



TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS

**Título: Sistema de Gestión de Citas para la oficina de trámites de
identificación de la Universidad de las Ciencias Informáticas (UCI).**

Autores

Elizabeth María López Rivera
Nilberto Rodríguez Rojas

Tutores

Ing. Annie Cubas González
Ing. Renier Pérez García

Dedicatoria y agradecimientos

- *A* Dios por estar a mi lado en los momentos felices y tristes, gracias a él hoy puedo ser la persona más feliz del mundo.
- *A* mis padres Elia y Humberto por apoyarme en todo lo que he deseado, por brindarme su amor incondicional y esfuerzo. A ellos dedico el sacrificio y la devoción de estos años de estudio.
- *A* mi hermano Humbertico por quererme tanto y por su ayuda.
- *A* la familia Bajuelo por aceptarme como una hija más durante todos mis estudios y brindarme todo el amor, apoyo y ayuda, sin los cuales esto no sería posible.
- *A* mi amiga Aimet por quererme como una hermana, por inculcarme valores los cuales hicieron de mi una mejor persona y por hacerme levantar cuando pensaba que no podía. Estoy agradecida a Dios por haber tenido el privilegio de conocerla y haber podido disfrutar esa amistad.
- *A* mis tios Paco y Annia por estar siempre a mi lado cuando más lo necesitaba.
- *A* mis hermanos de la fé por poder contar con ellos en todo momento y brindarme una familia durante mis estudios en la Universidad, fue maravilloso compartir con ellos.
- *A* mi tutora Annie por estar en todo tiempo a nuestro lado y por ser nuestra guía para lograr los frutos de este trabajo.
- *A* mi compañero de tesis Nilbert por su dedicación y entrega en la realización de este proyecto. Gracias a la amabilidad, compromiso y responsabilidad que lo caracteriza, ha sido agradable compartir este tiempo.
- *A* todos los profesores por ser nuestra guía en todos estos años de estudio.
- *A* todos las personas que de una forma u otra han colaborado con la realización de este trabajo.

Elizabeth

DEDICATORIA Y AGRADECIMIENTOS

- *A* mis padres Nilberto y Emelinda por su eterno amor y cariño, por toda su ayuda, comprensión, educación y consejos.
- *A* mi hermana Elianne por ser la mejor hermana del mundo, por ayudarme, apoyarme en todo y estar cerca de mis padres cuando yo no pude.
- *A* mi cuñado Luis Mario por su ayuda y apoyo.
- *A* mi sobrina Melanis por su cariño.
- *A* mi bisabuelo Santiago y mis abuelos Elieser y María por estar siempre orgullosos de mí.
- *A* toda mi familia sin la cual esto no sería posible.
- *A* mis amigos Herbert y Royli, por portarse como hermanos, por su amistad y ayuda incondicional, por sus oportunos consejos y hasta regaños cada vez que hizo falta.
- *A* mis amigos José Miguel, Daniel, Aynel, Karel, Yoevis, Juan, Tayler por contar con ellos en todo momento, para mí es un privilegio contar con ustedes.
- *A* mi tutora por estar siempre apoyándonos en lo que hiciera falta, por confiar en nosotros, por todo el tiempo que nos ha dedicado.
- *A* mi compañera de tesis por su dedicación y por estar siempre dispuesta a colaborar y a conseguir cualquier ayuda posible.
- *A* todas las personas que de una forma u otra han colaborado con la realización de este trabajo.

Nilberto

Declaración de Autoría

Se declara que los únicos autores de este trabajo son Elizabeth María López Rivera y Nilberto Rodríguez Rojas y se autoriza a la Universidad de las Ciencias Informáticas los derechos patrimoniales del mismo, con carácter exclusivo.

Para que así conste se firma a los ___ días del mes de _____ del año ____ el presente documento.

Elizabeth María López Rivera

Nilberto Rodríguez Rojas

Ing. Annie Cubas González

Ing. Renier Pérez García

Síntesis de los Tutores

Ing. Renier Pérez García

Ingeniero Informático, CUJAE 2002.

Categoría docente: Profesor Asistente.

Profesor del Departamento de Ingeniería y Gestión de Software, Facultad 1, Universidad de las Ciencias Informáticas.

Jefe de Proyecto del Departamento de Identificación del Centro de Identificación y Seguridad Digital.

Correo electrónico: **renier@uci.cu**

Ing. Annie Cubas González

Ingeniero en Ciencias Informáticas, UCI 2011

Categoría docente:

Analista del Proyecto Plataforma Modular de Identificación y Control de Acceso, del Departamento de Identificación del Centro de Identificación y Seguridad Digital.

Correo electrónico: **acubas@uci.cu**

Resumen

En la Universidad de las Ciencias Informáticas existe gran afluencia de personal, por lo que se hace complejo mantener un control organizado y agilizado de entrega de documentos de identificación en la oficina de trámites de identificación de la misma. Luego de un estudio de sistemas informáticos existentes capaces de automatizar este proceso en el mundo, se determinó como solución la implementación de un Sistema de Gestión de Citas de carácter flexible y genérico capaz de adaptarse a los requerimientos que demanda la Universidad de las Ciencias Informáticas y a los que cualquier otra institución necesite, contribuyendo así al proceso de informatización de la sociedad. Para el desarrollo de la solución se hizo necesario analizar, diseñar, implementar y probar este sistema informático, que a través de un conjunto de funcionalidades lográndose satisfacer las necesidades reales del cliente. Durante la realización de esta solución informática se siguieron las pautas establecidas para el proceso de desarrollo de software, se emplearon todos los procedimientos, metodologías, herramientas y prácticas que facilitaron la obtención de los resultados esperados en cuanto a calidad de *software*, tiempo de desarrollo, ahorro de recursos y esfuerzo.

Palabras Claves: Cita, Gestión o Administración, Identificación.

Índice de contenido

Introducción	1
Capítulo 1: Fundamentación Teórica	5
1.1 Introducción.....	5
1.2 Conceptos básicos asociados al dominio del problema	5
1.3 Estudio Preliminar	5
1.3.1 Análisis Externo.....	5
1.3.2 Análisis Interno.....	10
1.4 Análisis de los sistemas estudiados	12
1.5 Tecnologías, metodologías y herramientas a utilizar	12
1.6 Conclusiones.....	21
Capítulo 2: Características del sistema	22
2.1 Introducción.....	22
2.2 Propuesta del sistema	22
2.3 Arquitectura propuesta	22
2.4 Modelo de Procesos de Negocio	24
2.4.1 Descripción de las actividades del sistema.....	26
2.5 Especificación de los requisitos de <i>software</i>	27
2.5.1 Requisitos funcionales.....	28
2.5.2 Requisitos no funcionales	28
2.6 Descripción de las funcionalidades.....	30
2.7 Clasificación de las funcionalidades	34
2.8 Planeación de las iteraciones	35
2.9 Diseño de las funcionalidades	35
2.9.1 Diagrama de clases.....	36
2.9.2 Descripción de las clases	36
2.9.3 Diagrama de secuencia	37
2.10 Modelo de datos.....	38
2.10.1 Descripción de las entidades	39
2.11 Patrones.....	40
2.12 Conclusiones.....	42
Capítulo 3: Implementación y prueba.....	43
3.1 Introducción.....	43

3.2 Configuración del sistema	43
3.3 Método para la distribución de las solicitudes de citas.....	44
3.4 Diagrama de componentes.....	45
3.5 Diagrama de despliegue.....	46
3.6 Estándares de codificación.....	47
3.6.1 Principios generales de nomenclaturas	47
3.7 Validación de la propuesta de solución.....	50
3.7.1 Definición y descripción de las pruebas.....	50
3.7.2 Aplicación de las pruebas.....	51
3.7.3 Resultados de las pruebas	56
3.8 Elementos de seguridad en el sistema	58
3.9 Conclusiones.....	59
Conclusiones Generales.....	60
Recomendaciones	61
Bibliografía Referenciada.....	62
Bibliografía Consultada	64
Glosario de términos y siglas	67
Anexos.....	69

Índice de tablas

Tabla 1: Asignar Cita Médica	6
Tabla 2: Buscar Cita Médica	7
Tabla 3: Modificar Cita Médica.....	7
Tabla 4: Eliminar Cita Médica	7
Tabla 5: Atender Cita Médica.....	7
Tabla 6: Descripción de las actividades de los procesos de gestión de citas	26
Tabla 7: Descripción de la funcionalidad “gestionar citas”	30
Tabla 8: Clasificación de las funcionalidades	34
Tabla 9: Cronograma de diseño y construcción	35
Tabla 10: Descripción de la clase "BusinessFactory"	36
Tabla 11: Descripción de la clase "RepositoryFactory"	37
Tabla 12: Descripción de la entidad dCiudadano	39
Tabla 13: Criterios de calidad.....	47
Tabla 14: Principios de nomenclaturas	47
Tabla 15: Descripción de las variables del caso de prueba.....	53
Tabla 16: Caso de prueba “gestionar citas”	54
Tabla 17: Resultados de las Pruebas Unitarias.....	56
Tabla 18: Resultados de las Pruebas de Caja Negra.....	57

Índice de figuras

Figura 1: Modelo de Procesos de Negocio.....	25
Figura 2: Diagrama de clases de la funcionalidad “gestionar citas”	36
Figura 3: Diagrama de secuencia de la funcionalidad “gestionar citas”	38
Figura 4: Modelo de datos	39
Figura 5: Ejemplo del uso del patrón Singleton	40
Figura 6: Ejemplo del uso del patrón Creador	41
Figura 7: Diagrama de componentes	46
Figura 8: Diagrama de despliegue	46
Figura 9: Ejecución de la clase "BusinessFactoryTest"	52
Figura 10: Gráfico del resultado de las Pruebas Unitarias.....	56
Figura 11: Gráfico del resultado de las Pruebas de Caja Negra.....	57

Introducción

El desarrollo vertiginoso de las nuevas tecnologías, particularmente en las ramas de la informática y las telecomunicaciones, evidencia que es esta la era con mayor evolución científico-técnica de todos los tiempos. Este desarrollo tecnológico acelerado incide sustancialmente en el mundo referencial del ser humano, pues, si bien facilita la adquisición y almacenamiento de nuevos conocimientos, nuevos enfoques y/o perspectivas y acciones que ayer mismo parecían inaccesibles pero, de la misma manera, le están condicionando y obligando a adaptaciones y replanteamientos en todos los órdenes de su existencia.

En el campo del desarrollo de *software* también se constata un avance sin precedentes a nivel mundial, lo cual posibilita la construcción de aplicaciones y sistemas informáticos capaces de solventar la amplia amalgama de problemas teóricos y prácticos de la sociedad. Diversas áreas de la vida diaria se ven afectadas continuamente por la industria del *software*, tales como los aeropuertos, los bancos, las grandes empresas, las instituciones policiales y militares entre otras.

Los sistemas de gestión son una rama importante dentro de la industria del desarrollo de *software* puesto que resuelven un gran número de situaciones a través de la automatización de procesos, lo cual facilita el trabajo en oficinas, elimina el tiempo de respuesta por parte de los proveedores de servicios a la población, suprime engorrosos bloques de papeles, propicia el trabajo de los seres humanos con grandes volúmenes de información, el control y manipulación de estadísticas y una innumerable cantidad de beneficios y posibilidades más, que a la postre, repercuten de forma positiva en la economía y la calidad de vida de todos aquellos países que explotan este tipo de sistemas informáticos.

En Cuba, con la intención de utilizar la amplia gama de posibilidades que brinda la industria del *software* se trabaja hace ya algún tiempo en el desarrollo de este tipo de herramientas, donde la Universidad de las Ciencias Informáticas asume un rol protagónico en esta tarea. Justamente, la informatización de los procesos que se llevan a cabo en una empresa o entidad resulta un elemento clave. Un proceso que resulta prioritario automatizar por la importancia que tiene en la sociedad se encuentra relacionado con la gestión de citas pues de este dependerá el eficiente control y mejor planificación del tiempo en las oficinas de trámites.

La Universidad de las Ciencias Informáticas es una de las entidades que sufre afectaciones en la planificación de citas a la oficina de trámites de identificación. Cada año, al comenzar el curso escolar, en esta oficina se aglomeran muchos usuarios para adquirir su credencial (documento que los acredita como personal autorizado en la Universidad). Esto se debe a la amplia matrícula de estudiantes, profesores

recién graduados y trabajadores que posee la Universidad, además de que todo el proceso es realizado por una sola persona en la oficina.

Esta situación provoca consecuencias como: demora en el proceso de acreditación; pérdida de tiempo de las personas que solicitan el documento debido a las largas horas de espera para ser atendido en la oficina. Además de que existe un poco aprovechamiento de la capacidad máxima de tramitar documentos debido a la falta de planificación, pues hay horarios en los que asisten demasiadas personas a realizar el trámite y otros donde no hay fluctuación de personas accediendo al servicio, lo cual suscita el descontento entre los estudiantes y trabajadores de la Universidad.

Basado en lo antes expuesto se plantea como **problema de investigación** ¿Cómo organizar el flujo de personas para la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas?

Según el problema científico identificado, se define como **objeto de estudio**: Proceso de Gestión de citas.

La investigación tiene como **objetivo general**: Desarrollar un sistema que permita gestionar las citas para la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas, del cual se derivan los siguientes **objetivos específicos**:

- Desarrollar el marco teórico de la investigación
- Realizar análisis y diseño para la gestión de citas
- Implementar el Sistema de Gestión de Citas utilizando tecnología *web*
- Probar la solución implementada del Sistema de Gestión de Citas

De donde se deriva como **campo de acción** el proceso de gestión de citas en la Universidad de las Ciencias Informáticas.

Durante la investigación se asume como **idea a defender**: Con el desarrollo del sistema se gestionarán las citas para la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas organizando el flujo de personas que necesitará el servicio de dicha entidad.

Para el desarrollo de la investigación los **métodos científicos** empleados son:

Métodos teóricos:

- Histórico–lógico: Facilitó la aproximación a los referentes teóricos del tema para constatar teóricamente cómo el desarrollo de las tecnologías ha posibilitado mejoras en los procesos que intervienen en la gestión de citas.

- Modelación: Se realizó la modelación del proceso que interviene en la gestión de citas, ya que permite predecir las respuestas ante variaciones de algunos de sus parámetros, sin tener que ejecutarlos en la realidad.

Técnicas de obtención de información y recolección de datos:

- Revisión bibliográfica y documental. Para identificar las premisas teóricas de los temas que se abordan en este estudio, se realizó un análisis de la bibliografía nacional e internacional disponible.
- Las entrevistas se realizaron a profundidad en una sesión, en horario de la tarde y en condiciones de privacidad. Todas las entrevistas fueron guiadas por los autores de la presente investigación con el propósito de obtener información relevante referente a los procesos para la asignación de citas que no fueron encontrados en otras bibliografías.

Justifica la realización de la presente investigación la necesidad de resolver un problema coyuntural en la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas. Dicha solución hará más organizado y rápido el servicio para quienes lo solicitan. Para el cual se realizará un sistema genérico que podrá ser adaptable y configurable en otra entidad que requiera de este servicio.

Para cumplir con el objetivo se proponen las siguientes **tareas de la investigación:**

- Análisis de los principales conceptos asociados a la gestión de citas para obtener una base teórica necesaria para el desarrollo de la solución.
- Caracterización de las soluciones informáticas de gestión de citas existentes, teniendo en cuenta las soluciones en la industria del *software*, las soluciones de código abierto y las tendencias en el mundo académico.
- Comparación de las funcionalidades de estos sistemas.
- Caracterización de las técnicas de usabilidad y experiencia de usuario para el desarrollo de aplicaciones *web*.
- Definición de las herramientas informáticas y metodologías a usar para el desarrollo del Sistema de Gestión de Citas.
- Identificación de los procesos de negocio del Sistema de Gestión de Citas.
- Modelación y descripción de los procesos de negocio del Sistema de Gestión de Citas.
- Definición de los requisitos funcionales y no funcionales del Sistema de Gestión de Citas.
- Clasificación de los requisitos funcionales en básicos, estándar y empresarial.
- Diseño de prototipos de interfaz de usuario utilizando técnicas de usabilidad para la web.

- Descripción de los requisitos del Sistema de Gestión de Citas.
- Diseño de las clases del Sistema de Gestión de Citas.
- Implementación de los componentes que permitan la gestión de citas.
- Realización de las pruebas de caja negra al Sistema de Gestión de Citas en función de validar la propuesta de solución.

El presente documento consta de 3 capítulos:

Capítulo 1: Fundamentación teórica: Se caracteriza los conceptos fundamentales relacionados con el objeto de estudio. Se realiza un estudio detallado referente a los Sistemas de Gestión de Citas a escala nacional e internacional, así como las herramientas, lenguajes y metodologías a utilizar para el desarrollo de la solución.

Capítulo 2: Propuesta de solución: Se realiza un análisis de los procedimientos vinculados al campo de acción, obteniéndose el modelo de procesos para la gestión de citas con la especificación de todas sus actividades. Se define una lista de funcionalidades que son descritas detalladamente para guiar la implementación de la solución. Se realiza la planeación de las iteraciones que se llevará a cabo para diseñar y construir las funcionalidades. Se propone el diseño de las funcionalidades.

Capítulo 3: Implementación y prueba: Se incluye la implementación de las funcionalidades del proceso de gestión de citas, así como la validación de la propuesta de solución a través de los resultados obtenidos al ser aplicadas las pruebas.

Capítulo 1: Fundamentación Teórica

1.1 Introducción

Un sistema de gestión es un conjunto de etapas unidas en un proceso continuo, que permite trabajar ordenadamente una idea hasta lograr mejoras y su continuidad. La Universidad de las Ciencias Informáticas al emitir los documentos de identificación requiere de una óptima organización, por lo cual se hace necesario estudiar el proceso de gestión de citas y sistemas que automaticen procesos de emisión de documentos de identificación para comprender cómo se realiza la gestión de procesos en un sistema de información. El estudio de los sistemas de gestión de citas y la selección de las tecnologías, metodologías y herramientas permiten proponer una solución que se ajuste a sus características.

1.2 Conceptos básicos asociados al dominio del problema

Para poder entender la gestión de citas se debe tener conocimiento de los siguientes conceptos:

Gestión o Administración: El concepto de gestión hace referencia a la acción y a la consecuencia de administrar o gestionar algo. Al respecto, hay que decir que gestión informática es llevar a cabo la acción de adicionar, modificar y eliminar entidades en sistemas informáticos. Administrar, por otra parte, abarca las ideas de disponer, ordenar u organizar los activos que son gestionados por sistemas.

Cita: Una cita no es más que un encuentro previamente acordado entre dos o más personas en una fecha, hora y lugar determinados. (1)

Identificación: El término identificación se usa para designar al acto de identificar, reconocer o establecer los datos e información principal sobre una persona. La identificación, también puede ser el nombre que llevan determinadas documentaciones que tienen por objetivo justamente establecer la identidad de una persona o individuo. (2)

1.3 Estudio Preliminar

1.3.1 Análisis Externo

En el documento *"Manual de Usuario del Sistema de Cita Previa de DNI y Pasaporte Electrónico"* se describen los procesos de Cita Previa del DNI y Pasaporte Electrónico para el Nivel de Acceso desde *Internet*. (3) El mismo plantea:

Los ciudadanos a través de *Internet* podrán exclusivamente solicitar, verificar y anular citas para la expedición de DNI y/o pasaporte, así como recabar información sobre equipos de expedición disponibles

con sus horarios y nivel de ocupación a los efectos de ayudarles a elegir el punto de expedición más adecuado.

Los ciudadanos que deseen solicitar una cita para la renovación o expedición de DNI o pasaporte a través de *Internet* deben conectarse al portal “*www.citapreviadnie.es*”. Inicialmente se presentará una pantalla de Bienvenida, desde donde marcarán el enlace correspondiente para iniciar la solicitud. El sistema desglosa los procesos siguientes:

- Solicitar Cita

Se trata de una pantalla resumen con los datos asociados a una cita. Se divide en tres zonas: datos de la cita, datos del trámite y datos de contacto. Ver (Anexo 1-Figura 1)

- Confirmar Cita

Una vez completado el paso anterior, se mostrará la pantalla de confirmación de cita que incluirá además de los datos relativos a la cita un número de cita asignado. El usuario podrá comprobar que todos los datos de su cita son correctos y podrá obtener un documento de su cita en formato PDF pulsando en la opción “Imprimir en PDF”. Ver (Anexo 1-Figura 2)

- Gestión de Citas - Consulta y Anulación

Para consultar o anular una cita previamente solicitada el ciudadano deberá seleccionar la opción “Consultar Citas”. Si un ciudadano únicamente ha concertado una cita, accederá automáticamente a la pantalla de anulación individual. A ésta pantalla también accederá al marcar una cita en el listado de anulación si ha concertado más de una. Ver (Anexo 1-Figura 3 y 4)

La tesis “**Sistema de Información para la Gestión de las Citas en los Centros de Imágenes Diagnósticas, Pereira**” realizado por Gustavo Andrés Castro Guevara y Julián Alberto Ramírez Osorio el 5 de Mayo de 2009 plantea que ciertos centros de imágenes diagnósticas de la región no cuentan con un *software* adecuado para gestionar funciones operativas y administrativas del mismo, como es la gestión de las citas, debido a este problema se observa que puede estar afectando indirectamente la prestación del servicio hacia los pacientes, puesto que se tarda un tiempo considerable al momento de realizar la solicitud de una cita y no se optimiza el factor tiempo, para lograr atender mayor número de pacientes. Al implementarse un sistema de información orientado a la *web* para la gestión de citas de los pacientes, se brinda un canal adicional para gestionar sus citas. (4) El proceso se realiza de la siguiente manera:

Tabla 1: Asignar Cita Médica

Usuario	Sistema
---------	---------

1 Solicita la pantalla de asignar cita médica.	2 Presenta la pantalla para captura de datos.
3 Llena los datos solicitados.	4 Válida los datos ingresados.
	5 Verifica disponibilidad de fecha y hora.
6 Confirma registro de cita.	7 Guarda el registro de la cita.
	8 Presenta pantalla de operación exitosa.

Tabla 2: Buscar Cita Médica

Usuario	Sistema
1 Solicita la pantalla de buscar cita médica.	2 Presenta la pantalla para captura de datos.
3 Llena los datos pedidos.	4 Válida los datos ingresados.
	5 Presenta datos solicitados.

Tabla 3: Modificar Cita Médica

Usuario	Sistema
1 Solicita la pantalla de modificar cita médica.	2 Llama al caso de uso buscar cita médica.
3 Escoge fecha y hora para la cita.	4 Verifica disponibilidad de fecha y hora.
5 Confirma modificación cita.	6 Guarda datos de la cita.
	7 Presenta pantalla de operación exitosa.

Tabla 4: Eliminar Cita Médica

Usuario	Sistema
1 Solicita la pantalla de eliminar cita médica.	2 Llama al caso de uso buscar cita médica.
3 Confirma eliminar cita.	4 Modifica el registro de citas.
	5 Presenta pantalla de operación exitosa.

Tabla 5: Atender Cita Médica

Usuario	Sistema
1 Solicita la pantalla de atención cita.	2 Llama al caso de uso buscar cita médica.
3 Confirma asistencia.	4 Modifica el registro de citas.
	5 Presenta pantalla de operación exitosa.
	6 Genera factura de cobro.
7 Realiza pago correspondiente.	8 Registra pago de factura.

El “**Sistema Integral y Distribuido de Citas (Antizipa)**” proporciona la transparencia y la atención personalizada que el demandante de salud requiere, permite el acceso sólo a su casillero, incluso a su histórico de salud cumpliendo los requisitos de seguridad que para cada caso se decidan. El sistema, permite la asistencia remota a los pacientes con seguimiento de tratamientos e incidencias. El Sistema Integral de Citas convierte la naturaleza específica de cada acto sanitario en categorías acumulables y analizables en función de los informes que se requieran (informes en tiempo real, visibles o no, en función de las políticas de difusión que se determine). La novedad reside en que los mismos pacientes si así lo desean junto con centros, personal auxiliar y equipos médicos, utilizarán indistintamente la *web* en virtud de sus permisos y atributos y sin importar donde se encuentren. Todo el personal autorizado, no importa donde esté, podrá acceder al sistema integral de citas en función de sus atributos y con su respectivo nivel de privilegio y todo ello con los niveles de seguridad que cada situación requiera, crecientemente exigentes en la misma medida que se eleva del nivel de protección de los datos. *Antizipa*¹ ofrece un sistema integrado, robusto y seguro. (5)

El “**Software de Gestión de Citas y facturación para clínicas**” plantea el desarrollo de un *software* para el sector de las clínicas, dado que los habituales existentes en el mercado tenían diversas deficiencias de las cuales se quejaban los clientes del sector, tales como usar bases de datos poco eficientes, no tener completamente integradas todas las funciones a ejecutar en la secretaría de una clínica. Está desarrollado para clínicas dermatológicas y clínicas oculares, pero se puede hacer adaptaciones a todo tipo de clínicas, dado que el módulo de gestión de citas y facturación es común a todas. (6) Algunas características del sistema son:

- El sistema se basa en una potente base de datos en *SQL Server 2008*, para no tener mermas de rendimiento al manipular muchos pacientes.
- Se ha programado usando las últimas tecnologías, en *Visual Basic .NET*, con controles muy visuales y de fácil uso para reducir al mínimo la curva de aprendizaje para cualquier persona con unos conocimientos mínimos generales de informática a nivel de usuario.
- Se integra perfectamente el paso de una función a otra, es decir, mientras vemos la ficha de un contacto podemos crearle una cita directamente, mientras estamos viendo una cita podemos facturar o ver el historial completo del paciente.
- Se lleva un estricto control de usuarios, como emisión de informes para ejecutar los derechos de los pacientes.

¹ Sistema Integral y Distribuido de Citas.

