persona
idEmpresa	Char(10)	No null	Si	Si	Codigo de la empresa
usuCre	Varchar(20)	No null	No	No	Usuario que registro al cliente
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro al cliente
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA TIPO PAGO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idTipPago	Char(5)	No null	Si	Si	Codigo del tipo de pago
nomTipPago	varchar(20)	No null	Si	Si	Nombre del tipo de pago
usuCre	Varchar(20)	No null	No	No	Usuario que registro el tipo de pago
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el tipo pago
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación

pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA TIPO BANCO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idBanco	Char(10)	No null	Si	Si	Codigo del banco
nomBanco	Char (10)	No null	Si	Si	Nombre del banco
usuCre	Varchar(20)	No null	No	No	Usuario que registro al banco
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro al banco
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA UBICACIÓN VEHICULO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idUbiVehiculo	Char(10)	No null	Si	Si	Codigo de la ubicación
idTipParqueo	Char (10)	No null	Si	Si	Codigo tipo parqueo
nomUbicación	Varchar(80)	No null	Si	Si	Nombre de la ubicación
usuCre	Varchar(20)	No null	No	No	Usuario que registro la ubicación
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro

Flag	bit	No null	No	No	Estado de la tabla
------	-----	---------	----	----	--------------------

COLUMNAS DE LA TABLA TIPO PARQUEO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idTipParqueo	Char(10)	No null	Si	Si	Codigo del tipo de parqueo
nomZona	Varchar(25)	No null	Si	Si	Nombre de la zona
usuCre	Varchar(20)	No null	No	No	Usuario que registro del tipo de parqueo
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el tipo de parqueo
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA USUARIO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idUsuario	Char(8)	No null	Si	Si	Codigo del usuario
idPersona	Char(10)	No null	Si	Si	Codigo de la persona
Estado	Bit	No null	No	No	Estado del usuario
Logeo	Varchar(25)	No null	No	No	Login del usuario
Pass	Varchar(20)	No null	No	No	Contraseñas del usuario
usuCre	Varchar(20)	No null	No	No	Usuario que registro al usuario
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que registro el usuario
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación

pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PERFIL USUARIO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idPerfilUsuario	Char(10)	No null	Si	Si	Codigo de perfil por usuario
idPerfil	Char(10)	No null	Si	Si	Codigo del perfil
idUsuario	Char(5)	No null	Si	Si	Codigo del usuario
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PERFIL					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idPerfil	Char(5)	No null	Si	Si	Codigo de perfil
nomPerfil	Varchar(15)	No null	No	No	Nombre del perfil
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA MENU					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idMenu	int	No null	Si	Si	Codigo del menu
nomMenu	Varchar(80)	No null	No	No	Nombre del menu
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA SUB MENU					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idSubMenu	int	No null	Si	Si	Codigo del submenu
idMenu	int	No null	No	No	Codigo del menu
nomSubMenu	Varchar(80)	No null	No	No	Nombre del submenu
nomForm	Varchar(20)	No null	No	No	Nombre del formulario
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro

Flag	bit	No null	No	No	Estado de la tabla
------	-----	---------	----	----	--------------------

COLUMNAS DE LA TABLA DETALLE PERMISO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDetallePermiso	Char(10)	No null	Si	Si	Codigo del detalle permiso
idPerfil	Char(5)	No null	No	No	Codigo del perfil
idSubMenu	Int	No null	No	No	Codigo del subMenu
acceso	bit	No null	No	No	Estado del permiso
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifiko el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifiko el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA RESERVA_ALQUILER					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idReservaAlquiler	Char(10)	No null	Si	Si	Codigo del detalle reserva o alquiler
idHabitación	Char(8)	No null	Si	Si	Codigo de la habitación
idCliente	Char(10)	No null	Si	Si	Codigo del cliente
tipoReservaAlquiler	Char(1)	No null	No	No	Tipo de servicio alquiler o reserva
fechaReservaAlquiler	date	No null	No	No	Fecha del servicio
fechaIngreso	date	No null	No	No	Fecha de ingreso
fechaSalida	date	No null	No	No	Fecha de salida
CostoAlojamiento	Decimal(9,2)	No null	No	No	Fecha del alojamiento
observación	Varchar(200)	No null	No	No	Comentarios del alojamiento
flgHuesped	bit	No null	No	No	Estado el huésped

estadoRA	Char(1)	No null	No	No	Estado Reserva o alquiler
tipoDoc	Char(1)	No null	No	No	Tipo comprobante de pago
nroDocumento	Varchar(12)	No null	No	No	Número de comprobante
montolg	Decimal(4,2)	No null	No	No	Monto del Igv establecido para calcular
subTotal	Decimal(4,2)	No null	No	No	Subtotal del monto
Servicio	Decimal(4,2)	No null	No	No	Monto por los servicio
igv	Decimal(4,2)	No null	No	No	Igv
totalPagar	Decimal(4,2)	No null	No	No	Total a pagar
tipoPago	Char(2)	No null	No	No	Tipo de pago credito o contado
Acuenta	Decimal(4,2)	No null	No	No	Monto a cuenta
saldo	Decimal(4,2)	No null	No	No	Saldo total de la cuenta
estadoPago	Char(1)	No null	No	No	Estado del pago
tPago	Char(1)	No null	No	No	
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA COCHERA

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idCochera	Char(10)	No null	Si	Si	Codigo del servicio de cochera
idCliente	Char(10)	No null	Si	Si	Codigo del cliente
fechaIngreso	date	No null	No	No	Fecha de ingreso
fechaSalida	date	No null	No	No	Fecha de salida

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

CostoAlojamiento	Decimal(9,2)	No null	No	No	Fecha del alojamiento
observación	Varchar(200)	No null	No	No	Comentarios del alojamiento
estadoC	Char(1)	No null	No	No	Estado Reserva o alquiler
tipoDoc	Char(1)	No null	No	No	Tipo comprobante de pago
nroDocumento	Varchar(12)	No null	No	No	Número de comprobante
montolg	Decimal(4,2)	No null	No	No	Monto del Igv establecido para calcular
subTotal	Decimal(4,2)	No null	No	No	Subtotal del monto
Servicio	Decimal(4,2)	No null	No	No	Monto por los servicio
igv	Decimal(4,2)	No null	No	No	Igv
totalPagar	Decimal(4,2)	No null	No	No	Total a pagar
tipoPago	Char(2)	No null	No	No	Tipo de pago credito o contado
Acuenta	Decimal(4,2)	No null	No	No	Monto a cuenta
saldo	Decimal(4,2)	No null	No	No	Saldo total de la cuenta
estadoPago	Char(1)	No null	No	No	Estado del pago
tPago	Char(1)	No null	No	No	
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA DETALLE_COCHERA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDetCochera	Char(10)	No null	Si	Si	Codigo del detalle

idCochera	Char(10)	No null	Si	Si	Codigo de la cochera
idVehículo	Char(10)	No null	Si	Si	Codigo del vehículo
idUbiVehiculo	Char(10)	No null	Si	Si	Codigo de la ubicación del vehículo
descripción	Varchar(50)	No null	No	No	Comentarios del servicio
Precio	Decimal(4,2)	No null	No	No	Precio por el servicio
estado	Char(1)	No null	No	No	Estado del detalle
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA CONSUMO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idConsumo	Char(10)	No null	Si	Si	Codigo del consumo
idReservaAlquiler	Char(10)	No null	Si	Si	Codigo de la reservaAlquiler
idProducto	Char(10)	No null	Si	Si	Codigo del producto
cantidad	Decimal(4,2)	No null	No	No	Cantidad del producto
PrecioVenta	Decimal(4,2)	No null	No	No	Precio del producto
estado	Char(1)	No null	No	No	Estado de consumo
imp	Char(1)	No null	No	No	
bloq	Char(1)	No null	No	No	
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro

fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA DEUDAS

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDeuda	Char(10)	No null	Si	Si	Codigo de la deuda
idReservaAlquiler	Char(10)	No null	Si	Si	Codigo de la reservaAlquiler
Saldo	decimal(4,2)	No null	Si	Si	Saldo de la deuda
estado	char(1)	No null	No	No	Estado de la deuda
tPago	Decimal(4,2)	No null	No	No	
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA DETALLE_DEUDA

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDetalleDeuda	Char(10)	No null	Si	Si	Codigo del detalle deuda

idDeuda	Char(10)	No null	Si	Si	Codigo de la deuda
tipoDoc	Char(1)	No null	No	No	Tipo comprobante de pago
nroDocumento	Varchar(12)	No null	No	No	Número de comprobante
montolg	Decimal(4,2)	No null	No	No	Monto del Igv establecido para calcular
subTotal	Decimal(4,2)	No null	No	No	Subtotal del monto
igv	Decimal(4,2)	No null	No	No	Igv
Amortización	Decimal(4,2)	No null	No	No	Monto de amortización
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA APERTURA _CIERRE_ CAJA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idAperturaCierre	Char(15)	No null	Si	Si	Codigo de la apertura/cierre
idUsuario	Char(8)	No null	Si	Si	Codigo del usuario
idUsuEntregado	Char(8)	No null	No	No	Usuario a quién se le entrega caja
monto	Decimal(4,2)	No null	No	No	
descripción	Nvarchar(50)	No null	No	No	Monto del Igv establecido para calcular
saldo	Decimal(4,2)	No null	No	No	Monto del saldo
descripSaldo	Nvarchar(50)	No null	No	No	Descripción del saldo

totEfectivoServicio	Decimal(4,2)	No null	No	No	Monto de total de efectivos por servicios comestibles
totalIngreso	Decimal(4,2)	No null	No	No	Total de dinero ingresado por pagos a empresa
totResCochera	Decimal(4,2)	No null	No	No	Total de servicio de cochera
totDinero	Decimal(4,2)	No null	No	No	Total dinero
totEgreso	Decimal(4,2)	No null	No	No	Total egreso
totTarjetaBanco	Decimal(4,2)	No null	No	No	Total de pagos con banco
totDescDinero	Decimal(4,2)	No null	No	No	Total dinero descuento
totCajaEfectivo	Decimal(4,2)	No null	No	No	Total caja efectivo
totCajaEntregado	Decimal(4,2)	No null	No	No	Total de caja entregado
Estado	bit	No null	No	No	Estado del apertura/cierre caja
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA EGRESOS_CAJA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idEgreso	Char(10)	No null	Si	Si	Codigo del egreso
idCliente	Char(10)	No null	Si	Si	Codigo del cliente
idConcepto	Char(10)	No null	Si	Si	Codigo del concepto de egreso
IdPago	Decimal(4,2)	No null	Si	Si	

numRecibo	Char(10)	No null	Si	Si	Número de recibo de egreso
descripción	Nvarchar(50)	No null	No	No	Monto del Igv establecido para calcular
total	Decimal(4,2)	No null	No	No	Monto del saldo
Estado	bit	No null	No	No	Estado del apertura/cierre caja
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA INGRESO_CAJA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idIngreso	Char(10)	No null	Si	Si	Codigo del ingreso de caja
idCliente	Char(10)	No null	Si	Si	Codigo del cliente
idConcepto	Char(10)	No null	Si	Si	Codigo del concepto de egreso
IdBanco	Char(8)	No null	Si	Si	Codigo del banco
idDeuda	Char(10)	No null	Si	Si	Codigo de la deuda
tipoDoc	Char(1)	No null	No	No	Tipo documento de
nroDocumento	Varchar(12)	No null	No	No	Número del documento
nOp	Char(10)	No null	No	No	Número de operación
monto	Decimal(4,2)	No null	No	No	Monto del saldo
montolgv	Decimal(4,2)	No null	No	No	Monto del igv
subTotal	Decimal(4,2)	No null	No	No	Sub total
comisión	Decimal(4,2)	No null	No	No	Dinero o interés

igv	Decimal(4,2)	No null	No	No	Impuesto
total	Decimal(4,2)	No null	No	No	Total dinero
tPago	Decimal(4,2)	No null	No	No	Total del ingreso
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA CONCEPTO_IS					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idConcepto	Char(10)	No null	Si	Si	Codigo del concepto
concepto	Char(10)	No null	No	No	Descripción del concepto
tipo	Char(1)	No null	No	No	Tipo de concepto
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PAGO CONSUMO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idPagoConsumo	Char(10)	No null	Si	Si	Codigo del pago por el consumo
idReservaAlquiler	Char(10)	No null	Si	Si	Codigo de la reserva alquiler
idPago	Char(5)	No null	Si	Si	Codigo del Pago
IdBanco	Char(8)	No null	Si	Si	Codigo del banco
idDeuda	Char(10)	No null	Si	Si	Codigo de la deuda
tipoDoc	Char(1)	No null	No	No	Tipo de documento
fachaRegis	date	No null	No	No	Fecha del registro
fachaRegis	date	No null	No	No	Fecha de pago
nOp	Char(10)	No null	No	No	Número de operación
monto	Decimal(4,2)	No null	No	No	Monto del saldo
comisión	Decimal(4,2)	No null	No	No	Dinero o interés
nroDocumento	Varchar(12)	No null	No	No	Número de documento
montIgv	Decimal(4,2)	No null	No	No	Monto del igv
totReservaAlquiler	Decimal(4,2)	No null	No	No	Total del servicio
totConsumo	Decimal(4,2)	No null	No	No	Total del consumo
totLavanderia	Decimal(4,2)	No null	No	No	Total del servicio por lavandería
Igv	Decimal(4,2)	No null	No	No	
total	Decimal(4,2)	No null	No	No	Total dinero
tPago	Decimal(4,2)	No null	No	No	Total del ingreso
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA LAVANDERIA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idLavanderia	Char(10)	No null	Si	Si	Codigo de lavandería
idReserva	Char(10)	No null	Si	Si	Codigo de la reserva
idPrenda	Char(10)	No null	Si	Si	Codigo de la prenda
Cantidad	Decimal(4,2)	No null	Si	Si	Cantidad
PrecioVenta	Decimal(4,2)	No null	Si	Si	Precio
Estado	Char(1)	No null	Si	Si	Estado del registro
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA HUESPED					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idHuesped	Char(10)	No null	Si	Si	Codigo del huésped
idReserva	Char(10)	No null	Si	Si	Codigo de la reserva
idPersona	Char(10)	No null	Si	Si	Codigo de la persona
idVehículo	Char(10)	null	Si	Si	Codigo del vehiculo
descripción	Varchar(25)	No null	No	No	descripción
Estado	Char(1)	No null	No	No	Estado del registro
idUbiVehiculo	char(10)	No null	No	No	Codigo del ubicación del vehiculo

usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA DEUDAS_COCHERA

Nombre	Tipo Dato	NULL	PK	FK	Descripción
idDeudaCochera	Char(10)	No null	Si	Si	Codigo de la deuda por cochera
idCochera	Char(10)	No null	Si	Si	Codigo de la tabla cochera
Saldo	Decimal(4,2)	No null	No	No	Saldo del servicio
Estado	Char(1)	No null	No	No	Estado del registro
tPago	Char(2)	No null	No	No	Total pago
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA CAMBIOS PRECIOS					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idCambioPrecio	Char(10)	No null	Si	Si	Codigo del registro
idHabitación	Char(8)	No null	Si	Si	Codigo de la habitacion
precioActual	Decimal(4,2)	No null	No	No	Precio actual de la habitación
precioNuevo	Decimal(4,2)	No null	No	No	Nuevo precio
motivo	Varchar(100)	No null	No	No	Motivo del cambio del precio
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PROVEEDOR					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idProveedor	Char(10)	No null	Si	Si	Codigo del proveedor
idEmpresa	Char(10)	No null	Si	Si	Codigo de la empresa
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación

pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA PROVEEDOR					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idProveedor	Char(10)	No null	Si	Si	Codigo del proveedor
idEmpresa	Char(10)	No null	Si	Si	Codigo de la empresa
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

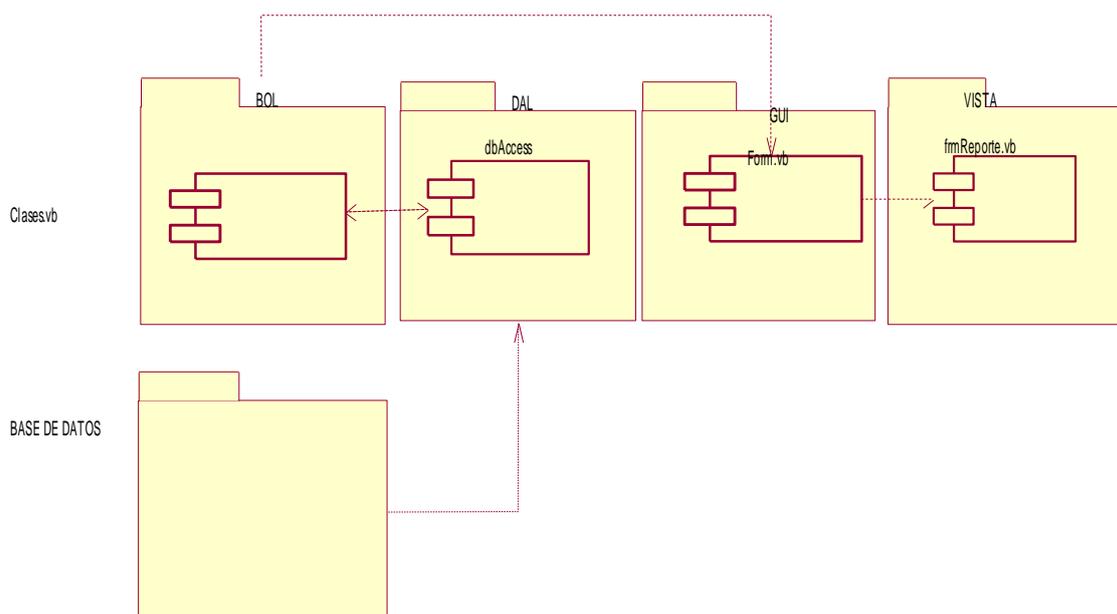
COLUMNAS DE LA TABLA PRODUCTO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idProducto	Char(10)	No null	Si	Si	Codigo del producto
nomProducto	Varchar(30)	No null	No	No	Nombre del producto
Descripcion	Varchar(50)	No null	No	No	Nombre del producto
Foto	Image	No null	No	No	Imagen del producto
Precio	Decimal(4,2)	No null	No	No	Precio del producto
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro

pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA EMPLEADO					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idEmpleado	Char(10)	No null	Si	Si	Codigo del producto
idPersona	Char(10)	No null	Si	Si	Codigo de la tabla persona
idArea	Char(10)	No null	Si	Si	Codigo del area
idPrefil	Char(5)	No null	Si	Si	Codigo del perfil
fechIngreso	date	No null	No	No	Fecha ingreso
fechaSalida	Date	No null	No	No	Fecha de salida
Salario	Decimal(4,2)	No null	No	No	Salario del empleado
Estado	Char(1)	No null	No	No	Estado del registro
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

COLUMNAS DE LA TABLA AREA					
Nombre	Tipo Dato	NULL	PK	FK	Descripción
idArea	Char(10)	No null	Si	Si	Codigo del area
nomArea	Char(10)	No null	Si	Si	Nombre del area
usuCre	Varchar(20)	No null	No	No	Usuario que realizó el registro
fecCre	Datetime	No null	No	No	Fecha que se realizó el registro
pcCre	Varchar(20)	No null	No	No	Estación que realizó el registro
usuMod	Varchar (20)	No null	No	No	Usuario que modifico el registro
fecMod	Datetime	No null	No	No	Fecha de ultima modificación
pcMod	Varchar(20)	No null	No	No	Estación de donde se modifico el registro
Flag	bit	No null	No	No	Estado de la tabla

4.3.3 IAGRAMA DE COMPONENTES



✓ **BOL CLASES.VB**

Clases de objetos del negocio definidas por el programador para ser reutilizadas, haciendo uso de sus atributos y métodos.

✓ **DAL DBACCESS**

Clase definida por el programador que contienen información de cadena de conexión e interfaces para consultas con la base de datos.

✓ **GUI FORM.VB**

Interfaz gráfica del usuario, muestra la información comprensible para el usuario, de forma que pueda interactuar con ella.

✓ **VISTA FRM REPORTE.VB**

Para la creación de reportes se uso crystal report view, aqui se muestran los reportes que servirán para la toma de decisiones de Alta dirección de la empresa.

✓ **BASE DE DATOS**

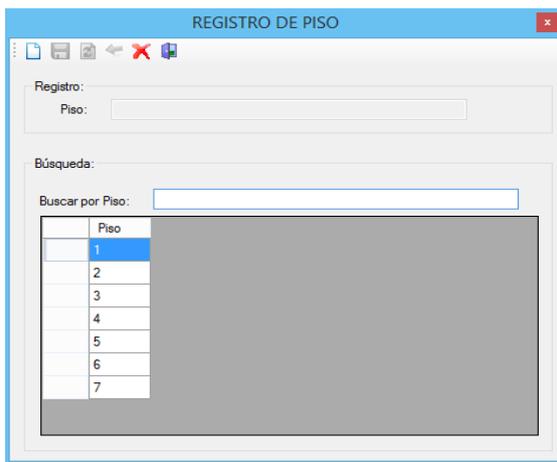
Es el componente que se encarga de almacenar sistemáticamente los datos.

