

Universidad de las Ciencias Informáticas



Portal de Trámites de la República de Cuba

*Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas.*

Autores:

Suleivis Bravo Tamayo.

Tayler Daniel García Hernández.

Tutores:

Ing. Yanet Silva Fernández.

Ing. Manuel Alejandro Quert Gómez.

Ciudad de la Habana

Junio 2013



“La responsabilidad nuestra es luchar porque la calidad del producto que aquí se haga sea de las mejores y la mejor posible...”

Ernesto Che Guevara

Declaración de autoría

DECLARACIÓN DE AUTORÍA

Declaramos ser los únicos autores del trabajo de diploma titulado Portal de Trámites de la República de Cuba, y autorizamos a la Universidad de las Ciencias Informáticas (UCI) los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los _____ días del mes _____ del año 2013.

Firma del autor

Suleivis Bravo Tamayo

Firma del autor

Tayler Daniel García Hernández

Firma del tutor

Ing. Yanet Silva Fernández

Firma del tutor

Ing. Manuel Alejandro Quert Gómez

Datos de contacto

DATOS DE CONTACTOS

Suleivis Bravo Tamayo

Correo: stamayo@estudiantes.uci.cu

Universidad de las Ciencias Informáticas, Ciudad Habana, Cuba.

Taylor Daniel García Hernández

Correo: tdhernandez@estudiantes.uci.cu

Universidad de las Ciencias Informáticas, Ciudad Habana, Cuba.

Ing. Yanet Silva Fernández

Correo: ysfernandez@uci.cu

Universidad de las Ciencias Informáticas, Ciudad Habana, Cuba.

Ingeniera en Ciencias Informáticas. Recién Graduado en Adiestramiento con 1 año de experiencia. Graduada en la UCI en el 2011.

Ing. Manuel Alejandro Quert Gómez

Correo: maquert@uci.cu

Universidad de las Ciencias Informáticas, Ciudad Habana, Cuba.

Ingeniero en Ciencias Informáticas. Profesor Instructor con 3 años de experiencia. Graduado en la UCI en el 2009.

Dedicatoria

DEDICATORIA

Quisiera dedicarles este trabajo a todas aquellas personas que hicieron posible su realización. En especial a mis padres por estar siempre cuando los necesito, a mi novio por estar siempre a mi lado, a mi hermano por sus consejos de no dejarme caer por los nervios, a mi cuñada por demostrarme su apoyo en todo momento, a mi bella princesita Gaby por brindarme su carita feliz, a mis suegros por brindarme energía positiva todo el tiempo y a mis abuelitos postizos por su infinito amor y comprensión.

Suleivís

Dedico este logro de mi vida en primer lugar a toda mi familia en especial a mi padrastro, mi hermano, mi papá y a lo más grande que tengo en mi vida, MI Mamá. A mi nueva familia por dejarme entrar en ella como un miembro más, en especial a mi Bebé. A todos mis amigos dentro y fuera de la universidad.

Tayler

Agradecimientos

AGRADECIMIENTOS



Agradecemos a todos los que de una forma u otra formaron parte de esta obra que se ha convertido en el fruto de cinco años de esfuerzo y dedicación. Nos sentimos eternamente agradecidos con las personas a quienes acudimos para obtener información y que desinteresadamente nos brindaron un poco de su ayuda.

Agradecimientos de Suleivis Bravo Tamayo:

Son sin duda estas líneas las más difíciles de escribir para mí, pues han sido muchas las personas que de una forma u otra me brindaron su mano en los momentos difíciles e hicieron posible que este día llegase.

Le quiero agradecer, a los obstáculos que han pasado por mi vida. Ellos son los que me han hecho fuerte, perseverante y finalmente, triunfadora.

A todas las personas que una vez me preguntaron “y cómo va, la tesis?”.

A mis tutores Yanet (mijita) por soportar mis malcriadeces y complacerme y Manuel por estar ahí cuando me hizo falta, a Liudnet por dedicarme su complicado tiempo en algunas ocasiones.

A mi compañero de tesis Tayler por darme muchos dolores de cabeza, y culminar la realización de este trabajo de diploma.

A mis compañeros de apartamento, por la paciencia que han tenido conmigo, el Yabo, Marquito y Ernesto.

A mis niñas del 92 105, chicas gracias por apoyarme y estar ahí cuando las necesitaba. Por compartir mis alegrías y sobre todo mis tristezas. En especial Lislien, por ser mi madre en la escuela (gracias por conocerte); a Lilibet por demostrarme carácter en la vida, a Alina por ser tan dulce y comprensiva, a Arianna por ser una parte tierna en mi vida y ayudarme a seguir mis sueños. A Lillian, Madonna, Yamílka, Amal, Yadíra mi Peguí, y a Yensy.

A mis amistades del edificio 97 y algunos infiltrados. A Odalys por compartir noches de novelas, a Yanet por ser una buena amiga, a Yuned, al Potín, al Calvo, a Javier, Avilés, Marinas, a

Agradecimientos

Panchí, Eudel por sus charlas instructivas, a Hack. En fin a todos aquellos de una forma u otra estuvieron ahí compartiendo conmigo.

A los compañeros del proyecto que me apoyaron en todo momento en especial Alíana, Anita, Lachí, Renier, Daymarís, Liliét, Karel, Diorge entre otros.

A mis vecinos queridos Olguita, Nidia, Carly y el Chino.

A mis amigas de la infancia Yaines, Yusel, Yanela, Yirka, por enseñarme a crecer. Y en especial a Naty mi hermanita chiquita, un beso para ti te quiero muchísimo.

A mis tíos y primos por estar siempre presentes y con sus lecciones me enseñaron a mirar al futuro y me vieron siempre en el lugar de una estudiante graduada de la universidad.

A mis suegros por su apoyo incondicional, por aceptarme tal y como soy, por ser tan comprensivos y quererme como una hija. Y a mis abuelitos por confiar en mí y darme mucho apoyo para seguir adelante en todo momento y aceptarme como una nieta más.

A mis abuelas que aunque ya no estén a mi lado siento que día a día me dicen que siga adelante.