El sistema "**Concertar Cita Consular**" realizado en el 2009. Está preparado para que una cita para optar a la nacionalidad española se pueda concertar vía *web*. (7) Para concertar una cita, el ciudadano debe realizar los siguientes pasos:

1. Seleccionar la Oficina Consular.
2. Rellenar un formulario con los datos personales del solicitante. Ver (Anexo 2-Figura 1)
 - Nombres y Apellidos
 - Teléfono y correo electrónico
 - Nombre de usuario
 - Contraseña
 - Repetir contraseña
3. Concertar Hora (Seleccionar fecha y hora para la cita): El solicitante podrá observar cuál es el primer día que tiene disponible para concertar su cita consular. Desde el calendario podrá cambiar de mes y seleccionar el día (casillas blancas) para observar qué horas hay disponibles. Ver (Anexo 2-Figura 2)
4. Datos de la Solicitud: Indica cuáles son las condiciones en las que se encuentra el solicitante para optar a la nacionalidad española. Ver (Anexo 2-Figura 3)
5. Confirmar Cita: Una vez realizados todos los pasos, al ciudadano se le presenta el resumen de su cita y deberá confirmarla. Ver (Anexo 2-Figura 4)

En el documento "**Manual de Usuario Paquete del Centro de Personalización e Impresión de Documentos (CPDI) versión 1.0**" se describen los procesos de Gestión de Cupos y Citas. (8) Este sistema proporciona las siguientes funcionalidades:

- Cambio de Citas

Las personas al entrar en el portal de pasaporte para solicitar una cita le son dadas una planilla con sus datos básicos y el serial de solicitud. Si por cuestiones ajenas a la persona no es posible iniciarle el trámite para el día asignado y a consideración de los encargados, es posible cambiar la cita para otro día. También puede darse el caso de algún feriado, o algún día que no se pudo trabajar en la oficina, el sistema permite cambiarle las citas a una oficina seleccionada. Ver (Anexo 3-Figura 1)

- Configurar horarios de oficinas

Los horarios de trabajo de las oficinas son configurables, al igual que los cupos asignados a ella. Para ello basta con ir al menú correspondiente, donde se mostrará un listado con las oficinas de trámites, al seleccionar alguna mostrará los horarios de atención y los cupos asignados.

Para configurar horarios es necesario seleccionar primero la oficina correspondiente, luego se selecciona el día al cual se le modificará el horario en caso de que no tenga ninguno asociado. Una vez modificado, en caso necesario, se indica la cantidad de cupos que tendrá. Ver (Anexo 3-Figura 2)

- Configurar horarios de entrega

Los horarios de entrega son los dedicados a las personas con el trámite concluido y con el pasaporte listo para ser entregado. Para ello se selecciona a la oficina correspondiente y seguidamente se mostrarán los horarios de atención los cuales podrán ser modificados. Ver (Anexo 3-Figura 3)

- Configurar estados de oficinas

Esta opción permite asignarle a una oficina los estados que atiende o quitárselos. Para ello se muestra un listado con las oficinas y al seleccionar alguna el sistema muestra los estados que atiende y los estados disponibles, permitiendo asignar o quitar estados en dependencia del caso. Ver (Anexo 3-Figura 4)

- Configurar solicitudes por estado

Esta opción permite asignarle a un estado las solicitudes que tendrá para los diferentes días. Para ello se muestra un listado con los estados y al seleccionar alguno el sistema muestra las solicitudes asignadas, además del total de cupos y los cupos disponibles. Para modificar los cupos seleccione el día correspondiente y escriba la cantidad de solicitudes a asignar en dependencia de la cantidad disponible. Ver (Anexo 3-Figura 5)

El "**Sistema de Atención Consular**" brinda la opción de iniciar trámites y servicios consulares mediante una cita. Está diseñado para que el cliente pueda solicitar una cita para optar a la nacionalidad chilena y se haga vía *web*. (9) Para concertar una cita, el ciudadano puede seleccionar las siguientes opciones:

1. Ingresar Solicitud (el usuario debe seleccionar el tipo de solicitud a realizar). Ver (Anexo 4-Figura 1)
2. Llenar formulario con los datos solicitados (los datos mostrados son en dependencia del tipo de solicitud seleccionada). Ver (Anexo 4-Figura 2)
3. Historial Solicitudes. Ver (Anexo 4-Figura 3)
4. Modificar Mis Datos. Ver (Anexo 4-Figura 4)
5. Cambiar Clave.

1.3.2 Análisis Interno

La tesis "**Módulo Citas del Sistema de Información Hospitalaria alas HIS**" realizada por Yosmel Martínez Díaz Alejandro y Tarafa Guzmán en junio de 2009 plantea que los Sistemas de Información Hospitalaria tienen entre sus componentes principales un módulo capaz de automatizar todo el proceso de planificación y asignación de citas dentro de los hospitales. El área de citas de las instituciones hospitalarias maneja grandes volúmenes de información que se procesa de forma manual, salvo en algunos casos que emplean aplicaciones aisladas que generalmente son propietarias. Además, permiten la planificación de citas solo para un determinado servicio del hospital. Por estas razones se desarrolló el Módulo de Citas del Sistema de Información Hospitalaria alas HIS, con el objetivo de facilitar la gestión de

información en las áreas de las instituciones hospitalarias. (10) Este sistema trae consigo los siguientes procesos:

- Asignar en el servicio cita de primera²

Inicia cuando el técnico(a) de registros y estadísticas de salud de consulta externa accede a la opción “Asignar cita de primera”, el sistema brinda la posibilidad de seleccionar especialidad, médico, fecha para citar al paciente. El técnico(a) selecciona la especialidad, el médico, la fecha y confirma la cita. El sistema crea la cita de primera y la tarjeta de cita.

- Asignar en el servicio cita de control³

Inicia cuando el técnico(a) de registros y estadísticas de salud de consulta externa accede a la opción “Asignar cita de control”, el sistema brinda la posibilidad de seleccionar especialidad, médico y fecha para citar al paciente. El técnico(a) selecciona la especialidad, el médico, la fecha y confirma la cita. El sistema crea la cita de control y la tarjeta de cita.

- Buscar cita para triaje⁴ especializado desde el servicio

Inicia cuando el técnico(a) de registros y estadísticas de salud de consulta externa accede a la opción “Buscar Cita para triaje especializado”, el sistema brinda la posibilidad de introducir criterios de búsqueda para localizar la cita. El técnico(a) introduce los datos que considera como criterios para realizar una búsqueda; el sistema busca y muestra las citas que cumplen con los criterios de búsqueda. El técnico(a) puede seleccionar la opción “Modificar o Eliminar una Cita para triaje especializado”.

- Buscar cita para charla desde el servicio

Inicia cuando el técnico(a) de registros y estadísticas de salud de consulta externa accede a la opción “Buscar Cita para charla”, el sistema brinda la posibilidad de introducir criterios de búsqueda para localizar la cita. El técnico(a) introduce los datos que considera como criterios para realizar una búsqueda; el sistema busca y muestra las citas que cumplen con los criterios de búsqueda. El técnico(a) puede seleccionar la opción “Modificar o Eliminar una Cita para charla”.

- Asignar cita de primera en la central de citas

Inicia cuando el técnico(a) de registros y estadísticas de salud de central de citas accede a la opción “Asignar cita de primera”, el sistema brinda la posibilidad de seleccionar servicio, especialidad, médico, y fecha para citar al paciente. El técnico(a) selecciona el servicio, la especialidad, el médico, la fecha y confirma la cita. El sistema crea la cita de primera y la tarjeta de cita.

²Consulta que se le hace a un paciente que viene por primera vez a atenderse determinada patología.

³Consulta que se le hace a un paciente que viene a atenderse por una patología anteriormente diagnosticada en el mismo servicio.

⁴Consulta donde el paciente sin historia clínica lo valora el médico y canaliza la situación.

- Buscar cita para triaje especializado desde la central de citas

Inicia cuando el técnico(a) de registros y estadísticas de salud de central de citas accede a la opción “Buscar Cita para triaje especializado”, el sistema brinda la posibilidad de introducir criterios de búsqueda para localizar la cita. El técnico(a) introduce los datos que considera como criterios para realizar una búsqueda, el sistema busca y muestra las citas que cumplen con los criterios de búsqueda, el técnico(a) puede seleccionar la opción “Modificar o Eliminar una Cita para triaje especializado”.

1.4 Análisis de los sistemas estudiados

Los sistemas de gestión de citas desarrollados en el mundo y en Cuba descritos anteriormente no se pueden utilizar pues están desarrollados única y exclusivamente para instalaciones específicas, además que son privativos en algunos casos y utilizan tecnologías muy costosas. De los sistemas de gestión de citas estudiados se seleccionaron funcionalidades, tecnologías y herramientas que posteriormente se utilizarán en la modelación y diseño del sistema propuesto por la presente investigación.

1.5 Tecnologías, metodologías y herramientas a utilizar

La informatización de la sociedad en los últimos años ha sido el motor impulsor del desarrollo de nuevas tecnologías, metodologías y herramientas para facilitar este propósito. En el ámbito mundial existen herramientas que permiten lograr en tiempo record el desarrollo de grandes sistemas que manipulan un amplio volumen de información. Además de posibilitar un alto grado de calidad gracias a las facilidades que implementan dichas herramientas.

En la actualidad para realizar el desarrollo de un *software* existen ciertos criterios a tener en cuenta como son los lenguajes de programación a utilizar, gestores de bases de datos, metodologías, entre otros. A continuación se presenta el resumen realizado durante la investigación y la propuesta tecnológica realizada por el proyecto Plataforma Modular de Identificación y Control de Acceso.

.NET Framework 4.0

El *Framework .Net* es una infraestructura sobre la que se reúne todo un conjunto de lenguajes y servicios que simplifican enormemente el desarrollo de aplicaciones, mediante la cual se ofrece un entorno de ejecución altamente distribuido, que permite crear aplicaciones robustas y escalables. (11)

Los principales componentes de este entorno son:

- Lenguajes de compilación
- Biblioteca de clases *.Net*
- CLR (*Common Language Runtime*)

A continuación se resumen las ventajas más importantes que proporciona .Net *Framework* 4.0:

- **Código administrado:** El CLR realiza un control automático del código para que este sea seguro, es decir, controla los recursos del sistema para que la aplicación se ejecute correctamente.
- **Interoperabilidad multilinguaje:** El código puede ser escrito en cualquier lenguaje compatible con .Net ya que siempre se compila en código intermedio (MSIL).
- **Compilación *just-in-time*:** El compilador JIT incluido en el *Framework* compila el código intermedio (MSIL) generando el código máquina propio de la plataforma. Se aumenta así el rendimiento de la aplicación al ser específico para cada plataforma.
- **Garbage collector:** El CLR proporciona un sistema automático de administración de memoria denominado recolector de basura (*garbage collector*). El CLR detecta cuándo el programa deja de utilizar la memoria y la libera automáticamente. De esta forma el programador no tiene por qué liberar la memoria de forma explícita aunque también sea posible hacerlo manualmente (mediante el método `dispose()` liberamos el objeto para que el recolector de basura lo elimine de memoria).
- **Seguridad de acceso al código:** Se puede especificar que una pieza de código tenga permisos de lectura de archivos pero no de escritura. Es posible aplicar distintos niveles de seguridad al código, de forma que se puede ejecutar código procedente de la *web* sin tener que preocuparse si esto va a estropear el sistema.
- **Despliegue:** Por medio de los ensamblados resulta mucho más fácil el desarrollo de aplicaciones distribuidas y el mantenimiento de las mismas. El *Framework* realiza esta tarea de forma automática mejorando el rendimiento y asegurando el funcionamiento correcto de todas las aplicaciones.

***NHibernate* 3.1.0**

En el desarrollo de gran número de aplicaciones empresariales se hace necesario persistir la información manejada por el sistema en bases de datos. En este proceso se genera una gran cantidad de código repetitivo y que, además, es obligatorio (código *boilerplate*) para manejar las conexiones a la base de datos, generar consultas entre otros. Esto provoca que el código de las aplicaciones se vuelva engorroso y que los programadores se desconcentren de los problemas que realmente tienen que resolver de lógica del negocio por atender detalles de una tecnología en específico.

NHibernate es un marco que permite hablar de una base de datos relacional de una manera orientada a objetos. Se puede almacenar (o como también se suele decir, "persistir") los objetos de una base de datos y cargar los objetos de la base de datos más adelante. *NHibernate* genera las sentencias SQL necesarias para insertar, actualizar, eliminar y cargar datos. No es el único marco ORM para .NET, pero es probablemente el más maduro y rico de las características de todos. (12)

Lenguaje de programación

C# es el lenguaje de propósito general diseñado por *Microsoft* para ser utilizado, por lo que programar usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes que existen. (13) Entre sus principales características se destacan:

- **Sencillez:** C# elimina elementos que otros lenguajes incluyen y que son innecesarios en .NET, por ejemplo: El código escrito en C# es auto contenido, lo que significa que no necesita de ficheros adicionales tales como ficheros de cabecera. El tamaño de los tipos de datos básicos es fijo e independiente del compilador.
- **Orientación a componentes:** La propia sintaxis de C# incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular mediante construcciones más o menos complejas. Es decir, la sintaxis de C# permite definir plácidamente propiedades (similares a campos de acceso controlado), eventos (asociación controlada de funciones de respuesta a notificaciones) o atributos (información sobre un tipo o sus miembros).
- **Eficiente:** En C# todo el código incluye numerosas restricciones para asegurar su seguridad y no permite el uso de punteros. Sin embargo, y a diferencia de *Java*, en C# es posible saltarse dichas restricciones manipulando objetos a través de punteros. Para ello basta marcar regiones de código como inseguras y podrán usarse en ellas punteros de forma similar a como se hace en C++, lo que puede resultar vital para situaciones donde se necesite una eficiencia y velocidad de procesamiento muy grande.

ASP.NET

ASP.NET es un modelo de desarrollo *web* unificado que incluye los servicios necesarios para crear aplicaciones *web* empresariales con el código mínimo. Forma parte de .NET *Framework* y al codificar las aplicaciones ASP.NET tiene acceso a las clases en .NET *Framework*. El código de las aplicaciones puede escribirse en cualquier lenguaje compatible con el *Common Language Runtime* (CLR), entre ellos *Microsoft Visual Basic* y C#. Estos lenguajes permiten desarrollar aplicaciones ASP.NET que se benefician del *Common Language Runtime* (CLR), seguridad de tipos, herencia, entre otros. (14)

ASP.NET ofrece varias ventajas como son:

- Mejor rendimiento
- Compatibilidad con herramientas de primer nivel
- Eficacia y flexibilidad
- Simplicidad
- Facilidad de uso

- Escalabilidad y disponibilidad
- Posibilidad de personalización y extensibilidad
- Seguridad

Lenguaje de Modelado

Business Process Modeling Notation o BPMN (en español Notación para el Modelado de Procesos de Negocio) es una notación gráfica que describe la lógica de los pasos de un proceso de negocio. Ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades. (15)

La utilización de BPMN permite la realización de diagramas muy simples con un conjunto pequeño de elementos gráficos, donde la combinación de ellos da la oportunidad de realizar un Diagrama de Proceso de Negocio. Debe notarse que uno de los objetivos del desarrollo de BPMN es brindar un mecanismo simple para crear modelos de procesos y al mismo tiempo que sea posible gestionar la complejidad inherente que puedan presentar. El método elegido para manejar estos dos conflictivos elementos fue organizar los aspectos gráficos de la notación en categorías específicas. Dentro de las categorías básicas de elementos, se puede añadir información y variaciones adicionales para dar soporte a los requerimientos complejos sin cambiar dramáticamente el flujo básico del diagrama.

Para definir el lenguaje de modelado a utilizar se realizó un análisis entre las características de UML (*Unified Modeling Language* cuyo significado en español es Lenguaje Unificado de Modelado) y BPMN, evaluando la medida en que podía satisfacer las necesidades del sistema, con lo cual se llegó a las siguientes conclusiones:

Desventajas de UML:

- UML no ha sido diseñado para modelar procesos de negocio, por lo que no está orientado a lo que necesita el experto en el dominio del negocio.
- Predispone un enfoque orientado a objetos lo que puede contradecir un enfoque orientado al negocio.
- UML suele estar más orientado a los arquitectos de sistemas y diseñadores de *software*: Está pensado para un público eminentemente técnico. (16)

Ventajas de BPMN:

- Considera un único diagrama para la representación de los procesos.
- Pensado para ser asignado con naturalidad a los lenguajes de ejecución.
- El estilo es similar a un diagrama de flujo que hace que sea amigable y familiar para los usuarios del negocio.

- Es independiente de cualquier metodología de modelado de procesos.

BPMN ha llegado para dar solución a una de las necesidades más importantes para las empresas en la actualidad, se trata del modelado de procesos de negocio que tiene un gran impacto en la competitividad y desarrollo de la misma.

Metodología de desarrollo

La metodología a utilizar para el desarrollo del sistema es FDD (*Feature Driven Development*), la cual es ágil y promueve un desarrollo basado en funcionalidades, entre sus características más importantes se encuentran: (17)

- Está pensada para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Se basa en un proceso iterativo con ciclos cortos que producen un *software* funcional que el cliente y la dirección de la empresa pueden ver y monitorizar.
- Se preocupa por la calidad, por lo que incluye un monitoreo constante del proyecto.
- Se obtienen resultados periódicos y tangibles.
- El proceso consta de cinco pasos secuenciales durante los cuales se diseña y se construye el sistema:
 - ✓ Desarrollo de un modelo global
 - ✓ Construcción de una lista de funcionalidades
 - ✓ Planeación por funcionalidad
 - ✓ Diseño por funcionalidad
 - ✓ Construcción por funcionalidad
- Se concentra en las fases de diseño y construcción.

FDD define una serie de roles claves, de soporte y adicionales que intervienen en el proceso de elaboración del sistema, cuyas funciones se mencionan a continuación:

Roles claves:

- **Project Manager / Director del Proyecto:** Líder administrativo y financiero del proyecto, protege al equipo de situaciones externas.
- **Chief Architect / Arquitecto jefe:** Diseño global del sistema, ejecución de todas las etapas.
- **Development Manager / Director de desarrollo:** Lleva diariamente las actividades de desarrollo, resuelve conflictos en el equipo y problemas referentes a recursos.
- **Chief Programmer / Programador jefe:** Analiza los requerimientos, diseña el proyecto y selecciona las funcionalidades a desarrollar de la última fase del FDD.

- **Class Owner / Propietario de clases:** Responsable del desarrollo de las clases que se le asignaron como propias. Participa en la decisión de que clase será incluida en la lista de funcionalidades de la próxima iteración.
- **Expertos de dominio:** Puede ser un usuario, un cliente, analista o una mezcla de estos. Poseen el conocimiento de los requerimientos del sistema, pasa el conocimiento a los desarrolladores para que se asegure la entrega de un sistema completo.

Roles de soporte:

- **Domain Manager / Director de dominio:** Lidera al grupo de expertos del dominio, resuelve sus diferencias de opinión concernientes a los requerimientos del sistema.
- **Release Manager / Director de liberaciones:** Controla el avance del proceso mediante la revisión de los reportes del programador jefe. Reporta resultados obtenidos semanalmente al gerente y al cliente donde incluye el porcentaje de avance de cada funcionalidad.
- **System Administrator / Administrador del sistema:** Configura, administra y repara los servidores, estaciones de trabajo y equipos de desarrollo y testeo utilizados por el equipo.

Roles adicionales:

- **Tester / Probador:** Verifica que el sistema recién creado cumpla con los requerimientos del cliente, puede llegar a ser una persona independiente del equipo del proyecto.
- **Technical Writer / Escritores de documentos técnicos:** Prepara la documentación para los usuarios, que pueden formar parte o no del equipo del proyecto.

En cada una de las fases de FDD, los diferentes roles realizan las funciones según les corresponde. Las fases son: (18)

1. **Desarrollo de un modelo global:** Los expertos del dominio presentan un tutorial inicial de alto nivel sobre el alcance del sistema y su contexto. A continuación, con la participación de los desarrolladores, construyen el esqueleto de un primer modelo del sistema bajo la tutela del arquitecto jefe. Luego, el dominio es dividido en distintas áreas y a cada subgrupo se le asigna un área de dominio a desarrollar. Una vez finalizada cada área, el grupo se reúne para realizar un modelo global en base a todas las alternativas.
2. **Construcción de una lista de funcionalidades:** El equipo identifica las funcionalidades, las agrupa y las prioriza. En iteraciones subsecuentes del proceso, el equipo se divide en subgrupos que se especializan en áreas relacionadas a las funcionalidades.

3. **Planeación por funcionalidad:** En base a la lista de funcionalidades de la etapa anterior, el director del proyecto, el director de desarrollo y el programador jefe establecen hitos, además, crean un cronograma de diseño y construcción.
4. **Diseño por funcionalidad:** El programador jefe toma la próxima funcionalidad a ser desarrollada, identifica las clases involucradas y contacta al propietario correspondiente, se hace una descripción de la clase y sus métodos.
5. **Construcción por funcionalidad:** En esta etapa cada propietario de clases construye los métodos para las funcionalidades y realiza la prueba de unidad a cada una de las clases. A su vez, se inspecciona el código y el propietario chequea el repositorio. Seguidamente se realiza una construcción principal en la cual se integra la funcionalidad implementada.

FDD, permite dejar satisfechos a los desarrolladores, gerentes y clientes sin afectar el proyecto, esto se debe a su buen manejo de las actividades, la disminución del riesgo del proyecto y el aseguramiento de la calidad que ofrece, respectivamente. (19)

IDE Microsoft Visual Studio .NET 2010

Visual Studio 2010 Professional 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#, y *Visual Basic .NET*, al igual que entornos de desarrollo *web* como ASP.NET. Aunque actualmente se han desarrollado las extensiones necesarias para muchos otros. *Visual Studio* permite a los desarrolladores crear aplicaciones, sitios y aplicaciones *web*, así como servicios *web* en cualquier entorno que soporte la plataforma .NET. Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas *web* y dispositivos móviles. *Visual Studio 2010 Professional* es la herramienta esencial para las personas que realizan tareas de desarrollo básico. Simplifica la compilación, la depuración y el despliegue de las aplicaciones en una variedad de plataformas incluyendo *SharePoint*. También viene con el soporte integrado para el desarrollo con pruebas y con las herramientas de depuración que ayudan a garantizar unas soluciones de alta calidad. (20)

AxureRP Pro 5.5

AxureRP es una aplicación ideal para crear prototipos y especificaciones muy precisas para páginas *web*. Es una herramienta especializada en la tarea, cuenta con todo lo que se puede necesitar para crear los prototipos de forma más eficiente. *AxureRP* permite componer la página *web* visualmente, añadiendo, quitando y modificando los elementos con suma facilidad demostrando su grado de especialización en las anotaciones. En este punto, permite especificar el estado de cada elemento (Propuesto, Aceptado, Incorporado), el beneficio esperado (Crítico, Importante, Útil), el riesgo, la estabilidad, a quién va dirigido y

a quién se le asignará la tarea. Es una herramienta imprescindible si el trabajo consiste en hacer prototipos. (21)

Visual Paradigm 8.0

Es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de *software*: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El *software* de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE⁵, proporciona además abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML. (22)

PostgreSQL 9.1

PostgreSQL es un sistema de gestión de bases de datos con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado. *PostgreSQL* utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. (23)

A continuación se enumeran las principales características de este gestor de bases de datos:

- Implementación del estándar SQL92/SQL99.
- Soporta distintos tipos de datos: Además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP) y cadenas de bits. También permite la creación de tipos propios.
- Incorpora una estructura de datos *array*.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas y orientadas a operaciones con redes.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

Ventajas

⁵ Computer-Aided Software Engineering

- **Diseñado para ambientes de alto volumen:** *PostgreSQL* usa una estrategia de almacenamiento de filas para conseguir una mejor respuesta en ambientes de grandes volúmenes. Los principales proveedores de sistemas de bases de datos comerciales usan también esta tecnología, por las mismas razones.
- **Herramientas gráficas de diseño y administración de BD:** Existen varias herramientas gráficas de alta calidad para administrar las bases de datos (pgAdmin, pgAccess) y para hacer diseño de bases de datos (Tora, Data Architect).
- **Instalación Ilimitada:** Con *PostgreSQL*, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del *software*.
- **Ahorros considerables en costos de operación:** *PostgreSQL* ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos, conservando todas las características, estabilidad y rendimiento.
- **Estabilidad y Confiabilidad Legendarias:** Es extremadamente común que compañías reporten que *PostgreSQL* nunca ha presentado caídas en varios años de operación de alta actividad.

1.6 Conclusiones

A lo largo del capítulo que aquí concluye se han abordado conceptos asociados a la gestión de citas, permitiendo una mejor comprensión del objeto de estudio. La fundamentación de las técnicas, herramientas y características de los sistemas de gestión de citas investigados demostró que existen soluciones similares a la requerida por el departamento de Identificación, las cuales son muy costosas y no se pueden utilizar pues están desarrolladas única y exclusivamente para instalaciones específicas, sin embargo poseen funcionalidades como: Gestionar Cita, Solicitar Cita, entre otras, que sirven de base para el sistema a desarrollar.

El análisis de las diferentes tecnologías, metodologías y herramientas según las necesidades de la solución determinó la utilización de la metodología FDD Ágil, como lenguaje de modelado BPMN, *Visual Paradigm* 8.0 como herramienta de modelado y *PostgreSQL* 9.1 como SGBD. Como tecnología para el mapeo de objetos *NHibernate* 3.1.0 y el lenguaje de programación C# usando el IDE de desarrollo Visual Studio 2010.

Capítulo 2: Características del sistema

2.1 Introducción

Cuando se define el modelo de procesos del negocio se describe cada una de las entidades que intervienen en el proceso de gestión de citas en aras de lograr un mayor entendimiento del mismo. A partir de este modelo se logran delimitar las funcionalidades que se automatizarán, las mismas se agrupan y clasifican con el objetivo de lograr un cronograma eficiente que permita diseñar a través de iteraciones las funcionalidades del sistema. Al presentarse la descripción de los requisitos funcionales y no funcionales que el sistema debe cumplir se obtiene una concepción general del sistema a implementar.

2.2 Propuesta del sistema

De acuerdo con las investigaciones realizadas y luego de materializar un profundo análisis del objeto de estudio, se ha concebido implementar un Sistema de Gestión de Citas para la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas con el objetivo de organizar el flujo de personas que necesitará el servicio de dicha entidad. Con el fin de cumplir los objetivos planteados, el sistema debe poseer dos niveles de acceso: el administrador del sistema que sería el usuario encargado de gestionar toda la información referente a los módulos de Administración, Configuración y Reportes y el cliente que gestiona la información concerniente al módulo Gestionar Citas. El sistema es una solución Cliente-Servidor que debe entrar en función desde que la persona solicita el servicio por cualquiera de las vías (dirigirse a la oficina, vía correo electrónico, vía telefónica, acceder al portal). Es imprescindible además que no solamente los solicitantes del servicio como principales beneficiarios de este sistema, noten una mejoría importante en los procesos del negocio, sino que además para los trabajadores y los demás usuarios del *software*, sea una ventaja su instalación y puesta en práctica. La propuesta de solución proporcionará ventajas como: eliminará la pérdida de tiempo para las personas en espera del documento, brindará un mejor aprovechamiento para tramitar documentos y posibilitará satisfacer al cliente. Para lograr esto, los requisitos funcionales se han capturado partiendo de la necesidad real de la Universidad.