4.4 CONSTRUCCIÓN DEL SISTEMA

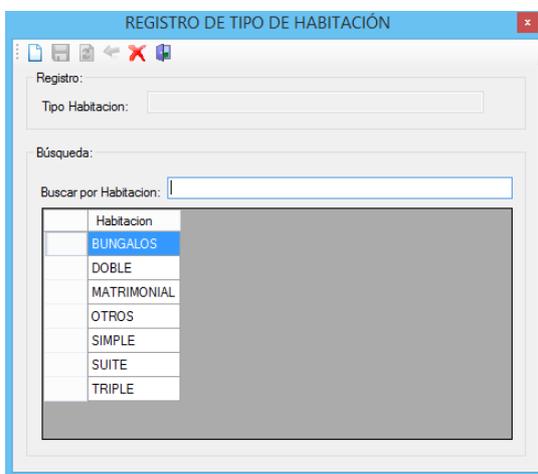
4.4.1 INCIÓN DE ITERACIONES

4.4.1.1 ITERACIÓN I

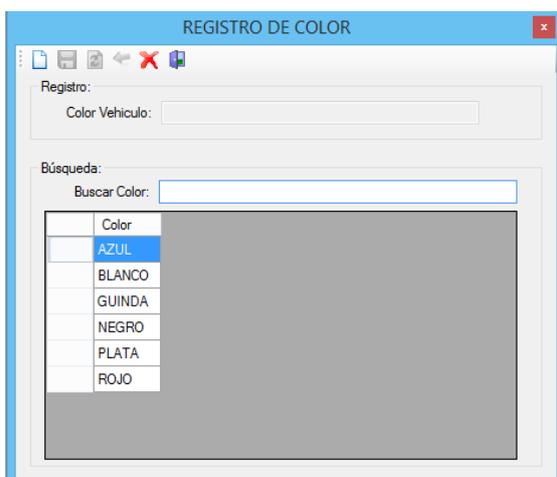
SPIKES PARA LAS HISTORIAS DEL USUARIO



Spike de mantenimiento de pisos



Spike de mantenimiento de habitación



Spike de mantenimiento de color

REGISTRO DE VEHÍCULO

Registro:

Nro Placa:

Tipo Vehículo: AUTOMOVIL

Marca: HYUNDAI Color: AZUL

Búsqueda:

Nro Placa:

	Nro Placa	Marca	Color	Tipo Vehículo
▶	PCL-523	HYUNDAI	NEGRO	CAMION
	PCL-159	HYUNDAI	AZUL	AUTOMOVIL
	HJR-021	HYUNDAI	AZUL	AUTOMOVIL
	PRI-254	HYUNDAI	AZUL	AUTOMOVIL
	PEL-128	HYUNDAI	AZUL	AUTOMOVIL
	PRL-122	HYUNDAI	AZUL	AUTOMOVIL
	PRL-235	NISSAN	BLANCO	CAMIONETA

Spike de mantenimiento de vehículo

REGISTRO DE PROVEEDOR

Registro:

Razón Social: RUC:

Dirección:

Teléfono: Celular: Departamento: AMAZONAS

Provincia: BAGUA Distrito: ARAMANGO

Búsqueda:

Buscar por Razón Social o RUC:

	Razón Social	RUC	Dirección	Teléfono 1	Teléfono 2	Departame
	ALICORP SA	12345678888	CALLE GRAU NRO 250	999999999		ICA
	GRUPO GLORIA SA	20100190797	REPUBLICA DE PANAMA - LIMA	95632147		LIMA

Spike de mantenimiento de Proveedor

**SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.**

REGISTRO DE CLIENTES PERSONA

Registro:

A. Paterno: A. Materno:

Nombres: Identificación: DNI Pasaporte Camé extranjera

Dirección: Fecha Nacimiento:

Estado Civil: Sexo: M F Celular: Teléfono:

Nacionalidad: PERU Departamento: AMAZONAS

Provincia: BAGUA Distrito: ARAMANGO

Búsqueda:

Buscar por Apellidos y Nombres o Identificación:

	A. Paterno	A. Materno	Nombres	TipIdentif	Tipo Identificación	Nro. Identificación	Dirección
	VILLA	PEREZ	CESAR IVAN	D	DNI	46184593	PROLOG MARISCAL BENA
	CASTILLON	SIGUAS	LUIS	D	DNI	45217896	AV. UNION NRO 363
	HUAYLLA	CABRERA	KATY	D	DNI	12345678	SAN AGUSTIN MZ T LT 8
	CARTAGENA	VILCHEZ	JOSE MIGUEL	D	DNI	22364295	CALLE LIMA N° 20 MZ "L"
	BOLAÑOS	BARRERA	LUCAS MANUEL	D	DNI	21563489	URB. LAS MAGNOLIAS N°
	PEREZ	CASTILLA	LUCAS LUIS	D	DNI	34567890	CALLE MIGUEL CARRILLO

Spike mantenimiento de los Clientes Naturales

REGISTRO DE CLIENTES EMPRESA

Registro:

Razón Social: RUC:

Dirección:

Celular: Teléfono:

Nacionalidad: PERU Departamento: AMAZONAS

Provincia: BAGUA Distrito: ARAMANGO

Búsqueda:

Buscar por Razón Social o RUC:

	Razón Social	RUC	Dirección	Teléfono 1	Teléfono 2
	COMPLEJO AGROINDUSTRIAL BETA SA	20297939131	CALLE LEOPOLDO CARRILLO NRO 160	956321478	
	ALICORP SA	12345678888	CALLE GRAU NRO 250	999999999	

Spike manteniendo de los Clientes Jurídicos

REGISTRO DE PRODUCTOS

Registro:

Producto:

Precio: Descripción:

Ruta Foto:  

Búsqueda:

Buscar producto:

Producto	Descripción	Precio	Foto
GASEOSA COCA COLA 1 LT		2.500	
DESAYUNO		7.500	
PIÑA COLADA	REALIZADO EN BASE A PIÑA Y HIELO CON JARABE DE GOMA	8.000	

Spike mantenimiento de Productos

REGISTRO DE HABITACIÓN

Registro:

N° Habitación: Tipo Habitación:

Piso: Precio Diario:

Descripción: Característica:

Estado: Disponible En Mantenimiento En Limpieza

Búsqueda:

Buscar por N° Habitación:

Nro. Habitación	Tipo Habitación	Nro. Piso	Precio Diario	Característica	Descripción
1	SIMPLE	1	60.00		AGUA FRIA Y CALIE
1	DOBLE	1	80.00		DUCHA PROPIA Y E
2	SIMPLE	1	60.00		AGUA FRIA + CALIE
3	DOBLE	1	80.00		AGUA FRIA + CALIE
4	SUITE	3	120.00		

Spike mantenimiento de Habitación

REGISTRO DE PRENDA

Registro:

Prenda:

Talla: Color: Precio:

Descripción:

Búsqueda:

Buscar Prenda:

	Prenda	Talla	Color	Descripción	Precio
	POLO RALF LAUREN	L	AZUL		10.000
	TOALLA	S	GUINDA	CASI LIMPIA	12.000
	VIBIRI	S	BLANCO		8.000
	SABANA	M	AZUL		20.000

Spike mantenimiento de Prenda

REGISTRO DE TALLA

Registro:

Talla:

Búsqueda:

Buscar Por Talla:

Talla
L
M
MML
O
R
S
V
XL
YLL

Spike mantenimiento tallas

REGISTRO DE TIPO DE VEHICULO

Registro:
Tipo Vehiculo:

Búsqueda:
Buscar Vehiculo:

Tipo Vehiculo
AUTOMOVIL
CAMION
CAMIONETA
MOTOCICLETA
TAXI

Spike mantenimiento tipo de vehículo

REGISTRO DE TIPO DE PARQUEO

Registro:
Tipo Parqueo:

Búsqueda:
Buscar Tipo Parqueo:

Tipo Parqueo	Precio
EXCLUSIVA	60.00
NORMAL	40.00
PREFERENCIAL	55.00
SUPER VIP	80.00
VIA VIP	90.00
VIP	75.00

Spike Mantenimiento de Tipo de Parqueo

REGISTRO DE UBICACION DE VEHICULO

Registro:
Tipo Parqueo:
Ubicacion:

Estado:
 Disponible En Mantenimiento En Limpieza

Búsqueda:
Buscar Ubicacion:

	Tipo Parqueo	Ubicacion Vehiculo	Estado
▶	EXCLUSIVA	2	Disponible
	EXCLUSIVA	1	Disponible
	EXCLUSIVA	3	Ocupado
	NORMAL	3	Ocupado
	NORMAL	1	Ocupado
	NORMAL	2	Disponible

Spike mantenimiento ubicación de vehículo

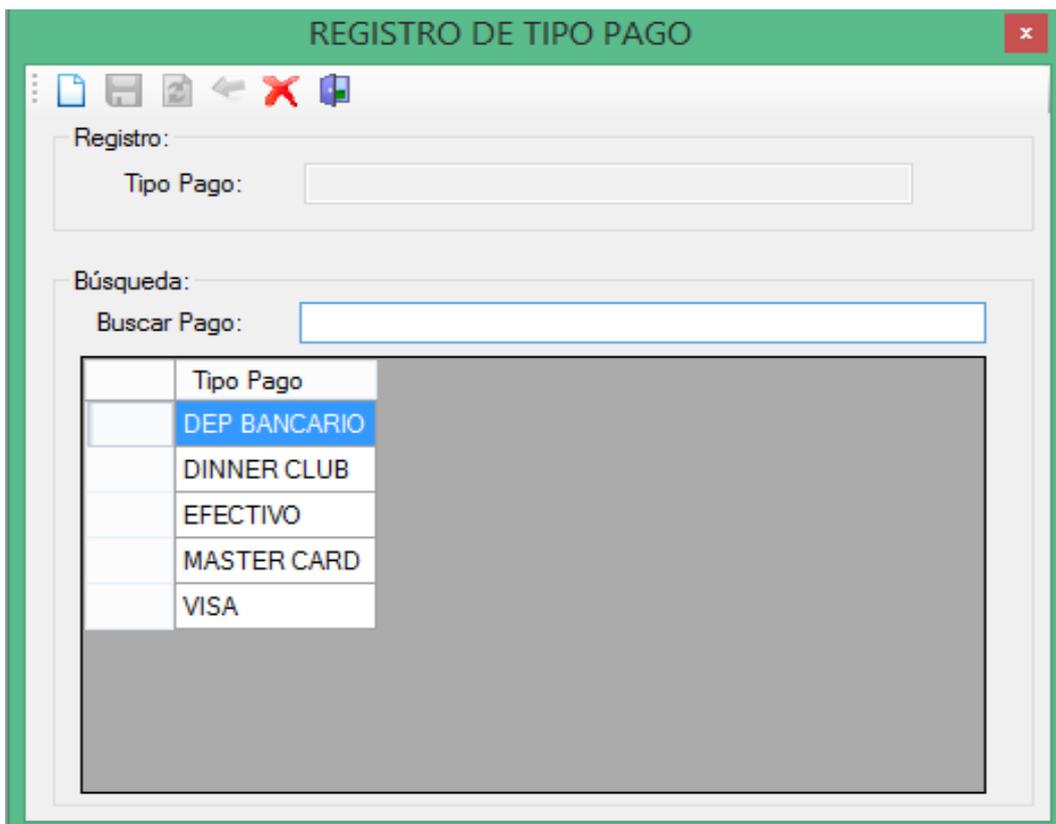
REGISTRO DE BANCO

Registro:
Tipo Banco:

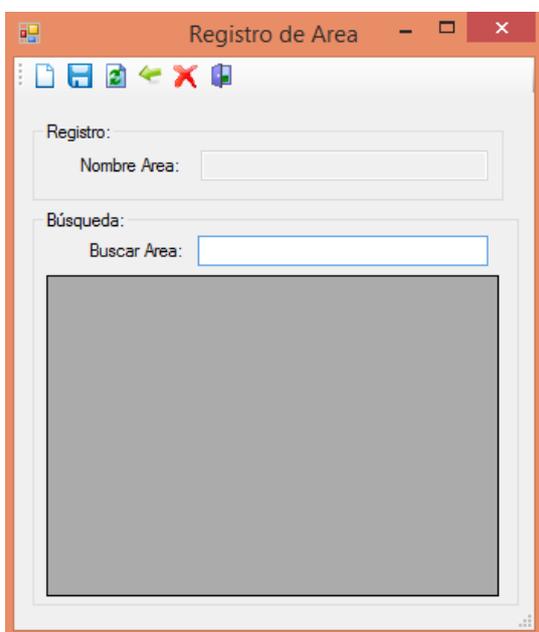
Búsqueda:
Buscar Banco:

	Tipo Banco
	BANBIF
	BANCO CONTINENTAL
	BANCO DE CREDITO DEL PERU
	CAJA AREQUIPA
	CAJA DE ICA
	CAJA HUANCAYO
	CAJA SULLANA
	CREDI SCOTIA
	MI BANCO

Spike mantenimiento de banco



Spike mantenimiento tipo de pago



Spike mantenimiento Area

Spike mantenimiento Empleado

TAREAS DE LAS HISTORIAS DE USUARIO

TAREA DE USUARIO	
Número de Tarea: 1	Historia de Usuario (Número y Nombre): Historia 01,Registro/Mantenimiento Piso
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de pisos de la empresa hotelera	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 27/03/2015	Fecha de Fin: 31/03/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los pisos del la empresa del área del hotel.	

TAREA DE USUARIO	
Número de Tarea: 2	Historia de Usuario (Número y Nombre): Historia 02,Registro/Mantenimiento Tipo

Habitación	
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los tipos de habitación que hay en la empresa.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 01/04/2015	Fecha de Fin: 03/04/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los tipos de habitación.	

TAREA DE USUARIO	
Número de Tarea: 3	Historia de Usuario (Número y Nombre): Historia 03,Registro/Mantenimiento Marca
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de las marcas de autos.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 06/04/2015	Fecha de Fin: 08/04/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar las marcas de los autos.	

TAREA DE USUARIO	
Número de Tarea: 4	Historia de Usuario (Número y Nombre): Historia 04,Registro/Mantenimiento color
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los colores, datos como que servirán para los autos, prendas.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 09/04/2015	Fecha de Fin: 13/04/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los colores.	

TAREA DE USUARIO	
Número de Tarea: 5	Historia de Usuario (Número y Nombre): Historia 05,Registro/Mantenimiento vehículo
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los vehículos.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra	

(Especificar)	
Fecha de Inicio: 14/04/2015	Fecha de Fin: 16/04/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los vehículos.	

TAREA DE USUARIO	
Número de Tarea: 6	Historia de Usuario (Número y Nombre): Historia 06,Registro/Mantenimiento Proveedor
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los proveedores.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 17/04/2015	Fecha de Fin: 21/04/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los proveedores de la empresa.	

TAREA DE USUARIO	
Número de Tarea: 7	Historia de Usuario (Número y Nombre): Historia 07,Registro/Mantenimiento Cliente
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los clientes.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 22/04/2015	Fecha de Fin: 24/04/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los clientes.	

TAREA DE USUARIO	
Número de Tarea: 8	Historia de Usuario (Número y Nombre): Historia 08,Registro/Mantenimiento Producto
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los productos de la empresa.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 30/04/2015	Fecha de Fin: 01/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	

Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de los productos que brinda la empresa.

TAREA DE USUARIO	
Número de Tarea: 9	Historia de Usuario (Número y Nombre): Historia 09,Registro/Mantenimiento Habitación
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de las habitaciones del hotel.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 04/05/2015	Fecha de Fin: 06/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar las habitaciones de la empresa.	

TAREA DE USUARIO	
Número de Tarea: 10	Historia de Usuario (Número y Nombre): Historia 10,Registro/Mantenimiento Talla
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de las tallas.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 07/05/2015	Fecha de Fin: 11/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los datos de las tallas de las prendas.	

TAREA DE USUARIO	
Número de Tarea: 11	Historia de Usuario (Número y Nombre): Historia 11,Registro/Mantenimiento Tipo vehículo
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los tipos de vehículos.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 12/05/2015	Fecha de Fin: 14/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los datos de los tipos de vehículos.	

TAREA DE USUARIO	
Número de Tarea: 12	Historia de Usuario (Número y Nombre): Historia 12,Registro/Mantenimiento Tipo Parqueo
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los tipos de parqueo.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 15/05/2015	Fecha de Fin: 15/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los datos del tipo de parqueo en lo que respecta al servicio de cochera.	

TAREA DE USUARIO	
Número de Tarea: 13	Historia de Usuario (Número y Nombre): Historia 13,Registro/Mantenimiento Ubicación vehículo
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de la ubicación del vehiculo.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 18/05/2015	Fecha de Fin: 18/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar datos de la ubicación de los vehículos, para el servicio de cochera.	

TAREA DE USUARIO	
Número de Tarea: 14	Historia de Usuario (Número y Nombre): Historia 14,Registro/Mantenimiento Banco
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de las entidades financieras.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 19/05/2015	Fecha de Fin: 20/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar datos de las entidades financieras con las que trabaja la empresa.	

TAREA DE USUARIO	
Número de Tarea: 15	Historia de Usuario (Número y Nombre): Historia 15,Registro/Mantenimiento tipo de

pago	
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de las tarjetas con las que se paga por el servicio.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 21/05/2015	Fecha de Fin: 21/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar información de las tarjetas con las que se paga un servicio.	

TAREA DE USUARIO	
Número de Tarea: 16	Historia de Usuario (Número y Nombre): Historia 16, Registro/Mantenimiento area.
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de las areas de la empresa.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 22/05/2015	Fecha de Fin: 22/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar las areas de las empresa.	

TAREA DE USUARIO	
Número de Tarea: 17	Historia de Usuario (Número y Nombre): Historia 17, Registro/Mantenimiento empleado.
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los empleados.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 25/05/2015	Fecha de Fin: 26/05/2015
Programador Responsable: Fernandez Cahuana Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar datos de los empleados.	

INTERFAZ DE LAS HISTORIAS DEL USUARIO

REGISTRO DE PISO

Registro De Nro Piso

Registro:
Piso:

Búsqueda:
Buscar por Piso:

Piso
1
2
3
4
5
6
7

Nuevo

Guardar

Modificar

Cancelar

Eliminar

Salir

Interfaz de registro de pisos

REGISTRO DE TIPO DE HABITACIÓN

Registro Tipo Habitación

Registro:
Tipo Habitación:

Búsqueda:
Buscar por Habitación:

Habitacion
BUNGALOS
DOBLE
MATRIMONIAL
OTROS
SIMPLE
SUITE
TRIPLE

Nuevo

Guardar

Modificar

Cancelar

Eliminar

Salir

Interfaz tipo de habitación

**SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.**



Interfaz Marca de vehículos



Interfaz registro de color



Interfaz Registro de Vehículo

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

REGISTRO DE PROVEEDOR

Registro De Proveedor

Registro:

Razón Social: RUC:

Dirección:

Teléfono: Celular: Departamento: AMAZONAS

Provincia: BAGUA Distrito: ARAMANGO

Búsqueda:

Buscar por Razón Social o RUC:

	Razón Social	RUC	Dirección	Teléfono 1	Teléfono 2	Departame
	ALICORP SA	12345678888	CALLE GRAU NRO 250	999999999		ICA
	GRUPO GLORIA SA	20100190797	REPUBLICA DE PANAMA - LIMA	95632147		LIMA

Nuevo

Guardar

Modificar

Cancelar

Eliminar

Salir

Interfaz Registro de Proveedor

REGISTRO DE CLIENTES PERSONA

Registro De Nueva Persona

Registro:

A. Paterno: A. Materno:

Nombres: Identificación: DNI Pasaporte Camé extranjera

Dirección: Fecha Nacimiento: 01/01/1915

Estado Civil: Sexo: M F Celular: Teléfono:

Nacionalidad: PERU Departamento: AMAZONAS

Provincia: BAGUA Distrito: ARAMANGO

Búsqueda:

Buscar por Apellidos y Nombres o Identificación:

	A. Paterno	A. Materno	Nombres	TipIdentf	Tipo Identificación	Nro. Identificación	Dirección
	VILLA	PEREZ	CESAR IVAN	D	DNI	46184593	PROLOG MARISCAL BENA
	CASTILLON	SIGUAS	LUIS	D	DNI	45217896	AV. UNION NRO 363
	HUAYLLA	CABRERA	KATY	D	DNI	12345678	SAN AGUSTIN MZ T LT 8
	CARTAGENA	VILCHEZ	JOSE MIGUEL	D	DNI	22364295	CALLE LIMA N° 20 MZ "L"
	BOLAÑOS	BARRERA	LUCAS MANUEL	D	DNI	21563489	URB. LAS MAGNOLIAS N°

Nuevo

Guardar

Modificar

Cancelar

Eliminar

Salir

Interfaz registro de clientes naturales

**SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.**

REGISTRO DE CLIENTES EMPRESA

Registro De Nueva Empresa

Logo: 

Registro:

Razón Social: RUC:

Dirección:

Celular: Teléfono:

Nacionalidad: Departamento:

Provincia: Distrito:

Búsqueda:

Buscar por Razón Social o RUC:

	Razón Social	RUC	Dirección	Teléfono 1	Teléfono 2
	COMPLEJO AGROINDUSTRIAL BETA SA	20297939131	CALLE LEOPOLDO CARRILLO NRO 160	956321478	
	ALICORP SA	12345678888	CALLE GRAU NRO 250	999999999	

Interfaz de registro de clientes Jurídicos

REGISTRO DE PRODUCTOS

Registro De Nuevo Producto

Logo: 

Registro:

Producto:

Precio: Descripción:

Ruta Foto:  

Búsqueda:

Buscar producto:

	Producto	Descripción	Precio	Foto
	GASEOSA COCA COLA 1 LT		2.500	
	DESAYUNO		7.500	
	PIÑA COLADA	REALIZADO EN BASE A PIÑA Y HIELO CON JARABE DE GOMA	8.000	

Interfaz Registro De Producto

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

REGISTRO DE HABITACIÓN

Registro De Habitación

Registro:
 N° Habitación: 1 Tipo Habitación: BUNGALOS
 Piso: 1 Precio Diario:
 Descripción:
 Característica:
 Estado: Disponible En Mantenimiento En Limpieza

Búsqueda:
 Buscar por N° Habitación:

Nro. Habitación	Tipo Habitación	Nro. Piso	Precio Diario	Característica	Descripción
1	SIMPLE	1	60.00		AGUA FRIA Y CALIE
1	DOBLE	1	80.00		DUCHA PROPIA Y F
2	SIMPLE	1	60.00		AGUA FRIA + CALIE
3	DOBLE	1	80.00		AGUA FRIA + CALIE
4	SUITE	3	120.00		

Nuevo
 Guardar
 Modificar
 Cancelar
 Eliminar
 Salir

Interfaz registro de habitación

REGISTRO DE PRENDA

Registro De Prenda

Registro:
 Prenda:
 Talla: L Color: AZUL Precio:
 Descripción:

Búsqueda:
 Buscar Prenda:

Prenda	Talla	Color	Descripción	Precio
POLO RALF LAUREN	L	AZUL		10.000
TOALLA	S	GUINDA	CASI LIMPIA	12.000
VIBIRI	S	BLANCO		8.000
SABANA	M	AZUL		20.000

Nuevo
 Guardar
 Modificar
 Cancelar
 Eliminar
 Salir

Interfaz registro de prenda

REGISTRO DE TALLA

Registro De Talla

Registro:
 Talla:
 13/12/2015

Búsqueda:
 Buscar Por Talla:

Talla
L
M
MML
O
R
S
V
XL
XLI

Nuevo
 Guardar
 Modificar
 Cancelar
 Eliminar
 Salir

Interfaz registro de tallas

REGISTRO DE TIPO DE VEHICULO

Registro De Tipo Vehiculo

Hotel "Princess"

Registro:
Tipo Vehiculo:

Búsqueda:
Buscar Vehiculo:

Tipo Vehiculo
AUTOMOVIL
CAMION
CAMIONETA
MOTOCICLETA
TAXI

Nuevo 

Guardar 

Modificar 

Cancelar 

Eliminar 

Salir 

Interfaz registro de tipo de vehículo

REGISTRO DE TIPO DE PARQUEO

Registro De Tipo Parqueo

Hotel "Princess"

Registro:
Tipo Parqueo:
Precio:

Búsqueda:
Buscar Tipo Parqueo:

Tipo Parqueo	Precio
EXCLUSIVA	60.00
NORMAL	40.00
PREFERENCIAL	55.00
SUPER VIP	80.00
VIA VIP	90.00
VIP	75.00

Nuevo 

Guardar 

Modificar 

Cancelar 

Eliminar 

Salir 

Interfaz registro tipo de parqueo

REGISTRO DE UBICACION DE VEHICULO

Registro De Ubicacion Vehiculo

Hotel "Princess"

Registro:
Tipo Parqueo:
Ubicacion:

Estado:
 Disponible En Mantenimiento

Búsqueda:
Buscar Ubicacion:

	Tipo Parqueo	Ubicacion Vehiculo	Estado
	EXCLUSIVA	2	Disponible
	EXCLUSIVA	1	Disponible
	EXCLUSIVA	3	Ocupado
	NORMAL	3	Ocupado
	NORMAL	1	Ocupado
	NORMAL	2	Disponible

Nuevo

Guardar

Modificar

Cancelar

Eliminar

Salir

Interfaz registro de ubicación vehículo

REGISTRO DE ENTIDAD FINANCIERA

Registro De Entidad Financiera

Hotel "Princess"

Registro:
Tipo Banco:

Búsqueda:
Buscar Banco:

	Tipo Banco
	BANBIF
	BANCO CONTINENTAL
	BANCO DE CREDITO DEL PERU
	CAJA AREQUIPA
	CAJA DE ICA
	CAJA HUANCAYO
	CAJA SULLANA
	CREDI SCOTIA
	MI BANCO