A mi cuñada por ser mi amiga y quererme como ella solo lo sabe hacer y ser tan dulce... por representar la hermana mayor que no tuve.

A mi hermano por sus consejos de no dejarme caer por los nervios, por demostrarme carácter ante las situaciones difíciles y por enseñarme a ser testaruda.

A mi novio por ser mi motor impulsor ante las dificultades y hacerme madurar. Por ayudarme a reconocer mis errores y nunca dejarme fallar. Por ser tan testarudo y sensible a la vez y por quererme como lo hace. Para ti amor un beso bien grande, gracias por ser la luz que alumbra mi camino. Te amo.

A mi papito lindo que lo adoro con la vida, para ti es este gran sueño, aquí estamos en la recta final. Quiero que sepas que eres el hombre que más amo en la vida, gracias a ti soy la mujer que soy ahora con mis defectos y virtudes, todo te lo debo a ti y mil gracias por apoyarme y defenderme en todo momento. Te adoro.

A mi mamuchi, no sé qué palabras decir ante la mujer que me dio la vida, sin ella no estuviese aquí. Gracias por hacer de mí lo que soy actualmente, por batallar conmigo en las buenas y en las malas, por aconsejarme y dármelo todo. Por enseñarme a ser persona en la vida, por tu apoyo

Agradecimientos

incondicional. Perdóname mamita por los malos ratos, y quiero que sepas que te amo con todo mi corazón y que nunca te voy a fallar. Te amo.

Agradecimientos de Tayler Daniel García Hernández:

A los tutores que de una forma u otra nos brindaron su apoyo y ayuda incondicional.

A mi compañera de tesis, Suly, por confiar en mí y por el esfuerzo realizado en el desarrollo de este trabajo de diploma.

A los drupaleros que me brindaron su ayuda en el momento que la necesité en especial a José Miguel.

A todos los compañeros que he tenido durante estos 5 años de estudios.

A los profesores que contribuyeron en mi formación como profesional.

Al piquete del proyecto, el Tuty, Naldo, Norky y otros.

A todos mis compañeros de cuarto que hemos formado una gran familia.

A todos mis amigos de la universidad que los quiero como hermanos, sin mencionar nombres pues no quiero que se quede alguno sin nombrar.

A la gente de mi barrio, de mi pueblo, Migue, Eduardito, Eriel, Arley, Yoán, a mis amigos de la Juan González.

A mi nueva familia, mi suegra, mis otros abuelos y tíos por aceptarme como un hijo.

A mi princesita linda, mi Bebé, te amo mucho.

A toda mi familia que les debo quien soy, mis primos, tíos, abuelos, en especial a mi abuela Enequina y mi abuelo Mengo, que ya no se encuentran entre nosotros y este era el gran sueño de ellos, vernos a mi hermano y a mí, graduados, y desde alguna parte de este mundo nos están viendo y sé que están muy orgullosos. A mis dos hermanitas que las quiero mucho. A mi padrastro por siempre estar halándome las orejas y encaminándome, para ser una mejor persona en la vida. A mi papá por ser tan recto. A mi hermano, por ser además un amigo, un maestro, porque es la persona que siempre he querido ser, es mi meta y por último y más importante de mi existencia, MI MADRE por ser todo lo que necesito en esta vida, porque eres lo más grande y especial que existe para mí, porque te debo todo lo que tengo y todo lo que soy, porque eres mi todo, porque te amo.

Resumen

RESUMEN

Con el uso de las Tecnologías de la Información y la Comunicación (TIC) cambia radicalmente la manera en que los gobiernos administran y ejecutan sus procesos internos, surgiendo la posibilidad de incrementar los canales de comunicación con los ciudadanos. De esta manera se vislumbra al gobierno electrónico como un medio para que los gobiernos modernicen sus procesos y mejoren la interacción con los ciudadanos. El gobierno cubano ha ido dando pasos agigantados en función de lograr esto, y un ejemplo de ello lo es la informatización de los procesos que se llevan a cabo en las direcciones del Ministerio del Interior (MININT).

La siguiente investigación tiene como objetivo desarrollar un portal de trámites que posibilite el control del flujo de personas en las Oficinas del Carné de Identidad de la República de Cuba, obtener el estado de opiniones de las personas respecto a los temas relacionados con la identificación de los ciudadanos, además de permitir al ciudadano solicitar y consultar un trámite utilizando para ello la información del Sistema Único de Identificación Nacional (SUIN). La comunicación con este sistema se realizó utilizando un servicio web, adaptándose a la arquitectura definida para los servicios comunes del SUIN.

Para implementar el portal de trámites se decidió utilizar el Sistema Gestor de Contenidos Drupal en su versión 7.0, el Sistema Gestor de Base de Datos Oracle 11g y como servidor web Xampp; mientras que para la implementación del servicio web de intercambio con el SUIN se utilizó las tecnologías *Windows Communication Foundation* y *Entity Framework*.

Palabras claves: Drupal, portal, servicio web, Sistema de Gestor de Contenidos.

Índice

ÍNDICE

Introducción	1
Capítulo 1 Fundamentación Teórica	6
1.1 Portal web.....	6
1.2 Trámite	6
1.2.1 Identificación.....	7
1.2.2 Portal para trámites de identificación en Cuba	7
1.2.3 Servicio administrativo de Identificación, Migración y Extranjería “SAIME”	8
1.2.4 Portal de Servicios al Ciudadano y Empresas. República de Perú.....	8
1.2.5 Portal Ciudadano del Gobierno del Estado de México	9
1.2.6 Portal de Trámite del Gobierno de Santa Fé	9
1.2.7 Conclusiones	10
1.3 Arquitectura para servicios web del SUIN.	10
1.4 Tecnologías web	11
1.4.1 Sistema de Gestión de Contenidos (CMS).....	11
1.4.2 Framework.....	12
1.4.3 Conclusiones	13
1.5 Sistemas de Gestión de Contenidos	13
1.5.1 Joomla	13
1.5.2 Drupal	14
1.5.3 WordPress.....	15
1.5.4 Conclusiones	15
1.6 Sistemas Gestores de Base de Datos (SGBD)	15
1.6.1 Oracle.....	16
1.7 Servidor Web	16
1.7.1 XAMPP	17
1.7.2 WAMP	17
1.7.3 Internet Information Service (IIS)	18
1.7.4 Conclusiones	18
1.8 Metodología de desarrollo	19