2.3 Arquitectura propuesta

La arquitectura de *software* es el diseño de más alto nivel de un sistema describiendo cada uno de los componentes que lo integran así como la comunicación entre los mismos. Se establecen los fundamentos para que analistas, diseñadores, programadores trabajen en una línea común que permita alcanzar los objetivos del sistema. El sistema se encuentra desarrollado sobre la plataforma .Net, en su versión 4.0, es una solución Cliente-Servidor. El servidor de aplicaciones *Internet Information Server 7.0* aloja la

aplicación haciendo uso de diferentes tecnologías como: *NHibernate* para mapear los objetos desde una aplicación .Net, entre otros.

El sistema en su vista más abstracta es una solución Cliente-Servidor que define la relación entre dos aplicaciones en las cuales una de ellas (Cliente) envía peticiones a la otra (Servidor) y este último le envía las respuestas. Los principales componentes del sistema serian: un conjunto de servidores locales que brindan servicios a otras aplicaciones, un conjunto de clientes que realizan peticiones a los servidores y una red que permite la conexión entre los servidores y clientes.

El estilo arquitectónico en capas se basa en una distribución jerárquica de los roles y las responsabilidades para proporcionar una división efectiva de los problemas a resolver. Este estilo permite asignar correctamente las funcionalidades a cada capa, pudiéndose reutilizar las capas inferiores que no tengan dependencias con las superiores, proveyendo un desacople entre las capas ya que la comunicación entre las mismas está basada en la abstracción, evitando que los cambios en una de ellas afecte directamente al resto.

El sistema es una solución Cliente-Servidor, compuesta por capas bien definidas y diseñadas con el objetivo de delegar responsabilidades como se muestra en el Anexo 5.

El sistema se encuentra lógicamente dividido en 4 capas definiendo claramente las responsabilidades de cada una, permitiendo reducir el acoplamiento y aumentar la reutilización de las mismas. Esta vista permite la realización de cambios en las capas sin realizar grandes cambios en las demás. La comunicación entre las capas se realiza a nivel de interfaces permitiendo trabajar de manera transparente a las instancias reales. Cada capa está integrada por un conjunto de componentes que encapsulan la mayor parte del comportamiento y escenarios en tiempo de desarrollo. Esto apoyado por especialistas en cada uno de estos componentes le da al desarrollador las herramientas y medios necesarios para la implementación del proceso correspondiente, agilizando el tiempo de desarrollo y disminuyendo las variables de aprendizaje.

Capa de Presentación: Es la capa donde el sistema interactúa con el usuario, haciendo uso de varias tecnologías para la validación de los datos de entrada así como el uso de componentes. En esta capa se encuentran todas las interfaces que serán mostradas a los usuarios, donde para su correcto funcionamiento se hace uso de código *JavaScript*, integrando los componentes de *JQuery*, así como archivos CSS que contienen los estilos de la aplicación.

Capa de Negocio: Es la capa que encapsula la lógica del sistema, compuesta por clases que manejan las entidades y que valiéndose de la capa de acceso a datos da respuesta a las funcionalidades requeridas para mostrarle correctamente las interfaces al usuario a través la capa de presentación.

Capa de Acceso a Datos: Es el componente que da soporte a las funcionalidades de la capa de negocio que se encuentran relacionadas con la fuente de datos. En esta capa se encuentra el ORM (*NHibernate*), herramienta que tiene como objetivo principal mapear los objetos desde una aplicación .Net a una base de datos relacional. La principal función de esta capa es realizar una implementación de las funcionalidades definidas en las interfaces de la capa de negocio y al mismo tiempo trabajar directamente con la fuente de datos.

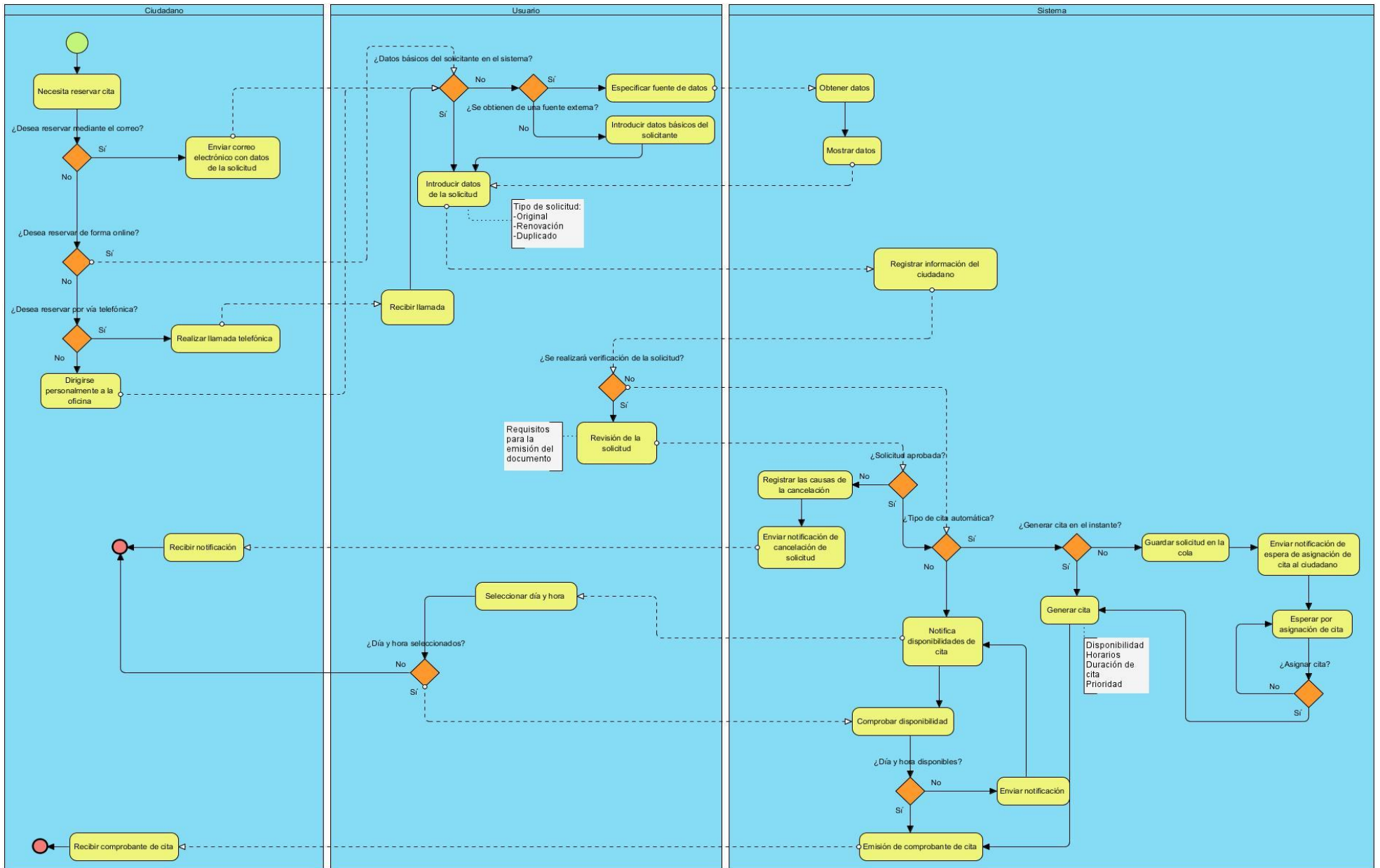
Capa de Entidades: Contiene las clases entidades del subsistema que se gestionan en la aplicación, persisten en la base de datos y se muestran en la presentación.

Base de datos: Está constituida por todo el conjunto de tablas y procedimientos que permiten el almacenamiento de la información recolectada y procesada por los procesos. Se encuentra dividida en varias vistas que separan la gestión de los procesos de los datos de negocio y la información de soporte.

2.4 Modelo de Procesos de Negocio

Luego de verificar la no existencia de un proceso de gestión de citas en la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas, se realiza una propuesta del flujo de actividades, a través de la modelación del proceso de gestión de citas para lograr un completo entendimiento del negocio. La Figura 1 muestra el diagrama general del proceso de gestión de citas.

Figura 1: Modelo de Procesos de Negocio



2.4.1 Descripción de las actividades del sistema

Para lograr un entendimiento claro y detallado del flujo de acciones que se llevan a cabo para gestionar las citas, se realiza la descripción de cada una de las actividades que conforman el proceso. La Tabla 6 muestra el flujo de actividades que se lleva a cabo.

Tabla 6: Descripción de las actividades de los procesos de gestión de citas

Actividad	Descripción	Responsable	Entrada	Salida
Necesita reservar cita.	El ciudadano necesita reservar una cita.	ciudadano	Necesidad de obtener documento de identificación.	El ciudadano utiliza una de las vías para realizar la reservación.
Dirigirse personalmente a la oficina.	El ciudadano se dirige a la oficina donde será atendido por un usuario autorizado en el sistema a registrar una solicitud.	ciudadano	Necesidad de obtener documento de identificación.	Se dirige personalmente a la oficina.
Introducir datos de la solicitud.	Una vez que el usuario autorizado a registrar una solicitud en el sistema tiene los datos necesarios, procede a introducirlos en la aplicación.	usuario	Ciudadano en contacto con el usuario.	Datos de la solicitud introducidos en el sistema.
Registrar información del ciudadano.	El sistema registra los datos de la solicitud introducidos por el usuario, realizando las validaciones necesarias en cada campo a introducir.	sistema	Datos de la solicitud introducidos en el sistema.	Datos de la solicitud registrados en el sistema.
Revisión de la solicitud.	El usuario responsable por las solicitudes verifica que según los datos de la solicitud, el ciudadano cumple con los requisitos necesarios de la institución para la emisión del documento, verificando además que todos los datos estén correctos.	usuario	Datos de la solicitud registrados en el sistema.	Solicitud revisada.
Notifica disponibilidad de cita.	En el caso de que la solicitud sea aprobada el sistema notifica las disponibilidades existentes para reservar citas.	sistema	Solicitud revisada.	Disponibilidades de días y horas.
Seleccionar día y hora.	El usuario selecciona el día y la hora de las disponibilidades mostradas.	usuario	Disponibilidad de horas y días.	Día y hora seleccionado por el usuario.
Comprobar disponibilidad.	El sistema verifica nuevamente que esté disponible la fecha y hora que el ciudadano seleccionó, para evitar que otra persona en el mismo instante de tiempo haya reservado para el mismo día y hora.	sistema	Día y hora seleccionado por el usuario.	Disponibilidad comprobada.
Emisión del comprobante de	El sistema emite al ciudadano un comprobante de cita donde queda	sistema	Disponibilidad comprobada.	Comprobante de cita emitido por el sistema.

cita.	reflejado el día y la hora en que debe presentarse en la oficina.			
Recibe comprobante de cita.	El ciudadano recibe el comprobante emitido por el sistema.	ciudadano	Comprobante de cita emitido por el sistema.	Comprobante de cita recibido por el ciudadano.
Realizar llamada telefónica.	El ciudadano realiza una llamada telefónica a una persona que es usuario del sistema.	ciudadano	Se dirige hacia un teléfono para realizar la llamada telefónica.	Llamada realizada.
Recibir llamada.	El usuario recibe la llamada realizada por el ciudadano.	usuario	Llamada telefónica realizada por el ciudadano.	Ciudadano en contacto con el usuario.
Enviar correo electrónico con datos de la solicitud.	El ciudadano envía un correo electrónico con sus datos.	ciudadano	Necesidad de obtener el documento de identificación.	Correo electrónico enviado con los datos de la solicitud.
Introducir datos del solicitante.	El usuario introduce los datos básicos del ciudadano, en caso de no tenerlos en el sistema se registran los mismos.	usuario	Ciudadano en contacto con el usuario.	Datos básicos del ciudadano introducidos en el sistema.
Generar cita.	El sistema genera la cita automáticamente en el instante en que es registrada y aprobada en caso de que se realice revisión de la solicitud.	sistema	Solicitud registrada en el sistema.	Cita generada por el sistema.
Guardar solicitud en la cola.	El sistema al no generar la cita en el instante guarda la solicitud en una cola.	sistema	Solicitud registrada en el sistema.	Solicitud guardada en la cola.
Esperar por asignación de cita.	El sistema asigna la cita según estime conveniente.	sistema	Solicitud guardada en la cola.	El sistema en espera de que se ejecute la acción que le indica que puede comenzar a generar las citas que están en la cola.
Enviar notificación.	El sistema envía una notificación al usuario indicando que la selección del día y la hora ya no está disponible.	sistema	Disponibilidad comprobada.	Notificación enviada.

2.5 Especificación de los requisitos de *software*

Lo más importante a la hora de crear un producto de *software* es obtener los requisitos o el análisis de los requisitos. Los clientes suelen tener una idea más bien abstracta del resultado final, pero no sobre las funciones que debería cumplir el *software*. Especificar los requisitos ayuda a los ingenieros a entender mejor el problema en cuya solución trabajarán. Los requisitos se pueden agrupar en: requisitos funcionales y requisitos no funcionales.

2.5.1 Requisitos funcionales

Una vez realizado el análisis del modelo de procesos de negocio con BPMN del Sistema de Gestión de Citas para la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas, fueron definidas las actividades que conforman el objeto de automatización, o sea, la base para definir los requisitos funcionales del sistema.

RF1 Registrar datos básicos del solicitante.

RF2 Registrar datos de la solicitud.

RF3 Mostrar solicitud.

RF4 Verificar solicitud.

RF5 Registrar cancelación de una solicitud.

5.1 Modificar el estado de la solicitud a “Resuelta”.

RF6 Mostrar disponibilidades de cita.

RF7 Permitir seleccionar día y hora.

RF8 Comprobar disponibilidad.

RF9 Guardar solicitud en la cola.

RF10 Generar cita.

RF11 Emitir comprobante de cita.

RF12 Cancelar cita.

RF13 Registrar información para generar cita automática.

- Tipos de documentos
- Tipo de cita
- Días no laborables
- Duración de la cita

RF14 Generar reportes.

- Mostrar reportes por día

2.5.2 Requisitos no funcionales

- **Ambiente**

Estaciones Clientes

Hardware: PC Pentium 4 a 1 GHz o superior, mínimo 512 MB de RAM.

Software: Cualquiera de los siguientes navegadores *web:* Mozilla Firefox v17.0 o superior, Internet Explorer v8 o superior, Google Chrome v20.0 o superior.

Otras: Local climatizado de ser posible.

Estaciones de Servidores

Hardware: PC Pentium 4 a 2 GHz o superior, mínimo 2 GB de RAM, 250 GB o superior de disco duro.

Software: Sistema Operativo *Windows Server 2008 R2* con SP2, antivirus actualizado, *Microsoft .Net framework v4.0*, *IIS 7.5* y *PostgreSQL 9.1*.

Otras: Climatizado obligatoriamente.

- **Usabilidad**

RnF1. El sistema podrá ser utilizado por cualquier usuario con las siguientes características:

- a) Conocimientos básicos relativos al uso de una computadora.
- b) Conocimientos básicos del sistema operativo *Windows*.
- c) Conocimientos sólidos relativos a los procesos de negocio acorde al rol que desempeñe.

RnF2. El sistema será distribuido en idioma español.

RnF3. Los términos utilizados se establecerán acorde al negocio correspondiente para facilitar la comprensión de la herramienta de trabajo.

RnF4. El sistema poseerá estructura y diseño homogéneos en todas sus pantallas, que facilite la navegación.

- a) Menús laterales y desplegados que permitan el acceso rápido a la información.

- **Fiabilidad**

RnF5. El sistema debe estar disponible las 24 horas del día los 7 días de la semana realizándose copias de seguridad semanalmente tanto de la aplicación como de la base de datos.

RnF6. El tiempo medio entre fallos no debe ser superior a 1 hora.

RnF7. El sistema estará 1 hora fuera de operación luego de haber fallado 3 veces.

- **Eficiencia**

RnF8. El sistema debe ser capaz de soportar una cantidad escalable de peticiones y dar respuestas efectivas y rápidas.

RnF9. El sistema debe hacer un uso eficiente de los recursos de *hardware* donde se encuentre desplegado.

- **Restricciones de diseño**

RnF10. Plataforma de desarrollo *.NET 4.0* utilizando *Visual Studio Ultimate 2010*.

RnF11. Biblioteca *jQuery* 1.7.2.

RnF12. El subsistema de identificación y gestión de citas utilizará ASP.NET *Web Forms* 4.0. Para el acceso a datos se utilizará el ORM *NHibernate* 3.1.0.

- **Interfaz**

RnF13. El sistema dispondrá de un diseño ameno e intuitivo acorde las características de la institución.

2.6 Descripción de las funcionalidades

La metodología FDD plantea que los requisitos funcionales del sistema se deben dividir en subconjuntos, agrupándolos según la afinidad y dependencia que exista entre ellos. Luego se describen estos requisitos detalladamente con el objetivo de definir paso a paso las acciones que deben seguir los desarrolladores, para garantizar una mayor coherencia de la aplicación con las expectativas del cliente.

La Tabla 7 muestra la descripción detallada de la funcionalidad “gestionar citas”, el resto de las descripciones se pueden ver en el Anexo 6.

Tabla 7: Descripción de la funcionalidad “gestionar citas”

Precondiciones	El usuario debe estar autenticado en el sistema con el rol solicitante.
Funcionalidades tratadas	RF1, RF2, RF3, RF4, RF5, RF6, RF7, RF8, RF9, RF11.
Conceptos tratados	Cita, Solicitud, Trámite, Notificación.
Descripción básica	<ol style="list-style-type: none"> 1. El sistema muestra un menú lateral izquierdo con las siguientes opciones: <ul style="list-style-type: none"> • Autenticación • Reportes • Administración • Configuración • Gestión 2. Luego de autenticarse el usuario, el sistema muestra por defecto la interfaz principal dónde se encuentran información general del sistema. 3. Si el solicitante selecciona la opción “Gestión”, el sistema despliega una lista con las siguientes pestañas: <ul style="list-style-type: none"> • Gestión de citas • Insertar cita <p>3.1 Si el usuario selecciona la opción “Insertar cita”, el sistema muestra una interfaz donde se encuentran los siguientes campos para ser llenados:</p> <ul style="list-style-type: none"> • Cita personal (El sistema carga los datos del usuario autenticado) • Cita a terceros (El sistema muestra una opción para buscar a la persona a quien se

	<p>le hace la solicitud)</p> <p>3.1.1 Si el usuario selecciona “Cita a terceros”, el sistema brinda la posibilidad de buscar a la persona por el carnet de identidad, al dar click en el botón: Aceptar el sistema llena todos los datos pertenecientes a ese solicitante.</p> <p><u>Datos Personales</u></p> <ul style="list-style-type: none">• Nombre• Apellidos• Identificador <p><u>Datos de la solicitud:</u></p> <ul style="list-style-type: none">• Tipo de solicitud (Original, Renovación, Duplicado)• Tipo de documento• Clasificación• Almanaque para seleccionar la fecha• Hora <p>3.1.2 El usuario selecciona una de las dos formas de generar la fecha:</p> <ul style="list-style-type: none">• Seleccionar fecha• Generar fecha <p>3.1.3 Si el usuario escoge “Seleccionar fecha”, el sistema muestra los campos para la selección de fecha y hora, notificando su disponibilidad.</p> <p>3.1.4 Si el usuario selecciona “Generar fecha”, el sistema le asigna una fecha según la configuración de la cita.</p> <p>3.1.5 El usuario da click en el botón: Aceptar.</p> <p>3.1.6 El sistema verifica nuevamente la disponibilidad de fecha y hora que seleccionó el usuario.</p> <p>3.1.7 En caso de existir aun la disponibilidad, el sistema registra la cita. (En caso contrario ver “Descripción alterna: Cita no disponible”)</p> <p>3.1.8 Una vez insertada la cita, el sistema verifica la solicitud para su aprobación o cancelación, en caso de que sea aprobada o cancelada el sistema lo notifica.</p> <p>3.1.7 En caso de ser aprobada, el sistema muestra la interfaz “Comprobante de Cita” con todos los datos de la cita asignada como son:</p> <ul style="list-style-type: none">• Nombre• Solapín• Fecha• Hora• Documento• Tipo de solicitud• Número de cita
--	---

- 3.1.8 El usuario tiene la opción de imprimir el comprobante de cita.
- 3.2 Si el solicitante selecciona la opción “Gestión de citas”, el sistema muestra una interfaz con un listado de las citas realizadas por el usuario autenticado y las siguientes opciones:
- Modificar cita
 - Ver detalles
 - Cancelar cita
- 3.2.1 Si el usuario selecciona la opción “Modificar cita”, el sistema muestra la fecha y la hora de la cita seleccionada de forma editables, para que el usuario los pueda actualizar.
- 3.2.2 Si el usuario selecciona la opción “Cancelar cita”, el sistema cancela la cita reservada y deja el día y hora correspondientes como disponibles.
- 3.2.3 Si el usuario selecciona la opción “Ver detalles”, el sistema muestra el comprobante correspondiente a esa cita.

Prototipos

The screenshot displays the 'SISTEMA DE GESTIÓN DE CITAS' web application. The header includes the system name and a user greeting: 'Bienvenido: Nilberto Rodríguez Rojas'. A navigation menu on the left lists: Autenticación, Administración, Reportes, Configuración, and Gestión. The main content area is titled 'Insertar Citas' and contains the following form fields:

- Tipo de Cita:** Radio buttons for 'Cita Personal' and 'Cita a Tercero'.
- Tipo de Documento:** A dropdown menu with 'CI' selected.
- Tipo de Solicitud:** A dropdown menu with 'original' selected.
- Nombre:** Text input field.
- Primer Apellido:** Text input field.
- Segundo Apellido:** Text input field.
- Carnet de Identidad:** Text input field.
- Fecha:** Radio buttons for 'Seleccionar Fecha' and 'Generar Fecha', followed by a date picker.
- Hora:** A time picker.

At the bottom of the form are two buttons: 'Aceptar' and 'Cancelar'.

SISTEMA DE GESTIÓN DE CITAS

GESTIÓN DE CITAS Bienvenido: Nilberto Rodriguez Rojas

- > Autenticación
- > Administración
- > Reportes
- > Configuración
- > Gestión

Comprobante de Cita

Comprobante de Cita		Cita Nro: 112
Nombre Y Apellidos: Nilberto Rodriguez Rojas		CI: 89041037508
Fecha: 01/01/0001	Hora: 0:00	Documento: solapín
		Tipo Solicitud: original

UCI, sábado, 18 de mayo de 2013

SISTEMA DE GESTIÓN DE CITAS

GESTIÓN DE CITAS Bienvenido: Nilberto Rodriguez Rojas

- > Autenticación
- > Administración
- > Reportes
- > Configuración
- > Gestión

Gestion de Citas

Listado de Citas

Nombre	Primer Apellido	Segundo Apellido	Fecha	Hora	Estado	Documento	Número			
Leonardo	Díaz	Novo	5/21/2013	11:30 AM	asignada	CI	96	🔍	✎	🗑️
Leonardo	Díaz	Novo	_//_	_:__	en espera	licencia	97	🔍	✎	🗑️
Nilberto	Rodriguez	Rojas	_//_	_:__	en espera	CI	86	🔍	✎	🗑️

1 Mostrando 1 a 4 de 4 registros

UCI, jueves, 16 de mayo de 2013

Descripción alterna: "Cita no disponible"

	3.1.5 En caso de no existir disponibilidad el sistema muestra un mensaje indicando “El día y hora seleccionados ya no están disponibles, por favor seleccione nuevamente su cita”.
Validaciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado en el sistema y no debe tener otra cita creada. • Los datos introducidos por el usuario deben estar correctos. • La fecha y hora seleccionada por el usuario deben estar disponibles.
Postcondiciones	Queda asignada la cita o denegada la solicitud por el Sistema de Gestión de Citas.

2.7 Clasificación de las funcionalidades

Luego de obtener las funcionalidades definidas y agrupadas, se continúa a clasificarlas en críticas y secundarias. Las críticas constituyen las funcionalidades de mayor impacto en el negocio por lo que son las primeras en ser diseñadas y construidas en cada iteración de desarrollo. Seguidamente se desarrollan las funcionalidades secundarias las cuales sirven al flujo principal del proceso. La Tabla 8 muestra la clasificación de las funcionalidades del Sistema de Gestión de Citas.

Tabla 8: Clasificación de las funcionalidades

Módulo	Funcionalidad	Clasificación
Sistema de Gestión de Citas	Registrar datos básicos del solicitante.	Crítica
	Registrar datos de la solicitud.	Crítica
	Mostrar solicitud.	Secundaria
	Verificar solicitud.	Crítica
	Registrar cancelación de una solicitud.	Crítica
	Mostrar disponibilidades de cita.	Crítica
	Permitir seleccionar día y hora.	Crítica
	Comprobar disponibilidad.	Crítica
	Guardar solicitud en la cola.	Secundaria
	Generar cita.	Secundaria
	Emitir comprobante de cita.	Crítica
	Cancelar cita.	Crítica
	Registrar información para generar cita automática.	Secundaria
	Generar reportes.	Secundaria

2.8 Planeación de las iteraciones

Una vez definida la lista de funcionalidades de la etapa anterior, se crea un cronograma de diseño y construcción. La Tabla 9 muestra el cronograma para el desarrollo de las funcionalidades del proceso de gestión de citas.