Nuevo

Guardar

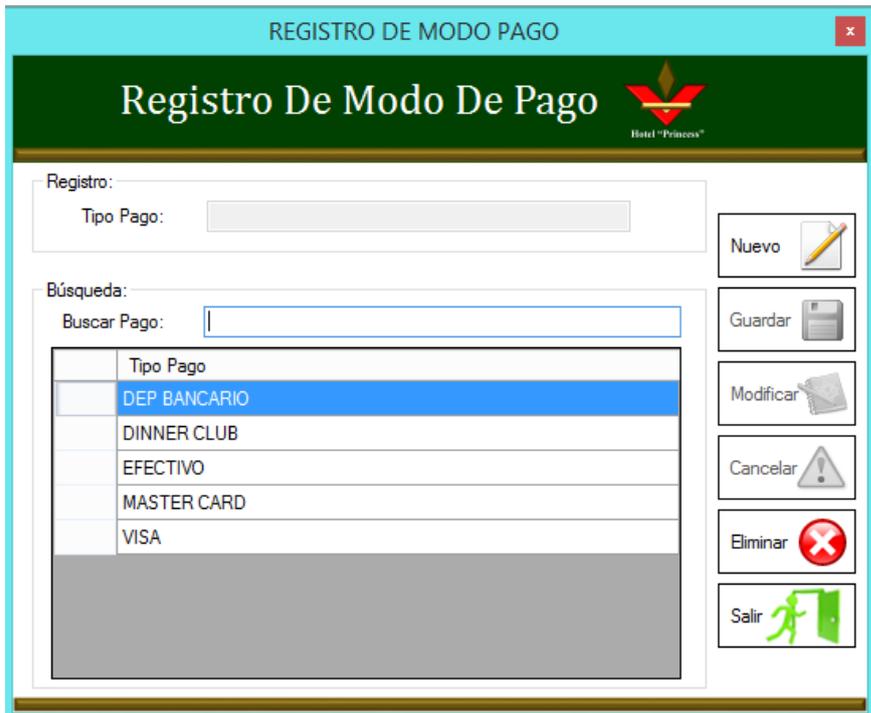
Modificar

Cancelar

Eliminar

Salir

Interfaz registro Banco (entidades financiera)



Interfaz registro tipo de pago



Interfaz registro de area

REGISTRO DE EMPLEADOS

Registro De Nuevo Empleado

Identificación:

Apellidos y Nombres:

Dirección:

Área: ADMINISTRACION Carga: ADMINISTRADOR

F. Ingreso: 13/12/2015 F. Baja: 13/12/2015 Salario: Inactivo

Búsqueda:

Buscar por Apellidos y Nombres o Identificación:

Identificación	Empleado	Dirección	Área
45217896	CASTILLON SIGUAS LUIS	AV. UNION NRO 363	ADMINISTRAC
12345678	HUAYLLA CABRERA KATY	SAN AGUSTIN MZ T LT 8	ADMINISTRAC
21536233	VARGAS BERNAOLA JIMENA	URB LAS MAGNOLIAS N " D" LT 120	HOTEL

Interfaz registro de Empleado

CODIFICACIÓN BOL

CLASE PISO

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsPiso
#Region "Variables"
Private sidPiso As String
Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _idPiso() As String Get
Return sidPiso
End Get
Set(ByVal value As String) sidPiso = value
End Set End Property

Public Property _nombre() As String Get
```

```
Return sNombre
End Get
Set(ByVal value As String)
    sNombre = value
End Set End Property
#End Region

Public Function PisoSelectAllByName(ByVal x As clsPiso) As
DataSet
Dim db As New dbAccess db.AddParameter("@NomPiso", x._nombre) Return
db.ExecuteDataSet("SP_MostrarPiso")
End Function

Public Sub MantenimientoPiso(ByVal x As clsPiso, ByVal
v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@idPiso", x._idPiso) db.AddParameter("@NomPiso", x._nombre)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Piso")
End Sub End Class
```

CLASE TIPO HABITACIÓN

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsPiso
#Region "Variables"
Private sidPiso As String Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _idPiso() As String
Get
Return sidPiso
End Get
Set(ByVal value As String) sidPiso = value
End Set
End Property

Public Property _nombre() As String Get
Return sNombre
End Get
Set(ByVal value As String)
sNombre = value
End Set End Property
#End Region
```

```
Public Function PisoSelectAllByName(ByVal x As clsPiso) As
DataSet
Dim db As New dbAccess db.AddParameter("@NomPiso", x._nombre) Return
db.ExecuteDataSet("SP_MostrarPiso")
End Function

Public Sub MantenimientoPiso(ByVal x As clsPiso, ByVal
v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@idPiso", x._idPiso) db.AddParameter("@NomPiso", x._nombre)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Piso")
End Sub End Class
```

CLASE MARCA AUTO

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsMarca
#Region "Variables"
Private sidMarca As String Private sNombre As String Private sTipo As Char
#End Region

#Region "Propiedades"
Public Property _idMarca() As String Get
Return sidMarca
End Get
Set(ByVal value As String) sidMarca = value
End Set
End Property

Public Property _nombre() As String Get
Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set End Property

Public Property _Tipo() As Char Get
Return sTipo
End Get
Set(ByVal value As Char)
sTipo = value
```

```
End Set End Property  
#End Region
```

```
Public Function MarcaSelectAllByName(ByVal x As clsMarca)  
As DataSet  
Dim db As New dbAccess db.AddParameter("@NomMarca", x._nombre) Return  
db.ExecuteDataSet("SP_MostrarMarca")  
End Function
```

```
Public Sub MantenimientoMarca(ByVal x As clsMarca, ByVal  
vlProceso As Char)  
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)  
db.AddParameter("@idMarca", x._idMarca) db.AddParameter("@NomMarca", x._nombre)  
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)  
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)  
  
db.ExecuteNonQuery("SP_Mant_Marca")  
End Sub End Class
```

CLASE COLOR

```
Imports System.Data  
Imports System.Data.SqlClient  
Public Class clsColor  
#Region "Variables"  
Private sidColor As String Private sNombre As String  
#End Region  
  
#Region "Propiedades"  
Public Property _idColor() As String  
Get  
Return sidColor  
End Get  
Set(ByVal value As String) sidColor = value  
End Set  
End Property  
  
Public Property _nombre() As String Get  
Return sNombre  
End Get  
Set(ByVal value As String)  
sNombre = value  
End Set End Property  
#End Region  
  
Public Function ColorSelectAllByName(ByVal x As clsColor)  
As DataSet
```

```
Dim db As New dbAccess db.AddParameter("@nomColor", x._nombre) Return  
db.ExecuteDataSet("SP_MostrarColor")  
End Function
```

```
Public Sub MantenimientoColor(ByVal x As clsColor, ByVal  
v1Proceso As Char)  
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)  
db.AddParameter("@idColor", x._idColor) db.AddParameter("@nomColor", x._nombre)  
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)  
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)  
  
db.ExecuteNonQuery("SP_Mant_Color")  
End Sub End Class
```

CLASE VEHÍCULO

```
Imports System.Data  
Imports System.Data.SqlClient  
Public Class clsColor  
#Region "Variables"  
Private sidColor As String Private sNombre As String  
#End Region  
  
#Region "Propiedades"  
Public Property _idColor() As String  
Get  
Return sidColor  
End Get  
Set(ByVal value As String) sidColor = value  
End Set  
End Property  
  
Public Property _nombre() As String Get  
Return sNombre  
End Get  
Set(ByVal value As String)  
sNombre = value  
End Set End Property  
#End Region  
  
Public Function ColorSelectAllByName(ByVal x As clsColor)  
As DataSet  
Dim db As New dbAccess db.AddParameter("@nomColor", x._nombre) Return  
db.ExecuteDataSet("SP_MostrarColor")  
End Function
```

```
Public Sub MantenimientoColor(ByVal x As clsColor, ByVal
v1Proceso As Char)
Dim db As New dbAccess
db.AddParameter("@Proceso", v1Proceso) db.AddParameter("@idColor", x._idColor)
db.AddParameter("@nomColor", x._nombre) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Color")
End Sub
End Class
```

CLASE PROVEEDORES

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsProveedor : Inherits clsEmpresa

#Region "Variables"
Private sidProveedor As String
#End Region

#Region "Propiedades"
Public Property _sidProveedor() As String Get
Return sidProveedor
End Get
Set(ByVal value As String)
sidProveedor = value
End Set End Property
#End Region

Public Function ProveedorSelectAllByName(ByVal xFiltro As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@filtro", xFiltro)
Return db.ExecuteDataSet("SP_MostrarProveedor")
End Function

Public Sub MantenimientoProveedor(ByVal x As clsProveedor,
ByVal v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@idProveedor", x._sidProveedor) db.AddParameter("@idEmpresa",
x._sIdEmpresa) db.AddParameter("@idDistrito", x._sidDistrito)
db.AddParameter("@razonSocial", x._srazonSocial) db.AddParameter("@ruc", x._sRuc)
db.AddParameter("@direccion", x._sDireccion) db.AddParameter("@telf1", x._sTelf1)
db.AddParameter("@telf2", x._sTelf2) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
```

```
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Proveedor")
End Sub

Public Function validarDobleIngreso(ByVal xRuc As String, ByVal xNop As String)
As String
Dim db As New dbAccess

db.AddParameter("@Ruc", xRuc) db.AddParameter("@nOp", xNop)

Dim xValor As Integer = db.ExecuteScalar("SP_ValidarDobleIngresoEmpresa")

Return xValor.ToString
End Function

Public Sub ActivarProveedor(ByVal xRuc As String) Dim db As New dbAccess

db.AddParameter("@Ruc", xRuc) db.ExecuteNonQuery("SP_ActivarProveedor")
End Sub
End Class
```

CLASE CLIENTE NATURAL

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsClienteNatural : Inherits clsPersona
#Region "Variables"
Private sidClienteN As String
#End Region

#Region "Propiedades"
Public Property _sidClienteN() As String Get
Return sidClienteN
End Get
Set(ByVal value As String) sidClienteN = value
End Set End Property
#End Region

Public Function ClienteSelectAllByName(ByVal xFiltro As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@filtro", xFiltro)
Return db.ExecuteDataSet("SP_MostrarClienteNatural")
End Function

Public Sub MantenimientoCliente(ByVal x As
clsClienteNatural, ByVal vlProceso As Char)
Dim db As New dbAccess
```

```
db.AddParameter("@Proceso", vlProceso) db.AddParameter("@idClienteN",  
x._sidClienteN) db.AddParameter("@idPersona", x._sidPersona)  
db.AddParameter("@idDistrito", x._sidDistrito) db.AddParameter("@apePat",  
x._sapePat) db.AddParameter("@apeMat", x._sapeMat) db.AddParameter("@nombres",  
x._snombres) db.AddParameter("@tipIden", x._ctipIden) db.AddParameter("@nroIden",  
x._snroIden) db.AddParameter("@direccion", x._sDireccion)  
db.AddParameter("@fecNac", x._dFechaNac) db.AddParameter("@sexo", x._cSexo)  
db.AddParameter("@eCivil", x._cEstCiv) db.AddParameter("@telf1", x._sTelf1)  
db.AddParameter("@telf2", x._sTelf2) db.AddParameter("@Usr", vgUsr)  
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")  
db.AddParameter("@Error", 0)
```

```
db.ExecuteNonQuery("SP_Mant_ClienteNatural")  
End Sub
```

```
Public Function validarDobleIngreso(ByVal xNroIden As String, ByVal xNop As  
String) As String  
Dim db As New dbAccess
```

```
db.AddParameter("@nroIden", xNroIden) db.AddParameter("@nOp", xNop)
```

```
Dim xValor As Integer = db.ExecuteScalar("SP_ValidarDobleClienteNatural")
```

```
Return xValor.ToString  
End Function
```

```
Public Sub ActivarCliente(ByVal xNroIden As String) Dim db As New dbAccess
```

```
db.AddParameter("@nroIden", xNroIden) db.ExecuteNonQuery("SP_ActivarClienteN")  
End Sub
```

```
Public Function MostrarPersona(ByVal x As  
clsClienteNatural) As DataSet  
'Dim db As New dbAccess 'db.AddParameter("@nroidentif", x._snroIden) 'Return  
db.ExecuteDataSet("SU_PersonaBuscar")  
End Function
```

```
Public Function mostrarHuespedReserva(ByVal xFiltro As String) As DataSet  
Dim db As New dbAccess  
db.AddParameter("@Filtro", xFiltro)  
Return db.ExecuteDataSet("SP_ListarPersonaReserva")  
End Function
```

```
Public Function mostrarPersonaEmpleado(ByVal xFiltro As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", xFiltro)
Return db.ExecuteDataSet("SP_ListarPersonaEmpleado")
End Function
End Class
```

CLASE CLIENTE JURÍDICO

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsClienteJuridico : Inherits clsEmpresa
#Region "Variables"
Private sidClienteJ As String
#End Region

#Region "Propiedades"
Public Property _sidClienteJ() As String Get
Return sidClienteJ
End Get
Set(ByVal value As String) sidClienteJ = value
End Set End Property
#End Region

Public Function ClienteSelectAllByName(ByVal xFiltro As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@filtro", xFiltro)
Return db.ExecuteDataSet("SP_MostrarClienteJuridico")
End Function

Public Sub MantenimientoCliente(ByVal x As
clsClienteJuridico, ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idClienteJ", x._sidClienteJ) db.AddParameter("@idEmpresa",
x._sidEmpresa) db.AddParameter("@idDistrito", x._sidDistrito)
db.AddParameter("@razonSocial", x._srazonSocial) db.AddParameter("@ruc", x._sRuc)
db.AddParameter("@direccion", x._sDireccion) db.AddParameter("@telf1", x._sTelf1)
db.AddParameter("@telf2", x._sTelf2) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_ClienteJuridico")
End Sub

Public Function validarDobleIngreso(ByVal xRuc As String, ByVal xNop As String)
As String
```

```
Dim db As New dbAccess

db.AddParameter("@Ruc", xRuc) db.AddParameter("@nOp", xNop)

    Dim xValor As Integer = db.ExecuteScalar("SP_ValidarDobleClienteJuridico")

Return xValor.ToString
End Function
```

```
Public Sub ActivarCliente(ByVal xRuc As String) Dim db As New dbAccess

db.AddParameter("@Ruc", xRuc) db.ExecuteNonQuery("SP_ActivarClienteJ")
End Sub End Class
```

CLASE PRODUCTO

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsProducto
#Region "Variables"
Private sidProducto As String
Private snomProducto As String Private sDescripcion As String Private dPrecio As
Decimal Private iFoto As Object
#End Region

#Region "Propiedades"
Public Property _sidProducto() As String Get
    Return sidProducto
End Get
Set(ByVal value As String) sidProducto = value
End Set End Property

Public Property _snomProducto() As String Get
    Return snomProducto
End Get
Set(ByVal value As String) snomProducto = value
End Set
End Property

Public Property _sDescripcion() As String Get
    Return sDescripcion
End Get
Set(ByVal value As String)
```

```

                                sDescripcion = value
        End Set End Property

Public Property _dPrecio() As Decimal Get
                                Return dPrecio
    End Get
    Set(ByVal value As Decimal)
                                dPrecio = value
        End Set End Property

Public Property _iFoto() As Object Get
                                Return iFoto
    End Get
    Set(ByVal value As Object) iFoto = value
        End Set End Property
#End Region

Public Function ProductoSelectAllByName(ByVal x As
clsProducto) As DataSet
Dim db As New dbAccess
db.AddParameter("@Producto", x._snomProducto)
Return db.ExecuteDataSet("SP_MostrarProductos")
End Function

Public Sub MantenimientoProducto(ByVal x As clsProducto,
ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idProducto", x._sidProducto) db.AddParameter("@nomPrd",
x._snomProducto) db.AddParameter("@descrip", x._sDescripcion)
db.AddParameter("@precio", x._dPrecio) db.AddParameter("@foto", x._iFoto)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Productos")
End Sub

Public Function listarConsumos(ByVal x As clsProducto) As
DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", x._snomProducto)
Return db.ExecuteDataSet("SP_ListarProductoConsumos")
End Function

Public Function listarProductoEliminados(ByVal fechaini As Date, ByVal fechafin
As Date, ByVal Concepto As String) As DataSet
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
db.AddParameter("@fechafin", fechafin)

```

```
db.AddParameter("@Filtro", Concepto)
Return
db.ExecuteDataSet("SP_ListadoProductosEliminados")
End Function End Class
```

CLASE HABITACIÓN

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsHabitacion
#Region "Variables"
Private sidHabitacion As String Private sidTipoHab As String Private sidPiso As
String Private snumeroHab As String Private sdescripcion As String
Private scaracteristicas As String Private sprecio_diario As Double Private sestado
As Char

#End Region

#Region "Propiedades"
Public Property _idHabitacion() As String Get
Return sidHabitacion
End Get
Set(ByVal value As String)
sidHabitacion = value
End Set End Property

Public Property _idTipoHab() As String Get
Return sidTipoHab
End Get
Set(ByVal value As String) sidTipoHab = value
End Set End Property

Public Property _idPiso() As String Get
Return sidPiso
End Get
Set(ByVal value As String) sidPiso = value
End Set
End Property

Public Property _numeroHab() As String Get
Return snumeroHab
End Get
Set(ByVal value As String)
```

```

                                snumeroHab = value
    End Set End Property

Public Property _descripcion() As String Get
                                Return sdescripcion
    End Get
    Set(ByVal value As String)
                                sdescripcion = value
    End Set End Property

Public Property _caracteristicas() As String Get
                                Return scaracteristicas
    End Get
    Set(ByVal value As String) scaracteristicas = value
    End Set End Property

Public Property _precio_diario() As Double Get
                                Return sprecio_diario
    End Get
    Set(ByVal value As Double) sprecio_diario = value
    End Set
End Property

Public Property _estado() As Char Get
                                Return sestado
    End Get
    Set(ByVal value As Char)
                                sestado = value
    End Set End Property

#End Region

Public Function HabitaciónSelectAllByName(ByVal x As
clsHabitacion) As DataSet
Dim db As New dbAccess
db.AddParameter("@numeroHab", x._numeroHab)
Return db.ExecuteDataSet("SP_MostrarHabitacion")
End Function

    Public Function EstadoHabitacion(ByVal tipoHabitacion As String, ByVal estado
As Char) As DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", tipoHabitacion) db.AddParameter("@Filtro_1", estado)
Return db.ExecuteDataSet("SP_Estado_ActualHabitacion")
End Function

Public Sub MantenimientoHabitacion(ByVal x As
                                clsHabitacion, ByVal v1Proceso As Char)
Dim db As New dbAccess
```

```
db.AddParameter("@Proceso", vlProceso) db.AddParameter("@idHabitacion",
x._idHabitacion) db.AddParameter("@idTipoHab", x._idTipoHab)
db.AddParameter("@idPiso", x._idPiso) db.AddParameter("@numeroHab", x._numeroHab)
db.AddParameter("@descripcion", x._descripcion) db.AddParameter("@caracteristicas",
x._caracteristicas) db.AddParameter("@precio_diario", x._precio_diario)
db.AddParameter("@estado", x._estado) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Habitacion")
End Sub

Public Sub cambiarPrecios(ByVal xIdHab As String, ByVal xPreAct As Decimal,
ByVal xNuePre As Decimal, ByVal xMotivo As String)
Dim db As New dbAccess

db.AddParameter("@idHabitacion", xIdHab) db.AddParameter("@PrecioAct", xPreAct)
db.AddParameter("@nuevoPrecio", xNuePre) db.AddParameter("@motivo", xMotivo)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)

db.ExecuteNonQuery("SP_InsertarNuevoPrecio")
End Sub

Public Function mostrarHabitacioneXlimpiar() As DataSet Dim db As New dbAccess
'db.AddParameter("@idPerfil", x._idPerfil) 'db.AddParameter("@Nop", xNop)
Return
db.ExecuteNonQuery("SP_ListarHabitacionesXlimpiar")
End Function

Public Sub GrabarHabitacionesLimpiar() Dim db As New dbAccess db.AddParameter("@Cad01",
vgCadValor)
'db.AddParameter("@idPerfil", x._idPerfil) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_GenerarHabitacionesLimpieza")
End Sub End Class
```

CLASE PRENDA

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsPrenda
```

```
#Region "Variables"
Private sidPrenda As String Private snomPrenda As String Private sidTalla As String
Private sidColor As String Private sDescripcion As String Private dPrecio As
Decimal
#End Region

#Region "Propiedades"
Public Property _idPrenda() As String Get
    Return sidPrenda
End Get
Set(ByVal value As String) sidPrenda = value
End Set End Property

Public Property _nomPrenda() As String Get
    Return snomPrenda
End Get
Set(ByVal value As String) snomPrenda = value
End Set
End Property

Public Property _idTalla() As String Get
    Return sidTalla
End Get
Set(ByVal value As String) sidTalla = value
End Set End Property

Public Property _idColor() As String Get
    Return sidColor
End Get
Set(ByVal value As String)
    sidColor = value
End Set End Property

Public Property _Descripcion() As String Get
    Return sDescripcion
End Get
Set(ByVal value As String) sDescripcion = value
End Set End Property

Public Property _Precio() As Decimal Get
    Return dPrecio
End Get
Set(ByVal value As Decimal)
```

```

                                dPrecio = value
    End Set End Property
#End Region

Public Function PrendaSelectAllByName(ByVal x As clsPrenda)
As DataSet
Dim db As New dbAccess db.AddParameter("@Prenda", x._nomPrenda) Return
db.ExecuteDataSet("SP_MostrarPrenda")
End Function

Public Sub MantenimientoPrenda(ByVal x As clsPrenda, ByVal
v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@idPrenda", x._idPrenda) db.AddParameter("@nomPrenda",
x._nomPrenda) db.AddParameter("@idTalla", x._idTalla) db.AddParameter("@idColor",
x._idColor) db.AddParameter("@descripcion", x._Descripcion)
db.AddParameter("@precio", x._Precio) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Prenda")
End Sub

Public Function listarPrendas(ByVal x As clsPrenda) As
DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", x._nomPrenda)
Return db.ExecuteDataSet("SP_ListarPrendasLavanderia")
End Function

End Class
```

CLASE TALLA

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsTalla
#Region "Variables"
Private sidTalla As String Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _idTalla() As String Get
Return sidTalla
End Get
Set(ByVal value As String)
sidTalla = value
End Set
```

```
End Property

Public Property _nombre() As String Get
    Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set
End Property
#End Region

Public Function TallaSelectAllByName(ByVal x As clsTalla)
    As DataSet
    Dim db As New dbAccess db.AddParameter("@nomTalla", x._nombre) Return
    db.ExecuteDataSet("SP_MostrarTalla")
End Function

Public Sub MantenimientoTalla(ByVal x As clsTalla, ByVal
    vlProceso As Char)
    Dim db As New dbAccess
    db.AddParameter("@Proceso", vlProceso) db.AddParameter("@idTalla", x._idTalla)
    db.AddParameter("@nomTalla", x._nombre) db.AddParameter("@Usr", vgUsr)
    db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
    db.AddParameter("@Error", 0)

    db.ExecuteNonQuery("SP_Mant_Talla")
End Sub
End Class
```

CLASE TIPO VEHICULOS

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsTipoVehiculo
    #Region "Variables"
    Private sidTipVehiculo As String Private snomTipVehiculo As String
    #End Region

    #Region "Propiedades"
    Public Property _idTipVehiculo() As String Get
        Return sidTipVehiculo
    End Get
    Set(ByVal value As String) sidTipVehiculo = value
    End Set
    End Property

    Public Property _nomTipVehiculo() As String Get
        Return snomTipVehiculo
    End Get
```

```
Set(ByVal value As String) snomTipVehiculo = value
End Set
End Property
#End Region
Public Function TiposVehiculoSelectAllByName(ByVal x As
clsTipoVehiculo) As DataSet
Dim db As New dbAccess db.AddParameter("@nomVehiculo", x._nomTipVehiculo) Return
db.ExecuteDataSet("SP_MostrarTipVehiculo")
End Function