Índice

1.9	Lenguajes de programación.....	21
1.9.1	PHP	21
1.9.2	JavaScript.....	22
1.9.3	CSS.....	22
1.9.4	C#.....	23
1.9.5	Linq.....	23
1.10	Lenguaje de modelado: UML.....	24
1.11	Herramientas y Tecnologías.....	24
1.11.1	Herramienta de modelado: Altova UModel 2010.....	24
1.11.2	Herramienta de Desarrollo: Microsoft Visual Studio 2010.....	25
1.11.3	Windows Communication Foundation.....	25
1.11.4	ADO.NET Entity Framework	26
1.12	Conclusiones Parciales.....	27
Capítulo 2 Análisis y Diseño.....		28
2.1	Presentación de la Solución.....	28
2.2	Modelo de Dominio.....	28
2.2.1	Descripción del Modelo de Dominio.....	29
2.3	Requerimientos del sistema.....	30
2.3.1	Descripción de los roles.....	30
2.3.2	Definición de requisitos funcionales.....	31
2.3.3	Especificación de requisitos funcionales.....	32
2.4	Arquitectura de la solución.....	39
2.5	Patrones de diseño en drupal.....	40
2.6	Diagrama de Clases	42
2.7	Modelo de Datos.....	43
2.7.1	Descripción de entidades.....	44
2.8	Conclusiones Parciales.....	45
Capítulo 3 Implementación y Pruebas.....		46
3.1	Estándares de codificación	46
3.2	Tratamiento de errores.....	49

Índice

3.3	Implementación	50
3.3.1	Diagrama de componentes	50
3.4	Diagrama de despliegue	52
3.5	Pruebas	53
3.5.1	Pruebas unitarias	53
3.5.2	Pruebas de Aceptación	55
3.5.3	Pruebas de Caja Negra	55
3.5.3.1	Diseño de casos de pruebas.....	56
3.5.4	Resultado de las Pruebas	58
3.6	Beneficios del portal.....	58
3.7	Conclusiones Parciales.....	59
	Conclusiones	60
	Recomendaciones	61
	Referencia Bibliográfica	62
	Glosario de Términos.....	66
	Anexos.....	67
Anexo I.	Especificación de Requisitos Funcionales	67
Anexo II.	Descripción de Entidades Modelo de Datos.....	91
Anexo III.	Prueba de Aceptación.....	93

Índice de Figuras

ÍNDICE DE FIGURAS

Ilustración 1 Estructura de los servicios	11
Ilustración 2 Fases de MSF for CMMI. Fuente: MSDN.....	21
Ilustración 3 Modelo de Dominio	29
Ilustración 4 Arquitectura de Drupal	40
Ilustración 5 Diagrama de Clases	42
Ilustración 6 Modelo de Datos.....	44
Ilustración 7 Estándar de Codificación: Indentación	47
Ilustración 8 Estándar de Codificación: Etiquetas de apertura y cierre de PHP	47
Ilustración 9 Estándar de Codificación: Punto y Coma.....	47
Ilustración 10 Estándar de Codificación: Comillas.....	48
Ilustración 11 Estándar de Codificación: Estructuras de Control	48
Ilustración 12 Estándar de Codificación: Funciones	49
Ilustración 13 Estándar de Codificación: Arrays	49
Ilustración 14 Tratamiento de errores: Consultar Estado del Trámite	50
Ilustración 15 Diagrama de Componentes	51
Ilustración 16 Diagrama de Despliegue.....	52
Ilustración 17 Prueba unitaria de la funcionalidad Solicitar Trámites del servicio web	54
Ilustración 18 Resultado de la prueba unitaria de la funcionalidad Solicitar Trámites	55
Ilustración 19 Resultados de las No Conformidades por Iteraciones.....	58
Ilustración 20 Prueba de Aceptación	93

Índice de Tablas

ÍNDICE DE TABLAS

Tabla 1 Descripción de roles	31
Tabla 2 Definición de los Requisitos Funcionales del Portal de Trámites.....	32
Tabla 3 Definición de los Requisitos Funcionales del servicio web ExchangePortalService	32
Tabla 4 Descripción del requisito del portal “Solicitar Trámites”	34
Tabla 5 Descripción de la Entidad DESTADO_TRAMITE_PORTAL	44
Tabla 6 Diseño de caso de pruebas "Solicitar Trámites"	58
Tabla 7 Descripción del requisito del portal “Registrar Usuario”	68
Tabla 8 Descripción del requisito del portal “Autenticar Usuario”	69
Tabla 9 Descripción del requisito del portal “Obtener requisitos para la realización de trámites”	70
Tabla 10 Descripción del requisito del portal “Consultar el estado de los trámites”	71
Tabla 11 Descripción del requisito del portal “Mostrar información de las oficinas”	73
Tabla 12 Descripción del requisito del portal “Cancelar solicitud del trámite”	74
Tabla 13 Descripción del requisito del portal “Publicar encuesta”	76
Tabla 14 Descripción del requisito del portal “Publicar quejas y sugerencias”	78
Tabla 15 Descripción del requisito del portal “Gestionar información del portal”	81
Tabla 16 Descripción del requisito del portal “Gestionar Usuario”	83
Tabla 17 Descripción del requisito del portal “Generar reporte”	84
Tabla 18 Descripción del requisito del servicio web “Obtener el estado de un trámite”	85
Tabla 19 Descripción del requisito del servicio web “Administrar turno”	87
Tabla 20 Descripción del requisito del servicio web “Obtener las solicitudes de una oficina”	89
Tabla 21 Descripción del requisito del servicio web “Obtener las solicitudes de una persona determinada”	90
Tabla 22 Descripción de la Entidad DPERSONA_TRAMITE_PORTAL	91
Tabla 24 Descripción de la Entidad DRESERVACION_CITAS	91
Tabla 25 Descripción de la Entidad DSOLICITUD_TRAMITE_PORTAL.....	92
Tabla 26 Descripción de la Entidad NCONFIG_TRAMITE_PORTAL.....	92
Tabla 27 Descripción de la Entidad NDIA_SEMANA.....	93