Tabla 9: Cronograma de diseño y construcción

Módulo	Funcionalidad	Fecha de ejecución	
		Diseño de la funcionalidad	Construcción de la funcionalidad
Sistema de Gestión de Citas	Registrar datos básicos del solicitante.	09/01/13 - 12/01/13	13/01/13 - 16/01/13
	Registrar datos de la solicitud.	09/01/13 - 12/01/13	13/01/13 - 16/01/13
	Mostrar solicitud.	09/01/13 - 12/01/13	13/01/13 - 16/01/13
	Verificar solicitud.	09/01/13 - 12/01/13	13/01/13 - 16/01/13
	Registrar cancelación de una solicitud.	20/01/13 - 2/02/13	3/02/13 - 13/02/13
	Mostrar disponibilidades de cita.	20/01/13 - 2/02/13	3/02/13 - 13/02/13
	Permitir seleccionar día y hora.	20/01/13 - 2/02/13	3/02/13 - 13/02/13
	Comprobar disponibilidad.	15/02/13 - 24/02/13	25/02/13 - 05/03/13
	Guardar solicitud en la cola.	15/02/13 - 24/02/13	25/02/13 - 05/03/13
	Generar cita.	15/02/13 - 24/02/13	25/02/13 - 05/03/13
	Emitir comprobante de cita.	10/03/13 - 18/03/13	27/03/13 - 30/04/13
	Cancelar cita.	10/03/13 - 18/03/13	27/03/13 - 30/04/13
	Registrar información para generar cita automática.	10/03/13 - 18/03/13	27/03/13 - 30/04/13
	Generar reportes.	10/03/13 - 18/03/13	27/03/13 - 30/04/13

2.9 Diseño de las funcionalidades

Los diagramas de clases son diagramas estáticos que describen la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos, son realizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro.

2.9.1 Diagrama de clases

La metodología FDD plantea que luego de realizar la planificación de las iteraciones se comienza, por orden, a diseñar cada una de las funcionalidades a ser desarrolladas, se identifican las clases involucradas y se describen, especificando los métodos que serán implementados en ellas. A continuación se presenta el diagrama de clases de la funcionalidad “gestionar citas” correspondiente al Sistema de Gestión de Citas (Figura 2), el resto de los diagramas de clases se pueden ver en el Anexo 7.

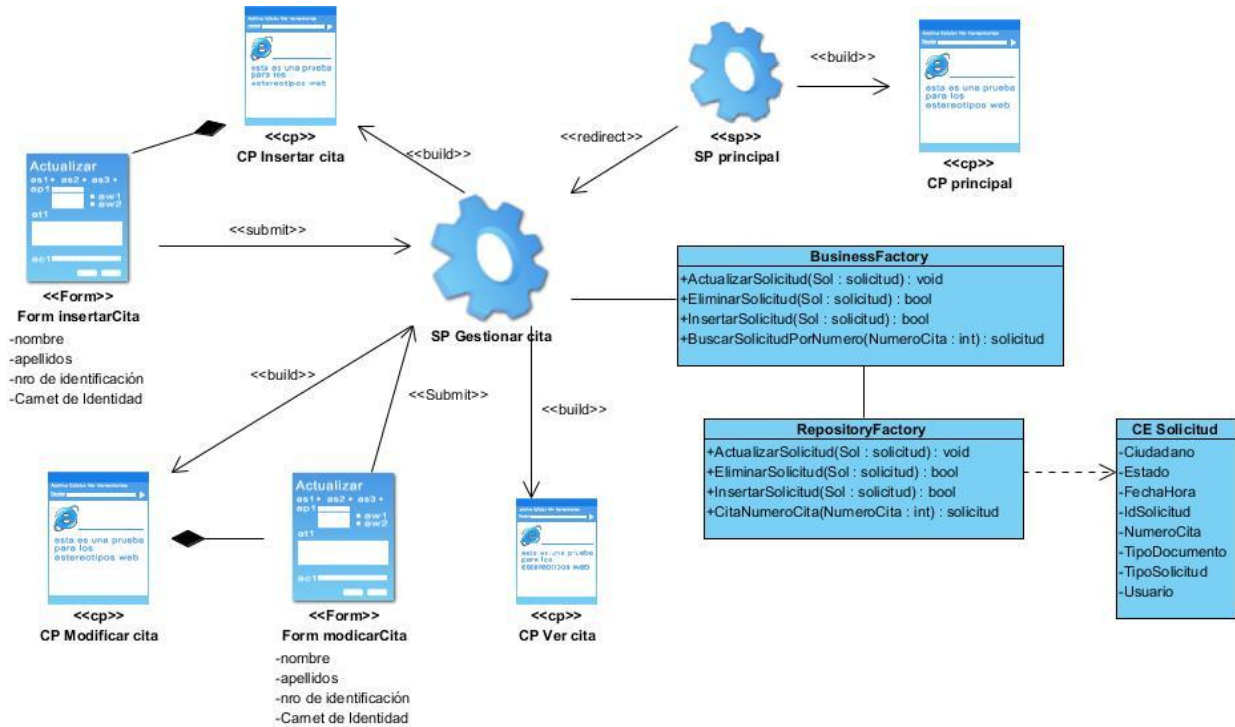


Figura 2: Diagrama de clases de la funcionalidad “gestionar citas”

Cada interfaz está asociada a una *server-page* con su *client-page* y para aquellas interfaces que poseen campos de captura de datos se le hace un formulario. Posee una clase Lógica del Negocio con sus pertinentes métodos, una clase Conexión y las clases Entidades referentes a la base de datos.

2.9.2 Descripción de las clases

A continuación se describen textualmente algunas de las clases más importantes del diseño.

Tabla 10: Descripción de la clase "BusinessFactory"

Nombre	<i>BusinessFactory</i>	
Descripción	Es la clase encargada de gestionar las operaciones de una solicitud.	
	Métodos	Descripción

ActualizarSolicitud(Sol:solicitud):void	Método encargado de actualizar una solicitud según la especificada.
EliminarSolicitud(Sol:solicitud):bool	Método encargado de eliminar una solicitud según la entrada por parámetros.
InsertarSolicitud(Sol:solicitud):bool	Método encargado de insertar la solicitud especificada.
BuscarSolicitudPorNumero(NumeroCita:int):solicitud	Método encargado de buscar una solicitud por un número de cita especificado.

Tabla 11: Descripción de la clase "RepositoryFactory"

Nombre	<i>RepositoryFactory</i>	
Descripción	Es la clase encargada de gestionar cada una de las operaciones para ejecutar las consultas en la base de datos.	
	Métodos	Descripción
	ActualizarSolicitud(Sol:solicitud):void	Método encargado de actualizar una solicitud determinada en la base de datos.
	EliminarSolicitud(Sol:solicitud):bool	Método encargado de eliminar una solicitud determinada en la base de datos.
	InsertarSolicitud(Sol:solicitud):bool	Método encargado de insertar una solicitud determinada en la base de datos.
	CitaNumeroCita(NumeroCita:int):solicitud	Método encargado de buscar una solicitud por un número de cita determinado en la base de datos.

2.9.3 Diagrama de secuencia

En un diagrama de secuencia se indican las clases que forman parte del programa y las llamadas que se hacen en cada una de ellas para realizar una tarea determinada. Se realizan diagramas de secuencia con el objetivo de definir acciones que se pueden realizar en la aplicación. (24)

La Figura 3 muestra el diagrama de secuencia de la funcionalidad “gestionar citas” correspondiente al Sistema de Gestión de Citas, el resto de los diagramas de secuencia se pueden ver en el Anexo 8.

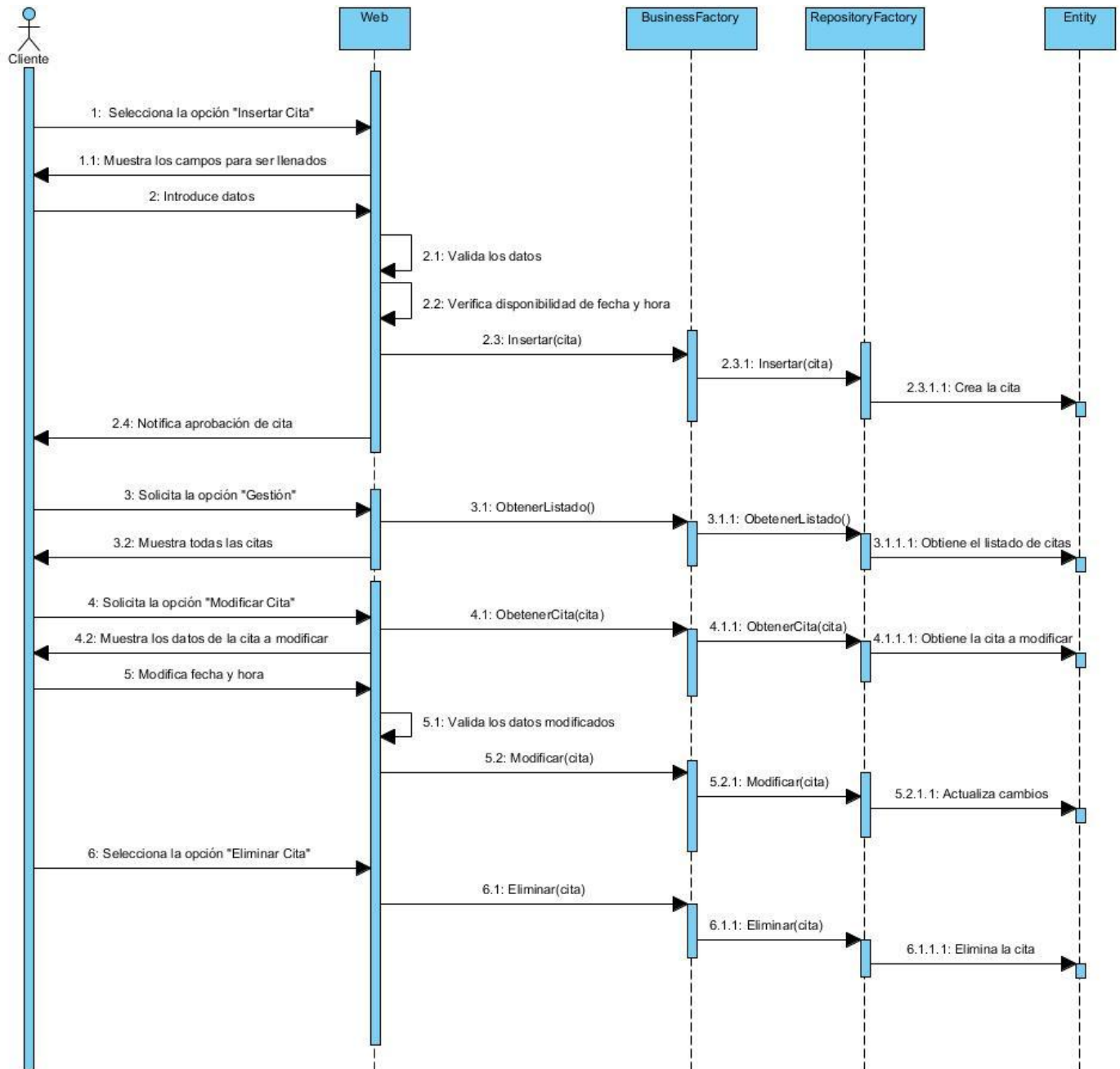


Figura 3: Diagrama de secuencia de la funcionalidad “gestionar citas”

2.10 Modelo de datos

El modelo de datos constituye la base final de la aplicación que se obtiene en el flujo de trabajo de análisis y diseño, en el cual se definen detalladamente todas las entidades persistentes, así como los atributos y relaciones entre ellas. El modelo contempla nomencladores cuyo objetivo es estandarizar el tratamiento sobre un determinado tipo de dato como es el tipo de solicitud, estado de cita y rol. Ver (Figura 4)

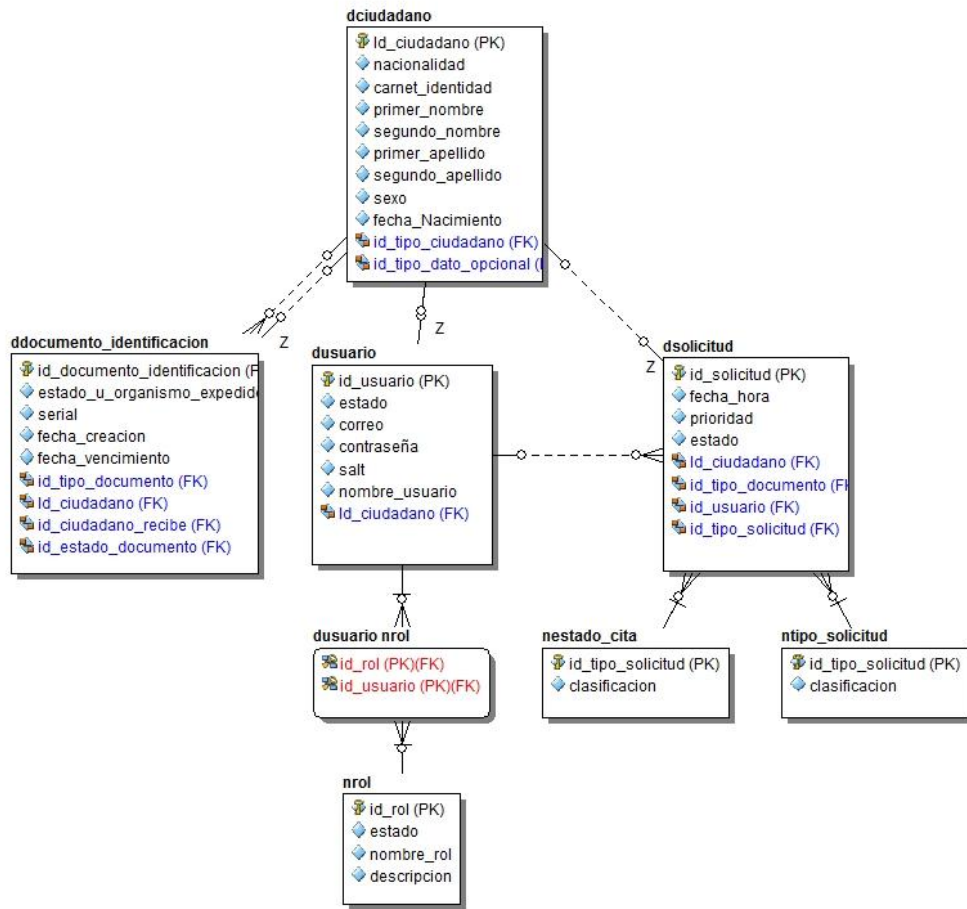


Figura 4: Modelo de datos

2.10.1 Descripción de las entidades

En la Tabla 12 se muestra la descripción de la entidad de mayor impacto en la aplicación, para realizar un estudio más detallado de las mismas ver Anexo 9.

dCiudadano: Entidad que guarda los datos de un ciudadano.

Tabla 12: Descripción de la entidad dCiudadano

Atributos	Tipo de Datos	Nulo	Definición
id_Ciudadano	<i>uuid</i>	No	Identificador del ciudadano
primer_Nombre	<i>varchar</i>	No	Primer nombre del ciudadano
segundo_Nombre	<i>varchar</i>	Si	Segundo nombre del ciudadano
primer_Apellido	<i>varchar</i>	No	Primer apellido del ciudadano

segundo_Apellido	<i>varchar</i>	No	Segundo apellido del ciudadano
lugar_Nacimiento	<i>varchar</i>	Si	Lugar de nacimiento del ciudadano
sexo	<i>char</i>	No	Sexo del ciudadano
fecha_Nacimiento	<i>timestamp</i>	No	Fecha de nacimiento del ciudadano
nacionalidad	<i>varchar</i>	No	Nacionalidad del ciudadano

2.11 Patrones

Los patrones brindan una estructura conocida para todos los programadores, de manera que no resulte distinta trabajar entre los mismos, permiten tener una estructura de código común, ahorran grandes cantidades de tiempo en la construcción de un *software*, son fáciles de comprender, mantener y extender, además de dar una mejor imagen de profesionalidad y calidad. Existen varios tipos de patrones entre los que se encuentran los patrones de creación y los patrones de asignación de responsabilidad. En el sistema desarrollado se utilizaron los siguientes patrones:

Patrones de creación:

Singleton: Garantiza que una clase sólo tenga una instancia, y proporciona un punto de acceso global a ella. (25)

Ejemplo:

```
public class BusinessFactory
{
    private static BusinessFactory instance = null;
    public static BusinessFactory Instance
    {
        get
        {
            if (BusinessFactory.instance == null)
                BusinessFactory.instance = new BusinessFactory();

            return BusinessFactory.instance;
        }
    }
}
```

Figura 5: Ejemplo del uso del patrón Singleton

Patrones de asignación de responsabilidad:

Creador: El patrón creador ayuda a identificar quién debe ser el responsable de la creación (en este caso, la clase *BusinessFactory*) de nuevos objetos o clases, la cual contiene la información necesaria para realizar la creación del objeto, además usar directamente las instancias creadas del objeto.

Ejemplo:

```
Usuario usuario = new Usuario(dUsuario.IdUsuario, dUsuario.NombreUsuario, dUsuario.Estado, dUsuario.Correo, CombertirdCiudadano_aCiudadano(dUsuario.Ciudadano));
```

Figura 6: Ejemplo del uso del patrón Creador

2.12 Conclusiones

A lo largo del presente capítulo se expuso el análisis y el diseño del Sistema de Gestión de Citas. Con la utilización de la arquitectura en capas se garantizó que al realizar cambios en una de las capas, las demás no se verían afectadas, pues el flujo de datos es en una dirección. Al realizar el modelo de procesos del negocio, permitió describir detalladamente los procesos que intervienen en el Sistema de Gestión de Citas, posibilitando definir las entradas y salidas de todas las actividades, así como los roles involucrados en cada instante del flujo, lográndose un mejor entendimiento.

Al definirse los requisitos que constituyen la base para la implementación del sistema, se realizaron los prototipos no funcionales de interfaz de usuario el cual visualiza la estructura de la futura aplicación. Se presentaron además las clases del diseño representando la estructura de clases y métodos que serán implementados para el producto final. Fueron abordados los patrones de diseño empleados para proveer a la solución una imagen de profesionalidad y calidad. Por último, se generó el diagrama entidad relación con las tablas que permitirán persistir los datos generados por el sistema.

Capítulo 3: Implementación y prueba

3.1 Introducción

La solución a desarrollar, debe ser fruto de un correcto diseño y construcción de las funcionalidades, las mismas se realizan de forma iterativa facilitando organizar el progreso del sistema e ir obteniendo pequeños fragmentos del producto. Durante la construcción se ejecuta la implementación y las pruebas de caja negra de las funcionalidades, teniendo estas listas para integrarse al sistema.

3.2 Configuración del sistema

Para guardar la configuración del sistema se utilizó XML, cuyo principal objetivo consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por toda la aplicación y soporte. El XML tiende a la globalización y la compatibilidad entre los sistemas, ya que es una tecnología que permite compartir la información de una manera segura, fiable y fácil. Además, XML permite al programador y los soportes dedicar sus esfuerzos a las tareas importantes cuando trabaja con los datos, ya que algunas tareas tediosas como la validación de estos o el recorrido de las estructuras corre a cargo del lenguaje y está especificado por el estándar, de modo que el programador no tiene que preocuparse por ello. (26)

El XML cumple los siguientes objetivos:

- Que fuera idéntico a la hora de servir, recibir y procesar la información que el HTML, para aprovechar toda la tecnología implantada para este último.
- Formal y conciso desde el punto de vista de los datos y la manera de guardarlos.
- Fácil de leer y editar.
- Fácil de implantar, programar y aplicar a los distintos sistemas.

Formato del XML:

Se escribe en un documento de texto ASCII, igual que el HTML y en la cabecera del documento se tiene que poner el texto:

```
<?xml version="1.0" encoding="utf-16"?>
```

En el documento se deben escribir las etiquetas que se deseen como las de HTML, como su nombre lo indica es un lenguaje de etiquetas extendido. Las etiquetas se escriben anidadas, unas dentro de otras.

```
<Label1>...<Label2>...</Label1>...</Label2>
```

Las etiquetas pueden tener atributos. Se le puede poner los atributos que deseen.

```
<Label atributo1="valor1" atributo2="valor2"...>
```

Para definir qué etiquetas y atributos se debe utilizar al escribir en XML, es necesario fijarse en la manera de guardar la información, es decir de una forma estructurada y ordenada. Por ejemplo, si se desea guardar la información relacionada con el calendario de trabajo en un documento XML se utiliza un esquema con las siguientes etiquetas:

```
<?xml version="1.0" encoding="utf-16"?>
<CalendarioTrabajo xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <DuracionCita>43</DuracionCita>
  <HoralInicioSeccionMannana>2013-04-26T19:00:00</HoralInicioSeccionMannana>
  <HoraFinSeccionMannana>2013-04-26T01:30:00</HoraFinSeccionMannana>
  <HoralInicioSeccionTarde>2013-04-26T06:00:00</HoralInicioSeccionTarde>
  <HoraFinSeccionTarde>2013-04-26T03:30:00</HoraFinSeccionTarde>
  <DiasTrabajoSemana>
    <string>Lunes</string>
    <string>Martes</string>
    <string>Miércoles</string>
    <string>Jueves</string>
    <string>Viernes</string>
    <string xsi:nil="true" />
    <string>Domingo</string>
  </DiasTrabajoSemana>
  <DiasNoLaborables>
    <dateTime>2013-04-08T00:00:00</dateTime>
    <dateTime>2013-04-15T00:00:00</dateTime>
    <dateTime>2013-04-09T00:00:00</dateTime>
  </DiasNoLaborables>
</CalendarioTrabajo>
```

3.3 Método para la distribución de las solicitudes de citas

En muchas ocasiones, un fenómeno muy común es la formación de colas; esto suele ocurrir cuando la demanda real de un servicio es superior a la capacidad que existe para dar dicho servicio. Ejemplo real de esa situación es la espera para realizar trámites de documentos de identificación en la Universidad de las Ciencias Informáticas.

La capacidad de la cola utilizada en el sistema, es el máximo número de clientes que pueden estar solicitando la cita (antes de comenzar a ser servidos), en el caso del Sistema de Gestión de Citas es una cola lo suficientemente grande que asegure que esta no se llene hasta desbordarse, debido a que no se sabe con seguridad cuantos clientes van a solicitar dicho servicio. El sistema da la opción de guardar la solicitud en una cola de espera, esto ocurre cuando la cita no se va a realizar en el instante y tampoco tiene definido los detalles de la misma, es decir, fecha y hora. En este caso el sistema tiene la responsabilidad de asignar la fecha y la hora a dicha cita.

El proceso de asignarle fecha y hora a cada una de las citas en cola se realiza de la siguiente manera: se busca en la base de datos, específicamente se obtiene de la tabla dsolicitud, la fecha y hora en la que se realizó la última cita y a partir de la misma se procede a asignar la fecha y hora de las citas en cola. Es importante decir que durante este proceso se tiene en cuenta el orden en que se le presta el servicio a cada uno de sus miembros. En este caso el orden que se tuvo en cuenta es el algoritmo FIFO (*First In First Out*) primero en entrar primero en salir, que no es más que atender primero a la cita que lleve más tiempo esperando, dígame la que primero entró a la cola.

3.4 Diagrama de componentes

Los diagramas de componentes muestran las organizaciones y dependencias lógicas entre componentes, además de ayudar a entender mejor el modelo de implementación. También puede contener paquetes utilizados para agrupar elementos del modelo. Los mismos prevalecen en el campo de la arquitectura de *software*.

El sistema está compuesto por el componente: <<GestionCitas.WebApp.dll>>, encargado de interactuar con los usuarios agrupando las clases de interfaz y servicios necesarios para la interacción. El mismo hace referencia a otros como: <<GestionCitas.Business.dll >> que contiene las clases controladoras del negocio. Las entidades y la capa de acceso a datos están contenidas dentro de <<GestionCitas.Entities.dll>> y <<GestionCitas.DAL.dll>> respectivamente. A continuación se presenta el diagrama de componentes correspondiente al Sistema de Gestión de Citas. Ver (Figura 7)

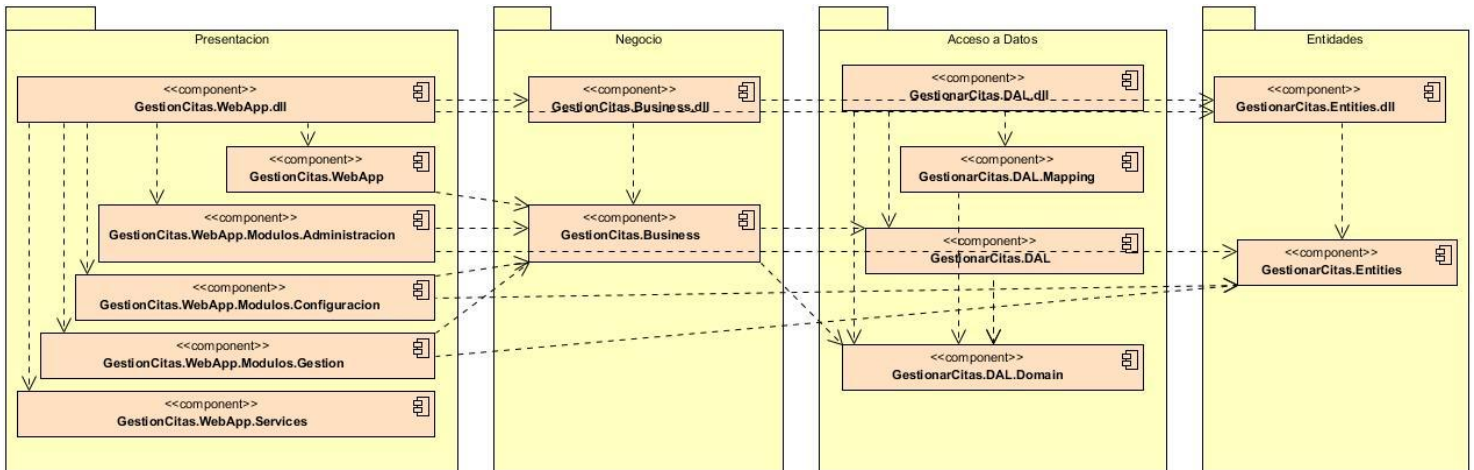


Figura 7: Diagrama de componentes

3.5 Diagrama de despliegue

Los diagramas de despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Los diagramas de despliegue son los complementos de los diagramas de componentes que, unidos, proveen la vista de implementación del sistema. Describen la topología del sistema, la estructura de los elementos de *hardware* y el *software* que ejecuta cada uno de ellos. Este tipo de diagrama representa a los nodos y sus relaciones; los nodos son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP/IP. (27) A continuación se presenta el diagrama de despliegue correspondiente al Sistema de Gestión de Citas. Ver (Figura 8)

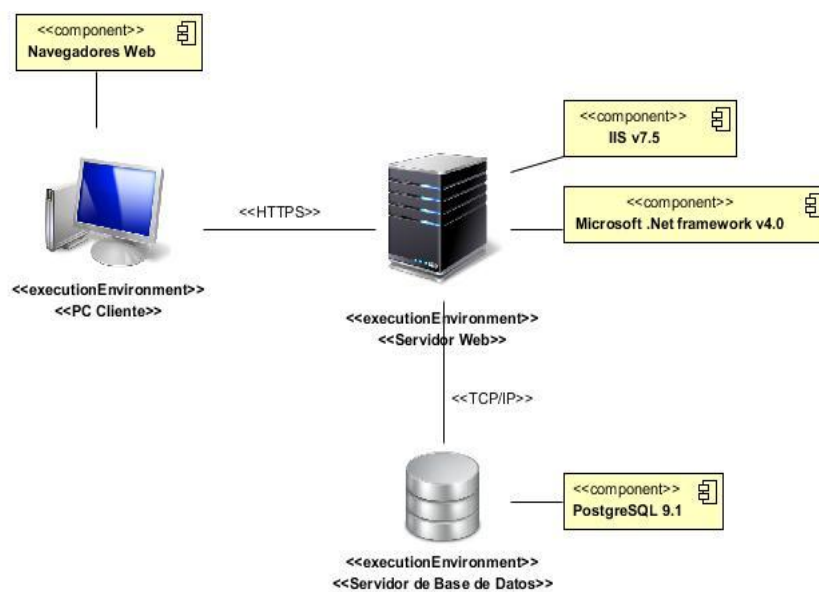


Figura 8: Diagrama de despliegue

3.6 Estándares de codificación

Un estándar de codificación es una forma de "normalizar" la programación de forma tal que logre un mayor entendimiento entre todas las personas que trabajan directamente sobre el código. Por lo general estos definen la forma en que deben ser declaradas las variables, las clases y los comentarios. Los estándares de codificación ayudan a garantizar que todos los usuarios tengan acceso a la información que el sitio *web* brinda a sus visitantes, además será más fácil para las personas con necesidades especiales utilizar la *web*.