Public Sub MantenimientoVehiculo(ByVal x As
clsTipoVehiculo, ByVal vlProceso As Char)
Dim db As New dbAccess
db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idTipVehiculo", x._idTipVehiculo)
db.AddParameter("@nomTipVehiculo", x._nomTipVehiculo) db.AddParameter("@Usr",
vgUsr) db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_TipoVehiculo")
End Sub End Class
```

CLASE TIPO PARQUEO

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsTipoParqueo
#Region "Variables"
Private sidTiPar As String Private sNombre As String Private sprecio As Decimal
#End Region

#Region "Propiedades"
Public Property _idTiPar() As String Get
Return sidTiPar
End Get
Set(ByVal value As String) sidTiPar = value
End Set End Property

Public Property _nombre() As String Get
Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set
End Property

Public Property _precio() As Decimal
```

```

        Get
            Return sprecio
    End Get
Set(ByVal value As Decimal) sprecio = value
End Set
End Property
#End Region

Public Function TipParqueoSelectAllByName(ByVal x As
clsTipoParqueo) As DataSet
Dim db As New dbAccess
db.AddParameter("@nomZona", x._nombre)
Return db.ExecuteDataSet("SP_MostrarTipParqueo")
End Function

Public Function EstadoParqueo(ByVal tipoParqueo As String, ByVal estado As
Char) As DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", tipoParqueo) db.AddParameter("@Filtro_1", estado)
Return db.ExecuteDataSet("SP_Estado_ActualParqueo")
End Function

Public Sub MantenimientoTipParqueo(ByVal x As
clsTipoParqueo, ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idTipParqueo", x._idTiPar) db.AddParameter("@nomZona", x._nombre)
db.AddParameter("@precio", x._precio) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_TipoParqueo")
End Sub End Class
```

CLASE UBICACION VEHÍCULO

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsUbicacionVehiculo
#Region "Variables"
Private sidUbiVehiculo As String Private sidTipParqueo As String Private
snomUbicacion As String Private sestado As Char
#End Region

#Region "Propiedades"
Public Property _idUbiVehiculo() As String Get
Return sidUbiVehiculo
End Get
Set(ByVal value As String)
sidUbiVehiculo = value
```

```
End Set End Property

Public Property _idTipParqueo() As String Get
    Return sidTipParqueo
End Get
Set(ByVal value As String) sidTipParqueo = value
End Set End Property

Public Property _nomUbicacion() As String Get
    Return snomUbicacion
End Get
Set(ByVal value As String) snomUbicacion = value
End Set
End Property

Public Property _estado() As Char Get
    Return sestado
End Get
Set(ByVal value As Char)
    sestado = value
End Set End Property
#End Region

Public Function UbiacionVehiculoSelectAllByName(ByVal x As
clsUbicacionVehiculo) As DataSet
Dim db As New dbAccess db.AddParameter("@nomUbicacion", x._nomUbicacion) Return
db.ExecuteDataSet("SP_MostrarUbiVehiculo")
End Function

Public Sub MantenimientoUbicacionVehiculo(ByVal x As
clsUbicacionVehiculo, ByVal vlProceso As Char) Dim db As New dbAccess
db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idUbiVehiculo", x._idUbiVehiculo)
db.AddParameter("@idTipParqueo", x._idTipParqueo) db.AddParameter("@nomUbicacion",
x._nomUbicacion) db.AddParameter("@estado", x._estado) db.AddParameter("@Usr",
vgUsr) db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_UbicacionVehiculo")
End Sub End Class
```

CLASE TIPO BANCO (ENTIDADES FINANCIERAS)

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsTipoBanco
#Region "Variables"
Private sidBanco As String Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _idBanco() As String Get
Return sidBanco
End Get
Set(ByVal value As String) sidBanco = value
End Set
End Property

Public Property _nombre() As String Get
Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set End Property
#End Region

Public Function TipoBancoSelectAllByName(ByVal x As
clsTipoBanco) As DataSet
Dim db As New dbAccess
db.AddParameter("@NomBanco", x._nombre)
Return db.ExecuteDataSet("SP_MostrarTipoBanco")
End Function

Public Sub MantenimientoTipoBanco(ByVal x As clsTipoBanco,
ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idBanco", x._idBanco) db.AddParameter("@nomBanco", x._nombre)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_TipoBanco")
End Sub End Class
```

CLASE TIPO PAGO

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsTipoPago
#Region "Variables"
```

```
Private sidPago As String Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _idPago() As String
Get
    Return sidPago
End Get
Set(ByVal value As String) sidPago = value
End Set End Property

Public Property _nombre() As String Get
    Return sNombre
End Get
Set(ByVal value As String)
    sNombre = value
End Set End Property
#End Region

Public Function TipoPagoSelectAllByName(ByVal x As
clsTipoPago) As DataSet
Dim db As New dbAccess db.AddParameter("@NomTipPago", x._nombre) Return
db.ExecuteDataSet("SP_MostrarTipoPago")
End Function

Public Sub MantenimientoTipoPago(ByVal x As clsTipoPago,
ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idPago", x._idPago) db.AddParameter("@nomPago", x._nombre)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_TipoPago")
End Sub End Class
```

CLASE AREA

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsArea
#Region "Variables"
Private sCodigo As String Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _codigo() As String
```

```

        Get
            Return sCodigo
    End Get
Set(ByVal value As String) sCodigo = value
    End Set
End Property

Public Property _nombre() As String Get
    Return sNombre
End Get
Set(ByVal value As String)
    sNombre = value
    End Set End Property
#End Region

Public Function AreaSelectAllByName(ByVal x As clsArea) As
DataSet
Dim db As New dbAccess
db.AddParameter("@Area", x._nombre)
Return db.ExecuteDataSet("SP_MostranArea")
End Function

Public Sub MantenimientoArea(ByVal x As clsArea, ByVal
v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@Codigo", x._codigo) db.AddParameter("@nomArea", x._nombre)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Area")
    End Sub End Class
```

CLASE EMPLEADO

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsEmpleado
#Region "Variables"
Private sidEmpleado As String Private sidPersona As String Private sidArea As
String Private sidPerfil As String Private sfechaIngreso As Date Private sfechaSalida
As Date Private sSalario As Double Private sestado As Char

#End Region

#Region "Propiedades"
```

```
Public Property _idEmpleado() As String Get
    Return sidEmpleado
End Get
Set(ByVal value As String) sidEmpleado = value
End Set End Property

Public Property _idPersona() As String Get
    Return sidPersona
End Get
Set(ByVal value As String) sidPersona = value
End Set
End Property

Public Property _idArea() As String Get
    Return sidArea
End Get
Set(ByVal value As String) sidArea = value
End Set End Property

Public Property _idPerfil() As String Get
    Return sidPerfil
End Get
Set(ByVal value As String)
    sidPerfil = value
End Set End Property

Public Property _fechIngreso() As Date Get
    Return sfechIngreso
End Get
Set(ByVal value As Date) sfechIngreso = value
End Set End Property

Public Property _fechSalida() As Date Get
    Return sfechSalida
End Get
Set(ByVal value As Date) sfechSalida = value
End Set
End Property

Public Property _Salario() As Double Get
    Return sSalario
End Get
Set(ByVal value As Double)
    sSalario = value
```

```
End Set End Property

Public Property _estado() As Char Get
    Return sestado
End Get
Set(ByVal value As Char) sestado = value
End Set End Property

#End Region

Public Function EmpleadoSelectAllByName(ByVal x As
clsEmpleado) As DataSet
Dim db As New dbAccess db.AddParameter("@NomEmpleado", x._idPersona) Return
db.ExecuteDataSet("SP_MostrarEmpleado")
End Function

Public Sub MantenimientoEmpleado(ByVal x As clsEmpleado,
ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idEmpleado", x._idEmpleado) db.AddParameter("@idPersona",
x._idPersona) db.AddParameter("@idArea", x._idArea) db.AddParameter("@idPerfil",
x._idPerfil) db.AddParameter("@fecIngreso", x._fechIngreso)
db.AddParameter("@fecSalida", x._fechSalida) db.AddParameter("@salario",
x._Salario) db.AddParameter("@estado", x._estado) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Empleado")
End Sub End Class
```

DAL

Cabe mencionar que el código en la clase dbaccess son códigos reutilizables por tal razón solo se hará mención en la primera iteración

CLASE DBACCESS

```
Imports System
Imports System.Data
Imports System.Data.SqlClient
```

```
Imports System.Configuration

Public Class dbAccess
Private cmd As IDbCommand = New SqlCommand() Private strConnectionString As String
= "" Private handleErrors As Boolean = False Private strLastError As String = ""

Public Sub New()
Dim vlServer, vlBD, vlUsr, vlPass, vlDes As String

vlServer = "SMITH" vlBD = "DB_Hotel" vlUsr = "luis0159" vlPass = "castillon"

strConnectionString = "Data source= " & vlServer & _
"; Initial catalog = " & vlBD &
_
"; user id = " & vlUsr & _
"; pwd = " & vlPass

Dim cnn As New SqlConnection() cnn.ConnectionString = strConnectionString
cmd.Connection = cnn
cmd.CommandType = CommandType.StoredProcedure

    vlDes = "SISTEMA HOTELERO VERSION - 1.0 - PRINCESS "

frmPrincipal.Text = (vlDes) frmPrincipal.Show()
End Sub

Public Function ExecuteReader() As IDataReader
Dim reader As IDataReader = Nothing
Try
Me.Open() reader =
cmd.ExecuteReader(CommandBehavior.CloseConnection)
Catch ex As Exception
    If handleErrors Then
strLastError = ex.Message
    Else
        Throw
    End If End Try Return reader
End Function

    Public Function ExecuteReader(ByVal commandtext As String) As IDataReader
Dim reader As IDataReader = Nothing Try
cmd.CommandText = commandtext
    reader = Me.ExecuteReader() Catch ex As Exception
        If (handleErrors) Then
strLastError = ex.Message
            Else
```

```
        Throw End If
    End Try
    Return reader
End Function

Public Function ExecuteScalar() As Object Dim obj As Object = Nothing
    Try
        Me.Open()
        obj = cmd.ExecuteScalar()
        Me.Close()
    Catch ex As Exception
        If handleErrors Then
            strLastError = ex.Message
        Else
            Throw
        End If End Try Return obj
End Function

    Public Function ExecuteScalar(ByVal commandtext As String) As Object
    Dim obj As Object = Nothing Try
    cmd.CommandText = commandtext obj = Me.ExecuteScalar()
    Catch ex As Exception
        If (handleErrors) Then
            strLastError = ex.Message
        Else
            Throw
        End If End Try Return obj
End Function

Public Function ExecuteNonQuery() As Integer Dim i As Integer = -1
    Try
        Me.Open()
        i = cmd.ExecuteNonQuery()
        Me.Close()
    Catch ex As Exception
        If handleErrors Then
            strLastError = ex.Message
        Else
            Throw
        End If End Try Return i
End Function

    Public Function ExecuteNonQuery(ByVal commandtext As String) As Integer
    Dim i As Integer = -1
    Try
        cmd.CommandText = commandtext
        i = Me.ExecuteNonQuery()
    End Try
```

```
Catch ex As Exception
    strLastError = ex.Message
    If handleErrors Then
        Else
            Throw
        End If
End Try Return i
End Function

Public Function ExecuteDataSet() As DataSet
Dim da As SqlDataAdapter = Nothing Dim ds As DataSet = Nothing
Try
da = New SqlDataAdapter() da.SelectCommand = CType(cmd, SqlCommand) ds = New
DataSet()
    da.Fill(ds)
Catch ex As Exception
    strLastError = ex.Message
    If (handleErrors) Then
        Else
            Throw
        End If End Try Return ds
End Function

Public Function ExecuteDataSet(ByVal commandtext As String) As DataSet
Dim ds As DataSet = Nothing Try
cmd.CommandText = commandtext ds = Me.ExecuteDataSet()
Catch ex As Exception
    strLastError = ex.Message
    If handleErrors Then
        Else
            Throw
        End If End Try Return ds
End Function

Public Property CommandText() As String Get
    Return cmd.CommandText
End Get
Set(ByVal value As String)
cmd.CommandText = value cmd.Parameters.Clear()
End Set
End Property

Public ReadOnly Property Parameters() As
    IDataParameterCollection
Get
    Return cmd.Parameters
End Get
End Property
```

```
Public Sub AddParameter(ByVal paramname As String, ByVal  
paramvalue As Object)  
    Dim param As SqlParameter = New SqlParameter(paramname, paramvalue)  
    cmd.Parameters.Add(param)  
End Sub  
  
Public Sub AddParameter(ByVal param As IDataParameter) cmd.Parameters.Add(param)  
End Sub  
  
Public Property ConnectionString() As String Get  
    Return strConnectionString  
End Get  
Set(ByVal value As String) strConnectionString = value  
End Set  
End Property  
  
Private Sub Open() cmd.Connection.Open()  
End Sub  
  
Private Sub Close() cmd.Connection.Close()  
End Sub  
  
Public Property HandleExceptions() As Boolean Get  
    Return handleErrors  
End Get  
Set(ByVal value As Boolean) handleErrors = value  
End Set  
End Property  
  
Public ReadOnly Property LastError() As String Get  
    Return strLastError  
End Get  
End Property  
  
Public Sub Dispose() cmd.Dispose()  
End Sub End Class
```

4.4.1.2 ITERACIÓN II

SPIKE DE LAS HISTORIAS DEL USUARIO

A con tinuacion se muestran los formulario prueba que se utilizaron en la iteración II

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

REGISTRO DE USUARIO

Registro:

A. Paterno: A. Materno:

Nombres: DNI: Sexo: M F

Dirección: Fecha Nacimiento: / /

Celular: Departamento:

Provincia: Distrito:

Login: Password: Confirmar Password:

Mostrar Password

Búsqueda:

Buscar por Razón Social o RUC:

	A. Paterno	A. Materno	Nombres	DNI	Dirección	F. Nacimiento
▶	VILLA	PEREZ	CESAR IVAN	46184593	PROLOG MARISCAL BENAVIDES 1173	21/02/1990
	CASTILLON	SIGUAS	LUIS	45217896	AV. UNION NRO 363	15/04/1993

Spike registro de usuario al Sistema

REGISTRO DE PERFIL

Registro:

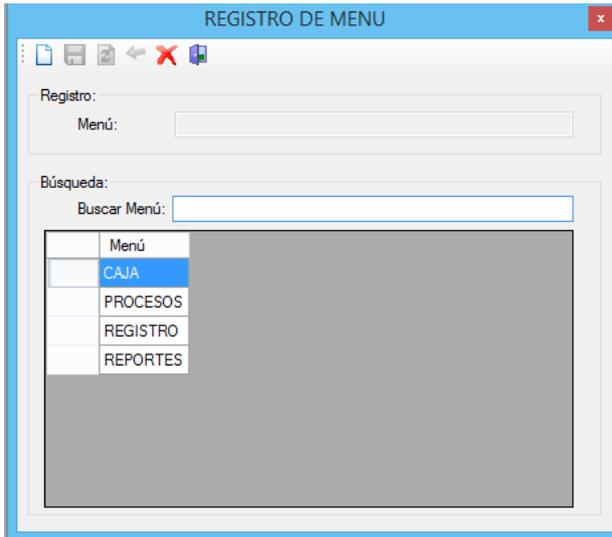
Nombre Perfil:

Búsqueda:

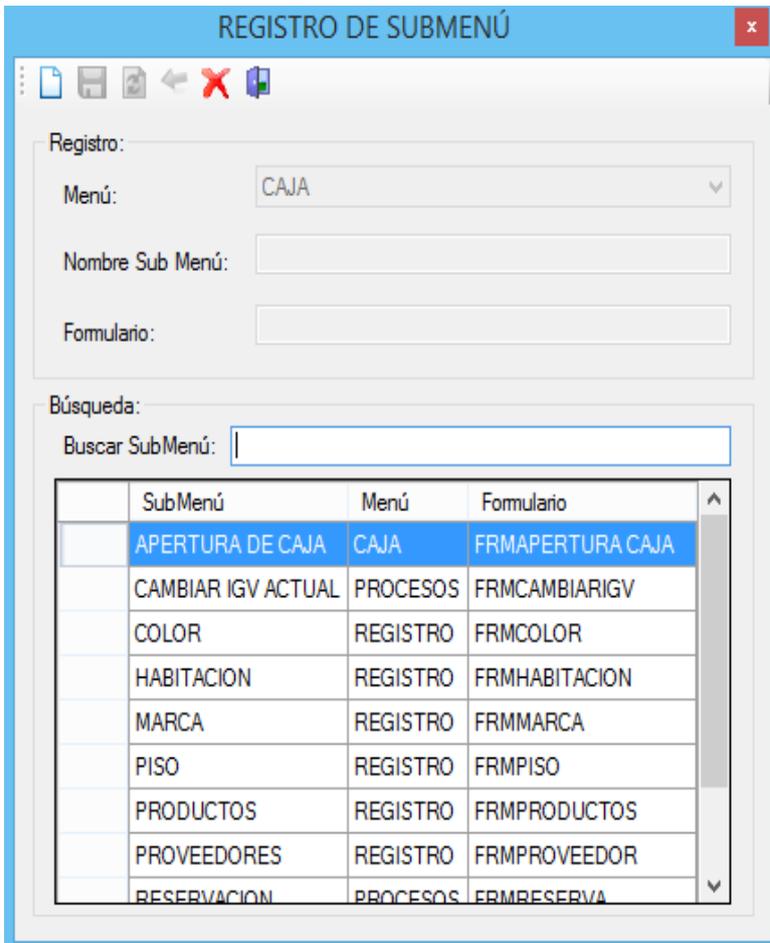
Buscar Perfil:

	Perfil
	ADMINISTRADOR
	DIGITADOR
	RECEPCIONISTA

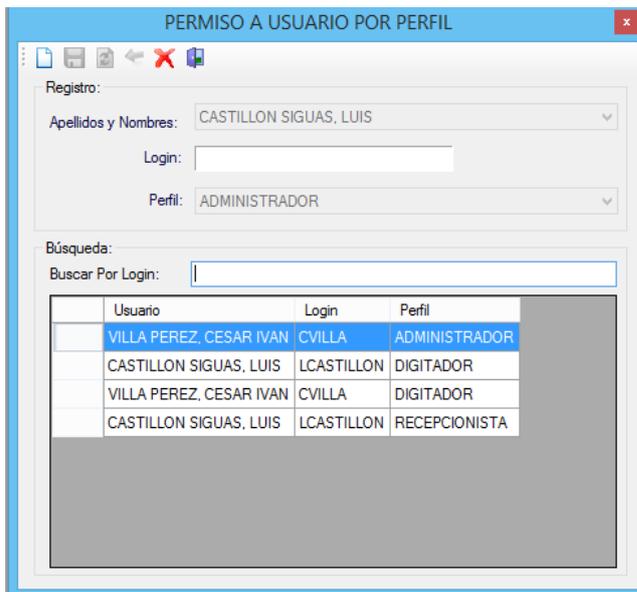
Spike registro de nuevo perfil de usuario



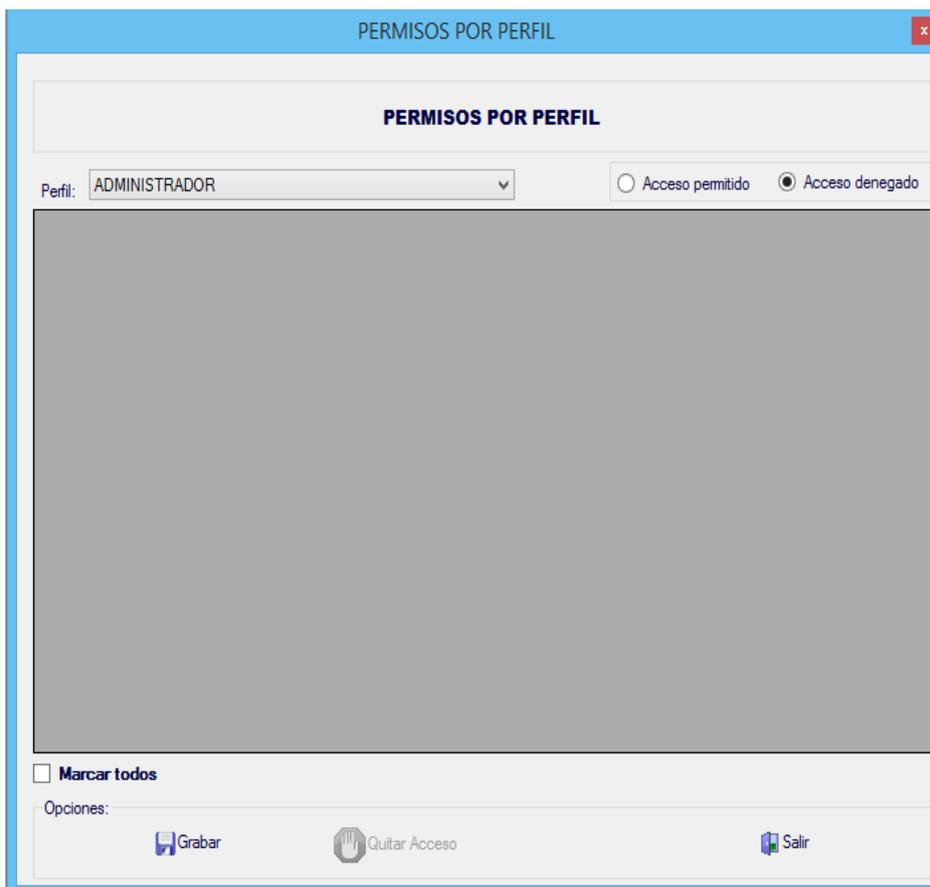
Spike registro de menus



Spike Registro de submenus



Spike asignación de perfil a usuario



Spike Permisos/operaciones por perfil



Spike cambio de IG

TAREAS DE LAS HISTORIAS DEL USUARIO

TAREA DE USUARIO	
Número de Tarea: 18	Historia de Usuario (Número y Nombre): Historia 18, Registro / Mantenimiento usuario al sistema.
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los usuarios del sistema	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 27/05/2015	Fecha de Fin: 01/06/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar los datos de las historias del usuario.	

TAREA DE USUARIO	
Número de Tarea: 19	Historia de Usuario (Número y Nombre): Historia 19, Registro / Mantenimiento de nuevo perfil.
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los perfil de usuario	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 02/06/2015	Fecha de Fin: 05/06/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar datos de los nuevos perfiles de usuario.	

TAREA DE USUARIO	
Número de Tarea: 20	Historia de Usuario (Número y Nombre): Historia 20, Registro / Mantenimiento de Menu
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los datos de los menus.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 08/06/2015	Fecha de Fin: 12/06/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar datos de los nuevos perfiles de usuario.	

TAREA DE USUARIO	
Número de Tarea: 21	Historia de Usuario (Número y Nombre): Historia 21, Registro / Mantenimiento operaciones(submenu)
Nombre de Tarea: Diseñar una estructura de datos que permita registrar las operaciones de los usuarios.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 15/06/2015	Fecha de Fin: 22/06/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar datos de las operaciones de usuarios.	

TAREA DE USUARIO	
Número de Tarea: 22	Historia de Usuario (Número y Nombre): Historia 22, Registro / Mantenimiento asignación de perfil a usuario.
Nombre de Tarea: Diseñar una estructura de datos que permita registrar las asignacion de perfil a usuario.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 23/06/2015	Fecha de Fin: 30/06/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar datos de las asignaciones de perfil a usuario.	

TAREA DE USUARIO	
Número de Tarea: 23	Historia de Usuario (Número y Nombre): Historia 22, Registro / Mantenimiento

asignación de operaciones a perfiles.	
Nombre de Tarea: Diseñar una estructura de datos que permita registrar las asignacion de operaciones a perfil	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 01/07/2015	Fecha de Fin: 17/07/2015
Programador Responsable: Castillon Siguas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita almacenar información de las asignaciones de operaciones a un determinado perfil.	

TAREA DE USUARIO	
Número de Tarea: 24	Historia de Usuario (Número y Nombre): Historia 24, Cambio de IGV
Nombre de Tarea: Diseñar una estructura de datos que permita registrar las asignacion de operaciones a perfil	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 20/07/2015	Fecha de Fin: 10/08/2015
Programador Responsable: Castillon Siguas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita actualizar el IGV.	

INTERFAZ DE LAS HISTORIAS DEL USUARIO

REGISTRO DE USUARIO

Registro De Nuevo Usuario

Registro:

A. Paterno: A. Materno:

Nombres: DNI: Sexo: M F

Dirección: Fecha Nacimiento: 01/01/1915

Celular: Departamento: AMAZONAS

Provincia: BAGUA Distrito: ARAMANGO

Login: Password: Confirmar Password:

Mostrar Password

Búsqueda:

Buscar por Apellido o DNI:

A. Paterno	A. Materno	Nombres	DNI	Dirección	F. Nacimiento
VILLA	PEREZ	CESAR IVAN	46184593	PROLOG MARISCAL BENAVIDES 1173	21/02/1990
CASTILLON	SIGUAS	LUIS	45217896	AV. UNION NRO 363	15/04/1993

Nuevo Guardar Modificar Cancelar Eliminar Salir Bloquear Usuario Desbloquear Usuario

Interfaz de registro de usuario



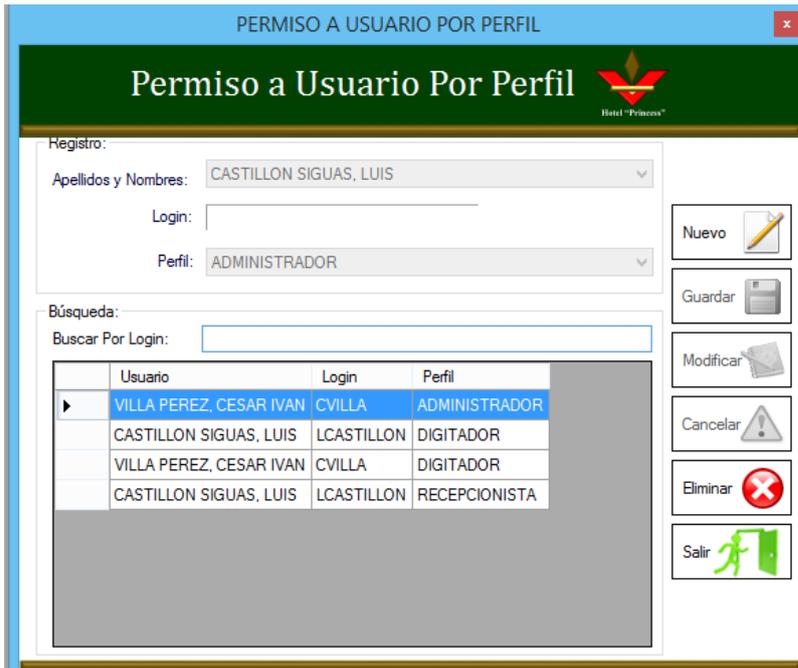
Interfaz de registro de nuevo perfil



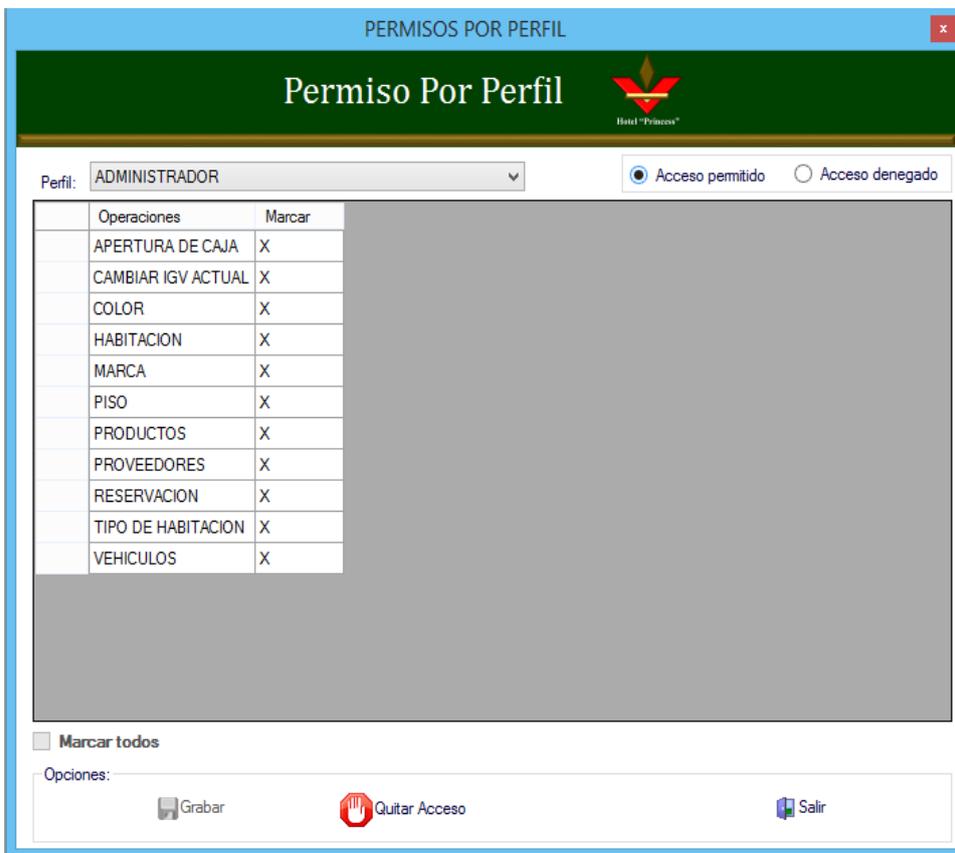
Interfaz registro de menus al sistema



Registro de operaciones (submenus)



Interfaz asignación de perfil a usuario



Interfaz asignación de permisos por perfil



Interfaz cambio de IGV

CODIFICACIÓN

BOL (OBJETOS DE LA LOGICA DEL NEGOCIO) CLASE USUARIO

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsUsuario : Inherits clsPersona
#Region "Variables"
Private sidUsuario As String Private sLogeo As String Private sClave As String
Private sEstado As Boolean Private iNroSesion As Integer
#End Region

#Region "Propiedades"
Public Property _idUsuario() As String Get
Return sidUsuario
End Get
Set(ByVal value As String)
sidUsuario = value
End Set End Property

Public Property _logeio() As String Get
Return sLogeo
End Get
Set(ByVal value As String) sLogeo = value
End Set End Property
```

```
Public Property _clave() As String Get
    Return sClave
End Get
Set(ByVal value As String) sClave = value
End Set End Property

Public Property _Estado() As Boolean Get
    Return sEstado
End Get
Set(ByVal value As Boolean) sEstado = value
End Set
End Property

Public Property _NroSesion() As Integer Get
    Return iNroSesion
End Get
Set(ByVal value As Integer) iNroSesion = value
End Set End Property
#End Region

Public Function verificarExistenciaUsuario(ByVal xLogeo As String) As Integer
Dim db As New dbAccess

db.AddParameter("@Login", xLogeo)

    Dim xValor As Integer = db.ExecuteScalar("sp_ValidarUsuarioExistente")

Return xValor
End Function

Public Function usuarioDevolverClave(ByVal xLogeo As String) As String
Dim db As New dbAccess

db.AddParameter("@Logeo", xLogeo)

Return db.ExecuteScalar("SP_Mostrar_Logeo").ToString
End Function

Public Function verificarEstado(ByVal xLogeo As String, ByVal xNop As Char) As
Integer
Dim db As New dbAccess

db.AddParameter("@nOP", xNop) db.AddParameter("@Login", xLogeo)

    Dim xValor As Integer = db.ExecuteScalar("SP_Verificar_Estado_Usuario")

Return xValor
```