Introducción

INTRODUCCIÓN

En la actualidad muchas organizaciones a nivel mundial utilizan los portales web para brindar información y servicios, esta práctica se ha hecho tan común, que incluso las instituciones que manejan la identificación de ciudadanos en muchos países se han movido en esta dirección y vienen incrementando sus actividades a través de *Internet*¹. Esto ha sido posible por los avances de las Tecnologías de la Información y de las Comunicaciones (TICs).[1]

Una de estas tecnologías es el portal web que ofrece servicios de valor añadido como enlaces, personalización de la información, documentos, aplicaciones, *e-mail*², foros para compartir opiniones, y buscadores.[2] Está dirigido a tratar informaciones específicas de un tema en particular y que el usuario pueda satisfacer todas sus necesidades, incentivándoles a utilizarlo continuamente.[3]

En Cuba el uso de las tecnologías web para informatizar varios servicios de identificación, no ha alcanzado el auge que tiene en otros países desarrollados, pero ha venido dando pasos agigantados en función de lograr un Gobierno en Línea³.

La Dirección de Identificación y Registro (DIR) del Ministerio del Interior (MININT) en Cuba, que desde 1997 gestiona los procesos de identificación, no está excepta de estos cambios, y por ello decide informatizar todos sus procesos. Ya en el 2008 en conjunto con la Universidad de las Ciencias Informáticas (UCI), desarrolla el Sistema Único de Identificación Nacional (SUIN), el cual está siendo desplegado por todo el país, garantizando digitalmente la inscripción e identificación de las personas, con procesos y documentos seguros, generando la base de datos única de identificación de la población como premisa indispensable para el Gobierno en Línea.

Este sistema, a pesar de sus beneficios, no cuenta con la capacidad de suplir las necesidades actuales que tienen los ciudadanos para documentarse, además de solicitar y supervisar sus trámites. Hoy en día si un ciudadano desea solicitar algún documento de identificación debe presentarse personalmente a la oficina. Se hace engorroso para la población la solicitud y

¹Conjunto descentralizado de redes de comunicación interconectadas, usando una familia de protocolos.

²Correo electrónico, es un servicio de red que permite a los usuarios enviar y recibir mensajes y archivos rápidamente.

³ Uso de las tecnologías de la información y el conocimiento en los procesos internos de gobierno.

Introducción

obtención de documentos de identificación debido al cúmulo de personas que se presentan en un mismo día en las oficinas, puesto que no existe un mecanismo que controle la organización de los turnos. Además los ciudadanos actualmente no tienen como realizar consultas o quejas en línea a especialistas que le ayuden a informarse del tema.

Dada la necesidad de crear un portal que tenga la información de los trámites y regulaciones actualizada, que permita al ciudadano solicitar cualquier documento de identificación a través de la *Intranet*⁴ o *Internet*, realizar un seguimiento del estado del trámite, así como realizar quejas o sugerencias, se plantea el siguiente **problema científico**: ¿Cómo controlar el flujo de personas en las Oficinas de Carné de Identidad y obtener el estado de opinión de la población respecto a los temas relacionados con la identificación de los ciudadanos de la República de Cuba?

A partir del problema antes planteado se definió como **objeto de estudio** los portales oficiales para trámites de identificación.

Para dar solución al problema planteado se propone el siguiente **objetivo general**, desarrollar un portal que permita controlar el flujo de personas y obtener el estado de opinión de los ciudadanos contribuyendo a mejorar la calidad en los servicios que se prestan en las Oficinas de Carné de Identidad de la República de Cuba.

Para dar cumplimiento a los objetivos generales se trazaron como **objetivos específicos**:

- Caracterizar portales de trámites que se utilicen en el mundo.
- Definir las herramientas a utilizar para la implementación del portal.
- Caracterizar la metodología y las herramientas definidas por el proyecto.
- Definir especificación de requerimientos funcionales y no funcionales.
- Diseñar la solución a implementar.
- Implementar la solución.
- Validar la solución implementada.

Como **campo de acción** los portales oficiales para trámites de identificación en la República de Cuba.

⁴ Una red de ordenadores privados que utiliza tecnología *Internet*.

Introducción

Para dar cumplimiento a los objetivos específicos se propone las siguientes **tareas de la investigación**:

- Reunión con el cliente para conocer el contenido referente al producto.
- Identificación de las vías existentes para la solicitud de trámites desde portales oficiales.
- Búsqueda de opinión profesional de diseñadores con experiencia en la elaboración de herramientas para portales.
- Identificación de un CMS o *Framework* para el desarrollo de la solución.
- Descripción de las herramientas y tecnologías a utilizar para el desarrollo de la solución.
- Caracterización de la metodología de desarrollo MSF *for* CMMI.
- Especificación de los requisitos funcionales y no funcionales para el desarrollo de la solución.
- Desarrollo del análisis y diseño de la solución.
- Realización de los prototipos de interfaz de usuario.
- Definición del modelo de dominio, diagrama de despliegue y de componentes para la solución.
- Implementar la solución.
- Realización de pruebas a la solución.

Para el desarrollo de la investigación se utilizarán los siguientes **métodos científicos de la investigación**:

Métodos teóricos:

- **Modelación:** se utiliza como herramienta para comprender el problema y crear un modelo abstracto del mismo que permita el diseño de la solución.
- **Histórico-Lógico:** Este método es utilizado para la identificación y seguimiento de la evolución de sistemas informáticos existentes similares, con el objetivo de lograr una definición coherente de la solución a implementar.
- **Analítico-Sintético:** se analiza la información recopilada a través de los diferentes medios bibliográficos que pueda servir para desarrollar mejor el diseño del sistema, y de esta

Introducción

forma aplicar estos conocimientos en la práctica para así adquirir una mayor preparación sobre el tema en cuestión, además se utiliza en el análisis de las herramientas para el desarrollo de la solución informática propuesta, teniendo en cuenta sus características, ventajas y desventajas.