El lenguaje de programación seleccionado para el desarrollo del sistema fue C#, a continuación se detallan algunos estándares establecidos a la hora de escribir el código en dicho lenguaje. Ver (Tabla 13)

Tabla 13: Criterios de calidad

Criterio	Objetivo
Facilidad de comunicación	Proporcionar a los usuarios y desarrolladores entradas y salidas fácilmente asimilables.
Descripción	Proporcionar y/o plasmar en el código detalles y explicaciones sobre la implementación realizada.
Simplicidad	La implementación realizada debe hacerse de la forma más comprensible posible.

3.6.1 Principios generales de nomenclaturas

Los nombres de cada uno de los elementos del programa, clase o interfaz deben ser significativos, su nombre debe explicar, siempre que sea posible, el uso o fin del elemento; se deben nombrar usando sustantivos (posiblemente compuestos) o formas verbales en imperativo. La forma de construir los nombres se basa en varias convenciones y guías propuestas por *Microsoft* entre otros. Los términos Convención Pascal⁶ y Convención Camel⁷ serán usados a lo largo de todo el desarrollo de la aplicación. A continuación, en la Tabla 14 se describen las reglas fundamentales:

Tabla 14: Principios de nomenclaturas

Tipos de identificadores	Reglas para nombres	Ejemplos
--------------------------	---------------------	----------

⁶ El primer carácter de cada palabra es en mayúscula y el resto en minúscula. Ejemplo: GestionarCitas

⁷ El primer carácter de cada palabra es en mayúscula (excepto la primera palabra) y el resto en minúscula. Ejemplo: gestionarCitas

Clases	Use la convención Pascal para el nombre de las clases.	Ejemplo: public class HelloWorld { ... }
Interfaces	Use el prefijo "I" con la convención Camel para las interfaces.	Ejemplo: IEntity
Métodos	Use la convención Pascal para el nombre de los métodos.	Ejemplo: void SayHello(string name) { ... }
Variables	<ul style="list-style-type: none"> • Use la convención Camel para el nombre de variables y parámetros. • Usa el significado de las palabras para el nombre de las variables, no uses abreviaturas. • No uses caracteres simples para el nombre de las variables i, n, s etc. Use nombres como index, temp. • No usar underscores (_) para los nombres de las variables locales. • Todas las variables miembros de las clases deben ser prefijadas con underscore (_) para ser diferenciadas de otras variables. • No usar nombres de variables que coincidan con palabras reservadas. • Las variables booleanas y las propiedades se les pueden poner como prefijos "is". 	Ejemplo: int totalCount = 0; void SayHello(string name) { string fullMessage = "Hello " + name; ... } Bien: string address int salary Mal: string nam string addr int sal Ejemplo: Una excepción de esta regla son las variables de los ciclos: for (int i = 0; i < count; i++) { ... } Ejemplo: Private bool _isFinished
Ficheros	Use la convención Pascal para el nombre de los ficheros.	Ejemplo: El fichero que contenga la clase "class HelloWorld", debe ser: helloworld.cs (or,helloworld.vb)
Indentación y espaciado	<ul style="list-style-type: none"> • Usar Tab para la Indentación. Define el tamaño del Tab como 4. • No usar SPACES. 	

Comentarios	<ul style="list-style-type: none"> • Comentarios deben estar en el mismo nivel del código. • Usar el mismo nivel de indentación. • No escribir comentarios para cada línea de código o para cada variable declarada. • Usar // o /// para los comentarios. Evitar usar /* ... */ 	<p>Bien:</p> <pre>// Format a message and display string fullMessage = "Hello " + name; DateTime currentTime = DateTime.Now; string message = fullMessage + ", the time is : " + currentTime.ToShortTimeString(); MessageBox.Show (message);</pre> <p>Mal:</p> <pre>// Format a message and display string fullMessage = "Hello " + name; DateTime currentTime = DateTime.Now; string message = fullMessage + ", the time is : " + currentTime.ToShortTimeString(); MessageBox.Show (message);</pre>
Llaves y líneas en blanco	<ul style="list-style-type: none"> • Las llaves se deben poner al mismo nivel del código que las contiene. • Usar una línea en blanco para separar agrupaciones lógicas del código. • Debe dejarse una y solo una línea en blanco entre cada método dentro de las clases. • Las llaves deben ser utilizadas sobre líneas separadas y no sobre la misma línea como en if y for. • Usar un espacio simple antes y después de cada operador y llave. • Usar #region para agrupar las partes del código relacionadas. • Mantener los miembros privados: variables, propiedades y métodos en la parte superior de los ficheros y clases, mantener los públicos en la parte de abajo. 	<p>Bien:</p> <pre>if (...) { // Do something }</pre> <p>Mal:</p> <pre>if (...){ // Do something }</pre> <p>Bien:</p> <pre>if (showResult == true) { for (int i = 0; i < 10; i++) { // } }</pre> <p>Mal:</p> <pre>if(showResult==true) { for(int i= 0;i<10;i++) { // } }</pre>

Elementos de entrada y salida	Usar un prefijo específico atendiendo al tipo de dato que procesa la variable de entrada salida siguiendo los prefijos de la siguiente tabla.	Ejemplo de los prefijos: <table border="1" data-bbox="1016 268 1453 674"> <thead> <tr> <th>Control</th> <th>Prefix</th> <th>Repeater</th> <th>rep</th> </tr> </thead> <tbody> <tr> <td>Label</td> <td>lbl</td> <td>Checkbox</td> <td>chk</td> </tr> <tr> <td>TextBox</td> <td>txt</td> <td>CheckBoxList</td> <td>cbi</td> </tr> <tr> <td>DataGrid</td> <td>dtg</td> <td>RadioButton</td> <td>rdo</td> </tr> <tr> <td>Button</td> <td>btn</td> <td>RadioButtonList</td> <td>rbi</td> </tr> <tr> <td>ImageButton</td> <td>imb</td> <td>Image</td> <td>img</td> </tr> <tr> <td>Hyperlink</td> <td>hik</td> <td>Panel</td> <td>pnl</td> </tr> <tr> <td>DropDownList</td> <td>ddl</td> <td>PlaceHolder</td> <td>phd</td> </tr> <tr> <td>ListBox</td> <td>lst</td> <td>Table</td> <td>tbi</td> </tr> <tr> <td>DataList</td> <td>dli</td> <td>Validators</td> <td>val</td> </tr> </tbody> </table>	Control	Prefix	Repeater	rep	Label	lbl	Checkbox	chk	TextBox	txt	CheckBoxList	cbi	DataGrid	dtg	RadioButton	rdo	Button	btn	RadioButtonList	rbi	ImageButton	imb	Image	img	Hyperlink	hik	Panel	pnl	DropDownList	ddl	PlaceHolder	phd	ListBox	lst	Table	tbi	DataList	dli	Validators	val
Control	Prefix	Repeater	rep																																							
Label	lbl	Checkbox	chk																																							
TextBox	txt	CheckBoxList	cbi																																							
DataGrid	dtg	RadioButton	rdo																																							
Button	btn	RadioButtonList	rbi																																							
ImageButton	imb	Image	img																																							
Hyperlink	hik	Panel	pnl																																							
DropDownList	ddl	PlaceHolder	phd																																							
ListBox	lst	Table	tbi																																							
DataList	dli	Validators	val																																							

3.7 Validación de la propuesta de solución

A lo largo de la historia los proyectos de *software* han sufrido problemas de calidad, tanto en el proceso de desarrollo como en los productos en sí. La deficiente gestión de configuración de *software* produce demoras en los plazos de entrega de los productos y en el nivel en que los mismos satisfacen las expectativas del cliente. Probar el *software* es de vital importancia ya que proporciona un alto grado de confianza y seguridad en el producto, a la vez que permite hacer estimaciones realistas sobre el comportamiento del producto una vez que se encuentre en funcionamiento.

Las pruebas son un conjunto de actividades en las cuales un sistema o componente es ejecutado bajo unas condiciones o requerimientos específicos, donde los resultados son observados y registrados para dar una evaluación de algún aspecto del sistema o componente que es evaluado y determinar la calidad del mismo.

Objetivos de las Pruebas:

- Encontrar y documentar los defectos que puedan afectar la calidad del *software*.
- Validar que el *software* trabaje como fue diseñado.
- Validar y probar los requisitos que debe cumplir el *software*.
- Validar que los requisitos fueron implementados correctamente.

3.7.1 Definición y descripción de las pruebas

Pruebas Unitarias

Las pruebas unitarias se realizan con el objetivo de ejecutar un código fuente llamando directamente a los métodos de una clase, pasándole a estos los parámetros apropiados. El objetivo de las pruebas unitarias

es el aislamiento de las partes del código y la demostración de que estas partes no contienen errores. Es por ello que se consideran a las pruebas unitarias como uno de los tipos de pruebas más importantes que se le aplican a los *software*, logrando como resultado que disminuya en un gran por ciento el número de errores existentes en los sistemas y por ende una mayor calidad y confiabilidad. Las pruebas unitarias se le realizaron a los servicios del sistema para validar que las salidas son las correctas y asegurarle al desarrollador que su solución no presenta errores en la lógica de programación. (28)

Pruebas de Caja Negra

Se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas que devuelve, sin importar su funcionamiento interno. Existe un tipo de pruebas que se denominan de Caja Negra en las cuales se analizan porciones de *software* atendiendo a sus funcionalidades, es decir, a qué hace sin importar como lo realiza. Por lo tanto, de una caja negra deben estar muy bien definidas sus entradas y salidas; pero no resulta necesario los detalles internos de su funcionamiento. (29)

En las pruebas de Caja Negra los casos de prueba pretenden demostrar que las funciones del *software* son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. Las pruebas de Caja Negra además:

- Verifican las especificaciones funcionales y no consideran la estructura interna del programa.
- Es hecha sin el conocimiento interno del producto.
- No validan funciones ocultas (por ejemplo funciones implementadas pero no descritas en las especificaciones funcionales del diseño) por tanto los errores asociados a ellas no serán encontrados.

Los casos de prueba fueron definidos según las descripciones de los requisitos para cada una de las iteraciones. Contaron con la definición de los valores que deben tomar cada una de las variables de entrada para la obtención de los resultados esperados. Luego de terminados los casos de prueba, se evaluó el sistema a través de la comparación entre los resultados obtenidos y los resultados esperados.

3.7.2 Aplicación de las pruebas

Pruebas Unitarias

Para cada iteración de desarrollo se realizaron las iteraciones de pruebas necesarias con el fin de lograr el correcto funcionamiento de cada una de las funcionalidades. Durante la ejecución de las pruebas se creó para cada clase un caso de prueba con los métodos necesarios para verificar su ejecución. A continuación se muestra un ejemplo de los casos de prueba creados:

Clase *BusinessFactoryTest*

Mediante la clase *BusinessFactoryTest* se verifica que los datos de un solicitante se registren correctamente. Esta clase cuenta con el método de prueba fundamental: *InsertarSolicitud()*, se realiza para insertar los datos básicos de un solicitante. A continuación se muestra el resultado de la ejecución de esta prueba:

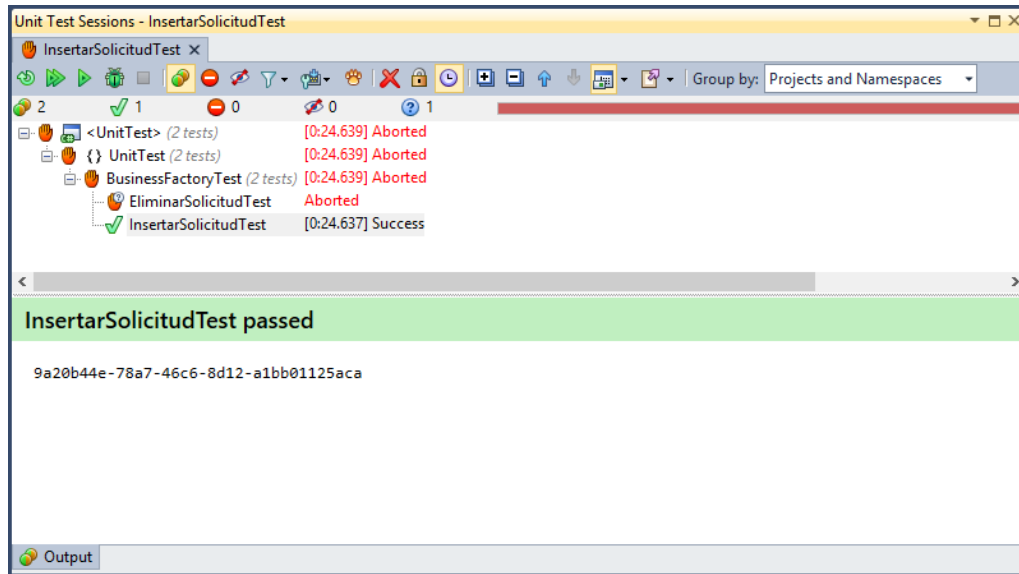


Figura 9: Ejecución de la clase "BusinessFactoryTest"

En la Figura 9 se aprecia el *test* ejecutado para comprobar que se insertan correctamente las solicitudes, usando la clase *BusinessFactoryTest*. El icono verde que se ubica al lado del nombre del *test* indica que se completó con éxito, en caso de haber fallado el color que se muestra es rojo. La barra de progreso en verde asegura que la totalidad de la prueba se ejecutó satisfactoriamente.

Pruebas de caja negra

Para la aplicación de las pruebas de caja negra se desarrollaron los casos de prueba correspondientes a todas las funcionalidades del sistema, estos constituyen un conjunto de condiciones o variables bajo las cuales el analista determinará si el requisito de una aplicación es parcial o completamente satisfactorio.

Los casos de prueba que no se detallan a continuación, fueron realizados y se encuentran registrados en la documentación oficial del proyecto Plataforma Modular de Identificación y Control de Acceso. La Tabla 15 muestra la descripción de las variables del caso de prueba "Gestionar Citas" y la Tabla 16 muestra el caso de prueba "Gestión de Citas".

Tabla 15: Descripción de las variables del caso de prueba

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre	campo de texto	No	En este campo se insertarán el nombre(s) del solicitante, no se deben introducir números, ni caracteres especiales.
2	Apellidos	campo de texto	No	En este campo se insertarán los apellidos del solicitante, no se deben introducir números, ni caracteres especiales.
3	Correo	campo de texto	Sí	Este campo puede o no quedar vacío, debido a que no todas las personas tienen correo electrónico.
4	Nro. Identificación	campo de texto	No	Puede ser un número de C.I (sólo 11 números) o un número de solapín (de los 7 dígitos, el primer o los dos primeros caracteres que sean letras y los restantes que sean números).
5	Modo de Cita	campo de selección	No	Se seleccionará solamente uno de dos modos de citas a elegir.
6	Tipo de Solicitud	lista desplegable	No	No se debe quedar en blanco, debe seleccionar alguno.
7	Fecha	calendario para seleccionar la fecha	No	No se debe quedar en blanco, debe seleccionar alguno.
8	Hora	lista desplegable	No	No se debe quedar en blanco, debe seleccionar alguno.
9	Tipos de documentos	lista desplegable	No	No se debe quedar en blanco, debe seleccionar algún tipo de documento.
10	Clasificación de documentos	lista desplegable	No	No se debe quedar en blanco, debe seleccionar alguna.
11	Tipo de cita	campo de selección	No	Se debe dejar marcado el tipo de cita que se desea.
12	Duración de la cita	campo de texto	No	La duración de la cita debe quedar registrada en minutos. No se deben ingresar letras, ni caracteres especiales.
13	Sesión Mañana (Hora inicio)	campo de texto	No	La hora debe estar en formato de 24h.
14	Sesión Mañana (Hora fin)	campo de texto	No	La hora debe estar en formato de 24h.
15	Sesión Tarde (Hora inicio)	campo de texto	No	La hora debe estar en formato de 24h.
16	Sesión Tarde (Hora fin)	campo de texto	No	La hora debe estar en formato de 24h.
17	Días no laborales	calendario para seleccionar la fecha	Sí	Se debe seleccionar del calendario los días que la oficina no trabajará.
18	Estado	campo de selección	No	No se debe quedar en blanco.

Tabla 16: Caso de prueba “gestionar citas”

Escenario	Descripción	Nombre	Apellidos	Nro. Identificación	Modo de cita	Tipo de cita	Fecha	Hora	Respuesta del sistema	Flujo central
EC 1.1 RF1. Registrar datos básicos del solicitante.	Se deben introducir los datos básicos del solicitante en el sistema para poder hacer la solicitud.	V	V	V	V	N/A	N/A	N/A	Una vez ingresados todos los datos el sistema devolverá una notificación de inserción exitosa.	Opción Gestionar cita. Nueva cita. Insertar cita.
		I	V	V	V	N/A	N/A	N/A	El sistema devolverá un mensaje de error.	
		V	I	V	V	N/A	N/A	N/A	El sistema devolverá un mensaje de error.	
		V	V	I	V	N/A	N/A	N/A	El sistema devolverá un mensaje de error.	
		V	V	V	I	N/A	N/A	N/A	El sistema devolverá un mensaje de error.	
I	I	I	I	N/A	N/A	N/A	N/A	El sistema devolverá un mensaje de error.		
EC 1.2 RF2. Registrar datos de la solicitud.	Se deben de introducir los datos necesarios para realizar la solicitud.	N/A	N/A	N/A	N/A	V	V	V	El sistema devolverá una notificación de inserción exitosa.	Opción Gestionar cita. Nueva cita. Insertar cita.
		N/A	N/A	N/A	N/A	I	V	V	El sistema devolverá un mensaje de error.	
		N/A	N/A	N/A	N/A	V	I	V	El sistema devolverá un mensaje de error.	
		N/A	N/A	N/A	N/A	I	I	I	El sistema devolverá un mensaje de error.	
N/A	N/A	N/A	N/A	V	V	I	El sistema devolverá un mensaje de error.			
EC 1.3 RF3. Mostrar solicitud.	El sistema permitirá ver la solicitud con todos sus datos.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	El sistema dará la opción de ver todos los datos de una cita.	Opción Gestionar cita. Ver.
		N/A	N/A	N/A	N/A	N/A	N/A	N/A		
EC 1.4 RF4. Verificar solicitud.	El sistema verifica que todos los datos sean correctos y la persona está autorizada para obtener el documento que está solicitando.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Una vez comprobados que los datos son correctos. En caso que haya algo incorrecto lanza un mensaje de error.	Opción Gestionar cita. Nueva cita. Insertar cita. Calendario.
		N/A	N/A	N/A	N/A	N/A	N/A	N/A		
EC 1.5 RF5. Registrar cancelación de una solicitud.	En caso de que el cliente no desee realizar una cita puede cancelarla.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	El sistema cambia el estado de la cita.	Opción Gestionar cita. Cancelar cita.
EC 1.6 RF6. Mostrar	El sistema debe mostrar solo los días y las horas disponibles para las citas.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	En el calendario se mostrará que cuando el día esté disponible estará desmarcado y en el caso de la hora, solo saldrán en la	Opción Gestionar cita. Nueva cita. Insertar cita.

disponibilidades de cita.		N/A	N/A	N/A	N/A	N/A	N/A	N/A	lista desplegable las horas disponibles.	Calendario.
EC 1.7 RF7. Permitir seleccionar día y hora.	En el calendario que te muestra la aplicación, el sistema permitirá seleccionar el día y la hora que no estén marcadas.	N/A	N/A	N/A	N/A	N/A	V	V	El sistema no mostrará los días y las horas reservados, es decir, no permitirá seleccionarlas.	Opción Gestionar cita. Nueva cita. Insertar cita. Seleccionar del calendario.
		N/A	N/A	N/A	N/A	N/A	I	I	En caso de que se desee seleccionar una fecha o una hora que estaban reservadas previamente, el sistema lanzará un mensaje de error.	
EC 1.8 RF8. Comprobar disponibilidad.	El sistema verifica si la fecha y hora introducidos por el usuario están disponibles en ese instante de tiempo, si están disponibles el sistema emite el comprobante de citas, si no están disponibles se emite una notificación al usuario indicando que debe seleccionar otra fecha y hora.	N/A	N/A	N/A	N/A	N/A	V	V	Si el día y la hora seleccionados por el usuario están disponibles, el sistema mostrara comprobante de cita.	Opción Gestionar cita. Nueva cita. Insertar cita. Calendario.
		N/A	N/A	N/A	N/A	N/A	I	I	En caso de que el día o la hora no estén disponibles, el sistema enviará un mensaje de error indicando que debe seleccionar otro día u otra hora.	
EC 1.9 RF9. Guardar solicitud en la cola.	Cuando el tipo de cita es automática, la solicitud se guarda en una cola.	N/A	N/A	N/A	V	N/A	N/A	N/A	El sistema guardará las solicitudes automáticamente.	Opción Configuración. Tipo de Cita. Cita automática. Guardar en cola.
		N/A	N/A	N/A	I	N/A	N/A	N/A	No se guardarán automáticamente.	
EC 1.10 RF12. Cancelar cita	Da la posibilidad de borrar la cita completamente del sistema.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se eliminará la cita totalmente del sistema.	Gestionar cita. Cancelar. Aceptar.
		N/A	N/A	N/A	N/A	N/A	N/A	N/A		

3.7.3 Resultados de las pruebas

Pruebas Unitarias

Durante la ejecución de las pruebas unitarias para la primera y segunda iteración de desarrollo, se realizaron dos iteraciones de prueba.

En la Tabla 17 se representan los resultados alcanzados en la aplicación de las pruebas unitarias realizadas durante las 2 iteraciones de desarrollo.

Tabla 17: Resultados de las Pruebas Unitarias

Iteraciones	Funcionalidades		
	Funcionalidades con errores	Funcionalidades correctas	Funcionalidades no implementadas
1	5	6	3
1,1	3	9	2
2	2	11	1
2,1	0	14	0

En la Figura 10 se muestra de forma gráfica los resultados especificados anteriormente:

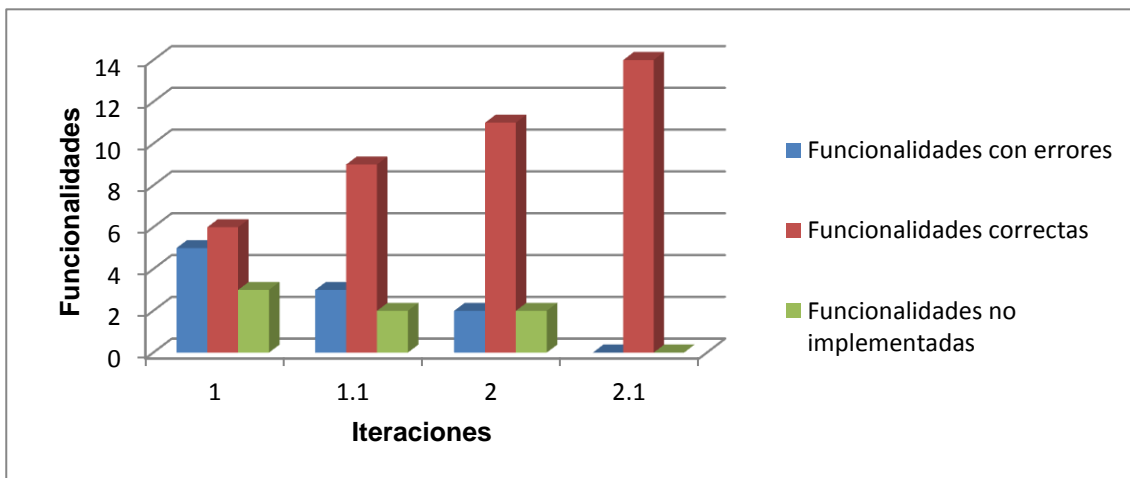


Figura 10: Gráfico del resultado de las Pruebas Unitarias

Pruebas de Caja Negra

Las pruebas de caja negra fueron aplicadas sobre una versión estable del producto, realizándose 4 iteraciones de prueba para cada uno de las funcionalidades. La Tabla 18 muestra los resultados arrojados luego de su aplicación.