End Function

```
Public Function UsuarioSelectAllByName(ByVal xFiltro As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@filtro", xFiltro)
Return db.ExecuteDataSet("SP_MostrarUsuario")
End Function
```

```
Public Sub MantenimientoUsuario(ByVal x As clsUsuario,
ByVal v1Proceso As Char)
Dim db As New dbAccess
```

```
db.AddParameter("@Proceso", v1Proceso) db.AddParameter("@idUsuario", x._idUsuario)
db.AddParameter("@idPersona", x._sidPersona) db.AddParameter("@idDistrito",
x._sIdDistrito) db.AddParameter("@apePat", x._sapePat) db.AddParameter("@apeMat",
x._sapeMat) db.AddParameter("@nombres", x._snombres) db.AddParameter("@Dni",
x._snroIden) db.AddParameter("@direccion", x._sdireccion)
db.AddParameter("@fecNac", x._dFechaNac) db.AddParameter("@sexo", x._cSexo)
db.AddParameter("@telf1", x._sTelf1) db.AddParameter("@Login", x._logeio)
db.AddParameter("@Password", x._clave) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)
```

```
db.ExecuteNonQuery("SP_Mant_Usuario")
End Sub
```

```
Public Function validarDobleIngreso(ByVal xDni As String, ByVal xNop As String)
As String
Dim db As New dbAccess
```

```
db.AddParameter("@Dni", xDni) db.AddParameter("@nOp", xNop)
```

```
Dim xValor As Integer = db.ExecuteScalar("SP_ValidarDobleUsuario")
```

```
Return xValor.ToString
End Function
```

```
Public Sub ActivarUsuario(ByVal xNroIden As String)
Dim db As New dbAccess
```

```
db.AddParameter("@Dni", xNroIden) db.ExecuteNonQuery("SP_ActivarUsuario")
End Sub
```

```
Public Function mostrarUsuarioPerfil() As DataSet Dim db As New dbAccess
db.AddParameter("@nOp", "1") db.AddParameter("@idUsuario", "")
```

```
Return db.ExecuteDataSet("sp_MostrarUsuarioPerfil")
End Function
```

```
Public Function mostrarloginUsuario(ByVal x As clsUsuario)
As String
Dim db As New dbAccess
db.AddParameter("@nOp", "2") db.AddParameter("@idUsuario", x._idUsuario)
Return
db.ExecuteScalar("sp_MostrarUsuarioPerfil").ToString
End Function
```

```
Public Sub BloquearUsuario(ByVal xLogeo As String) Dim db As New dbAccess

db.AddParameter("@login", xLogeo) db.ExecuteNonQuery("SP_BloquearUsuario")
End Sub
```

```
Public Function usuarioMostrarNonbre(ByVal x As clsUsuario)
As DataSet
Dim db As New dbAccess

db.AddParameter("@Proceso", "0") db.AddParameter("@idUsuario", x._idUsuario)
db.AddParameter("@Login", "") db.AddParameter("@PassWord", "")

Return db.ExecuteDataSet("SP_MostrarUsuarioClave")
End Function
```

```
Public Sub usuarioResetearClave(ByVal x As clsUsuario)
Dim db As New dbAccess

db.AddParameter("@Proceso", "1") db.AddParameter("@idUsuario", x._idUsuario)

Dim xLogin As New SqlParameter("@Login", SqlDbType.NVarChar, 20)
xLogin.Direction = ParameterDirection.Output db.Parameters.Add(xLogin)

Dim xPass As New SqlParameter("@PassWord", SqlDbType.NVarChar, 15)
xPass.Direction = ParameterDirection.Output db.Parameters.Add(xPass)

db.ExecuteNonQuery("SP_MostrarUsuarioClave") vgLogin = xLogin.Value.ToString
vgPassword = xPass.Value.ToString
End Sub
```

```
Public Sub usuarioProcesarCambioClave(ByVal x As
clsUsuario)
Dim db As New dbAccess

db.AddParameter("@idUsuario", x._idUsuario) db.AddParameter("@nuevoPass", x._clave)
```

```
db.ExecuteNonQuery("SP_CambiarClaveUsuario")
End Sub
End Class
```

CLASE PERFIL

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsPerfil
#Region "Variables"
Private sCodigo As String
Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _codigo() As String Get
Return sCodigo
End Get
Set(ByVal value As String) sCodigo = value
End Set End Property

Public Property _nombre() As String Get
Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set
End Property
#End Region

Public Function PerfilSelectAllByName(ByVal x As clsPerfil)
As DataSet
Dim db As New dbAccess
db.AddParameter("@Perfil", x._nombre)
Return db.ExecuteDataSet("SP_MostrarPerfil")
End Function

Public Sub MantenimientoPerfil(ByVal x As clsPerfil, ByVal
v1Proceso As Char)
Dim db As New dbAccess
db.AddParameter("@Proceso", v1Proceso) db.AddParameter("@Codigo", x._codigo)
db.AddParameter("@nomPerfil", x._nombre) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Perfil")
End Sub
End Class
```

CLASE MENU

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsMenu
#Region "Variables"
Private sidMenu As Integer
Private sNombre As String
#End Region

#Region "Propiedades"
Public Property _idMenu() As Integer Get
Return sidMenu
End Get
Set(ByVal value As Integer) sidMenu = value
End Set End Property

Public Property _nombre() As String Get
Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set
End Property
#End Region

Public Function MenuSelectAllByName(ByVal x As clsMenu) As
DataSet
Dim db As New dbAccess
db.AddParameter("@nomMenu", x._nombre)
Return db.ExecuteDataSet("SP_MostrarMenu")
End Function

Public Sub MantenimientoMenu(ByVal x As clsMenu, ByVal
v1Proceso As Char)
Dim db As New dbAccess
db.AddParameter("@Proceso", v1Proceso) db.AddParameter("@idMenu", x._idMenu)
db.AddParameter("@nomMenu", x._nombre) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Menu")
End Sub
End Class
```

CLASE SUBMENÚ (OPERACIONES)

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsSubMenu
```

```
#Region "Variables"
Private sCodigo As Integer Private sidMenu As Integer Private sNombre As String
Private snomForm As String
#End Region

#Region "Propiedades"
Public Property _codigo() As Integer
    Get
        Return sCodigo
    End Get
Set(ByVal value As Integer)
    sCodigo = value
End Set End Property

Public Property _sIdMenu() As Integer
    Get
        Return sidMenu
    End Get
Set(ByVal value As Integer)
    sidMenu = value
End Set End Property

Public Property _nombre() As String
    Get
        Return sNombre
    End Get
Set(ByVal value As String)
    sNombre = value
End Set End Property

Public Property _snomForm() As String
    Get
        Return snomForm
    End Get
Set(ByVal value As String)
    snomForm = value
End Set
End Property
#End Region

Public Function SubMenuSelectAllByName(ByVal x As clsSubMenu) As DataSet
    Dim db As New dbAccess
    db.AddParameter("@SubMenu", x._nombre)
    Return db.ExecuteDataSet("SP_MostrarSubMenu")
End Function

Public Sub MantenimientoSubMenu(ByVal x As clsSubMenu,
    ByVal vlProceso As Char)
    Dim db As New dbAccess
    db.AddParameter("@Proceso", vlProceso)
    db.AddParameter("@Codigo", x._codigo)
    db.AddParameter("@idMenu", x._sIdMenu)
    db.AddParameter("@NomSubMenu", x._nombre)
    db.AddParameter("@nomForm", x._snomForm)
```

```
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_SubMenu")
End Sub End Class
```

CLASE PERFIL USUARIO

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsPerfilUsuario
#Region "Variables"
Private sidPermiso As String Private sidUsuario As String Private sidPerfil As String
#End Region

#Region "Propiedades"
Public Property _idPermiso() As String Get
Return sidPermiso
End Get
Set(ByVal value As String) sidPermiso = value
End Set End Property

Public Property _idUsuario() As String Get
Return sidUsuario
End Get
Set(ByVal value As String) sidUsuario = value
End Set
End Property

Public Property _idPerfil() As String Get
Return sidPerfil
End Get
Set(ByVal value As String) sidPerfil = value
End Set
End Property

#End Region

Public Function PerfilSelectAllByName(ByVal xLogeo As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@logeio", xLogeo)
Return db.ExecuteDataSet("SP_MostrarioUsuarioPorPerfil")
End Function
```

```
Public Sub MantenimientoPerfilUsuario(ByVal x As
clsPerfilUsuario, ByVal vlProceso As Char) Dim db As New dbAccess
db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idPermiso", x._idPermiso) db.AddParameter("@idUsuario",
x._idUsuario) db.AddParameter("@idPerfil", x._idPerfil) db.AddParameter("@Usr",
vgUsr) db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Perminos")
End Sub

Public Function validarDobleIngreso(ByVal xIdUsuario As String, ByVal xIdPerfil
As String, ByVal xNop As String) As String
Dim db As New dbAccess

db.AddParameter("@idUsuario", xIdUsuario) db.AddParameter("@idPerfil", xIdPerfil)
db.AddParameter("@nOp", xNop)

Dim xValor As Integer = db.ExecuteScalar("SP_ValidarDobleUsuarioPerfil")

Return xValor.ToString
End Function

Public Sub ActivarPermiso(ByVal xIdUsuario As String, ByVal
xIdPerfil As String)
Dim db As New dbAccess

db.AddParameter("@idUsuario", xIdUsuario) db.AddParameter("@idPerfil", xIdPerfil)

db.ExecuteNonQuery("SP_ActivarPermisoUsuario")
End Sub
End Class
```

CLASE OPERACIONES POR PERFIL

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsPermisoPerfil
#Region "Variables"
Private sidDetalle As String Private sidPerfil As String Private sidSubMenu As
String
#End Region

#Region "Propiedades"
Public Property _idDetalle() As String
Get
Return sidDetalle
End Get
Set(ByVal value As String)
```

```

        sidDetalle = value
    End Set End Property

Public Property _idPerfil() As String Get
    Return sidPerfil
End Get
Set(ByVal value As String)
    sidPerfil = value
End Set End Property

Public Property _idSubMenu() As String Get
    Return sidSubMenu
End Get
Set(ByVal value As String) sidSubMenu = value
End Set End Property
#End Region

Public Function mostrarPermisos(ByVal x As
clsPermisoPerfil, ByVal xNop As Byte) As DataSet Dim db As New dbAccess
db.AddParameter("@idPerfil", x._idPerfil) db.AddParameter("@Nop", xNop)
Return db.ExecuteDataSet("SP_MostrarPermisosXPerfil")
End Function

Public Sub GrabarPermiso(ByVal x As clsPermisoPerfil)
Dim db As New dbAccess db.AddParameter("@Cad01", vgCadValor)
db.AddParameter("@idPerfil", x._idPerfil) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_GenerarPermisoPerfil")
End Sub

Public Function verificarPermiso(ByVal xSubMenu As String) As Byte
Dim db As New dbAccess

db.AddParameter("@Login", vgUsr) db.AddParameter("@idSubMenu", xSubMenu)

Dim xValor As Byte = db.ExecuteScalar("SP_VerificarPermiso")

Return xValor
End Function

Public Sub QuitarPermiso(ByVal x As clsPermisoPerfil) Dim db As New dbAccess
db.AddParameter("@idPerfil", x._idPerfil) db.AddParameter("@idSubMenu",
x._idSubMenu)

```

```

db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_QuitarAccesoPerfil")
End Sub End Class

```

CLASE IGV

```

Imports System.Data
Imports System.Data.SqlClient

Public Class clsIGV
Public Function verIGVInicial() As Double Dim db As New dbAccess

    Dim xValor As Double = db.ExecuteScalar("SP_IGVInicial")

Return xValor
End Function

Public Sub cambiar_nuevo_IGV(ByVal xIGV As String) Dim db As New dbAccess
db.AddParameter("@newIGV", xIGV)