Métodos empíricos:

- **Entrevista:** Se manifiesta a través de la realización de entrevistas a los proveedores de requisitos válidos, además es necesario el intercambio con el personal capacitado para obtener información referente al tema, criterios y corregir malas concepciones para un mayor entendimiento del problema a resolver y determinar las necesidades del cliente.

El presente trabajo prevé como **posible resultados:**

- Especificación de requisitos de *software*.
- Versión estable del Portal de Trámites de la República de Cuba.
- Manual de usuario para el portal desarrollado.

Como **posibles aportes:**

- Contribución a un Gobierno en Línea.
- Sistema informático que controle las solicitudes de los ciudadanos y permita el seguimiento por parte del ciudadano del estado del trámite iniciado.
- Módulos para realizar quejas, sugerencias y encuestas.
- Control del flujo de personas.
- Actualización de la información referente a los trámites y las regulaciones.

El presente documento consta de tres capítulos, desarrollados a partir del estudio realizado de la información bibliográfica e institucional existente. La descripción de los mismos se presenta a continuación:

- **Capítulo 1 Fundamentación Teórica:** Se describen los conceptos fundamentales asociados al dominio del problema y el objeto de estudio, realizándose un análisis de la situación actual. Hace referencia al estado del arte del tema tratado, tanto en el ámbito

Introducción

nacional como internacional. Se presenta la fundamentación de las tecnologías utilizadas para el diseño del sistema y las propuestas para su implementación. Además, se realiza la descripción de las tecnologías y herramientas empleadas en el desarrollo de la solución.

- **Capítulo 2: Análisis y diseño:** Se realiza el desarrollo de la solución donde se analizan los elementos principales que intervienen en la aplicación. Se hace el levantamiento de requisitos que contiene la definición de requisitos funcionales y no funcionales del portal a desarrollar. Se identifica el modelo de dominio y el modelo de datos del sistema que se va a implementar y se establece los prototipos de interfaz gráfica. Además se describe la arquitectura a utilizar, así como los patrones de diseño para el desarrollo del portal.
- **Capítulo 3: Implementación y Pruebas:** Se describe la implementación del portal a través del diagrama de componentes, los estándares de codificación y el tratamiento de excepciones. Se expone la distribución de los nodos necesarios para el despliegue del portal y se valida la solución presentada.

Capítulo 1 Fundamentación Teórica

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

Este capítulo hace un recorrido por las características típicas de los portales web, haciendo referencia a los de trámites de identificación. Se realiza un estudio de algunos portales homólogos al que se necesita, definiéndose en caso de ser necesarias las tecnologías y herramientas a utilizar para realizar el Portal de Trámites de la República de Cuba. Se caracteriza la metodología y el Gestor de Base de Datos definidos por el proyecto.

1.1 PORTAL WEB

Según David Morrison⁵ enunció sobre un portal web: que puede ayudar a reconocer un portal frente a otro tipo de páginas web. Para ello el autor utiliza las iniciales del término:[4]

Personalización para usuarios finales.

Organización del escritorio.

Recursos informativos divididos y organizados.

Trayectoria o seguimiento de las actividades de los usuarios.

Acceso a base de datos.

Localización de cosas importantes.

Un portal web está dirigido a resolver necesidades de información específica de un tema en particular. Ofrece servicios para la navegación en *Internet*, logrando incrementar la intensidad de tráfico en el mismo. Es considerado un intermediario de información que tiene como fuente de ingreso la de tener una forma simple de acceder a toda y no sólo a una parte de la información referida al tema del mismo.[5]

Según los autores del presente trabajo de diploma un portal web: es un sitio web que ofrece de forma íntegra, una variedad de servicios al usuario, basándose en una misma temática y de forma organizada.

1.2 TRÁMITE

⁵Técnico especialista web de la compañía Lotus, en las oficinas centrales de la compañía en Cambridge.

Capítulo 1 Fundamentación Teórica

Según los autores del presente trabajo de diploma un trámite es: la gestión que realiza una o varias personas en función de resolver un asunto y en dependencia del trámite tendrá estados por los que pasar. Regularmente los trámites se realizan en las administraciones públicas y en menor grado en el sector privado; el ciudadano tiene que hacer trámites en forma permanente para desenvolverse en la sociedad, es por ello que existen muchos organismos públicos creados para tal fin. La DIR⁶ por encargarse de la identificación de los ciudadanos cubanos es quien rige todos los trámites de este tipo, que pueden ser: carné de identidad, tarjeta de menor, carné transitorio, pasaportes, registro de vehículo, licencia de conducción y armas de fuego.

1.2.1 IDENTIFICACIÓN

Según el Dr. Osvaldo Brito Febles: la identificación consistente en establecer la Identidad, características, individualidad y/o semejanza de las personas.[6]

Según la Real Academia: la identificación está unida a la identidad de las personas, que es el conjunto de los rasgos propios de un sujeto, dichos rasgos caracterizan a un individuo y los diferencia de los demás.

Según los autores del presente trabajo de diploma la identificación es: el conjunto de características particulares que hacen que una persona sea ella misma. La identificación vendría a ser la operación mediante la cual se determina ese grupo de detalles peculiares de una persona o grupo de personas.

1.2.2 PORTAL PARA TRÁMITES DE IDENTIFICACIÓN EN CUBA

Según los autores del presente trabajo de diploma un portal de trámites de identificación, según las necesidades que tiene la República de Cuba: se refiere a una aplicación web que permite a los ciudadanos conocer información sobre los trámites de identificación, y ofrece al usuario de forma fácil e íntegra, el acceso a una variedad de trámites y servicios, donde puedan conocer la información requerida para la realización de los mismos. Además debe brindar información sobre las Oficinas de Carné de Identidad donde se realizará el trámite, en cuanto a su ubicación y hora de apertura y cierre. Debe permitir reservar citas evitando al ciudadano tener

⁶ Dirección de Identificación y Registro.