Tabla 18: Resultados de las Pruebas de Caja Negra

Iteraciones	Escenarios			
	Gestión	Configuración	Administración	Reportes
1	8	7	10	4
2	5	4	5	2
3	1	2	2	1
4	0	0	0	0

En la Figura 11 se muestra de forma gráfica los resultados especificados anteriormente:

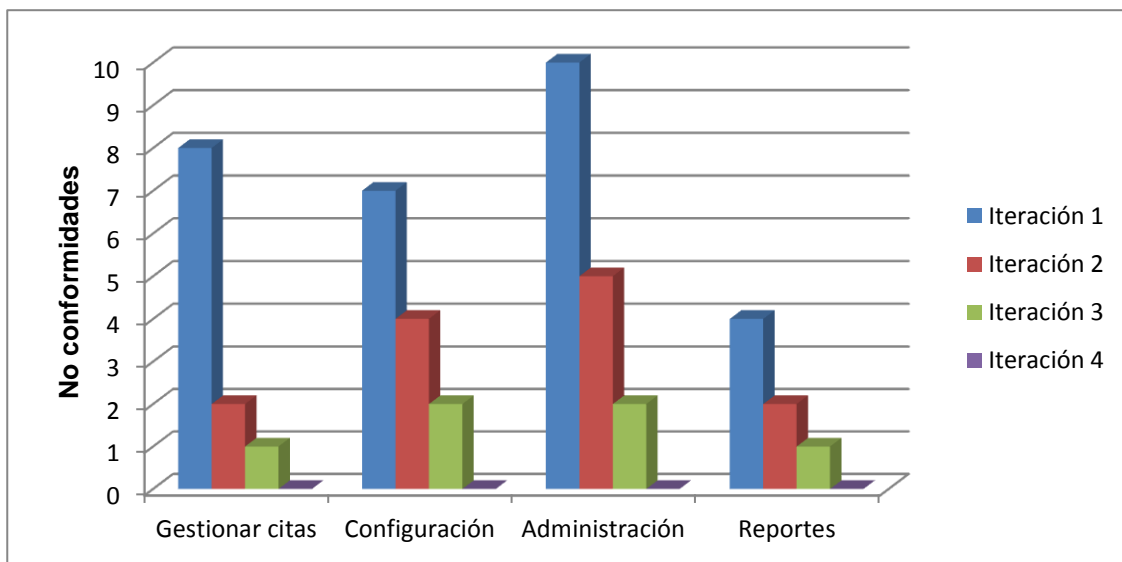


Figura 11: Gráfico del resultado de las Pruebas de Caja Negra

3.8 Elementos de seguridad en el sistema

La seguridad en las aplicaciones *web* son de vital importancia, por lo que se debe comprender e implementar las mismas para protegerlas. (30)

ASP.NET funciona junto con *Microsoft .NET Framework* e *Internet Information Services* (IIS) para ayudar a proporcionar aplicaciones *web* seguras. Para dar seguridad al Sistema de Gestión de Citas, se llevó a cabo dos funciones principales:

Autenticación: Ayuda a comprobar que el usuario es precisamente quien dice ser. La aplicación obtiene las credenciales (diversas formas de identificación, como nombre y contraseña) de un usuario, y las valida consultando a una autoridad determinada. Si las credenciales son válidas, se considera a la entidad que ha enviado las credenciales como una entidad autenticada.

Autorización: Limita los derechos de acceso mediante la concesión o negación de permisos específicos a una identidad autenticada.

Internet Information Services (IIS) puede conceder o negar el acceso en función de la dirección IP o del nombre de *host* del usuario. ASP.NET se basa en *Microsoft .NET Framework*, el desarrollador de aplicaciones ASP.NET también tiene acceso a todas las características de seguridad integradas de *.NET Framework*, como la seguridad de acceso a código y la seguridad de acceso basada en funciones.

3.9 Conclusiones

El desarrollo del Sistema de Gestión de Citas perteneciente al subsistema de Identificación, permitió incorporar a la solución una mejor estructura, al tener ya definidos los componentes que habrían de ser programados, así como la relación de los nodos físicos que los componen. Además, se definieron los estándares de codificación a utilizar para el desarrollo de la aplicación con el objetivo de lograr una buena organización en el código y un mayor entendimiento de cada sentencia.

Las pruebas de caja negra y de unidad realizadas para el Sistema de Gestión de Citas, posibilitaron evaluar la calidad del producto, desde las respuestas de la aplicación ante las peticiones del usuario hasta el comportamiento de las clases y métodos de los procesos. La implementación del sistema cumplió con los requisitos definidos para la solución.

Conclusiones Generales

Con el desarrollo del presente trabajo de diploma se ha logrado dar cumplimiento a los objetivos trazados, proporcionando una solución al proceso de entrega de documentos de Identificación en la Universidad de las Ciencias Informáticas. A continuación se muestran los principales resultados obtenidos:

- El estudio del estado en que se encuentra la gestión de citas a nivel nacional e internacional contribuyó de manera importante en el desarrollo del sistema, lo que permitió adquirir conocimientos fundamentales sobre el tema, pues de éstos se obtuvieron funcionalidades válidas para el análisis y diseño del sistema implementado. Una vez detalladas las herramientas y tecnologías utilizadas permitieron un desarrollo claro y fluido de la solución, construida sobre bases sólidas y un entorno bien definido.
- La utilización de la metodología seleccionada posibilitó documentar ampliamente la solución, lo que brindó guías en el proceso de desarrollo y agilizó la implementación del mismo. Luego de definir los requerimientos funcionales y no funcionales que el sistema debía cumplir, se realizó el análisis y diseño de la solución, definiéndose las clases necesarias, así como la relación entre ellas, brindándole una solución a todas las funcionalidades descritas.
- Al identificarse el estilo arquitectónico en capas, el cual tiene como característica la definición clara de las responsabilidades, permitió reducir el acoplamiento y aumentar la reutilización de las mismas. Los patrones de diseño y los estándares de codificación empleados permitieron garantizar una mayor flexibilidad en la ejecución de la propuesta de solución, trayendo como resultado una implementación de manera organizada, permitiendo un mayor entendimiento de la misma.
- Probar la solución es un requisito de obligatorio cumplimiento para garantizar la calidad del producto, por esto fue la atención que se le prestó a dicha actividad. Las pruebas se realizaron de forma incremental, permitiendo la detección de no conformidades y su corrección, obteniéndose así un producto de mayor calidad, lo cual solidifica las bases y brinda confianza y experiencia para las versiones posteriores.
- La solución obtenida confirma que el desarrollo del Sistema de Gestión de Citas cumple con los requisitos definidos y con la calidad requerida, resolviendo los problemas existentes en la oficina de trámites de la Universidad de las Ciencias Informáticas.

Recomendaciones

Teniendo en cuenta los resultados obtenidos con el desarrollo del Sistema de Gestión de Citas para la oficina de trámites de identificación de la Universidad de las Ciencias Informáticas, se recomienda:

- Implementar la funcionalidad de obtener los datos del solicitante por una fuente de datos externa.
- Agregar al sistema funcionalidades que le permitan gestionar citas para otro tipo de trámites.
- Desarrollar un manual de usuario que ayude a interactuar con el Sistema de Gestión de Citas.

Bibliografía Referenciada

1. diccionario de la lengua española. [En línea] [Citado el: 4 de mayo de 2013.] <http://www.wordreference.com/definicion/>.
2. definición abc. [En línea] 2013. [Citado el: 25 de abril de 2013.] <http://www.definicionabc.com/general/identificacion.php>.
3. Manual de Usuario del Sistema de Cita Previa de DNI y Pasaporte electrónico. [En línea]
4. **Gustavo Andrés Castro Guevara, Julián Alberto Ramírez Osori.** *Sistema de Información para la Gestión de las Citas en los Centros de Imágenes Diagnósticas, Pereira* . 2009.
5. Antizipa. [En línea] <http://www.cazurrabbit.com/antizipa/sanidad.html>.
6. Software de Gestión de Citas y facturación para clínicas. *Cideco*. [En línea] <http://www.cideco.es/index.php/otros-proyectos/38-otros-proyectos/102-software-de-gestion-de-citas-y-facturacion-para-clinicas.html>.
7. SuCita.con. [En línea] 2009. <http://citapreviaconsular.maec.es/>.
8. Manual de Usuario Paquete del Centro de Personalización e Impresión de Documentos (CPDI) versión 1.0. [En línea]
9. Sistema de Atención Consular. [En línea] <http://chileabroad.gov.cl/la-habana/en>.
10. **Yosmel Martínez Díaz Alejandro, Tarafa Guzmán.** *Módulo Citas del Sistema de Información Hospitalaria alas HIS*. 2009.
11. desarrolloweb.com. [En línea] <http://www.desarrolloweb.com/articulos/1328.php>.
12. **Dr.Gabriel Nicolas Schenker.** NHibernate 3. [En línea] [Citado el: 5 de abril de 2013.]
13. [En línea] <http://www.programacion.com/>.
14. msdn. [En línea] [Citado el:] <http://msdn.microsoft.com/es-es/library/4w3ex9c2%28v=vs.100%29.aspx>.
15. **BizAgi.** *Business Process Modeling Notation*. Colombia : s.n., 2009.
16. **Rodríguez, Alfonso, Fernández-Medina, Eduardo y Piattini, Mario.** *Hacia la definición de Procesos de Negocios Seguros basados en una Arquitectura Dirigida por Modelos*. Chile : s.n., 2005.
17. **Gutiérrez, Sarah, y otros, y otros.** *FDD: Desarrollo Basado en Funcionalidades*. Colombia : Universidad Nacional de Colombia, 2009.
18. **Universidad ORT Uruguay, Cátedra de Ingeniería de Software.** *Metodología FDD*. Uruguay : s.n., 2003.
19. scribd. [En línea] <http://www.scribd.com/doc/84488342/Practica-3-Otras-metodologias-Agiles>.

20. [En línea] <http://www.microsoft.com/en-us/download/details.aspx?id=2890>.
21. softonic. [En línea] <http://axure-rp.m.softonic.com/>.
22. freedownloadmanager. [En línea]
http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/.
23. PostgreSQL-es. [En línea] <http://www.postgresql.org.es/documentacion>.
24. **Jesús Cáceres Tello**. Diagramas de secuencia. [En línea] 2013. [Citado el: 2 de mayo de 2013.] <http://www2.uah.es/jcaceres/capsulas/DiagramaSecuencia.pdf>.
25. msdn. [En línea] <http://msdn.microsoft.com/es-es/library/bb972272.aspx>.
26. **Alvarez, Miguel Angel**. *Cómo y para qué se creó el XML y una breve descripción de sus usos*.
27. **Ing. Jose Luis Jurado**. Scribd. [En línea] <http://www.scribd.com/doc/91245075/Diagramas-de-Despliegue>.
28. Microsoft. [En línea] [Citado el: 4 de Mayo de 2013.] <http://msdn.microsoft.com/es-es/library/ms182516.aspx>.
29. Prueba de Programas. [En línea]
http://repointo.upr.edu.cu/textuales/Elaborados%20por%20la%20academia/PIV_Prueba_de_programas.pdf.
30. [En línea] [http://msdn.microsoft.com/es-es/library/ks310b8y\(v=vs.100\).aspx](http://msdn.microsoft.com/es-es/library/ks310b8y(v=vs.100).aspx).

Bibliografía Consultada

1. diccionario de la lengua española. [Online] [Cited: mayo 4, 2013.]
<http://www.wordreference.com/definicion/>.
2. definición abc. [Online] 2013. [Cited: abril 25, 2013.]
<http://www.definicionabc.com/general/identificacion.php>.
3. Manual de Usuario del Sistema de Cita Previa de DNI y Pasaporte electrónico. [Online]
4. **Gustavo Andrés Castro Guevara, Julián Alberto Ramírez Osori.** *Sistema de Información para la Gestión de las Citas en los Centros de Imágenes Diagnósticas, Pereira* . 2009.
5. Antizipa. [Online] <http://www.cazurrabit.com/antizipa/sanidad.html>.
6. Software de Gestión de Citas y facturación para clínicas. *Cideco*. [Online]
<http://www.cideco.es/index.php/otros-proyectos/38-otros-proyectos/102-software-de-gestion-de-citas-y-facturacion-para-clinicas.html>.
7. SuCita.con. [Online] 2009. <http://citapreviaconsular.maec.es/>.
8. Manual de Usuario Paquete del Centro de Personalización e Impresión de Documentos (CPDI) versión 1.0. [Online]
9. Sistema de Atención Consular. [Online] <http://chileabroad.gov.cl/la-habana/en>.
10. **Yosmel Martínez Díaz Alejandro, Tarafa Guzmán.** *Módulo Citas del Sistema de Información Hospitalaria alas HIS*. 2009.
11. desarrolloweb.com. [Online] <http://www.desarrolloweb.com/articulos/1328.php>.
12. **Dr. Gabriel Nicolas Schenker.** NHibernate 3. [Online] [Cited: abril 5, 2013.]
13. [Online] <http://www.programacion.com/>.
14. msdn. [Online] [Cited:] <http://msdn.microsoft.com/es-es/library/4w3ex9c2%28v=vs.100%29.aspx>.
15. **BizAgi.** *Business Process Modeling Notation*. Colombia : s.n., 2009.
16. **Rodríguez, Alfonso, Fernández-Medina, Eduardo and Piattini, Mario.** *Hacia la definición de Procesos de Negocios Seguros basados en una Arquitectura Dirigida por Modelos*. Chile : s.n., 2005.
17. **Gutiérrez, Sarah, et al., et al.** *FDD: Desarrollo Basado en Funcionalidades*. Colombia : Universidad Nacional de Colombia, 2009.
18. **Universidad ORT Uruguay, Cátedra de Ingeniería de Software.** *Metodología FDD*. Uruguay : s.n., 2003.
19. scribd. [Online] <http://www.scribd.com/doc/84488342/Practica-3-Otras-metodologias-Agiles>.

20. [Online] <http://www.microsoft.com/en-us/download/details.aspx?id=2890>.
21. softonic. [Online] <http://axure-rp.m.softonic.com/>.
22. freedownloadmanager. [Online]
http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/.
23. PostgreSQL-es. [Online] <http://www.postgresql.org.es/documentacion>.
24. **Jesús Cáceres Tello**. Diagramas de secuencia. [Online] 2013. [Cited: mayo 2, 2013.]
<http://www2.uah.es/jcaceres/capsulas/DiagramaSecuencia.pdf>.
25. msdn. [Online] <http://msdn.microsoft.com/es-es/library/bb972272.aspx>.
26. **Alvarez, Miguel Angel**. *Cómo y para qué se creó el XML y una breve descripción de sus usos*.
27. **Ing. Jose Luis Jurado**. Scribd. [Online] <http://www.scribd.com/doc/91245075/Diagramas-de-Despliegue>.
28. Microsoft. [Online] [Cited: Mayo 4, 2013.] <http://msdn.microsoft.com/es-es/library/ms182516.aspx>.
29. Prueba de Programas. [Online]
http://repolinfo.upr.edu.cu/textuales/Elaborados%20por%20la%20academia/PIV_Prueba_de_programas.pdf.
30. [Online] [http://msdn.microsoft.com/es-es/library/ks310b8y\(v=vs.100\).aspx](http://msdn.microsoft.com/es-es/library/ks310b8y(v=vs.100).aspx).
31. The free dictionary. [Online] <http://es.thefreedictionary.com/cita>.
32. Desarrollo Basado en Funcionalidades. [Online]
<http://rousselz.wordpress.com/2010/08/08/metodologia-fdd-feature-driven-development-desarrollo-basado-en-funciones/>.
33. Ingeniería de Software. [Online] http://ingenieriadesoftware.mex.tl/61162_FDD.html.
34. Programación en Castellano. [Online] 2013.
http://www.programacion.com/articulo/disenio_de_software_con_patrones_parte_2_121.
35. Programación gratis .Net. [Online] <http://www.programas-gratis.net/b/citas-medicas>.
36. Scribd. [Online] <http://www.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>.
37. Slideshare. [Online] <http://www.slideshare.net/joshell/diagramas-uml-componentes-y-despliegue>.
38. **Larman, Craig**. Patrones GRASP Parte1_El mundo informático. [Online]
<http://jorgesaavedra.wordpress.com/2006/08/17/patrones-grasp-craig-larman/>.

39. **Universidad Carlos III de Madrid.** *Introducción a los sistemas de información: El modelo Cliente/Servidor.* Madrid : s.n., 2007.
40. **BPM, Club.** Club - BPM Business Process Management. Centro Oficial del BPM. [Online] Club BPM, 2006. [Cited: 04 26, 2011.] <http://www.club-bpm.com/Mision.htm>.
41. *Introducción a BPM para Dummies.* **KIRAN GARIMELLA, M. L., BRUCE WILLIAMS.** s.l. : Wiley Publishing, 2008. ISBN: 978-0-470-37359-0.
42. **Visual Paradigm.** Visual Paradigm. [Online] 2011. [Cited: Febrero 18, 2011.] <http://www.visual-paradigm.com/>.
43. Calidad del Software. [Online] 2011. www.calidaddelsoftware.com.
44. **David Garlaw, Mary Shaw.** *An introduction to software architecture.* 1994.
45. **Marca Huallpara, Quisbert Limachi.** *Análisis y Diseño de Sistemas.*
46. **CISED.** Centro de Identificación y Seguridad Digital. *Centro de Identificación y Seguridad Digital.* [Online] 2012. [Cited: Enero 13, 2013.] portal.proiden.uci.cu.
47. **Scribd.** Scribd. *Scribd.* [Online] [Cited: Enero 15, 2013.] <http://es.scribd.com/doc/7411856/Caracteristicas-de-C>.
48. **Barros, Óscar.** *Experiencias de empresas en BPM y arquitectura empresarial.* Chile : s.n., 2006.
49. **Club-BPM.** *Apuntes BPM. ¿Qué es BPM?* 2009.

Glosario de términos y siglas

B

BPM (*Business Process Management*): Conjunto de métodos, herramientas y tecnologías utilizados para diseñar, representar, analizar y controlar procesos de negocio operacionales. BPM es un enfoque centrado en los procesos para mejorar el rendimiento que combina las tecnologías de la información con metodologías de proceso y gobierno. BPM es una colaboración entre personas de negocio y tecnólogos para fomentar procesos de negocio efectivos, ágiles y transparentes. BPM abarca personas, sistemas, funciones, negocios, clientes, proveedores y socios.

C

CASE: Herramienta para el modelado de los procesos.

CLR: Es el centro neurálgico del Framework .NET encargado de gestionar la ejecución de las aplicaciones, aplicar parámetros de seguridad y ejecutar el denominado recolector de basuras.

D

Diagrama de componentes: Esquema o diagrama que muestra las interacciones y relaciones de los componentes de un modelo.

Diagrama de despliegue: Disposición física de los distintos nodos que componen un sistema.

F

Framework: Conjunto de clases modeladas de forma general para resolver problemas relacionados en un contexto específico.

I

IDE: *Integrated Development Enviroment* (Entorno de Desarrollo Integrado).

Internet: Red mundial de comunicación compuesta por miles de redes telefónicas e informáticas que se encuentran conectadas entre sí para transmitir información.

Iteraciones: Es un ciclo de desarrollo completo que genera como resultado una entrega de producto ejecutable (interna o externa).

R

Requerimiento: Condición o capacidad que tiene que ser alcanzada por un sistema o los componentes de un sistema para satisfacer un contrato o documento formalmente.

Requerimiento Funcional: Capacidades o condiciones que el sistema debe cumplir.

Requerimiento no Funcional: Propiedades o cualidades que el sistema debe tener.

Roles: Personas con privilegios de interactuar con el sistema solamente en la parte que se le es asignada.

M

Microsoft: Empresa multinacional estadounidense, fundada en 1975 por Bill Gates y Paul Allen. Dedicada al sector de la informática, con sede en Redmond, Washington, Estados Unidos.

O

ORM: Mapeo Objeto-Relacional, técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.

P

Poscondiciones: Requisitos cumplidos una vez que se realiza la acción.

Precondiciones: Requisitos que deben cumplirse antes de que suceda cualquier actividad.

Proceso de negocio: Ordenación lógicamente interrelacionada de tareas desarrolladas en tiempo y espacio (con comienzo y fin, con entradas y salidas definidas) y que se orienta al logro de un objetivo de negocio, generando un output de valor (total o parcial) para el cliente del proceso.

S

SGBD: Sistema gestor de base de datos.

Anexos

Anexo 1: Imágenes tomadas del "Manual de Usuario del Sistema de Cita Previa de DNI y Pasaporte electrónico"

Figura 1

The screenshot shows the 'SOLICITUD DE TRÁMITE PARA DNI' page. At the top, there is a header with the Spanish coat of arms, 'MINISTERIO DEL INTERIOR', and 'Cuerpo Nacional de Policía'. Navigation links include 'Ayuda', 'Mape Web', and 'Contactar'. A secondary navigation bar contains 'Selección de Trámite' and 'SALIR'. The main content area is titled 'SOLICITUD DE TRÁMITE PARA DNI' and contains three sections: 'Datos de la cita', 'Datos del Trámite', and 'Datos de Contacto'. The 'Datos de la cita' section shows 'Equipo: OFICINA DE DON BENITO (06259L6D1)', 'Fecha: Lunes, 16 de Enero de 2012.', and 'Hora: 09:00.', with a 'Volver' button. The 'Datos del Trámite' section shows 'Trámite: DNI.', 'Tipo de Expedición: Individual.', and 'Número de Expediciones: 1.'. The 'Datos de Contacto' section has a 'Teléfono Móvil:' field and a 'Confirmar' button. A footer bar at the bottom reads 'MINISTERIO DEL INTERIOR. Dirección General de la Policía. Cuerpo Nacional de Policía.'

Figura 2

The screenshot shows the 'CONFIRMACIÓN DE LA CITA' page. The header is identical to Figure 1. The secondary navigation bar includes 'Selección de Trámite', 'Consulta de Citas', 'Anulación de Citas', 'Imprimir en PDF', and 'SALIR'. The main content area is titled 'CONFIRMACIÓN DE LA CITA' and contains a confirmation message: 'Si ha rellenado el número de teléfono móvil, debe de haber recibido un mensaje corto confirmando la cita que nos ha solicitado.' Below this, it lists 'Los datos de su cita son:' followed by: 'Documento: 20000005X', 'Equipo: OFICINA DE DON BENITO (06259L6D1)', 'Fecha: Lunes, 16 de Enero de 2012.', 'Hora: 09:00.', 'Datos del Trámite: DNI (Individual).', 'Nº de Expediciones: 1.', 'Número de cita: D090001.', 'Documentación necesaria: [Documentación para la expedición del DNI.](#) [Alt + e]', 'Utilice este enlace si desea [anular la cita.](#) [Alt + c]', 'Si quiero puedo [imprimir su cita en formato PDF pulsando aquí.](#) [Alt + f]', and 'Si lo desea puede [solicitar otro trámite.](#) [Alt + t]'. On the right side, there is a box titled 'INFORMACIÓN ADICIONAL' containing links for 'DNELECTRÓNICO' and 'WEB POLICÍA'. A footer bar at the bottom reads 'MINISTERIO DEL INTERIOR. Dirección General de la Policía. Cuerpo Nacional de Policía.'

Figura 3

MINISTERIO DEL INTERIOR | Cuerpo Nacional de Policía | Ayuda | Mapa Web | Contactar

Selección de Trámite | Consulta de Citas | Anulación de Citas | SALIR

Usted se encuentra en: Inicio | Listado de Citas.

LISTADO DE CITAS. (CONSULTA)

A continuación se muestran las citas asociadas a su documento, seleccione cita.

Oficina	Agen- da	Trámite	Fecha	Hora	Estado
DON BENITO [Alt + d]	DNI	DNI	16/01/2012	09:00	No realizado
MADRID SANTA ENGRACÍA [Alt + d]	Pasaporte	Pasaporte	10/01/2012	09:20	No realizado

[Volver](#)

MINISTERIO DEL INTERIOR, Dirección General de la Policía.
Cuerpo Nacional de Policía.

Figura 4

MINISTERIO DEL INTERIOR | Cuerpo Nacional de Policía | Ayuda | Mapa Web | Contactar

Selección de Trámite | Consulta de Citas | Anulación de Citas | SALIR

ANULACIÓN DE TRÁMITE (DNI)

Datos de la cita

Equipo: DON BENITO (06259L5D1)

Fecha: Lunes, 16 de Enero de 2012.

Hora: 09:00.

Nº de Cita: D090001.

Datos del Trámite

Trámite : DNI.

Número de Expediciones : 1.

Datos de Contacto

Teléfono Móvil: .

[Anular Cita](#)

MINISTERIO DEL INTERIOR, Dirección General de la Policía.
Cuerpo Nacional de Policía.

Anexo 2: Imágenes tomadas del sistema "Concertar Cita Consular"

Figura 1

Gobierno de España | MINISTERIO DE ASUNTOS EXTERIORES Y DE COOPERACION

SuCITA.com Citas Consulares
Oficina Consular en Sildavia
lunes, 12 de enero de 2009 | ayuda | desconectar

Solicitud de Opción a la Nacionalidad Española

Ciudadano: _____ Día: _____ Hora: _____

Para acceder a la consulta de una cita previamente concertada introduzca sus datos de usuario y contraseña. Recuerde que los datos con asterisco (*) son obligatorios.

* Nombre: JUAN ANTONIO
 * Apellidos: EJEMPLO DE APELLIDO
 Teléfono contacto: 99 565 345 23
 Correo electrónico: juan.antonio@mail.com
Se recomienda cumplimentar un correo electrónico, al cual se enviará la información de las notificaciones que realice la oficina consular.
 * Nombre de usuario: juan.antonio@mail.com
6 caracteres mínimo. No se aceptan espacios ni caracteres especiales. Se recomienda utilizar el correo electrónico como nombre de usuario.
 * Contraseña: ●●●●●●
8 caracteres mínimo y al menos un número y una letra en mayúsculas.
 * Repetir Contraseña: ●●●●●●
 * Escriba el texto que ve en la imagen: 948150 [948150]

Paso 1 de 4 [siguiente]

Figura 2

Gobierno de España | MINISTERIO DE ASUNTOS EXTERIORES Y DE COOPERACION

SuCITA.com Citas Consulares
Oficina Consular en Sildavia
lunes, 12 de enero de 2009 | ayuda | desconectar

Solicitud de Opción a la Nacionalidad Española

Ciudadano: JUAN ANTONIO EJEMPLO DE APELLIDO Día: 27/01/2009 Hora: 09:40

Seleccione mes y día

dic		enero de 2009					feb
lun	mar	mie	jue	vie	sab	dom	
5	6	7	1	2	3	4	
12	13	14	15	16	17	18	
19	20	21	22	23	24	25	
26	27	28	29	30	31		

primer día con horas disponibles 19/01/2009

día disponible
 día ND disponible
 día seleccionado

Seleccione una de las horas disponibles del 27-01-2009

09:00 AM	09:00	09:08	09:16	09:32	09:40	09:48	09:56
10:00 AM	10:12	10:20	10:28	10:36	10:52		

[anterior] Paso 2 de 4 [siguiente]

Figura 3

Gobierno de España | **Ministerio de Asuntos Exteriores y de Cooperación**

SuCITA.com
Citas Consulares

Oficina Consular en Sildavia
lunes, 12 de enero de 2009 | ayuda | desconectar

Solicitud de Opción a la Nacionalidad Española

Ciudadano: JUAN ANTONIO EJEMPLO DE APELLIDO Día: 27-01-2009 Hora: 09:40

1. datos personales
2. concertar hora
3. datos de la solicitud
4. confirmar cita

Solicita optar a la nacionalidad española...