db.ExecuteNonQuery("SP_CambiarNuevoIGV")
End Sub End Class

```

4.4.1.3 INTERACCIÓN III

SPIKE PARA LAS HISTORIAS DEL USUARIO

En esta itación se omitio realizar spike o prototipos de la interfaz, si omitieron estas ya que en las iteraciones anteriores se concretaron con el interfaz final.

TAREAS DE LAS HISTORIAS DEL USARIO

TAREA DE USUARIO	
Número de Tarea: 25	Historia de Usuario (Número y Nombre): Historia 25, Reservación / Alquiler
Nombre de Tarea: Diseñar una estructura de datos que permita registrar las asigancion de operaciones de reservación y alquiler.	
Tipo de Tarea: Desarrollo	

Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 11/08/2015	Fecha de Fin: 13/08/2015
Programador Responsable: Castillon Siguas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los operaciones de reservación y alquiler de habitaciones.	

TAREA DE USUARIO	
Número de Tarea: 26	Historia de Usuario (Número y Nombre): Historia 26, Serevicio cochera
Nombre de Tarea: Diseñar una estructura de datos que permita registrar las asignacion de operaciones de servicio de cochera.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 15/08/2015	Fecha de Fin: 17/08/2015
Programador Responsable: Fernandez Cahuan Smith Ruben	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los operaciones del servicio de cochera.	

TAREA DE USUARIO	
Número de Tarea: 27	Historia de Usuario (Número y Nombre): Historia 27, Cambiar estado de habitaciones.
Nombre de Tarea: Diseñar una estructura de datos que permita cambiar el estado de las habitaciones.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 18/08/2015	Fecha de Fin: 21/08/2015
Programador Responsable: Castillon Siguas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita cambiar o actualizar el estado de las habitaciones.	

TAREA DE USUARIO	
Número de Tarea: 28	Historia de Usuario (Número y Nombre): Historia 28, apertura de caja
Nombre de Tarea: Diseñar una estructura de datos que permita realizar la apertura de caja del dia.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 24/08/2015	Fecha de Fin: 26/08/2015
Programador Responsable: Castillon Siguas Luis	

Descripción: Se diseña e implementa una estructura de datos que permita realizar la apertura de caja.

TAREA DE USUARIO	
Número de Tarea: 29	Historia de Usuario (Número y Nombre): Historia 29, cierre de caja
Nombre de Tarea: Diseñar una estructura de datos que permita realizar el cierre de caja del día.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 27/08/2015	Fecha de Fin: 28/08/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita realizar el cierre de caja.	

TAREA DE USUARIO	
Número de Tarea: 30	Historia de Usuario (Número y Nombre): Historia 30, ingreso de caja
Nombre de Tarea: Diseñar una estructura de datos que permita realizar el ingreso de caja.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 31/08/2015	Fecha de Fin: 02/09/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los ingresos día fuera de los alquiers.	

TAREA DE USUARIO	
Número de Tarea: 31	Historia de Usuario (Número y Nombre): Historia 31, egreso caja
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los egresos de caja.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 03/09/2015	Fecha de Fin: 04/09/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los egresos de caja durante el día.	

TAREA DE USUARIO	
Número de Tarea: 32	Historia de Usuario (Número y Nombre): Historia 32, concepto ingreso y salida
Nombre de Tarea: Diseñar una estructura de datos que permita registrar los diferentes tipos de conceptos de I/S.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 07/09/2015	Fecha de Fin: 08/09/2015
Programador Responsable: Castellon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita registrar los conceptos de ingreso y salida.	

TAREA DE USUARIO	
Número de Tarea: 33	Historia de Usuario (Número y Nombre): Historia 33, Listado de caja detallado
Nombre de Tarea: Diseñar una estructura de datos que permita mostrar el listado de movimientos de caja.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 09/09/2015	Fecha de Fin: 10/09/2015
Programador Responsable: Castellon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que mostrar los movimientos de caja durante el día.	

TAREA DE USUARIO	
Número de Tarea: 34	Historia de Usuario (Número y Nombre): Historia 34, Listado de cierre de caja realizados por usuario.
Nombre de Tarea: Diseñar una estructura de datos que permita listar los cierre de caja realizados.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 11/09/2015	Fecha de Fin: 15/09/2015
Programador Responsable: Castellon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita listar los cierres de caja realizados.	

TAREA DE USUARIO	
Número de Tarea: 35	Historia de Usuario (Número y Nombre): Historia 35, resumen del día.
Nombre de Tarea: Diseñar una estructura de datos que permita mostrar los datos del día (caja).	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 16/09/2015	Fecha de Fin: 18/09/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita mostrar todos los movimientos de caja del día.	

TAREA DE USUARIO	
Número de Tarea: 36	Historia de Usuario (Número y Nombre): Historia 36, Anulación de servicio.
Nombre de Tarea: Diseñar una estructura de datos que permita anular servicios.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 21/09/2015	Fecha de Fin: 24/09/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita anular servicio, por errores que puedan ocurrir dentro de la empresa.	

TAREA DE USUARIO	
Número de Tarea: 37	Historia de Usuario (Número y Nombre): Historia 37, Listado de servicios anulados.
Nombre de Tarea: Diseñar una estructura de datos que permita anular servicios.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 21/09/2015	Fecha de Fin: 24/09/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita listar los servicios anulados.	

TAREA DE USUARIO	
Número de Tarea: 38	Historia de Usuario (Número y Nombre): Historia 38, Listado de documentos generados
Nombre de Tarea: Diseñar una estructura de datos que permita listar los tipos de comprobante generados.	
Tipo de Tarea: Desarrollo Desarrollo/Corrección/Mejora/Otra (Especificar)	
Fecha de Inicio: 01/10/2015	Fecha de Fin: 05/10/2015
Programador Responsable: Castillon Sigvas Luis	
Descripción: Se diseña e implementa una estructura de datos que permita listar los documentos generados por rango de fecha ya sean boletas o facturas.	

INTERFAZ DE LAS HISTORIAS DEL USUARIO

Interfaz reservación/ alquiler

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

SERVICIO DE COCHERA

Registro De Cochera

Registro | Listado

Tipo Documento:
 Boleta Factura

Cliente: Dirección: Identificación:

Tipo Vehículo: Nº Placa: Marca:

Tipo Parqueo: Ubicación: Precio:

Fecha Ingreso: 14/12/2015 Fecha Salida: 15/12/2015 Costo: Tipo Pago:
 Contado A cuenta: 0.00 Crédito Saldo: 0.00

Sub Total: 0.00
Servicios 10%: 0.00
I.G.V. 18.00 %: 0.00
Total a Pagar: 0.00

Descripción:

Vehículo	Descripcion	Parqueo	Ubicacion	Precio
----------	-------------	---------	-----------	--------

Interfaz servicio cochera

HABITACIONES POR LIMPIAR

Habitaciones Por Limpiar

ID	Nro. Habitación	Nro. Piso	Tipo Habitación	Precio	Marcar
----	-----------------	-----------	-----------------	--------	--------

Marcar todos

Opciones:

Interfaz cambiar estado habitaciones

APERTURA DE CAJA

Apertura De Caja

Hotel "Princess"

Usuario: CVILLA

Monto:

Observacion:

Guardar Salir

Interfaz apertura caja

CIERRE DE CAJA

Registro De Cierre De Caja por Usuario

Hotel "Princess"

FECHA INICIO: 14/12/2015

USUARIO: CVILLA

Mostrar Datos

TOTAL PAGO EFECTIVO: S/. 0.00

TOTAL APERTURA CAJA: S/. 0.00

TOTAL INGRESOS: S/. 0.00

TOTAL RESERVA Y COCHERA: S/. 0.00

TOTAL DINERO: S/. 0

TOTAL EGRESOS: S/. 0.00

TOTAL TARJETAS: S/. 0.00

TOTAL DESCUENTO: S/. 0.00

TOTAL EN CAJA: S/. 0.00

TOTAL SALDO: S/. 0.00

TOTAL ENTREGADO: S/. 0.00

Arqueo Caja

Salir

SALDO DE APERTURA PARA EL SIGUIENTE TURNO

Interfaz cierre de caja

REGISTRO DE INGRESOS

Ingresos De Caja

Tipo Documento: Boleta Factura

Forma De Pago: EFECTIVO

Entidad Financiera: BANBIF N° OP:

Identificación:

Ciente:

Dirección:

Concepto: CANCELACION DE CREDITOS MENSUAL

Detalle

Sub Total: 0.00
Servicios 10%: 0.00
I.G.V. 18.00 %: 0.00
Total a Pagar: 0.00

Monto a Pagar:

Salir

Interfaz ingreso caja

REGISTRO DE EGRESOS DE CAJA

Registro De Egresos

Documento: Numero Recibo: Dinero Entregado: EFECTIVO

Ciente:

Dirección:

Identificación:

Concepto: COMPRA DE ARTICULOS DE LIMPIEZA PARA LAS HABITACIONES

Descripción:

Total a Pagar:

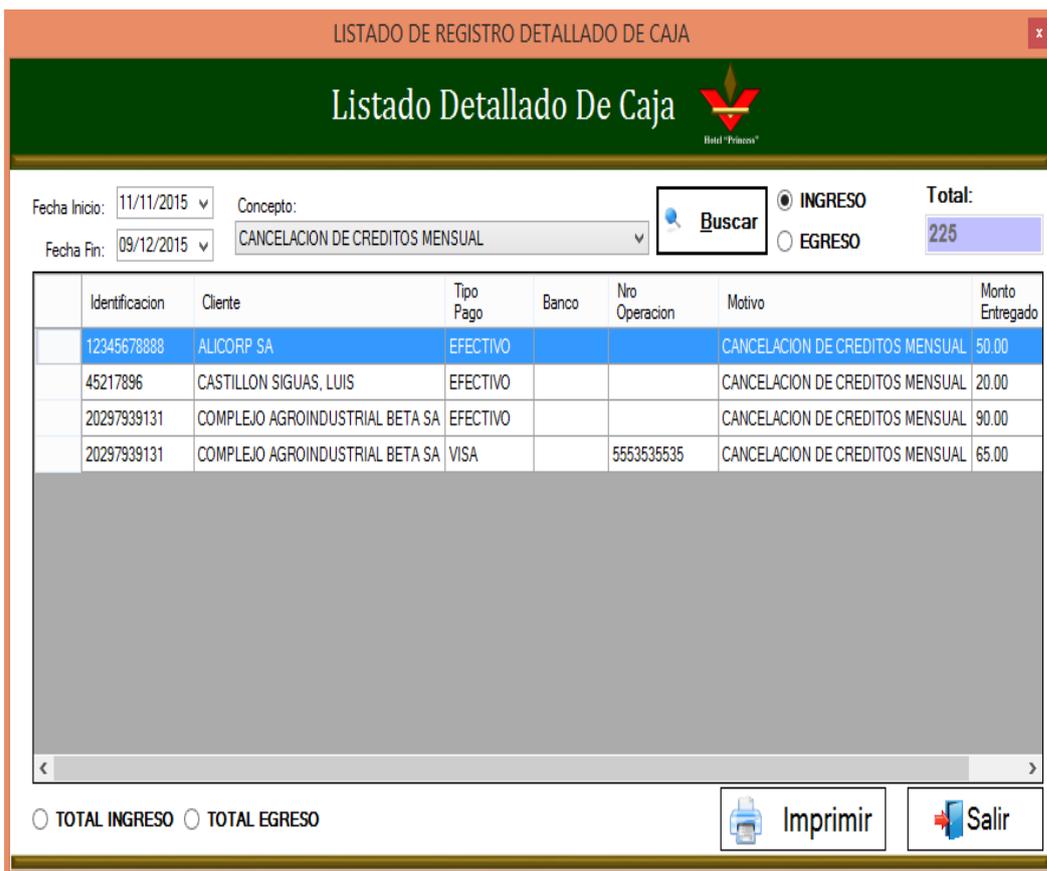
Nuevo
Guardar
Cancelar
Salir

Interfaz egreso caja

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.



Interfaz Concepto ingreso y salida



Interfaz Listado detallado de caja

**SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.**

LISTADO DE CIERRE CAJA REALIZADO

Listado De Cierre De Caja Realizados

Fecha Inicio: 01/12/2015 Fecha Fin: 14/12/2015 Usuario: CVILLA

Fecha	Usuario	Saldo	Saldo Entregado	Descripcion Saldo	Total Apertura	Pago Efectivo	Total Ingreso	Reserva y Cochera	Total Dinero
08/12/2015 04:15 p.m.	CVILLA	45.00	LCASTILLON	TODO EXACTO	5.00	0.00	0.00	2300.00	2305.00

TOTAL : Saldo: 45 Apertura Caja: 5 Pago Efectivo: 0 Total Ingreso: 0 Reserva y Cochera: 2300 Total Dinero: 2305 Total Egreso: 0 Tarjeta y Banco: 0 Total Dcto: 0 Total Caja: 2305

Mostrar Todos Los Cierres De Caja

Interfaz listado de cierre de caja realizados

DINERO TOTAL DEL DIA

Resumen De Dinero Ingresado Diario

Fecha Inicio: 01/12/2015 Fecha Fin: 14/12/2015

Tipo	Total Reserva	Total Consumo	Servicio Lavandería	Total Neto
EFFECTIVO	2780.99	479.00	2858.00	6117.99
VISA	0.00	115.00	116.00	231.00
RESERVA CONTADO	1740.00	0.00	0.00	1740.00
RESERVA CREDITO	0.00	0.00	0.00	0.00
COCHERA CONTADO	3740.30	0.00	0.00	3740.30
COCHERA CREDITO	0.00	0.00	0.00	0.00
INGRESO DEUDA RESERVA	0.00	0.00	0.00	0.00
INGRESO DEUDA COCHERA	0.00	0.00	0.00	0.00
INGRESO OTROS	0.00	0.00	0.00	0.00
EGRESOS TOTAL	0.00	0.00	0.00	0.00
APERTURA CAJA	0.00	0.00	0.00	214.40

Total Consumo: 594 Servicio Lavandería: 2974 Total Dinero: 12043.69

Interfaz resumen del dia

**SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.**

ANULACION DE SERVICIO GENERADOS

Anulacion De Servicios Generados

Buscar:

Legenda: ■ Por cancelar ■ Cancelado

	Ciente	Dirección	Identificación	Nro. Habitación	Piso	Tipo Habitación	Precio Diario
	VILLA PEREZ, CESAR IVAN	PROLOG MARISCAL BENAVIDES 1173	46184593	5	1	DOBLE	80.00
	ALICORP SA	CALLE GRAU NRO 250	12345678888	3	1	DOBLE	80.00

Interfaz anulación de servicios

LISTADO DE PRODUCTO ELIMINADOS

Listado De Productos Anulados

Fecha Inicio: 09/12/2015 Fecha Fin: 14/12/2015
 CONSUMO LAVANDERIA

	Identificación	Ciente	Tipo Habitación	Nro Habitación	Nro Piso	Cantidad	Producto	P. Unitario	Total	Resp	Fecha
	46184593	VILLA PEREZ, CESAR IVAN	DOBLE	5	1	1.00	PIÑA COLADA	8.00	8.0000	CVILLA	09/12/2015

Mostrar Todos Los Productos Total Dinero: 8

Interfaz Listado de servicios anulados

SISTEMA COMPUTARIZADO PARA LA ADMINISTRACIÓN DEL RUBRO HOTELERO
DE LA EMPRESA DE SERV. TURÍSTICO PRINCESS S.A.

LISTADO DE DOCUMENTOS GENERADOS													
Listado De Documento Generados													
Fecha Inicio:		01/10/2015		Fecha Fin:		14/12/2015							
Cliente:				<input type="text"/>		<input checked="" type="radio"/> BOLETA <input type="radio"/> FACTURA		<input type="button" value="Buscar"/>					
Identificación:		<input type="text"/>		Nro Documento:		<input type="text"/>							
fecha	Tipo	N° Documento	Identificación	Cliente	Dirección	Total Reserva	Total Consumo	Total Lavande					
16/11/2015	BOLETA	0042-0000005	12345678888	ALICORP SA	CALLE GRAU NRO 250	70.00	0.00	0.00					
16/11/2015	BOLETA	0006-0000006	12345678888	ALICORP SA	CALLE GRAU NRO 250	80.00	0.00	0.00					
19/11/2015	BOLETA	0443-0000005	20297939131	COMPLEJO AGROINDUSTRIAL BETA SA	CALLE LEOPOLDO CARRILLO NRO 160	90.00	0.00	0.00					
19/11/2015	BOLETA	0005-0000005	12345678888	ALICORP SA	CALLE GRAU NRO 250	80.00	0.00	0.00					
19/11/2015	BOLETA	0056-0000005	12345678888	ALICORP SA	CALLE GRAU NRO 250	80.00	0.00	0.00					
19/11/2015	BOLETA	0056-0000006	20297939131	COMPLEJO AGROINDUSTRIAL BETA SA	CALLE LEOPOLDO CARRILLO NRO 160	90.00	0.00	0.00					
19/11/2015	BOLETA	0005-0000005	12345678888	ALICORP SA	CALLE GRAU NRO 250	70.00	0.00	0.00					
19/11/2015	BOLETA	0005-0000006	20297939131	COMPLEJO AGROINDUSTRIAL BETA SA	CALLE LEOPOLDO CARRILLO NRO 160	120.00	0.00	0.00					
19/11/2015	BOLETA	0005-0000005	20297939131	COMPLEJO AGROINDUSTRIAL BETA SA	CALLE LEOPOLDO CARRILLO NRO 160	80.00	0.00	0.00					
19/11/2015	BOLETA	0005-0000005	20297939131	COMPLEJO AGROINDUSTRIAL BETA SA	CALLE LEOPOLDO CARRILLO NRO 160	90.00	0.00	0.00					
19/11/2015	BOLETA	0065-0000005	12345678888	ALICORP SA	CALLE GRAU NRO 250	50.00	0.00	0.00					
Total registros: 162						TOTAL :		Total Consumo: 474	Servicio Lavandera: 3250	Sub Total: 0.00	Total Comision: 0.00	Total I.G.V.: 0.00	Total Dinero: 15192.3
						<input type="button" value="Imprimir"/>		<input type="button" value="Salir"/>					

Interfaz listado de documento generados

CODIFICACIÓN

BOL (OBJETOS DE LA LÓGICA DEL NEGOCIO)

CLASE RESERVACION/COCHERA

Imports System.Data

Imports System.Data.SqlClient

Public Class clsReserva

#Region "Variables"

Private sidReserva As String Private sidHabitacion As String Private sidCliente As String Private cTipoR As Char

Private dFecRva As Date Private dFecIng As Date Private dFecSal As Date Private

dCosto As Decimal Private sObs As String Private sTipoDoc As Char Private sNroDoc

As String Private dMontoIgv As Double Private dSubTotal As Double Private dServicio

As Double Private dIgv As Double

```
Private dTotalPagar As Double Private sTipoPago As String Private dAcuenta As
Double Private dSaldo As Double
#End Region

#Region "Propiedades"
Public Property _sidReserva() As String Get
    Return sidReserva
End Get
    Set(ByVal value As String) sidReserva = value
End Set End Property
Public Property _sidHabitacion() As String
Get
    Return sidHabitacion
End Get
Set(ByVal value As String) sidHabitacion = value
End Set
End Property
Public Property _sidCliente() As String Get
    Return sidCliente
End Get
    Set(ByVal value As String) sidCliente = value
End Set End Property
    Public Property _cTipoR() As Char
Get
    Return cTipoR
End Get
Set(ByVal value As Char)
    cTipoR = value
End Set End Property
Public Property _dFechaRva() As Date Get
    Return dFecRva
End Get
Set(ByVal value As Date) dFecRva = value
End Set End Property
Public Property _dFechaIng() As Date Get
    Return dFecIng
End Get
Set(ByVal value As Date)
    dFecIng = value
End Set End Property
Public Property _dFecSal() As Date Get
    Return dFecSal
End Get
Set(ByVal value As Date)
```

```

                                dFecSal = value
        End Set End Property
Public Property _dCosto() As Decimal Get
                                Return dCosto
        End Get
Set(ByVal value As Decimal) dCosto = value
        End Set End Property
Public Property _sObs() As String Get
                                Return sObs
        End Get
Set(ByVal value As String)
                                sObs = value
        End Set End Property
Public Property _sTipoDoc() As Char Get
                                Return sTipoDoc
        End Get
Set(ByVal value As Char) sTipoDoc = value
        End Set
End Property
Public Property _sNroDoc() As String Get
                                Return sNroDoc
        End Get
Set(ByVal value As String)
                                sNroDoc = value
        End Set End Property
Public Property _dMontoIgv() As Double
Get
                                Return dMontoIgv
        End Get
Set(ByVal value As Double) dMontoIgv = value
        End Set
End Property
Public Property _dSubTotal() As Double Get
                                Return dSubTotal
        End Get
Set(ByVal value As Double) dSubTotal = value
        End Set End Property
Public Property _dServicio() As Double
Get
                                Return dServicio
        End Get
Set(ByVal value As Double) dServicio = value
        End Set End Property
        Public Property _dIgv() As Double
```

```

                Get
                    Return dIgv
            End Get
        Set(ByVal value As Double) dIgv = value
        End Set
    End Property
    Public Property _dTotalPagar() As Double Get
        Return dTotalPagar
    End Get
    Set(ByVal value As Double) dTotalPagar = value
    End Set End Property
    Public Property _sTipoPago() As String
    Get
        Return sTipoPago
    End Get
    Set(ByVal value As String) sTipoPago = value
    End Set
    End Property
    Public Property _dAcuenta() As Double Get
        Return dAcuenta
    End Get
    Set(ByVal value As Double) dAcuenta = value
    End Set End Property
    Public Property _dSaldo() As Double
    Get
        Return dSaldo
    End Get
    Set(ByVal value As Double)
        dSaldo = value
    End Set End Property
#End Region

Public Function mostrarClientes(ByVal xFiltro As String) As
DataSet
Dim db As New dbAccess
db.AddParameter("@filtro", xFiltro)
Return db.ExecuteDataSet("sp_MostrarCliente")
End Function

Public Function listarReserva(ByVal xFiltro As String) As
DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", xFiltro)
Return db.ExecuteDataSet("SP_listarReservaciones")
End Function

Public Sub MantenimientoReserva(ByVal x As clsReserva,
ByVal v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@Cad01", vgCadValor)

```

```
db.AddParameter("@idReserva", x._sidReserva) db.AddParameter("@idHabitacion",  
x._sidHabitacion) db.AddParameter("@idCliente", x._sidCliente)  
db.AddParameter("@TipoR", x._cTipoR) db.AddParameter("@FechaR", x._dFechaRva)  
db.AddParameter("@FechaIng", x._dFechaIng) db.AddParameter("@FechaSal", x._dFecSal)  
db.AddParameter("@Costo", x._dCosto) db.AddParameter("@Obs", x._sObs)  
db.AddParameter("@TipoDoc", x._sTipoDoc) db.AddParameter("@nroDocumento",  
x._sNroDoc) db.AddParameter("@montoIgv", x._dMontoIgv) db.AddParameter("@subTotal",  
x._dSubTotal) db.AddParameter("@servicio", x._dServicio) db.AddParameter("@igv",  
x._dIgv) db.AddParameter("@totalPagar", x._dTotalPagar)  
db.AddParameter("@tipoPago", x._sTipoPago) db.AddParameter("@aCuenta", x._dAcuenta)  
db.AddParameter("@saldo", x._dSaldo) db.AddParameter("@Usr", vgUsr)  
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")  
db.AddParameter("@Error", 0)
```

```
db.ExecuteNonQuery("SP_Mant_Reserva")  
End Sub
```

```
Public Function reservaDisponiblesAlquiler(ByVal xTipo As String, ByVal xFiltro  
As String) As DataSet  
Dim db As New dbAccess db.AddParameter("@TipoHab", xTipo)  
db.AddParameter("@Filtro", xFiltro)  
Return  
db.ExecuteDataSet("SP_ListarHabitacionesDisponibles_Alquiler")  
End Function
```

```
Public Function reservaDisponiblesReserva(ByVal xTipo As String, ByVal xFiltro  
As String, ByVal xFecha As DateTime) As DataSet  
Dim db As New dbAccess db.AddParameter("@TipoHab", xTipo)  
db.AddParameter("@Filtro", xFiltro) db.AddParameter("@Fecha", xFecha)  
Return  
db.ExecuteDataSet("SP_ListarHabitacionesDisponibles_Reserva")  
End Function
```

```
Public Sub updateSalidaReserva(ByVal x As clsReserva)  
Dim db As New dbAccess
```

```
db.AddParameter("@idHabitacion", x._sidHabitacion) db.AddParameter("@idReserva",  
x._sidReserva) db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
```

```
db.ExecuteNonQuery("SP_SalidaReserva")  
End Sub
```

```
Public Sub salidaDeudas(ByVal x As clsReserva)  
Dim db As New dbAccess
```

```
db.AddParameter("@idReserva", x._sidReserva) db.AddParameter("@saldo", x._dSaldo)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)
```

```
db.ExecuteNonQuery("SP_InsertarDeudaReserva")
End Sub
```

```
Public Function verMontosPagar(ByVal xNop As Integer, ByVal
xIdReserva As String) As Double
Dim db As New dbAccess
```

```
db.AddParameter("@nOp", xNop) db.AddParameter("@idReserva", xIdReserva)
```

```
Dim xValor As Double = db.ExecuteScalar("SP_MostrarMontosPagos")
```

```
Return xValor
End Function
```

```
Public Sub cancelaReservaConsumo(ByVal xidReser As String, ByVal xTipoDoc As
Char, ByVal xidPago As String, ByVal xidBanco As String, ByVal xfecRegistro As
Date, ByVal xfechaPago As Date, ByVal xnOP As String, ByVal xmonto As Double, ByVal
xcomision As Double, ByVal xNroDoc As String, ByVal xMonIgv As Double, _
ByVal xTotReser As Double, ByVal xtotCon As
Double, ByVal xTotLavanderia As Double, ByVal xSubTot As Double, _
ByVal xIgv As Double, ByVal xTot As Double,
ByVal xTpago As String)
```

```
Dim db As New dbAccess db.AddParameter("@idReserva", xidReser)
db.AddParameter("@tipoDoc", xTipoDoc) db.AddParameter("@idPago", xidPago)
db.AddParameter("@idBanco", xidBanco) db.AddParameter("@fechaRegis", xfecRegistro)
db.AddParameter("@fechaPago", xfechaPago) db.AddParameter("@nOP", xnOP)
db.AddParameter("@monto", xmonto) db.AddParameter("@comision", xcomision)
db.AddParameter("@nroDoc", xNroDoc) db.AddParameter("@MontoIgv", xMonIgv)
db.AddParameter("@totReserva", xTotReser) db.AddParameter("@totConsumo", xtotCon)
db.AddParameter("@totLavanderia", xTotLavanderia) db.AddParameter("@subTotal",
xSubTot) db.AddParameter("@igv", xIgv) db.AddParameter("@total", xTot)
db.AddParameter("@tPago", xTpago) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)
```

```
db.ExecuteNonQuery("SP_CancelarReservacionConsumo")  
End Sub
```

```
Public Function mostrarHuespedProgramados(ByVal x As  
clsReserva) As DataSet  
Dim db As New dbAccess  
db.AddParameter("@idReserva", x._sidReserva)  
Return  
db.ExecuteDataSet("SP_MostrarHuespedProgramados")  
End Function
```

```
Public Sub actualizarIngresoReserva(ByVal xidReserva As String, ByVal  
xidPersona As String, ByVal xidVehiculo As String)  
Dim db As New dbAccess
```