Capítulo 1 Fundamentación Teórica

que hacer colas, debido a la aglomeración de personas. El usuario va a poder solicitar, consultar y supervisar algún documento de identificación de manera *online*⁷ a través del portal. Para poder consultar el estado de la solicitud realizada el portal debe comunicarse con el SUIN, sistema que se encarga de gestionar todos estos trámites. Según la arquitectura definida para la realización del SUIN, esta comunicación debe llevarse a cabo utilizando un servicio web utilizando la tecnología *Windows Communication Foundation*.

A continuación se hace un estudio de los distintos portales de trámites que se utilizan en los gobiernos de México, Venezuela, Perú y Santa Fé.

1.2.3 SERVICIO ADMINISTRATIVO DE IDENTIFICACIÓN, MIGRACIÓN Y EXTRANJERÍA “SAIME”

El portal del SAIME es utilizado para la realización de diferentes trámites en la República Bolivariana de Venezuela. Entre los diferentes servicios que brinda se encuentran: [7]

- Identificación.
- Misión Identidad.
- Pasaportes.
- Dactiloscopia.
- Extranjería.
- Migración.
- Módulos de Cedulación.

De la mayoría de estos servicios se ofrece información requerida, así como sus objetivos. El servicio de pasaporte además propone opciones de registro para la solicitud, consulta del estado del trámite entre otras funcionalidades. El servicio Misión Identidad brinda la ubicación de las oficinas en los diferentes estados del país, así como el horario de cada día de la semana.

1.2.4 PORTAL DE SERVICIOS AL CIUDADANO Y EMPRESAS. REPÚBLICA DE PERÚ

Este portal, se utiliza para la realización de trámites a los ciudadanos y a las empresas. Los trámites brindados a los ciudadanos se engloban en las siguientes categorías: [8]

- Nacimiento.

⁷ Hace referencia a un estado de conectividad.

Capítulo 1 Fundamentación Teórica

- Juventud.
- Adulto.
- Adulto mayor.

En estas cuatro categorías se ofrece los diferentes trámites que se pueden realizar en la República de Perú. Cada uno muestra información de cómo es el procedimiento para la obtención de los mismos, su costo, objetivos, y documentos a presentar.

1.2.5 PORTAL CIUDADANO DEL GOBIERNO DEL ESTADO DE MÉXICO

El portal brinda información de los diferentes trámites que necesitan los ciudadanos mexicanos. Los trámites se encuentran en un catálogo que se puede mostrar de diferentes formas, ordenado alfabéticamente, por temas y por personas. Además contiene la opción para hacer sugerencias, reportar quejas contra un servidor público, o denunciar delitos. Los trámites brindados a los ciudadanos se engloban en las siguientes categorías: [9]

- Nacimiento.
- Juventud.
- Adulto.
- Adulto mayor.

En estas categorías se brindan informaciones de diferentes trámites que se pueden realizar en el Estado de México. Cada uno muestra información de cómo es el procedimiento para la obtención de los mismo, su costo, objetivos y documentos a presentar.

1.2.6 PORTAL DE TRÁMITE DEL GOBIERNO DE SANTA FÉ

El portal ofrece una variedad de trámites con los que el ciudadano se encuentra identificado, contiene un listado extenso y amplio sobre cada proceso, permite realizar consultas y obtener turnos *online* para la confección del pasaporte y del documento de identificación nacional (DNI). Entre los diferentes trámites que brinda este portal se cuenta con: [10]

- Registro Civil-Trámites.
- Patente Única sobre Vehículos.
- Impuesto Inmobiliario.

Capítulo 1 Fundamentación Teórica

En estas categorías se hace especificación a cada trámite que el ciudadano desee. En el punto de Registro Civil-Trámites se realiza la solicitud de turnos web para atender de forma rápida y segura, en la Patente Única sobre Vehículos se visualiza o se liquida la deuda sin beneficio y la impresión de boletas para el mismo, así como en el trámite de Impuesto de Inmobiliario.

1.2.7 CONCLUSIONES

Los portales antes mencionados no satisfacen las necesidades anteriormente expuestas, pues no cuentan con la documentación de los trámites que se realizan en el país. Ninguno de estos portales permite la comunicación con el SUIN imposibilitando la consulta del estado de un trámite iniciado por este sistema. La mayoría de estos sistemas son privados por lo que el país tendría que hacer un gasto monetario para poder adquirirlos. Por lo anterior expuesto se hace necesario un portal de trámites que cumpla con todo lo requerido; donde los ciudadanos puedan conocer la información para la realización de trámites de carné de identidad, tarjeta de menor, carné transitorio, pasaportes, licencia de conducción, registro de vehículos y armas de fuego; además de un servicio web que permita la comunicación con el SUIN utilizando para ello la arquitectura de los servicios comunes de este sistema.

1.3 ARQUITECTURA PARA SERVICIOS WEB DEL SUIN.

Los servicios que utiliza el SUIN están diseñados siguiendo un patrón de capas bien definidas para reducir al máximo el acoplamiento y aumentar la reutilización entre cada una de las capas. A continuación se describe cada una de ellas.[11]

- **Presentación:** La capa superior del CommonCore contiene la presentación de los servicios *Windows Communication Foundation* (WCF) los cuales interactúan con los servicios base que son los que invocan a las capas inferiores. En esta capa se incluyen las entidades del negocio de los servicios, se define la interfaz de los conectores, lo que permite que esta capa interactúe con más facilidad con dichos componentes y obliga al modelo a implementar las responsabilidades definidas en esta capa.[11]
- **Controlador:** En el controlador se reciben las peticiones de la capa superior y se implementa la lógica de negocio del servicio realizando las peticiones al conector de la

Capítulo 1 Fundamentación Teórica

capa de acceso a datos y manejando los datos devueltos por este, necesarios para el negocio del servicio. Esta capa es la intermediaria de la arquitectura de los servicios.[11]

- **Conector:** Los conectores permiten establecer una conexión entre las entidades del negocio y la base de datos. Interactúan con el modelo y las entidades generadas por el *Entity Framework*. En el conector se realizan las operaciones básicas del acceso a datos utilizando LINQ.[11]
- **Modelo:** Conjunto de funcionalidades y operaciones genéricas que facilitan el trabajo de acceso a datos con las entidades mapeadas, además se definen los tipos de datos de cada una de ellas.[11]
- **Entidades:** Conjunto de clases mapeadas por el *Entity Framework* que representan cada una de las entidades existentes en la base de datos.[11]