- Anexo I (por tener padre o madre que hubiese sido originariamente español)
- Anexo II (por ser nieto de quienes perdieron o tuvieron que renunciar a la nacionalidad española como consecuencia del exilio)
- Anexo III (sólo si usted es español no de origen)

[descargar formulario](#)

[anterior](#) Paso 3 de 4 [siguiente](#)

Figura 4

Gobierno de España | **Ministerio de Asuntos Exteriores y de Cooperación**

SuCITA.com
Citas Consulares

Oficina Consular en Sildavia
lunes, 12 de enero de 2009 | ayuda | desconectar

Solicitud de Opción a la Nacionalidad Española

Ciudadano: JUAN ANTONIO EJEMPLO DE APELLIDO Día: 27-01-2009 Hora: 09:40

1. datos personales
2. concertar hora
3. datos de la solicitud
4. confirmar cita

Para acceder a la consulta de una cita previamente concertada introduzca sus datos de usuario y contraseña. Recuerde que los datos introducidos deben coincidir con los de su documento de identidad; además, no se debe repetir una solicitud y este hecho podría ocasionar la denegación de la misma.

Estos son los datos de su cita. Si está de acuerdo con ellos, pulse el botón de **confirmar cita** y, a continuación, imprima su resguardo.

Nombre JUAN ANTONIO

Apellidos EJEMPLO DE APELLIDO

Día de la cita martes, 27 de enero de 2009

Hora de la cita 09:40

[confirmar cita](#)

[anterior](#) Paso 4 de 4

Anexo 3: Imágenes tomadas del "Manual de Usuario Paquete del Centro de Personalización e Impresión de Documentos (CPDI) versión 1.0"

Figura 1

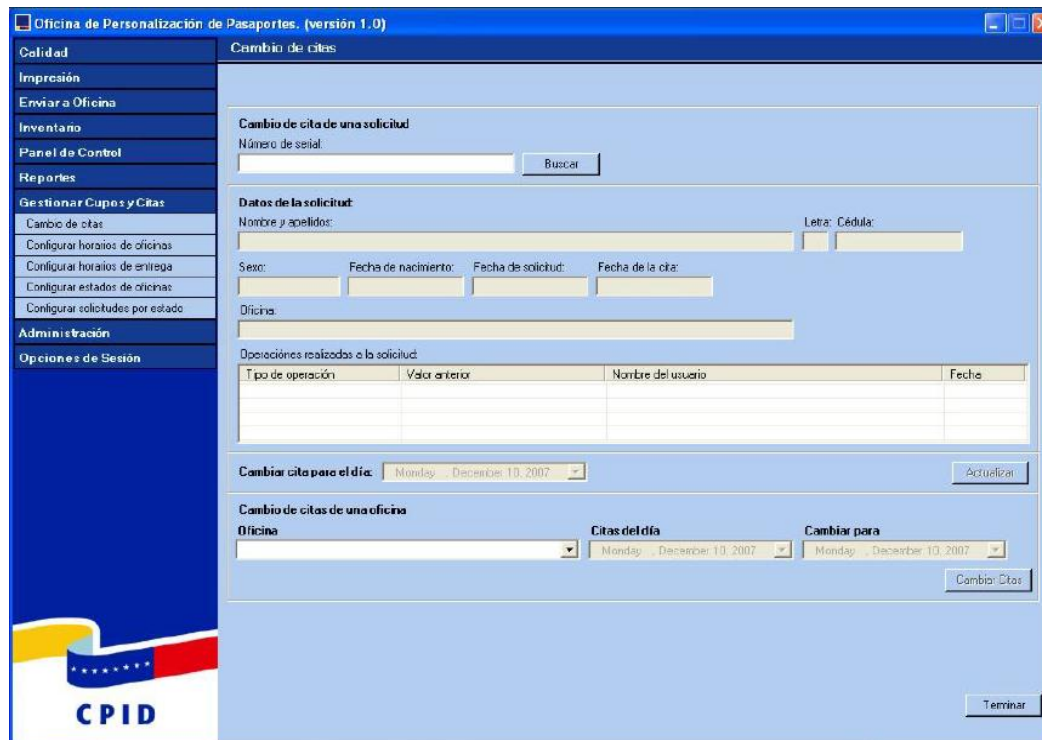


Figura 2

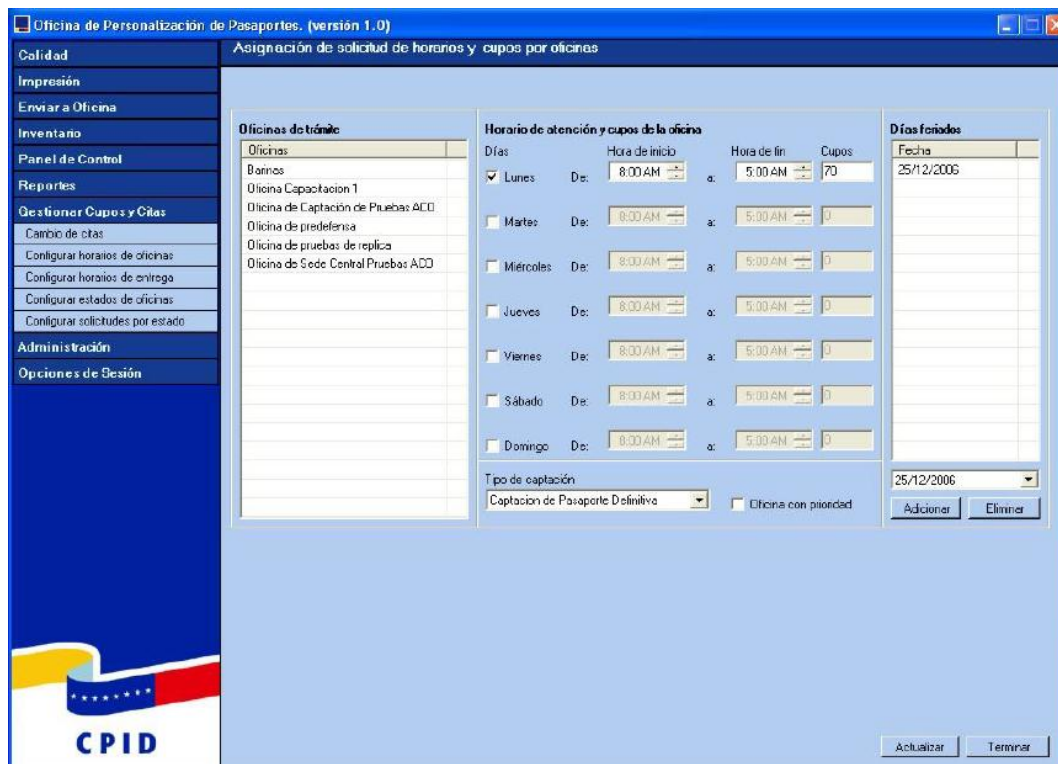


Figura 3

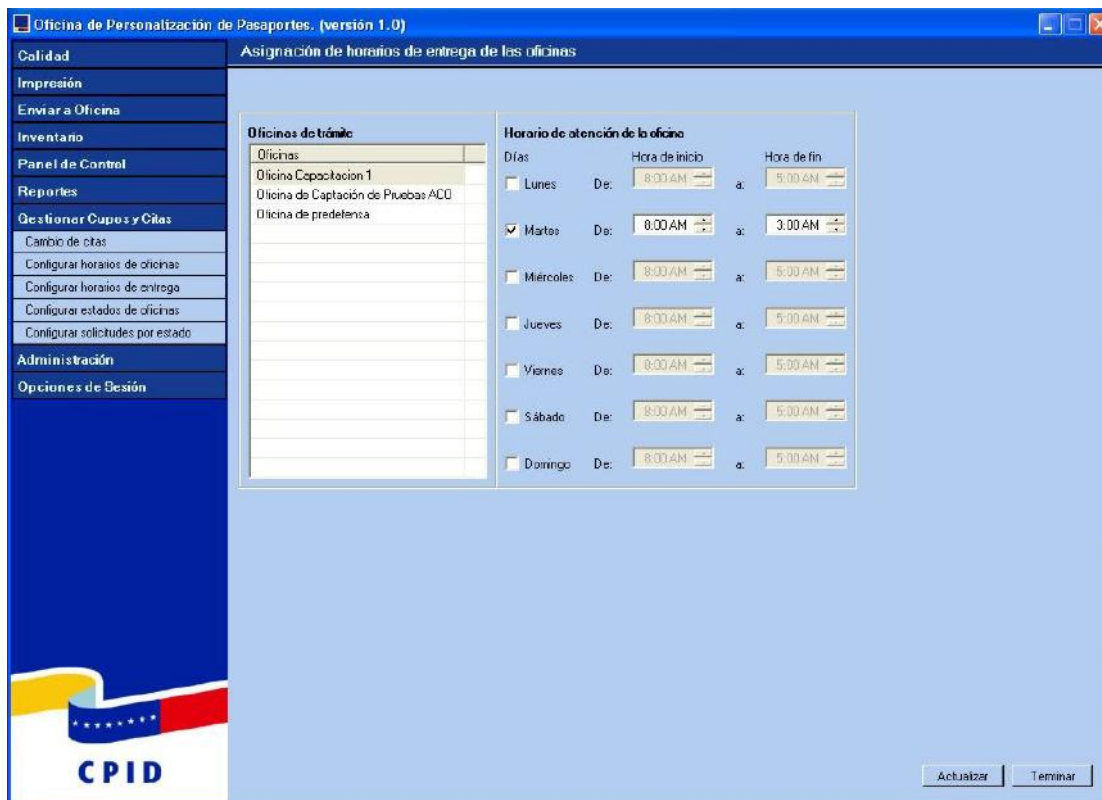


Figura 4

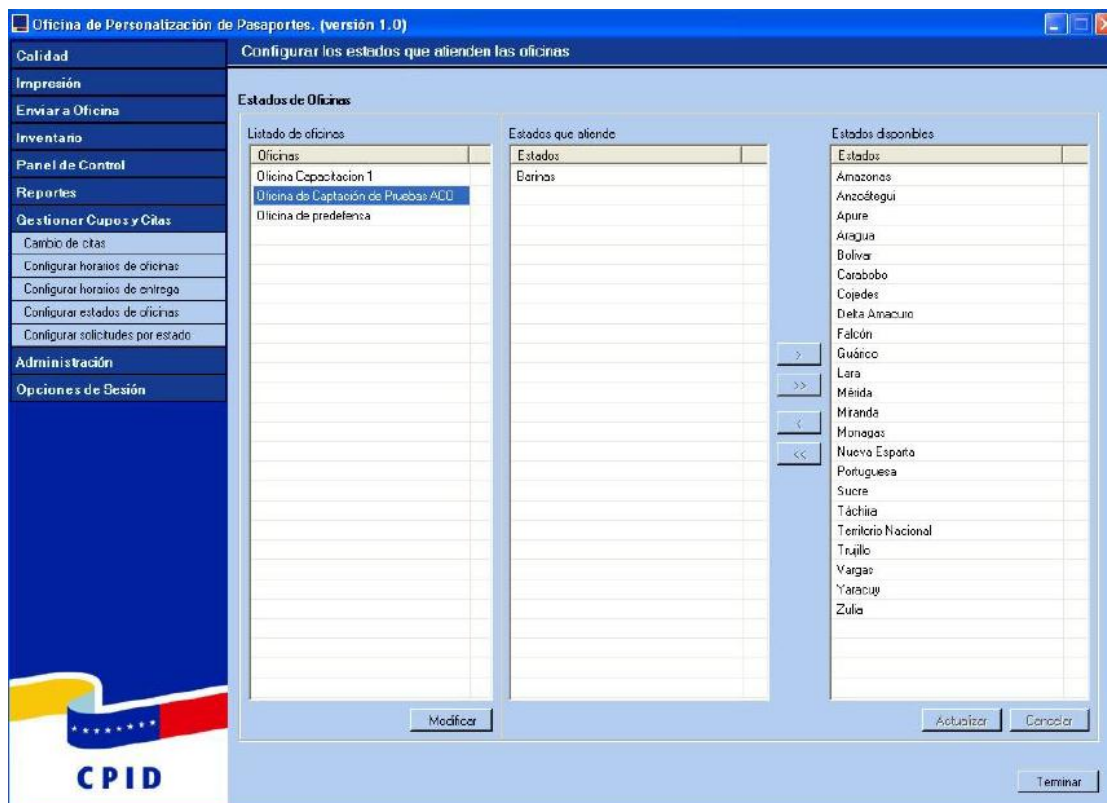
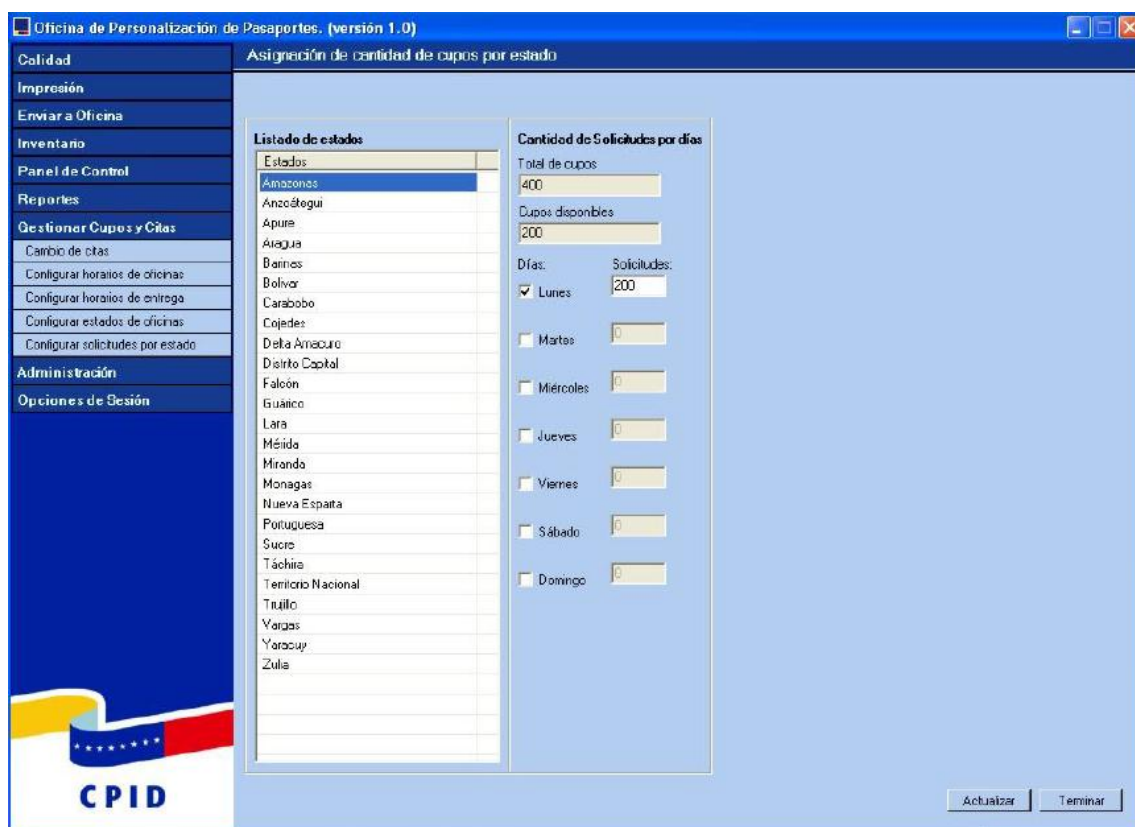


Figura 5



Anexo 4: Imágenes tomadas del sistema "Sistema de Atención Consular"

Figura 1



Figura 2

MINISTERIO DE RELACIONES EXTERIORES DE CHILE

Cerrar Sesión Inicio 23/05/2013 Versión : 1.0.0

Opciones

Información Usuario

Nombre : Renier, Pérez
 Consulado : La Habana, Seccion Consular
 Perfil : Ciudadano

[Modificar Mis Datos](#)
[Cambiar Clave](#)

Solicitudes

[Ingresar Solicitud](#)
[Seguimiento Solicitud](#)

Visas (*) Información Obligatoria

Fecha Solicitud: 23-05-2013

Lugar Inicio Trámite: La Habana, Seccion Consular
 Importante: Usted será atendido solamente por este consulado

I. Confirmar Datos del Solicitante

Cédula de Identidad
 Número Pasaporte
 DNI: 78111929363
 Nombre: Renier Pérez García
 Género: Masculino
 Domicilio de residencia *: Ave. 51 Edificio 2 Apto 8 % Autopista y Arroyo Arenas La Lisa
 Nacionalidad *: Cuba
 País de residencia *: Cuba
 Correo *: renier.perez@gmail.com
 Teléfono de residencia: +5353722280
 Teléfono Móvil

[Modificar Datos Solicitante](#)

II. Datos del Interesado

Relación con el solicitante *: Misma Persona

Primer Apellido *: Pérez

Segundo Apellido: García

Nombre del Interesado *: Renier

DNI: 78111929363

Género *: Masculino

Fecha de Nacimiento *: 19-11-1978 Formato (dd-mm-aaaa)

Figura 3

MINISTERIO DE RELACIONES EXTERIORES DE CHILE

Cerrar Sesión Inicio 23/05/2013 Versión : 1.0.0

Opciones

Información Usuario

Nombre : Renier, Pérez
 Consulado : La Habana, Seccion Consular
 Perfil : Ciudadano

[Modificar Mis Datos](#)
[Cambiar Clave](#)

Solicitudes

[Ingresar Solicitud](#)
[Seguimiento Solicitud](#)

Historial Solicitudes

Tipo de solicitud: Todos Estado: Todos

Número de solicitud:

[Buscar](#)

No hay registros que mostrar

Registros: 0 Página: 0 de 0

[Salir](#)

Figura 4

MINISTERIO DE RELACIONES EXTERIORES DE CHILE

Cerrar Sesión Inicio 23/05/2013 Versión : 1.0.0

Opciones

Información Usuario

Nombre : Renier, Pérez
 Consulado : La Habana, Sección Consular
 Perfil : Ciudadano

» [Modificar Mis Datos](#)
 » [Cambiar Clave](#)

Solicitudes

» [Ingresar Solicitud](#)
 » [Seguimiento Solicitud](#)

Modificar Mis Datos

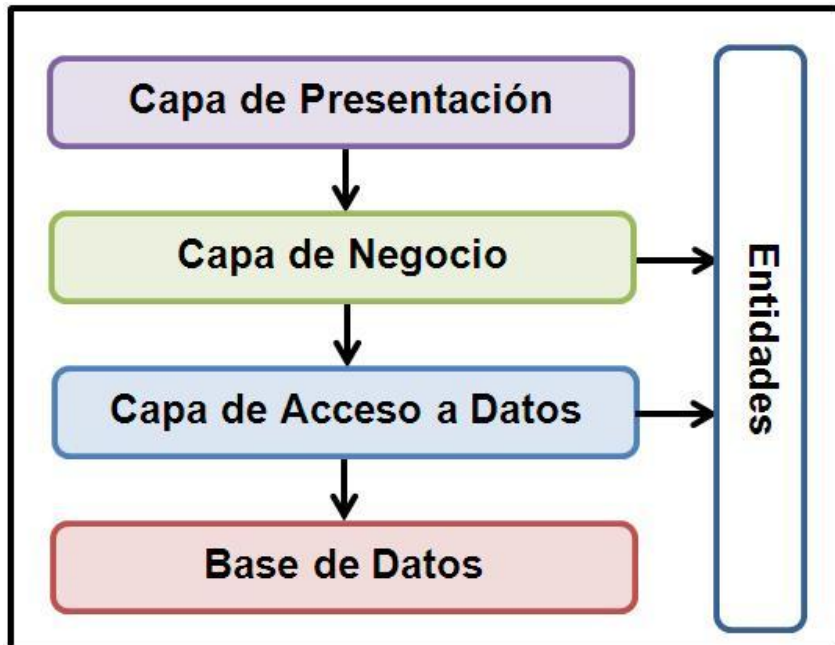
Seleccione modificar datos perfil y luego complete todos los campos obligatorios * del siguiente formulario

Cédula de Identidad	
Pasaporte	
Dni	78111929363
Nombres	Renier Pérez García
Género	Masculino
Domicilio Residencia *	Ave. 51 Edificio 2 Apto 8 % Autopista y Arroyo Arenas La Lisa
Nacionalidad *	Cuba
País Residencia *	Cuba
Correo *	renier.perez@gmail.com
Teléfono Residencia *	+5353722280
Teléfono Móvil	

(*) Campos Obligatorios

Modificar Datos Perfil Salir

Anexo 5: Vista lógica de la arquitectura de software



Anexo 6: Descripción de las funcionalidades

Tabla 1: Descripción de la funcionalidad "Administración"

Precondiciones	El usuario debe estar autenticado en el sistema con el rol administrador.
Funcionalidades tratadas	RF10, RF11.
Conceptos tratados	Cita, Solicitud, Trámite, Notificación.
Descripción básica	<p>1 El sistema muestra un menú lateral izquierdo con las siguientes opciones:</p> <ul style="list-style-type: none"> • Autenticación • Administración • Reportes • Configuración • Gestión <p>2 Luego de autenticarse el usuario, el sistema muestra por defecto la interfaz principal dónde se encuentran información general del sistema.</p> <p>3 Si el usuario selecciona la opción "Administración", el sistema despliega una lista con las siguientes pestañas:</p> <ul style="list-style-type: none"> • Buscar Cita • Administrar Citas • Resolver citas en espera <p>3.1 Si el usuario selecciona la opción "Buscar Cita" el sistema muestra la interfaz con el siguiente campo:</p> <ul style="list-style-type: none"> • Entre el número de la cita <p>3.1.1 Si el usuario entra el número de la cita, al dar click en el botón: Buscar, aparece la cita correspondiente a ese número.</p> <p>3.2 Si el usuario selecciona la opción "Administrar Cita", el sistema muestra una interfaz con un listado de las solicitudes que están registradas sin asignación de citas en el orden en que se encuentran en la cola, con las siguiente opciones:</p> <ul style="list-style-type: none"> • Eliminar Cita • Subir o bajar solicitud en la lista <p>3.2.1 Si el usuario selecciona la opción "Subir o bajar solicitud en la lista", el sistema reorganiza la cola y actualiza el listado.</p> <p>3.2.2 Si el usuario selecciona la opción "Eliminar Cita", el sistema elimina la cita de la lista de solicitudes y reorganiza la cola.</p> <p>3.3 Si el usuario selecciona la opción "Resolver citas en espera" el sistema ejecuta el algoritmo para generar todas las citas de la cola según los parámetros definidos en la configuración.</p>
Prototipos	

SISTEMA DE GESTIÓN DE CITAS

GESTIÓN DE CITAS Bienvenido: Nilberto Rodriguez Rojas

- > Autenticación
- ▼ Administración
 - Buscar cita**
 - Administrar citas
 - Resolver citas en espera
- > Reportes
- > Configuración
- > Gestión de citas

Buscar Cita

Entre el número de la cita:

Buscar

Nombre y apellidos: Elizabeth Lopez Rivera		Número de cita: 219
Fecha: 01/01/0001	Hora: 0:00	Tipo de documento: licencia

Resolver Cita

UCI, viernes, 31 de mayo de 2013

SISTEMA DE GESTIÓN DE CITAS

GESTIÓN DE CITAS Bienvenido: Nilberto Rodriguez Rojas

- > Autenticación
- ▼ Administración
 - Buscar cita
 - Administrar citas**
 - Resolver citas en espera
- > Reportes
- > Configuración
- > Gestión de citas

Administrar Citas

Listado de Citas

Nombre	1er Apellido	2do Apellido	Fecha	Hora	Estado	Documento	Número	
Aliana	Hernandez	Rodriguez	22/05/2013	12:00 PM	asignada	licencia	155	
Elizabeth	Lopez	Rivera	_//_/	_:__	en espera	licencia	219	
Elizabeth	Lopez	Rivera	03/06/2013	8:00 AM	asignada	CI	218	
Herbert	Tamayo	Valero	03/06/2013	8:00 AM	asignada	CI	215	
Herbert	Tamayo	Valero	03/06/2013	8:30 AM	asignada	solapín	216	

1 2 3 > >| Mostrando 1 a 5 de 11 registros

UCI, viernes, 31 de mayo de 2013

Descripción alterna:

Validaciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado en el sistema. • Los datos introducidos por el usuario deben estar correctos.
Postcondiciones	Queda asignada la cita o denegada la solicitud por el Sistema de Gestión de Citas.

Tabla 2: Descripción de la funcionalidad "Reportes"

Precondiciones	El usuario debe estar autenticado en el sistema con el rol administrador.
Funcionalidades tratadas	RF24.
Conceptos tratados	Cita, Solicitud, Trámite, Notificación.

Descripción básica	<ol style="list-style-type: none">1. El sistema muestra un menú lateral izquierdo con las siguientes opciones:<ul style="list-style-type: none">• Autenticación• Administración• Reportes• Configuración• Gestión2. Luego de autenticarse el usuario, el sistema muestra por defecto la interfaz principal dónde se encuentran información general del sistema.3. Si el usuario selecciona la opción “Reportes” el sistema despliega una lista con la siguiente pestaña:<ul style="list-style-type: none">• Cita para el Día3.1 Si el usuario selecciona la pestaña “Cita para el Día” el sistema muestra una interfaz con el listado de todas las citas del día con los siguientes datos:<ul style="list-style-type: none">• Estado• Fecha• Hora• Nombre• Tipo de trámite• Apellidos• Documento3.1.1 El sistema genera el reporte correspondiente, con una opción para ver los detalles de cada cita.
Prototipos	

Descripción alterna:

Validaciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado en el sistema. • Los datos introducidos por el usuario deben estar correctos.
Postcondiciones	Queda asignada la cita o denegada la solicitud por el Sistema de Gestión de Citas.