```
db.AddParameter("@idReserva", xidReserva) db.AddParameter("@idPersona", xidPersona)  
db.AddParameter("@idVehiculo", xidVehiculo) db.AddParameter("@Usr", vgUsr)  
db.AddParameter("@wks", vgPc)
```

```
db.ExecuteNonQuery("SP_UpdateEstadoHuesped")  
End Sub End Class
```

CLASE COCHERA

```
Imports System.Data  
Imports System.Data.SqlClient  
Public Class clsCochera  
#Region "Variables"  
Private sidCochera As String Private sidUbicacion As String Private sidCliente As  
String Private dFecIng As Date Private dFecSal As Date Private dCosto As Decimal  
Private sObs As String  
Private sTipoDoc As Char Private sNroDoc As String Private dMontoIgv As Double  
Private dSubTotal As Double Private dServicio As Double Private dIgv As Double  
Private dTotalPagar As Double Private sTipoPago As String Private dAcuenta As  
Double Private dSaldo As Double  
#End Region  
  
#Region "Propiedades"  
Public Property _idCochera() As String Get  
Return sidCochera  
End Get  
Set(ByVal value As String)  
sidCochera = value
```

```
End Set End Property

Public Property _idUbicacion() As String Get
    Return sidUbicacion
End Get
Set(ByVal value As String) sidUbicacion = value
End Set End Property

Public Property _sidCliente() As String Get
    Return sidCliente
End Get
Set(ByVal value As String) sidCliente = value
End Set
End Property

Public Property _dFechaIng() As Date Get
    Return dFecIng
End Get
Set(ByVal value As Date)
    dFecIng = value
End Set End Property

Public Property _dFecSal() As Date Get
    Return dFecSal
End Get
Set(ByVal value As Date) dFecSal = value
End Set End Property

Public Property _dCosto() As Decimal Get
    Return dCosto
End Get
Set(ByVal value As Decimal) dCosto = value
End Set End Property

Public Property _sObs() As String Get
    Return sObs
End Get
Set(ByVal value As String) sObs = value
End Set
End Property

Public Property _sTipoDoc() As Char Get
    Return sTipoDoc
```

```
End Get
        Set(ByVal value As Char) sTipoDoc = value
End Set End Property

Public Property _sNroDoc() As String Get
        Return sNroDoc
End Get
Set(ByVal value As String) sNroDoc = value
End Set
End Property

Public Property _dMontoIgv() As Double Get
        Return dMontoIgv
End Get
Set(ByVal value As Double) dMontoIgv = value
End Set
End Property

Public Property _dSubTotal() As Double Get
        Return dSubTotal
End Get
Set(ByVal value As Double)
        dSubTotal = value
End Set End Property

Public Property _dServicio() As Double Get
        Return dServicio
End Get
Set(ByVal value As Double) dServicio = value
End Set End Property

Public Property _dIgv() As Double Get
        Return dIgv
End Get
Set(ByVal value As Double) dIgv = value
End Set
End Property

Public Property _dTotalPagar() As Double Get
        Return dTotalPagar
End Get
Set(ByVal value As Double) dTotalPagar = value
End Set End Property
```

```

Public Property _sTipoPago() As String Get
    Return sTipoPago
End Get
Set(ByVal value As String) sTipoPago = value
End Set End Property

Public Property _dAcuenta() As Double Get
    Return dAcuenta
End Get
Set(ByVal value As Double) dAcuenta = value
End Set
End Property

Public Property _dSaldo() As Double Get
    Return dSaldo
End Get
Set(ByVal value As Double) dSaldo = value
End Set End Property
#End Region

Public Function listarAlquilerCochera(ByVal xFiltro As String) As DataSet
Dim db As New dbAccess
db.AddParameter("@Filtro", xFiltro)
Return db.ExecuteDataSet("SP_listarAlquilerCochera")
End Function

Public Sub MantenimientoCochera(ByVal x As clsCochera,
ByVal v1Proceso As Char, ByVal xCadena As String) Dim db As New dbAccess
db.AddParameter("@Proceso", v1Proceso) db.AddParameter("@Cad01", xCadena)
db.AddParameter("@idCochera", x._idCochera) db.AddParameter("@idCliente",
x._sidCliente) db.AddParameter("@FechaIng", x._dFechaIng)
db.AddParameter("@FechaSal", x._dFecSal) db.AddParameter("@Costo", x._dCosto)
db.AddParameter("@Obs", x._sObs) db.AddParameter("@TipoDoc", x._sTipoDoc)
db.AddParameter("@nroDocumento", x._sNroDoc) db.AddParameter("@montoIgv",
x._dMontoIgv) db.AddParameter("@subTotal", x._dSubTotal)
db.AddParameter("@servicio", x._dServicio) db.AddParameter("@igv", x._dIgv)
db.AddParameter("@totalPagar", x._dTotalPagar) db.AddParameter("@tipoPago",
x._sTipoPago) db.AddParameter("@aCuenta", x._dAcuenta) db.AddParameter("@saldo",
x._dSaldo) db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "")

```

```
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Cochera")
End Sub

Public Sub updateSalidaCochera(ByVal x As clsCochera)
    Dim db As New dbAccess

    ' db.AddParameter("@idUbiVehiculo", x._idUbicacion) db.AddParameter("@idCochera",
    x._idCochera) db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)

    db.ExecuteNonQuery("SP_SalidaCochera")
End Sub

Public Sub salidaDeudasCochera(ByVal x As clsCochera)
    Dim db As New dbAccess

    db.AddParameter("@idCochera", x._idCochera) db.AddParameter("@saldo", x._dSaldo)
    db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
    db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

    db.ExecuteNonQuery("SP_InsertarDeudaCochera")
End Sub End Class
```

CAMBIAR ESTADO DE HABITACIÓN

La clase cambiar estado de habitacion se menciona en la primera iteración, el cual incluye el procedimientos almacenado Cambiar estado de habitación

CLASE APERTURA CAJA / CIERRE DE CAJA

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsAperturaCaja
    #Region "Variables"
    Private sidApertura As String
    Private sidUsuario As String Private sidUsuEntregado As String Private smonto As
    Decimal
    Private sdescripcion As String
    Private ssaldo As Double Private sdescripSaldo As String
    Private stotEfectivoServicio As Double
    Private stotIngreso As Double Private stotResCochera As Double Private stotDinero
    As Double
```

```
Private stotEgreso As Double Private stotTarjetaBanco As Double Private
stotDescDinero As Double Private stotCajaEfectivo As Double Private
stotCajaEntregado As Double Private sestado As Boolean

#End Region
#Region "Propiedades"
Public Property _idApertura() As String Get
    Return sidApertura
End Get
Set(ByVal value As String) sidApertura = value
End Set
End Property

Public Property _idUserario() As String Get
    Return sidUsuario
End Get
Set(ByVal value As String) sidUsuario = value
End Set End Property

Public Property _idUsuEntregado() As String Get
    Return sidUsuEntregado
End Get
Set(ByVal value As String)
    sidUsuEntregado = value
End Set End Property

Public Property _monto() As Decimal Get
    Return smonto
End Get
Set(ByVal value As Decimal) smonto = value
End Set End Property

Public Property _descripcion() As String Get
    Return sdescripcion
End Get
Set(ByVal value As String) sdescripcion = value
End Set
End Property

Public Property _saldo() As Double Get
    Return ssaldo
End Get
Set(ByVal value As Double)
    ssaldo = value
```

```
End Set End Property

Public Property _descripSaldo() As String Get
    Return sdescripSaldo
End Get
Set(ByVal value As String) sdescripSaldo = value
End Set End Property

Public Property _totEfectivoServicio() As Double Get
    Return stotEfectivoServicio
End Get
Set(ByVal value As Double) stotEfectivoServicio = value
End Set
End Property

Public Property _totIngreso() As Double Get
    Return stotIngreso
End Get
Set(ByVal value As Double)
    stotIngreso = value
End Set End Property

Public Property _totResCochera() As Double Get
    Return stotResCochera
End Get
Set(ByVal value As Double) stotResCochera = value
End Set End Property

Public Property _totDinero() As Double Get
    Return stotDinero
End Get
Set(ByVal value As Double) stotDinero = value
End Set End Property

Public Property _totEgreso() As Double Get
    Return stotEgreso
End Get
Set(ByVal value As Double) stotEgreso = value
End Set
End Property

Public Property _totTarjetaBanco() As Double Get
    Return stotTarjetaBanco
```

```
End Get
Set(ByVal value As Double) stotTarjetaBanco = value
End Set End Property

Public Property _totDescDinero() As Double Get
Return stotDescDinero
End Get
Set(ByVal value As Double) stotDescDinero = value
End Set
End Property

Public Property _totCajaEfectivo() As Double Get
Return stotCajaEfectivo
End Get
Set(ByVal value As Double) stotCajaEfectivo = value
End Set
End Property

Public Property _totCajaEntregado() As Double Get
Return stotCajaEntregado
End Get
Set(ByVal value As Double)
stotCajaEntregado = value
End Set End Property

Public Property _estado() As Boolean Get
Return sestado
End Get
Set(ByVal value As Boolean) sestado = value
End Set End Property
#End Region

Public Function UsuarioSelectAllByName(ByVal x As
clsAperturaCaja) As DataSet
Dim db As New dbAccess
db.AddParameter("@usuario", x._idUserio)
'x._idUserio)
Return db.ExecuteDataSet("SP_SetDevolUsuarioApertura")
End Function

Public Sub MantenimientoApertura(ByVal x As
clsAperturaCaja, ByVal vlProceso As Char)
Dim db As New dbAccess
db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idAperturaCierre", x._idApertura) db.AddParameter("@idUserio",
x._idUserio) db.AddParameter("@idUsuEntregado", x._idUsuEntregado)
db.AddParameter("@monto", x._monto) db.AddParameter("@descripcion", x._descripcion)
```

```
db.AddParameter("@saldo", x._saldo) db.AddParameter("@descripSaldo",  
x._descripSaldo) db.AddParameter("@totEfectivoServicio",  
x._totEfectivoServicio)  
db.AddParameter("@totIngreso", x._totIngreso) db.AddParameter("@totResCochera",  
x._totResCochera) db.AddParameter("@totDinero", x._totDinero)  
db.AddParameter("@totEgreso", x._totEgreso) db.AddParameter("@totTarjetaBanco",  
x._totTarjetaBanco) db.AddParameter("@totDescDinero", x._totDescDinero)  
db.AddParameter("@totCajaEfectivo", x._totCajaEfectivo)  
db.AddParameter("@totCajaEntregado",  
x._totCajaEntregado)  
db.AddParameter("@estado", x._estado) db.AddParameter("@Usr", vgUsr)  
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")  
db.AddParameter("@Error", 0)  
  
db.ExecuteNonQuery("SP_Mant_AperturaCaja")  
End Sub End Class
```

CLASE INGRESO CAJA

```
Imports System.Data  
Imports System.Data.SqlClient  
Public Class clsIngresoCaja  
#Region "Variables"  
Private sidIngreso As String Private sidCliente As String Private sidConcepto As  
String Private sidPago As String Private sidBanco As String Private sidDeuda As  
String Private stipoDoc As Char Private snroDocumento As String Private snOP As  
String  
Private smonto As Double  
Private sMontoIgv As Double Private ssubTotal As Double Private scomision As Double  
Private sigv As Double Private stotal As Double Private stPago As String  
  
#End Region  
#Region "Propiedades"  
Public Property _idIngreso() As String  
Get  
Return sidIngreso  
End Get  
Set(ByVal value As String) sidIngreso = value  
End Set End Property
```

```
Public Property _idCliente() As String Get
    Return sidCliente
End Get

    Set(ByVal value As String)
        sidCliente = value
    End Set End Property

Public Property _idConcepto() As String Get
    Return sidConcepto
End Get
Set(ByVal value As String) sidConcepto = value
End Set End Property

Public Property _idPago() As String Get
    Return sidPago
End Get
Set(ByVal value As String) sidPago = value
End Set
End Property

Public Property _idBanco() As String Get
    Return sidBanco
End Get
Set(ByVal value As String) sidBanco = value
End Set End Property

Public Property _idDeuda() As String Get
    Return sidDeuda
End Get
Set(ByVal value As String)
    sidDeuda = value
End Set End Property

Public Property _tipoDoc() As Char Get
    Return stipoDoc
End Get
    Set(ByVal value As Char) stipoDoc = value
End Set End Property

Public Property _nroDocumento() As String Get
    Return snroDocumento
End Get
Set(ByVal value As String)
```

```

                                snroDocumento = value
    End Set End Property

Public Property _nOP() As String Get
                                Return snOP
    End Get
    Set(ByVal value As String)
                                snOP = value
    End Set End Property

Public Property _monto() As Double Get
                                Return smonto
    End Get
    Set(ByVal value As Double) smonto = value
    End Set End Property

Public Property _MontoIgv() As Double Get
                                Return sMontoIgv
    End Get
    Set(ByVal value As Double) sMontoIgv = value
    End Set
    End Property

Public Property _subTotal() As Double Get
                                Return ssubTotal
    End Get
    Set(ByVal value As Double)
                                ssubTotal = value
    End Set End Property

Public Property _comision() As Double Get
                                Return scomision
    End Get
    Set(ByVal value As Double)
                                scomision = value
    End Set End Property

Public Property _igv() As Double Get
                                Return sigv
    End Get
    Set(ByVal value As Double) sigv = value
    End Set End Property

Public Property _total() As Double Get
```

```

Return stotal
End Get
Set(ByVal value As Double)
    stotal = value
End Set End Property

Public Property _tPago() As String Get
    Return stPago
End Get
Set(ByVal value As String) stPago = value
End Set End Property
#End Region

Public Sub MantenimientoIngresoCaja(ByVal x As
clsIngresoCaja)
Dim db As New dbAccess
'db.AddParameter("@Proceso", vlProceso)
' db.AddParameter("@idIngreso", x._idIngreso)
db.AddParameter("@idCliente", x._idCliente) db.AddParameter("@idConcepto",
x._idConcepto) db.AddParameter("@idPago", x._idPago) db.AddParameter("@idBanco",
x._idBanco) db.AddParameter("@idDeuda", x._idDeuda) db.AddParameter("@tipoDoc",
x._tipoDoc) db.AddParameter("@nroDocumento", x._nroDocumento)
db.AddParameter("@nOP", x._nOP) db.AddParameter("@monto", x._monto)
db.AddParameter("@MontoIgv", x._MontoIgv) db.AddParameter("@subTotal", x._subTotal)
db.AddParameter("@comision", x._comision) db.AddParameter("@igv", x._igv)
db.AddParameter("@total", x._total) db.AddParameter("@tPago", x._tPago)
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Ingreso_Caja")
End Sub End Class

```

CLASE EGRESOS CAJA

```

Imports System.Data
Imports System.Data.SqlClient
Public Class clsEgresosCaja
#Region "Variables"
Private sidEgreso As String Private sidCliente As String Private sidConcepto As
String Private sidPago As String

```

```
Private snumRecibo As String Private sdescripcion As String Private sTotal As
Decimal Private sestado As Boolean

#End Region
#Region "Propiedades"
Public Property _idEgreso() As String Get
    Return sidEgreso
End Get
Set(ByVal value As String) sidEgreso = value
End Set End Property

Public Property _idCliente() As String Get
    Return sidCliente
End Get
Set(ByVal value As String) sidCliente = value
End Set End Property

Public Property _idConcepto() As String Get
    Return sidConcepto
End Get
Set(ByVal value As String) sidConcepto = value
End Set
End Property

Public Property _idPago() As String Get
    Return sidPago
End Get
Set(ByVal value As String)
    sidPago = value
End Set End Property

Public Property _numRecibo() As String Get
    Return snumRecibo
End Get
Set(ByVal value As String) snumRecibo = value
End Set End Property

Public Property _descripcion() As String Get
    Return sdescripcion
End Get
Set(ByVal value As String) sdescripcion = value
End Set
End Property
```

```
Public Property _Total() As Decimal Get
    Return sTotal
End Get
Set(ByVal value As Decimal)
    sTotal = value
End Set End Property

Public Property _estado() As Boolean Get
    Return sestado
End Get
Set(ByVal value As Boolean) sestado = value
End Set End Property
#End Region

Public Sub MantenimientoEgresosCaja(ByVal x As
clsEgresosCaja, ByVal vlProceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", vlProceso)
db.AddParameter("@idEgreso", x._idEgreso) db.AddParameter("@idCliente",
x._idCliente) db.AddParameter("@idConcepto", x._idConcepto)
db.AddParameter("@idPago", x._idPago) db.AddParameter("@numRecibo", x._numRecibo)
db.AddParameter("@descripcion", x._descripcion) db.AddParameter("@total", x._Total)
db.AddParameter("@estado", x._estado) db.AddParameter("@Usr", vgUsr)
db.AddParameter("@wks", vgPc) db.AddParameter("@Mensaje", "")
db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_EgresosCaja")
End Sub

End Class
```

CONCEPTO INGRESO Y SALIDA

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsConcepto_IS
#Region "Variables"
Private sidConcepto As String Private sNombre As String Private stipo As String
#End Region

#Region "Propiedades"
```

```
Public Property _idConcepto() As String Get
    Return sidConcepto
End Get
Set(ByVal value As String) sidConcepto = value
End Set End Property

Public Property _nombre() As String Get
    Return sNombre
End Get
Set(ByVal value As String) sNombre = value
End Set
End Property

Public Property _tipo() As String Get
    Return stipo
End Get
Set(ByVal value As String) stipo = value
End Set End Property
#End Region

Public Function ConceptoSelectAllByName(ByVal x As
clsConcepto_IS) As DataSet
Dim db As New dbAccess
db.AddParameter("@concepto", x._nombre)
Return db.ExecuteDataSet("SP_MostrarConcepto")
End Function

Public Function TipoEfectivoSelectAllByName(ByVal x As
clsConcepto_IS) As DataSet
Dim db As New dbAccess
Return db.ExecuteDataSet("SP_Get_TipoEfectivo")
End Function

Public Function TipoEgresosSelectAllByName(ByVal x As
clsConcepto_IS) As DataSet
Dim db As New dbAccess
Return db.ExecuteDataSet("SP_MostrarConceptoEgresos")
End Function

Public Function TipoIngresosSelectAllByName(ByVal x As
clsConcepto_IS) As DataSet
Dim db As New dbAccess
Return db.ExecuteDataSet("SP_MostrarConceptoIngresos")
End Function

Public Sub MantenimientoConcepto(ByVal x As clsConcepto_IS,
ByVal v1Proceso As Char)
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)
db.AddParameter("@idConcepto", x._idConcepto) db.AddParameter("@concepto",
x._nombre) db.AddParameter("@tipo", x._tipo)
```

```
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)

db.ExecuteNonQuery("SP_Mant_Concepto_IS")
End Sub End Class
```

CLASE CAJA

Cabe mencionar que las operaciones listado detallado de caja, listado de caja realizados, resumen del día, listados de documentos generados, o nacen de la clase caja.

```
Imports System.Data
Imports System.Data.SqlClient
Public Class clsCaja
    Public Function ConceptoSelectAllByName(ByVal xTipo As String) As DataSet
    Dim db As New dbAccess
    db.AddParameter("@Tipo", xTipo)
    Return
    db.ExecuteNonQuery("Sp_Get_Caja_ListadoConceptosByTipo")
    End Function

    Public Function ListarDetalleCaja(ByVal fechaini As Date, ByVal fechafin As
    Date, ByVal Concepto As String) As DataSet
    Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
    db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", Concepto)

    Return db.ExecuteNonQuery("SP_Listar_Detalle_Caja")
    End Function

    Public Function ListarDetalleCaja_Ingreso(ByVal fechaini As Date, ByVal
    fechafin As Date, ByVal Concepto As String) As DataSet
    Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
    db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", Concepto)

    Return
    db.ExecuteNonQuery("SP_Listar_Detalle_Caja_Ingreso")
    End Function

    Public Function ListarCierreCajaxUsuario(ByVal fechaini As Date, ByVal fechafin
    As Date, ByVal xUsuario As String) As DataSet
    Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
    db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", xUsuario)
```

```
Return db.ExecuteDataSet("SP_ListarCierreCaja_Usuario")  
End Function
```

```
Public Function ListarCierreCajaTarjeta(ByVal fechaini As Date, ByVal fechafin  
As Date, ByVal xUsuario As String) As DataSet  
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)  
db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", xUsuario)
```

```
Return db.ExecuteDataSet("SP_ListarCierreCaja_TARJETA")  
End Function
```

```
Public Function ListarCierreCajaEfectivo(ByVal fechaini As Date, ByVal fechafin  
As Date, ByVal xUsuario As String) As DataSet  
Dim db As New dbAccess  
db.AddParameter("@fechaIni", fechaini) db.AddParameter("@fechafin", fechafin)  
db.AddParameter("@Filtro", xUsuario)
```

```
Return  
db.ExecuteDataSet("SP_ListarCierreCaja_Efectivo")  
End Function
```

```
Public Function ListarCierreCajaIngreso(ByVal fechaini As Date, ByVal fechafin  
As Date, ByVal xUsuario As String) As DataSet  
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)  
db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", xUsuario)
```

```
Return db.ExecuteDataSet("SP_ListarCierreCaja_Ingreso")  
End Function
```

```
Public Function ListarCierreCajaEgreso(ByVal fechaini As Date, ByVal fechafin  
As Date, ByVal xUsuario As String) As DataSet  
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)  
db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", xUsuario)
```

```
Return db.ExecuteDataSet("SP_ListarCierreCaja_Egreso")  
End Function
```

```
Public Function ListarCierreCajaCocheraReserva(ByVal fechaini As Date, ByVal  
fechafin As Date, ByVal xUsuario As String) As DataSet  
Dim db As New dbAccess  
db.AddParameter("@fechaIni", fechaini) db.AddParameter("@fechafin", fechafin)  
db.AddParameter("@Filtro", xUsuario)
```

```
Return  
db.ExecuteDataSet("SP_ListarCierreCaja_ReservaCochera")  
End Function
```

```
Public Function ListarCierreCajaApertura(ByVal fechaini As Date, ByVal fechafin
As Date, ByVal xUsuario As String) As DataSet
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", xUsuario)

Return
db.ExecuteDataSet("SP_ListarCierreCaja_Apertura")
End Function

Public Function ListarCierreDiario(ByVal fechaini As Date, ByVal fechafin As
Date, ByVal xUsuario As String) As DataSet
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
db.AddParameter("@fechafin", fechafin) db.AddParameter("@usuario", xUsuario)

Return db.ExecuteDataSet("SP_CierreCaja_Turno")
End Function

Public Function ResumenDiario(ByVal fechaini As Date, ByVal
fechafin As Date) As DataSet
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
db.AddParameter("@fechafin", fechafin)

Return db.ExecuteDataSet("SP_DineroPorDia")
End Function

Public Function HospesajeRealizadoxUsuario(ByVal fechaini As Date, ByVal fechafin
As Date, ByVal xUsuario As String) As DataSet
Dim db As New dbAccess db.AddParameter("@fechaIni", fechaini)
db.AddParameter("@fechafin", fechafin) db.AddParameter("@Filtro", xUsuario)

Return db.ExecuteDataSet("SP_HuespedPorUsuario")
End Function End Class
```

CLASE CONSUMO

Clase en la cual se realizan las anulacion de servicio, listado de servicios anulados.