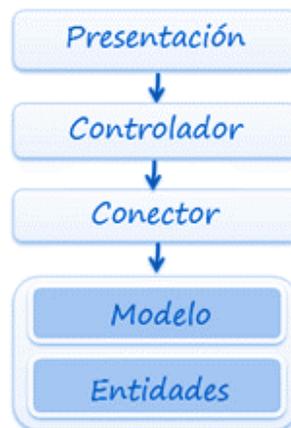


Ilustración 1 Estructura de los servicios

1.4 TECNOLOGÍAS WEB

Las tecnologías web son un conjunto de conocimientos específicos y procesos para transformar la realidad y resolver algún problema, estas implican un conjunto de herramientas que facilitarán lograr mejores resultados a la hora del desarrollo.[12] [13]

1.4.1 SISTEMA DE GESTIÓN DE CONTENIDOS (CMS)

Capítulo 1 Fundamentación Teórica

Un CMS⁸ se utiliza principalmente para facilitar la gestión de grandes sitios web, y también son conocidos como sistema de gestión de contenidos web (WCMS⁹). Hay que tener en cuenta, que la aplicación de los CMS no se limita solo a la web, sino también a la gestión de todo tipo de documentos y registros electrónicos. Permite crear una estructura de soporte para la creación y administración de contenidos, contiene un conjunto de herramientas de *software* que controla una o varias bases de datos, gestiona de manera uniforme los sitios dinámicos proporcionándole un entorno que facilita la presentación, el mantenimiento, la ampliación y sus actualizaciones periódicas.[14]

Según los desarrolladores de Joomla un CMS es: un *software* que permite crear y administrar páginas web fácilmente, separando la creación de su contenido de la mecánica necesaria para presentarlo en la web, almacenándose este, en una base de datos. El *software* de Joomla reúne la plantilla y el contenido para crear páginas, sitios y portales web.[15]

Para los autores del presente trabajo de diploma un CMS: es un *software* que se utiliza principalmente para la gestión web. Se encarga de gestionar aspectos como los menús de navegación y permite manejar de forma independiente el contenido. Permite crear, modificar y administrar el contenido de un sitio web sin necesidad de conocimiento técnico alguno.

1.4.2 FRAMEWORK

Según Jerome Lafosse¹⁰ un *framework*¹¹ es: un conjunto de bibliotecas, herramientas y normas a seguir que ayudan a desarrollar aplicaciones. Los *frameworks* los desarrollan los programadores de sistemas. Un *framework* está compuesto por varios segmentos/componentes que interactúan los unos con los otros. Las aplicaciones pueden escribirse de manera más eficaz si se utiliza un *framework* adaptado al proyecto. En proyectos de desarrollo a gran escala y de diseño en equipo, los *frameworks* son muy útiles, incluso imprescindibles. Permiten la reutilización de código, la estandarización del desarrollo y la utilización del ciclo de desarrollo.[17]

⁸Content Management Systems, por sus siglas en inglés.

⁹Web Content Management System, por sus siglas en inglés.

¹⁰Ingeniero en Informática, obtuvo su diploma en el conservatorio nacional de artes y oficios (CNAM), especialista en las tecnologías web.

¹¹Marco de trabajo.

Capítulo 1 Fundamentación Teórica

Según George Shephred¹² un *framework* 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.[18]

Para los autores del presente trabajo de diploma un *framework* es: un marco de trabajo el cual permite desarrollar de forma más rápida y organizada, y contiene funcionalidades que no requiere su implementación, sino que ya se encuentran implementadas. Está compuesto de componentes personalizables para el desarrollo del sistema, contiene un conjunto de clases que forman una aplicación genérica y reutilizable.

1.4.3 CONCLUSIONES

Por ser los CMS sistemas cuyo objetivo principal es la gestión de contenidos y los *frameworks* marcos de trabajo lo mejor es utilizar un CMS para la implementación del portal web pues facilita el desarrollo del mismo, teniendo todo al alcance de un clic e implementando solo los requisitos que sean específicos para el portal, reduciendo a su vez el tiempo de desarrollo.

1.5 SISTEMAS DE GESTIÓN DE CONTENIDOS

1.5.1 JOOMLA

Según los desarrolladores de Joomla: Joomla es un sistema de gestión de contenidos (CMS) que se destaca por su sencillez y rapidez, es gratuito y el más difundido en el mundo de la web, es un administrador de contenidos para la construcción de sitios web dinámicos que ofrecen una amplia gama de posibilidades y servicios, lo cual permite gestionar con mucha facilidad la web y además requiere de una base de datos MySQL¹³. [19], [20],[21]

Para los autores de este trabajo de diploma: Joomla es un *software* que está creciendo rápidamente, con facilidad de administración y flexibilidad en el uso de las plantillas, tiene un número de propiedades, contiene sencillos formularios de contacto, y controla la edición y publicación del contenido. Joomla es una solución de código abierto que posee un sistema de

¹²Experto en *Microsoft.NET Framework* y desarrolla algunos de los líderes de la industria de terceros. NET basados en herramientas.

¹³ Sistema de Gestión de Bases de Datos relacional, multihilo y multiusuario.

Capítulo 1 Fundamentación Teórica

registro que permite a los usuarios configurar las opciones personales y los usuarios pueden acceder, editar, publicar y administrar.

1.5.2 DRUPAL

Drupal ha tenido un tremendo impacto en el panorama de *Internet*, como sistema de gestión de contenidos, ha permitido la creación de sitios y portales web con brillantes funciones y contenidos para pequeñas y grandes organizaciones. Ofrece nuevas y numerosas soluciones y mayor facilidad de uso. Tanto si está diseñando una web para un negocio, una organización sin fines de lucro, o por el contrario, si desarrolla un sitio web de grandes dimensiones.[15] Brinda una amplia gama de rasgos y servicios incluyendo la administración de usuarios, publicaciones y funcionalidades que usan vocabularios controlados. Equipado con una mezcla poderosa de novedades y configuraciones, puede apoyar una diversidad de proyectos web, manejados por grandes comunidades.[22],[23] Posee características particulares como son:[24]

- Control sobre el flujo de creación, edición, borrado y publicación de contenidos.
- Categorización y organización de contenidos.
- Gestión de usuarios y roles. Privacidad, control de acceso.
- Presentación de contenidos en diversos formatos.
- Sistema ampliable y extensible.
- Autenticación de los usuarios.