Tabla 3: Descripción de la funcionalidad "Configuración"

Precondiciones	El usuario debe estar autenticado en el sistema con el rol administrador.
Funcionalidades tratadas	RF13.
Conceptos tratados	Cita, Solicitud, Trámite, Notificación.
Descripción básica	<ol style="list-style-type: none"> 1. El sistema muestra un menú lateral izquierdo con las siguientes opciones: <ul style="list-style-type: none"> • Autenticación • Administración • Reportes • Configuración • Gestión 2. Luego de autenticarse el usuario, el sistema muestra por defecto la interfaz principal dónde se encuentran información general del sistema.

	<p>3. Si el usuario selecciona la opción “Configuración”, el sistema despliega una lista con las siguientes pestañas:</p> <ul style="list-style-type: none"> • Calendario • Tipo de cita automática • Tipo de Documento • Correo <p>3.1 Si el usuario selecciona la opción “Calendario”, el sistema muestra una interfaz con los siguientes campos para ser llenados:</p> <ul style="list-style-type: none"> • Duración aproximada de la cita • Cantidad máxima de citas por turno • Semanas de trabajo: el usuario tiene la opción de Agregar Semana, Configurar Semana o Eliminar Semana. • Días de trabajo semanales <ul style="list-style-type: none"> ➤ Lunes ➤ Martes ➤ Miércoles ➤ Jueves ➤ Viernes ➤ Sábado ➤ Domingo • Para cada día seleccionado se configura las sesiones de trabajo. <ul style="list-style-type: none"> ✓ Horario Mañana <ul style="list-style-type: none"> ➤ Horario de Inicio ➤ Horario Fin ➤ Laborable ✓ Horario Tarde <ul style="list-style-type: none"> ➤ Horario de Inicio ➤ Horario Fin ➤ Laborable <p>3.1.1 El usuario introduce los días no laborables, y da click en el botón “Agregar Día” o “Eliminar Fecha”.</p> <p>3.1.2 El usuario introduce los datos correspondientes y da click en el botón “Guardar configuración”.</p> <p>3.2 Si el usuario selecciona la opción “Tipo de cita automática”, el sistema muestra una interfaz con los siguientes campos para ser seleccionados:</p> <ul style="list-style-type: none"> • Generar solicitud en el instante
--	--

	<ul style="list-style-type: none">• Guardar solicitud en cola <p>3.2.1 El sistema registra la información seleccionada por el usuario al dar click en el botón: Aceptar.</p> <p>3.3 Si el usuario selecciona la opción “Tipo de Documento” el sistema muestra una interfaz con los documentos existentes y brinda la opción de poder seleccionar los que el sistema permitirá hacer solicitudes.</p> <p>3.4 Si el usuario selecciona la opción “Correo” el sistema muestra una interfaz con los datos correspondientes.</p> <ul style="list-style-type: none">• Servidor• Puerto• Usuario• Contraseña• Enviar correo de prueba
Prototipos	

SISTEMA DE GESTIÓN DE CITAS

 Bienvenido: Nilberto Rodríguez Rojas ✖ ?

GESTIÓN DE CITAS

- > Autenticación
- > Administración
- > Reportes
- ▼ Configuración
 - Calendario**
 - Tipo de cita automática
 - Tipo de documentos
 - Correo
- > Gestión de citas

Configuración Calendario

Duración aproximada de la cita

Cantidad máxima de citas por turno

Semanas de trabajo	
23 (02/06/2013 - 08/06/2013)	<input checked="" type="checkbox"/>
24 (09/06/2013 - 15/06/2013)	<input type="checkbox"/>
25 (16/06/2013 - 22/06/2013)	<input type="checkbox"/>
26 (23/06/2013 - 29/06/2013)	<input type="checkbox"/>

Agregar semana

Configurar Semana
Eliminar Semana

Días de trabajo semanales

lunes
 martes
 miércoles
 jueves
 viernes
 sábado
 domingo

Configuración general ▼

lunes ▼

martes ▼

miércoles ▼

jueves ▼

viernes ▼

sábado ▼

domingo ▼

Guardar configuración

Días no Laborables

Días no laborables	
31/05/13	<input type="checkbox"/>
24/05/13	<input type="checkbox"/>

Agregar día

Eliminar fecha

Guardar configuración

UCI, viernes, 31 de mayo de 2013

SISTEMA DE GESTIÓN DE CITAS

GESTIÓN DE CITAS Bienvenido: Nilberto Rodríguez Rojas

- > Autenticación
- > Administración
- > Reportes
- > Configuración
- > Gestión

Configuración tipo de citas automáticas

Generar cita en el instante
 Guardar solicitud en cola

[Aceptar](#)

UCI, jueves, 16 de mayo de 2013

SISTEMA DE GESTIÓN DE CITAS

GESTIÓN DE CITAS Bienvenido: Nilberto Rodríguez Rojas

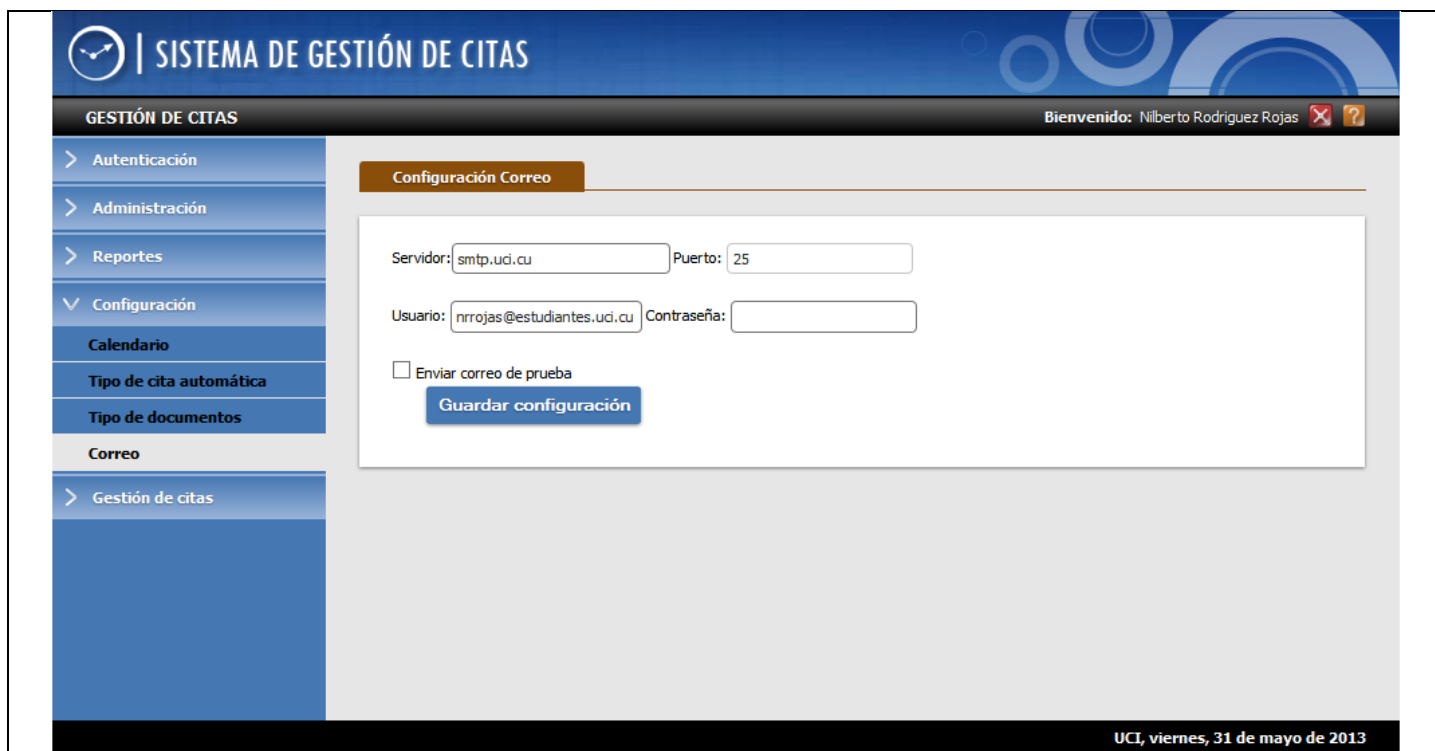
- > Autenticación
- > Administración
- > Reportes
- ▼ Configuración
 - Calendario
 - Tipo de cita automática
 - Tipo de documentos**
 - Correo
- > Gestión de citas

Tipos de Documentos

Tipos de documentos	
CI	<input checked="" type="checkbox"/>
licencia	<input checked="" type="checkbox"/>
solapín	<input checked="" type="checkbox"/>

[Aceptar](#)

UCI, viernes, 31 de mayo de 2013



Descripción alterna:

Validaciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado en el sistema. • Los datos introducidos por el usuario deben estar correctos.
Postcondiciones	Queda asignada la cita o denegada la solicitud por el Sistema de Gestión de Citas.

Anexo 7: Diagrama de clases

Figura 1

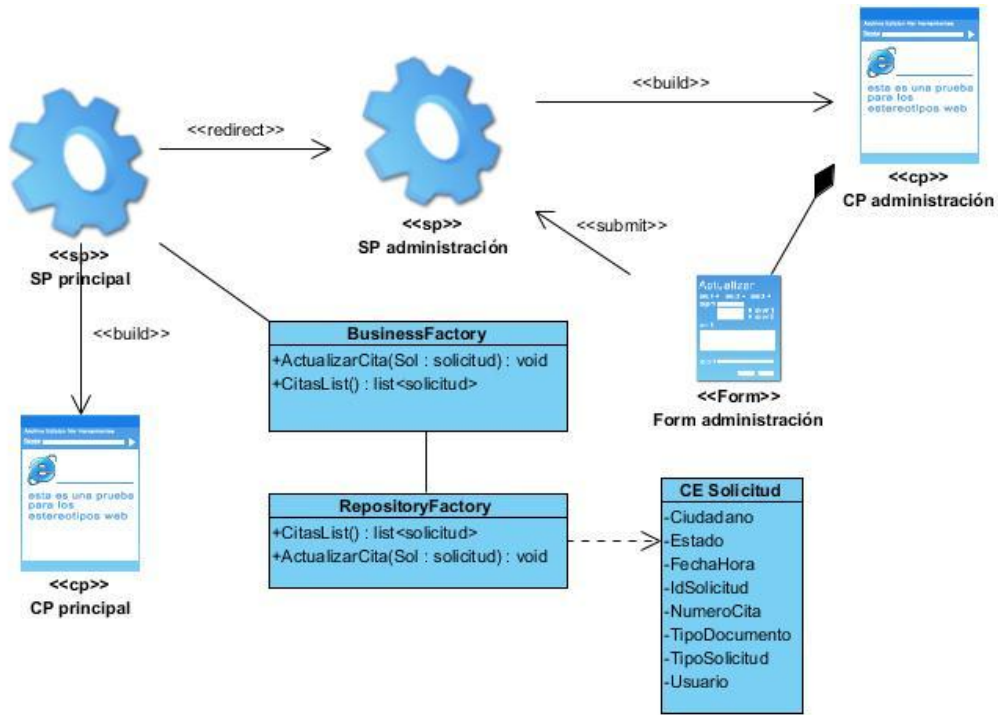


Figura 2

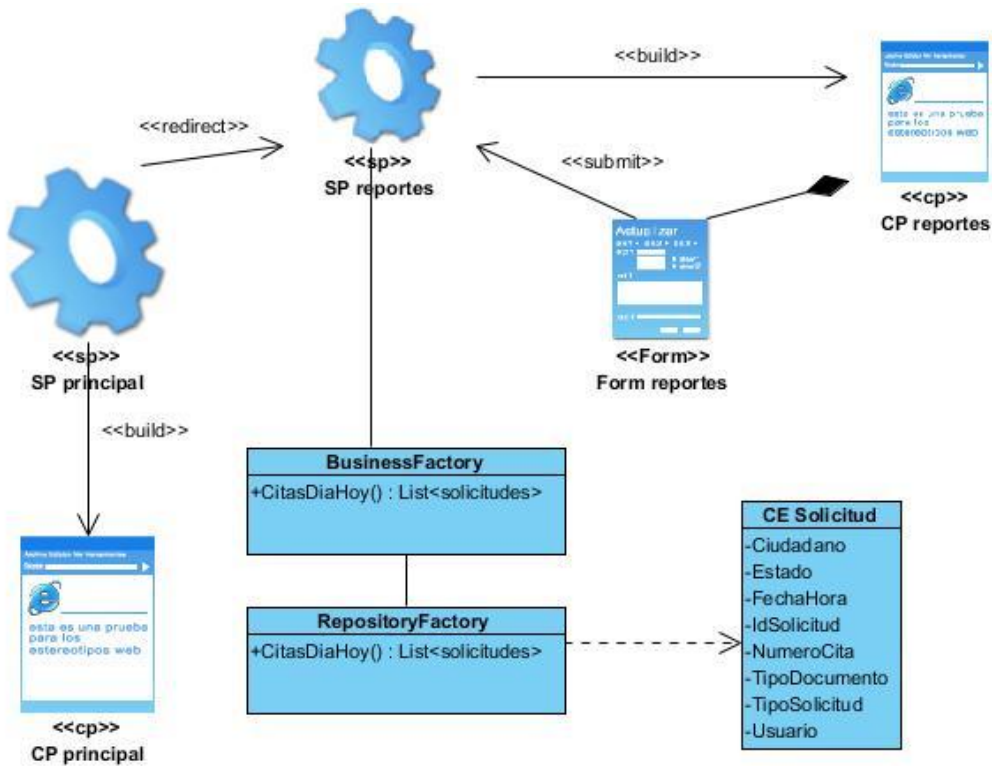
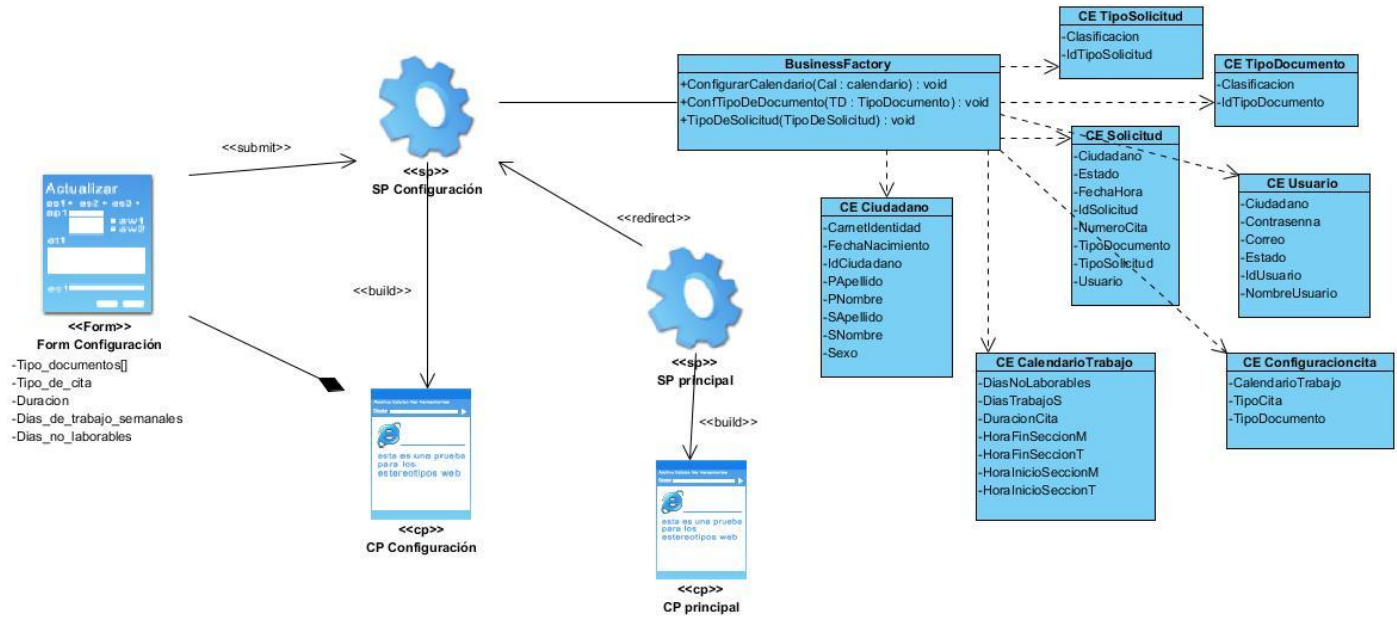


Figura 3



Anexo 8: Diagrama de secuencia

Figura 1

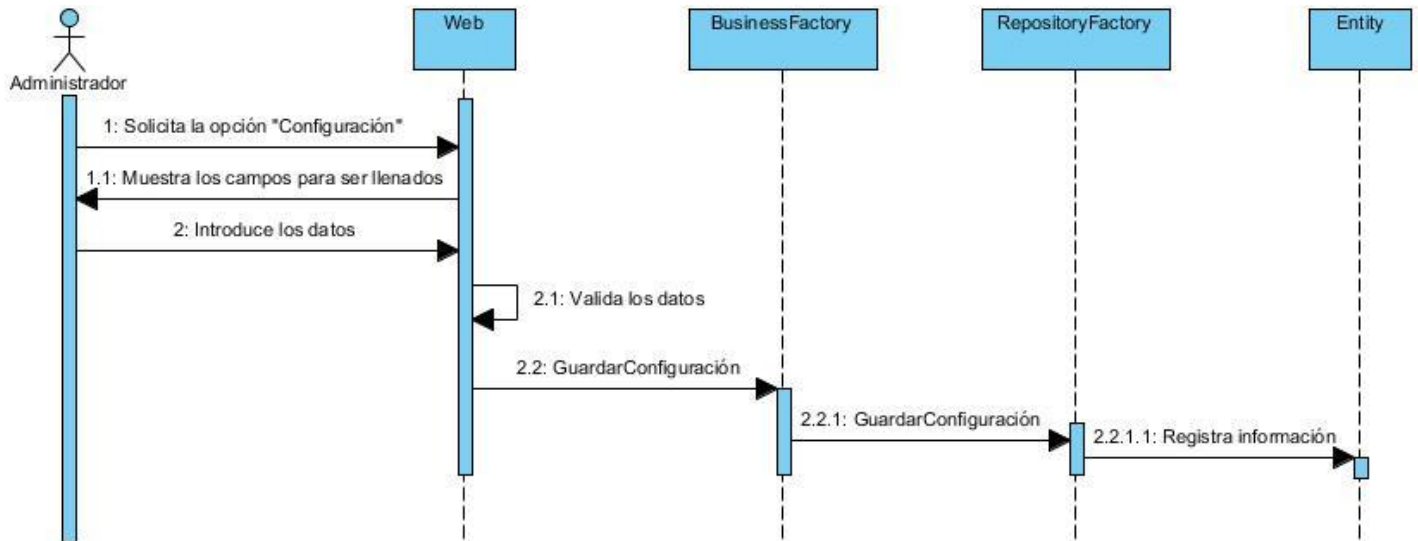


Figura 2

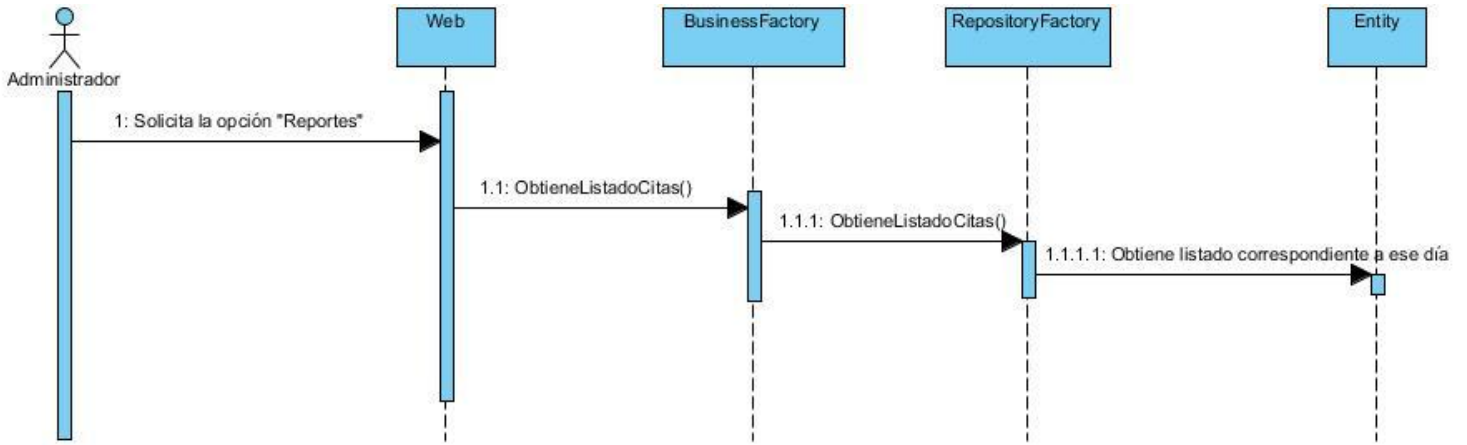
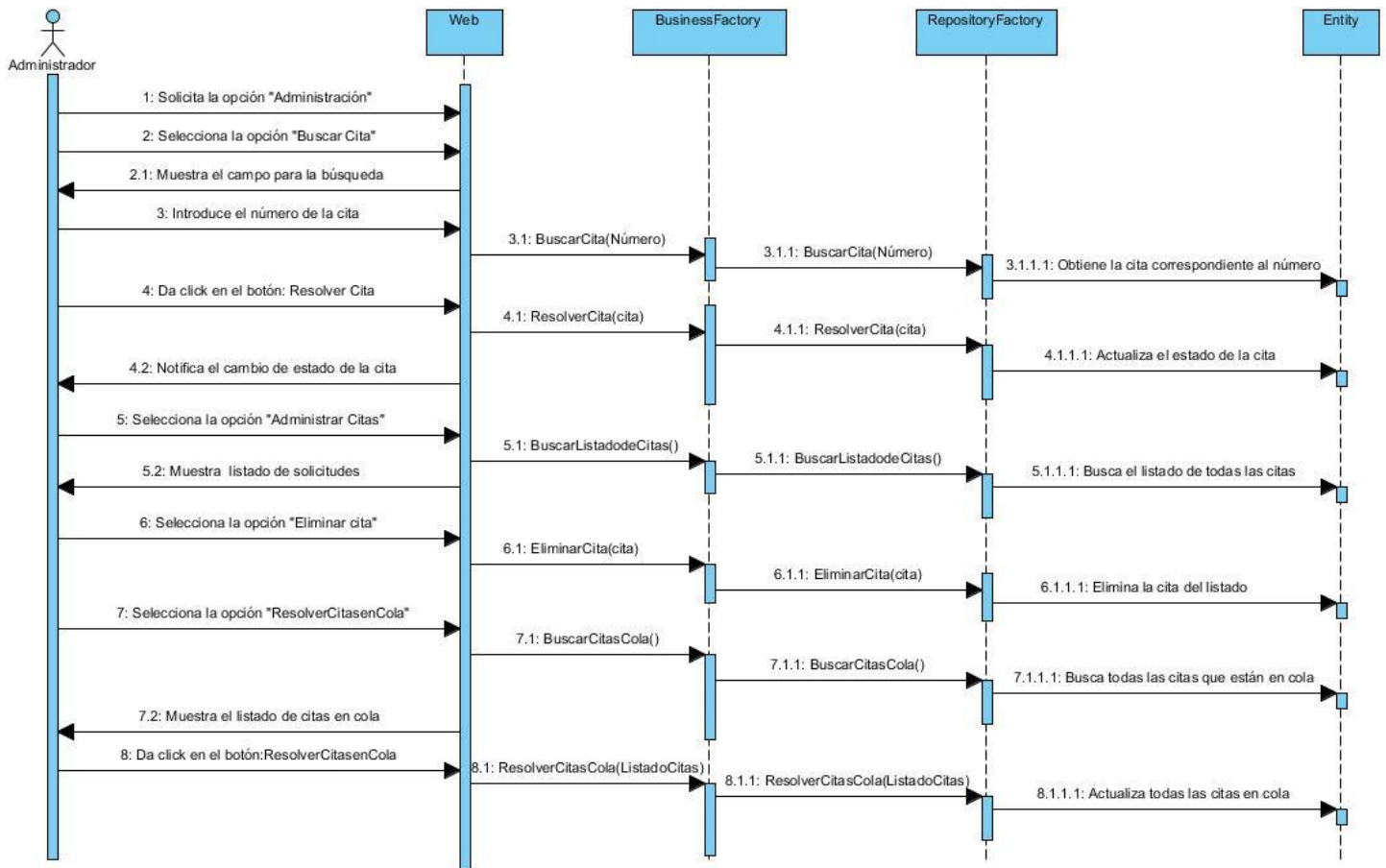


Figura 3



Anexo 9: Descripción de las entidades

Tabla dSolicitud: Entidad que guarda los datos de una solicitud

Atributos	Tipo de Datos	Nulo	Definición
id_solicitud	<i>uuid</i>	No	Identificador de la solicitud
fecha	<i>timestamp</i>	No	Fecha de la solicitud
hora	<i>timestamp</i>	No	Hora de la solicitud
estado	<i>char</i>	Si	Estado de la solicitud
id_tipo_solicitud (FK)	<i>serial</i>	No	Identificador del tipo de solicitud
id_ciudadano (FK)	<i>uuid</i>	No	Identificador del ciudadano

Tabla dUsuario: Entidad que guarda los datos del usuario

Atributos	Tipo de Datos	Nulo	Definición
id_ciudadano (FK)	<i>uuid</i>	No	Identificador del ciudadano
estado	<i>char</i>	Si	Estado del usuario
correo	<i>varchar</i>	No	Correo del usuario
contraseña	<i>varchar</i>	No	Contraseña del usuario
nombre_usuario	<i>varchar</i>	Si	Nombre del usuario
id_rol (FK)	<i>serial</i>	No	Identificador del rol

Tabla dDocumento_Identificacion: Entidad que guarda los datos de los documentos de identificación

Atributos	Tipo de Datos	Nulo	Definición
id_documento_identificacion	<i>uuid</i>	No	Identificador del documento
id_ciudadano (FK)	<i>uuid</i>	No	Identificador del ciudadano
id_tipo_documento (FK)	<i>serial</i>	No	Identificador del tipo de documento
id_estado_documento (FK)	<i>serial</i>	No	Identificador del estado del documento

Tabla dUsuario_nrol: Entidad encargada de asignarle un rol al usuario

Atributos	Tipo de Datos	Nulo	Definición
id_rol (FK)	<i>serial</i>	No	Identificador del documento
id_usuario (FK)	<i>uuid</i>	No	Identificador del ciudadano