```
Imports System.Data
Imports System.Data.SqlClient

Public Class clsConsumos
#Region "Variables"
Private sidConsumo As String
Private sidReserva As String
```

```
Private sidProducto As String Private iCantidad As Integer Private dPrecio As
Decimal
#End Region

#Region "Propiedades"
Public Property _sidConsumo() As String Get
    Return sidConsumo
End Get
    Set(ByVal value As String) sidConsumo = value
End Set
End Property
Public Property _sidReserva() As String Get
    Return sidReserva
End Get
    Set(ByVal value As String)
        sidReserva = value
    End Set End Property
Public Property _sidProducto() As String Get
    Return sidProducto
End Get
Set(ByVal value As String) sidProducto = value
End Set
End Property
Public Property _iCantidad() As Integer Get
    Return iCantidad
End Get
Set(ByVal value As Integer) iCantidad = value
End Set End Property
Public Property _dPrecio() As Decimal
Get
    Return dPrecio
End Get
Set(ByVal value As Decimal) dPrecio = value
End Set
End Property
#End Region

Public Function mostrarConsumos(ByVal x As clsConsumos) As
DataSet
Dim db As New dbAccess db.AddParameter("@idReserva", x._sidReserva) Return
db.ExecuteDataSet("SP_ListarConsumos")
End Function

Public Function mostrarConsumosAnulados(ByVal x As
clsConsumos) As DataSet
Dim db As New dbAccess
db.AddParameter("@idReserva", x._sidReserva)
Return db.ExecuteDataSet("SP_ListarConsumosAnulados")
```

End Function

```
Public Sub grabarCosumos(ByVal x As clsConsumos, ByVal  
v1Proceso As Char)  
Dim db As New dbAccess db.AddParameter("@Proceso", v1Proceso)  
db.AddParameter("@idConsumo", x._sidConsumo) db.AddParameter("@idReserva",  
x._sidReserva) db.AddParameter("@idProducto", x._sidProducto)  
db.AddParameter("@Cantidad", x._iCantidad) db.AddParameter("@precio", x._dPrecio)  
db.AddParameter("@Usr", vgUsr) db.AddParameter("@wks", vgPc)  
db.AddParameter("@Mensaje", "") db.AddParameter("@Error", 0)  
  
db.ExecuteNonQuery("SP_GrabarConsumos")  
End Sub End Class
```

CAPITULO V INTEGRACIÓN

5.1 CONSIDERACIONES DE SEGURIDAD

5.1.1 CONSIDERACIONES DE SEGURIDAD EN LA BASE DE DATOS

5.1.1.1 ENCRIPCIÓN

Las redes públicas como Internet no proporcionan un medio de comunicación segura entre entidades. La comunicación en esas redes es susceptible de que terceras personas, sin autorización, tengan acceso a ella o la modifiquen. Además de permitir el cifrado de los archivos de los discos locales, la criptografía ayuda a crear medios de comunicación seguros sobre canales que, de otro modo, serían inseguros, proporcionando a su vez integridad de datos y autenticación. Las clases del espacio de nombres de criptografía de .NET Framework se ocupan de administrar muchos de los detalles de la criptografía.

La encriptación que se maneja es mediante la clase DESCryptoServiceProvider la cual permite cifrar los datos mediante el cifrado estándar de datos (DES) o los algoritmos de Triple DES.

Este método lo utilizamos en el registro de la contraseña del usuario, mediante una función de encriptación, guardamos la contraseña encriptada en la base de datos, de esta manera evitamos que terceros se hagan de la información privada que maneja la institución.

5.1.1.2 INYECCION SQL

La inyección de código SQL es un ataque en el cual se inserta código malicioso en las cadenas que posteriormente se pasan a una instancia de SQL Server para su análisis y ejecución. Todos los procedimientos que generan instrucciones SQL deben revisarse en busca de vulnerabilidades de inyección de código, ya que SQL Server ejecutará todas las consultas

recibidas que sean válidas desde el punto de vista sintáctico. En nuestra aplicación manejamos procedimientos almacenados parametrizados la cual evita la inyección sql.

5.1.2 CONSIDERACIONES DE SEGURIDAD EN LA APLICACIÓN

5.1.2.1 AUTENTICACIÓN DE USUARIOS

El sistema asigna permisos a las opciones del menú según el area al que pertenece el usuario autenticado. Esto, para evitar que un usuario realice operaciones no autorizadas.

5.1.2.2 OPCIONES DE MENÚ

En esta aplicación se han asignado permisos según el area al que pertenezca el usuario. El área de administración General tiene acceso a todas las opciones del menú y las demás áreas solo tendran acceso a las operaciones relacionadas con su perfil de usuario. Esto por requerimiento de la empresa.

5.2 PUESTA EN MARCHA

5.2.1 ESPECIFICACIONES TÉCNICAS

5.2.1.1 CLIENTE

REQUERIMIENTOS MÍNIMOS DE HARDWARE

- ✓ **Procesador:** Core 2 duo de 1.250 GHZ.
- ✓ **MemoriaRAM:** 4GB DDR3.
- ✓ **Disco duro:** 500GB 7200 RPM.
- ✓ **Monitor:** 17".
- ✓ Impresora.
- ✓ Tarjetadedred10MB.

REQUERIMIENTOS MÍNIMOS DE SOFTWARE

- ✓ Windows 7 Home Premium
- ✓ Controlador de impresora.
- ✓ MS Office 2010
- ✓ Adobe reader 11.0

CAPITULO VI DISCUSIÓN

6.1 RESULTADOS DE LA PRIMERA ENCUESTA

Nuestra población es de 17 trabajadores dentro de la empresa, con muestra tenemos 06 trabajadores futuros usuarios, del Sistema computarizado para la Administración del rubro hotelero de la Empresa de Serv. Turístico Princess s.a.. La realización de la encuesta, obtención de los datos y el análisis de los mismos fue realizado durante el mes de septiembre del 2014, desde ese momento se ha venido trabajando con dichos trabajadores seleccionados durante todo el desarrollo de la investigación.

6.1.1 PREGUNTA 1: TIEMPO EN MINUTOS PARA REGISTRAR UN CLIENTE

6.1.1.1 PREGUNTA ASOCIADA

¿cual es el tiempo promedio en minutos para registrar a un cliente?

6.1.1.2 RESPUESTAS PROPORCIONADAS POR LOS USUARIOS

El resultado obtenidos para determinar el tiempo promedio en minutos que tarda un usuario para registrar un cliente es:

Nº USUARIO	TIEMPO EN MINUTOS
U1	5
U2	6
U3	5
U4	5
U5	6
U6	5

6.1.1.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el tiempo promedio en minutos que tarda un usuario en registrar un expediente para el total de usuarios encuestados utilizamos la siguiente formula:

$$PTR = \sum_{i=1}^{10} \frac{T_i}{N}$$

$$PTR = 5.3$$

Donde:

PTR: promedio de tiempo de registro. T: Tiempo para registrar un cliente. N: Número de usuarios encuestados.

6.1.1.4 RESULTADOS

Como resultados determinamos que el tiempo promedio en minutos que tarda un usuario en registrar un cliente para el total de usuarios encuestados es de 5.3.

6.1.2 PREGUNTA 2: TIEMPO PROMEDIO EN MINUTOS PARA REALIZAR UN COMPROBANTE

6.1.2.1 PREGUNTA ASOCIADA

¿cual es el tiempo promedio en minutos para realizar un comprobante de pago?

6.1.2.2 RESPUESTAS PROPORCIONADAS POR LOS USUARIOS

El resultado obtenidos para determinar el tiempo promedio en minutos que tarda un usuario para realizar un comprobante de pago.

Nº USUARIO	TIEMPO EN MINUTOS
U1	5

U2	5
U3	5
U4	5
U5	5
U6	5

6.1.2.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el tiempo promedio en minutos que tarda un usuario en realizar un comprobante de pago para el total de usuarios encuestados utilizamos la siguiente formula:

$$PTR = \sum_{i=1}^{10} \frac{T_i}{N}$$

$$PTR = 5$$

Donde:

PTR: promedio de tiempo de registro.

T: Tiempo para realizar un comprobante. N: Número de usuarios encuestados.

6.1.2.4 RESULTADOS

Como resultados determinamos que el tiempo promedio en minutos que tarda un usuario en realizar un comprobante de pago para el total de usuarios encuestados es de 5.

6.1.3 PREGUNTA 3: TIEMPO PROMEDIO EN MINUTOS PARA LA BUSQUEDA DE INFORMACIÓN PARA LOS REPORTES QUE SE ENVIAN A MINCETUR

6.1.3.1 PREGUNTA ASOCIADA

¿cual es el tiempo promedio en minutos para la búsqueda de información para los reportes que se envían a MINCETUR?

6.1.3.2 RESPUESTAS PROPORCIONADAS POR LOS USUARIOS

El resultado obtenidos para determinar el tiempo promedio en minutos que tarda un usuario en buscar información para para los reportes que se envían e MINCETUR.

Nº USUARIO	TIEMPO EN MINUTOS
U1	10
U2	10
U3	11
U4	11
U5	10
U6	10

6.1.3.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el tiempo promedio en minutos que tarda un usuario en buscar información para el total de usuarios encuestados utilizamos la siguiente formula:

$$PTR = \sum_{i=1}^{10} \frac{T_i}{N}$$

$$PTR = 10.3$$

Donde:

PTR: promedio de tiempo de registro.

T: Tiempo para la búsqueda de información N: Número de usuarios encuestados.

6.1.3.4 RESULTADOS

Como resultados determinamos que el tiempo promedio en minutos que tarda un usuario en realizar búsqueda de información para el total de usuarios encuestados es de 10.3.

6.1.4 PREGUNTA 4: CUALES SON LOS PROBLEMAS COMUNES QUE SE PRESENTAN EN LA EMPRESA.

6.1.4.1 PREGUNTA ASOCIADA

¿cuales son los problemas comunes que se presentan en la emprsas?

6.1.4.2 RESULTADOS

- ✓ Demora en la búsqueda de información.
- ✓ Mal cálculo de cobro por el servicio.
- ✓ Confusión de los datos al momento de realizar un comprobante de pago.

6.2 RESULTADOS DE LA SEGUNDA ENCUESTA

La segunda encuesta fue realizada a 4 usuarios, empleados y el administrador de la empresa del sistema computarizado para la Administración del Rubro Hotelero de la empresa de Serv. Turístico Princess S.A. la realización de la encuesta, obtención de los datos y el análisis de los mismos fue realizado durante el mes de Diciembre de 2015, dichos datos fueron recolectados después de que el sistema fue puesto en marcha.

Esta primera encuesta consta de de preguntas a continuación presentamos los resultados obtenidos por cada una de ellas.

6.2.1 PREGUNTA 1: TIEMPO PROMEDIO EN MINUTOS PARA EL REGISTRO UN CLIENTE HACIENDO USO DEL SISTEMA

6.2.1.1 PREGUNTA ASOCIADA

Haciendo uso del sistema ¿cual es el tiempo promedio en minutos para registrar un cliente?

6.2.1.2 RESPUESTAS PROPORCIONADAS POR LOS USUARIOS

El resultado obtenido para determinar el tiempo promedio en minutos que tarda un usuario en registrar un cliente haciendo uso del sistema es:

N° USUARIO	TIEMPO EN MINUTOS
U1	2
U2	2
U3	3
U4	2

6.2.1.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el tiempo promedio en minutos que tarda un usuario en registrar un cliente haciendo uso del sistema de información para el total de usuarios encuestados utilizamos la siguiente formula:

$$PTR = \sum_{i=1}^{10} \frac{T_i}{N}$$

$$PTR = 2.25$$

Donde:

PTR: promedio de tiempo de registro. T: Tiempo para el registro.

N: Número de usuarios encuestados.

6.2.1.4 RESULTADOS

Como resultados determinamos que el tiempo promedio en minutos que tarda un usuario en registrar un nuevo cliente haciendo uso del sistema para el total de usuarios encuestados es de 2.25.

6.2.2 PREGUNTA 2: TIEMPO PROMEDIO EN MINUTOS PARA REALIZAR UN COMPROBANTE DE PAGO HACIENDO USO DEL SISTEMA

6.2.2.1 PREGUNTA ASOCIADA

Haciendo uso del sistema ¿cual es el tiempo promedio en

minutos para realizar un comprobante de pago?

6.2.2.2 RESPUESTAS PROPORCIONADAS POR LOS USUARIOS

El resultado obtenido para determinar el tiempo promedio en minutos que tarda un usuario en realizar un comprobante de pago haciendo uso del sistema es:

Nº USUARIO	TIEMPO EN MINUTOS
U1	2
U2	2
U3	3
U4	2

6.2.2.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el tiempo promedio en minutos que tarda un usuario en realizar un comprobante de pago haciendo uso del sistema para el total de usuarios encuestados utilizamos la siguiente formula:

$$PTR = \sum_{i=1}^{10} \frac{T_i}{N}$$

$$PTR = 2.25$$

Donde:

PTR: promedio de tiempo de registro. T: Tiempo para el registro.

N: Número de usuarios encuestados.

6.2.2.4 RESULTADOS

Como resultados determinamos que el tiempo promedio en minutos que tarda un usuario en realizar un comprobante de pago haciendo uso del sistema para el total de usuarios encuestados es de 2.25.

6.2.3 PREGUNTA 3: TIEMPO PROMEDIO EN MINUTOS PARA REALIZAR LA BÚSQUEDA DE INFORMACIÓN PARA LOS REPORTES QUE SE ENVIAN A MINCETUR HACIENDO USO DEL SISTEMA

6.2.3.1 PREGUNTA ASOCIADA

Haciendo uso del sistema ¿cual es el tiempo promedio en minutos para la búsqueda de información para los reportes que se envian a MINCETUR?

6.2.3.2 RESPUESTAS PROPORCIONADAS POR LOS USUARIOS

El resultado obtenido para determinar el tiempo promedio en minutos que tarda un usuario en buscar información para los reportes que se envian a MINCETUR haciendo uso del sistema es:

Nº USUARIO	TIEMPO EN MINUTOS
U1	5
U2	5
U3	6
U4	5

6.2.3.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el tiempo promedio en minutos que tarda un usuario buscar información para los reportes que se envian ha MINCETUR haciendo uso del sistema para el total de usuarios encuestados utilizamos la siguiente formula:

$$PTR = \sum_{i=1}^{10} \frac{T_i}{N}$$

$$PTR = 5.25$$

Donde:

PTR: promedio de tiempo de registro. T: Tiempo para la búsqueda.

N: Número de usuarios encuestados.

6.2.3.4 RESULTADOS

Como resultados determinamos que el tiempo promedio en minutos que tarda un usuario en buscar información para los reportes que se envían a MINCETUR haciendo uso del sistema para el total de usuarios encuestados es de 5.25.

6.2.4 PREGUNTA 4: PORCENTAJE DE USUARIOS QUE CREE QUE SE SOLUCIONARON LOS PROBLEMAS COMUNES DENTRO DENTRO DE LA EMPRESA.

6.2.4.1 PREGUNTA ASOCIADA

Haciendo uso del sistema ¿cree que se solucionaron los problemas comunes que se presentan en la empresa?

6.2.4.2 RESPUESTAS PROPORCIONADA POR LOS USUARIOS

Los resultados obtenidos de los usuarios para determinar el porcentaje de usuarios que cree se solucionaron los problemas es:

Nº USUARIO	UMERO DE USUARIO	RCENTAJE DE USUARIOS
SI	4	100
NO	0	00

6.2.4.3 ANÁLISIS DE LOS DATOS OBTENIDOS

Para determinar el porcentaje de usuarios que cree se solucionaron los problemas en el manejo de expedientes

haciendo uso del sistema, se uso la siguiente fórmula:

$$PU = \frac{cU * 100}{N}$$

Donde:

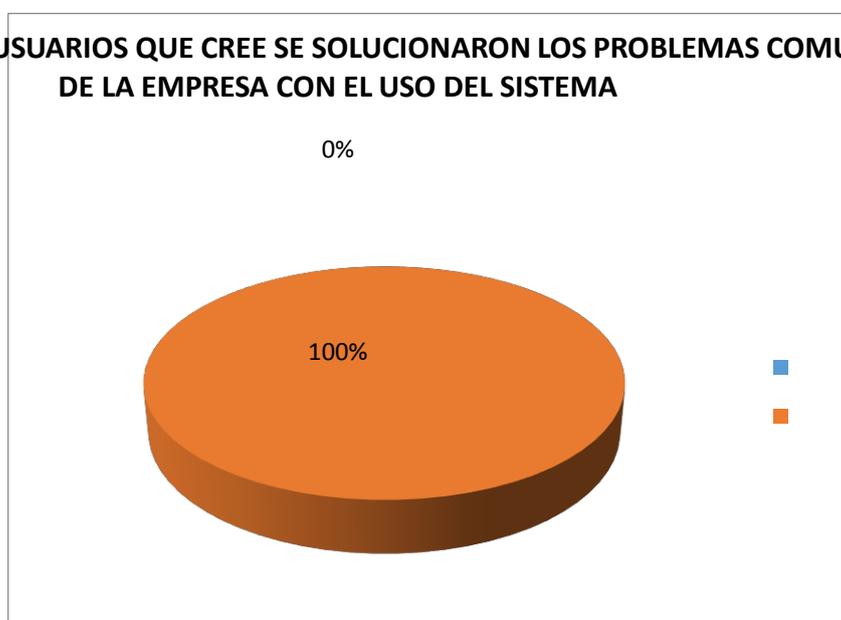
PU: Porcentaje de usuarios que cree se solucionaron los problemas dentro de la empresa.

CU: Cantidad de usuarios que cree se solucionaron los problemas dentro de la empresa.

N: número de usuarios encuestados.

PORCENTAJE DE USUARIOS QUE CREE SE SOLUCIONARON LOS PROBLEMAS COMUNES DE LA EMPRESA CON EL USO DEL SISTEMA

NO SI



6.2.4.4 RESULTADOS

Como resultados determinamos que el 100% de usuarios cree que con el uso del sistema se solucionaron los problemas comunes dentro de la empresa como:

- ✓ Demora en la búsqueda de información.
- ✓ Mal cálculo de cobro por el servicio.
- ✓ Confusión de los datos al momento de realizar un comprobante de pago

CAPITULO VII CONCLUSIONES Y RECOMENDACIONES

7.1 CONCLUSIÓN

- ✓ Con la implementación del Sistema Computarizado para la Administración del Rubro Hotelero de la Empresa de Serv. Turístico Princess S.A., se han obtenido resultados satisfactorios, con lo que se cumple con el objetivo general ya que ha contribuido de manera eficiente en la automatización de la información y la administración de la empresa.
- ✓ La metodología XP nos permite obtener un software de calidad y garantía, ya que ayudan a gestionar los cambios que se puedan presentar durante la implementación.
- ✓ La participación del cliente ha sido un aspecto clave en el éxito de la implementación del sistema. Las reuniones que se realizaron ayudaron a la definición y validación de los requerimientos del sistema.
- ✓ El desarrollo de los prototipos del sistema, agilizó la implementación del sistema ayudando al entendimiento del cliente, para las modificaciones requeridas tanto de interfaz como de funcionalidad.
- ✓ El desarrollo de los reportes permiten visualizar información la cual ayuda a una toma de decisiones adecuada.

7.2 RECOMENDACIONES

- ✓ El Sistema computarizado para la administración del rubro hotelero nos permite ver la necesidad automatizar las demás áreas de la empresa. Por lo que se recomienda crear una cartera de proyectos el cual le permita identificar, priorizar, administrar y controlar para lograr los objetivos de la empresa.

- ✓ Realizar una evaluación periódica de la interacción entre el usuario y el sistema, con el objetivo de realizar mejoras en las funcionalidades del sistema.

CAPITULO VIII GLOSARIO DE TÉRMINOS

8.1 GLOSARIO DE TÉRMINOS

- ✓ **Autenticación de usuario:** es el proceso de verificar la identidad digital del usuario y así conectarse a la aplicación.
- ✓ **Encriptación:** es el proceso mediante el cual cierta información es cifrada de forma que el resultado sea ilegible a menos que se conozcan los datos necesarios para su interpretación.
- ✓ **Factibilidad:** se refiere a la disponibilidad de los recursos necesarios para llevar a cabo los objetivos o metas señaladas.
- ✓ **Historia de usuario:** descripción de una funcionalidad que debe incorporar un sistema de software, y cuya implementación aporta valor al cliente. Se utiliza en las metodologías de desarrollo ágil.
- ✓ **Interfaz:** es el medio con que el usuario y la aplicación pueden interactuar.
- ✓ **Iteración:** cuando se hace referencia a un proyecto se refiere al desarrollo y entrega de componentes incrementales de funcionalidades de un negocio.
- ✓ **Prototipo:** es una representación de un sistema, aunque no es un sistema completo, permite entender ciertos aspectos de él y así clarificar los requerimientos.
- ✓ **Recepcionar:** recibir el expediente y verificar su estado.

CAPITULO IX REFERENCIAS BIBLIOGRÁFICAS

9.1 REFERENCIAS BIBLIOGRÁFICAS

LIBROS: [L01]

Charte, Francisco. *Programando Visual Basic 2010*. Madrid: Anaya Multimedia 2010.

[L02]

De La Mora, Eyssautier. *Metodología de la investigación y desarrollo de la inteligencia*,
México: ECASA, 1993.

[L03]

Freeman, A., y Steven Sanderson. *Pro asp.net MVC 3 Framework*. New York: Apress. 2011.

[L04]

Hernández, R., Fernández C. y Pilar Baptista. *Metodología de la Investigación Científica*. Mexico, D. F.: McGraw.Hill Interamericana, 2003.

[L05]

Pando, Yuri. *SQL Server 2008*. Lima: Editorial Macro. 2009.

[L06]

Palacio J., Ruata C. *Scrum Manager Gestion de Proyectos*. SafeCreative. 2011

[L07]

Cortizo J., Expósito D. y Ruiz Miguel. *eXtreme Programming*. 2008

SITIOS WEB: [WWW01]

Calvert, David (2009). Rapid Application Development.

<http://hebb.cis.uoguelph.ca/~dave/27320/new/rad.html>

[WWW02]

Microsoft (2012). Lo nuevo en .net framework <http://msdn.microsoft.com/es-es/library/ms171868.aspx>

[WWW03]

Microsoft (2012). Lo más destacado de Visual Studio 2010
<http://msdn.microsoft.com/es-es/library/dd547188.aspx>

[WWW04]

New Web Star (2011). AllFusion ERwin - Data Modeler V7
<http://www.newwebstar.com/programas/38412-allfusion-erwin-data-modeler-v7rs.html>

[WWW05]

Microsoft (2012). Bases de Datos del Sistema. <http://msdn.microsoft.com/es-es/library/ms178028%28v=sql.90%29.aspx>

[WWW06]

Microsoft (2012). Ediciones y componentes de SQL Server 2008
<http://msdn.microsoft.com/es-es/library/ms144275%28v=sql.100%29.aspx>

[WWW07]

Microsoft (2012). Introducción Técnica a Windows Server 2008.
<http://www.microsoft.com/spain/windowsserver2008/evaluation/overview.w.msp>

[WWW08]

Microsoft (2012). Diez motivos para adquirir Windows 7.
<http://windows.microsoft.com/es-VE/windows7/products/10-reasons-to-buy-Windows-7>

[WWW09]

Wikipedia (2014). Windows Server 2008.

http://es.wikipedia.org/wiki/Windows_Server_2008

[WWW10]

Wikipedia (2013). jQuery. <http://es.wikipedia.org/wiki/JQuery>

[WWW11]

Microsoft (2010). Project Professional 2010.

<http://www.microsoft.com/project/es/es/project-professional-2010.aspx>

[WWW12]

John Charles Olamendy (2008). Reporting Services. http://www.c-sharpcorner.com/uploadfile/john_charles/reporting-services2/

[WWW13]

WiWi Weblog (s.f.). IIS7

<http://wiwiloz.wordpress.com/iis-internet-information-server/>

ARCHIVOS PDF: [PDF01]

(Garcia, Conceptos Basicos de Base de Datos 2008)

[PDF02]

(Letelier y Penadés, Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP) 2006)

[PDF03]

(Palacio, El modelo Scrum 2006)