Para los autores de este trabajo de diploma Drupal es: un sistema de gestión de contenido configurable que permite desarrollar desde sitios sencillos a desarrollos complejos, además de publicar artículos, imágenes u otros archivos y servicios añadidos como foros, encuestas, administración de usuarios y permisos. Es un programa de código abierto, desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de las páginas generadas, el respeto de los estándares de la web, y un énfasis especial en la usabilidad y consistencia de todo el sistema. Los principios del proyecto de Drupal fomentar la modularidad, las normas, la colaboración y la facilidad de uso. Si se tiene que comparar con Joomla, Drupal está más estructurado y es más potente para desarrollar sitios complejos, pero tiene una curva

Capítulo 1 Fundamentación Teórica

de aprendizaje mayor. Posee un módulo que permite la comunicación con el Sistema Gestor de Base de Datos Oracle.

1.5.3 WORDPRESS

Aunque WordPress es la plataforma de blogs más popular, la mayoría de sus usuarios desconoce sus opciones de uso avanzado y están desaprovechando muchas oportunidades para ser mejores blogueros. El blog es el espejo de la persona si tienes muchas cosas que contar, quieres compartir tu opinión sobre el mundo o quieres que tus jefes vean lo buen profesional que eres, un blog es tu ventana al universo.[25]

Para los autores de este trabajo de diploma WordPress es: una plataforma de publicación personal orientada a la estética, los estándares web y la usabilidad que a pesar de haber evolucionado para ser utilizado como CMS, actualmente se enfoca más como sistema de blogs.

1.5.4 CONCLUSIONES

Teniendo en cuenta las principales características de los CMS estudiados y según las condiciones que debe cumplir el portal de trámites, se considera que lo mejor es utilizar el Sistema de Gestor de Contenidos (CMS) Drupal (versión 7), ya que se integra mejor con el Gestor de Base de Datos Oracle *Database* 11g que define y utiliza el proyecto, mientras que Joomla no tiene un mecanismo que permita la comunicación con este gestor. Por otra parte WordPress es un CMS utilizado principalmente para realizar sistemas blogs.

1.6 SISTEMAS GESTORES DE BASE DE DATOS (SGBD)

Según Olga Pons Capote¹⁴un Sistema de Gestión de Base de Datos (SGBD): es un conjunto de elementos *software* con capacidad para definir, mantener y utilizar una base de datos.[26]

Según Olga Pons Capote una Base de Datos (BD): es un fondo común de información almacenada en una computadora para cualquier persona o programa autorizado pueda acceder a ella, independientemente de su procedencia y del uso que haga.[26]

¹⁴ Ingeniera Técnica en Informática de Sistemas. En la Universidad de Granada, en el departamento de Ciencias Informáticas e Inteligencia Artificial.

Capítulo 1 Fundamentación Teórica

Para los autores de este trabajo de diploma SGBD son: un tipo de *software* muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

1.6.1 ORACLE

Es un Sistema de Gestión de Base Datos relacional. Básicamente es una herramienta cliente/servidor para la gestión de bases de datos. Posee muchas características que garantizan la seguridad e integridad de los datos. Se considera uno de los sistemas de bases de datos más completos, destacándose por su:

- Soporte de transacciones.
- Estabilidad.
- Escalabilidad.
- Soporte multiplataforma.

*Oracle Database 11g*¹⁵ proporciona funcionalidades que garantizan un alto rendimiento, escalabilidad, fiabilidad y seguridad mediante el uso de plataformas, asegurando altos niveles de calidad de servicios e incremento de la flexibilidad de negocio. Incorpora seguridad de archivos que permite la gestión de todo tipo de datos. Es el primer gestor de base datos en incluir funcionalidades que permiten hacer pruebas de cambios en aplicaciones simulando las cargas reales generadas por los usuarios en los entornos de producción. Ofrece soluciones concretas a las demandas de los usuarios respecto a las bases de datos actuales y de próxima generación.[27]

1.7 SERVIDOR WEB

Un servidor web, es en términos generales una máquina equipada con *software*, que utiliza el protocolo http¹⁶, para responder a las peticiones de los clientes web. Es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente, generando una respuesta en cualquier

¹⁵Sistema Gestor de Base de Datos Relacional: acrónimo de *Relational Data Base Management System*.

¹⁶*Hypertext Transfer Protocol*, por sus siglas en inglés; en español (protocolo de transferencia de hipertexto).

Capítulo 1 Fundamentación Teórica

lenguaje o aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web.[28]

1.7.1 XAMPP

XAMPP, su nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos). Es un servidor independiente de plataforma, consiste principalmente en la base de datos MySQL, el servidor web Apache y el intérprete para lenguaje de script PHP¹⁷. Esta liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar paginas dinámicas. Actualmente se encuentra disponible para *Microsoft Windows*, *GNU/Linux*, *Solari* y *MacOs X*. XAMPP solamente requiere según sus características descargar y ejecutar un archivo, con unas pequeñas configuraciones en alguno de sus componentes que el servidor web necesitará. Además se actualiza regularmente para incorporar las últimas versiones y también incluye otros módulos. Oficialmente los diseñadores de XAMPP, solo pretenden su uso como herramienta de desarrollo, para permitir el diseño de sitios web, sin embargo, en la práctica es usado como servidor de sitios web con algunas modificaciones, es generalmente lo suficientemente seguro. Hasta el momento existen cuatro distribuciones de XAMPP:

- **XAMPP para Linux:** La versión para sistemas Linux (testado para SuSE, RedHat, Mandrake y Debian) contiene: Apache, MySQL, PHP y PEAR, Perl, ProFTPD, phpMyAdmin y OpenSSL.
- **XAMPP para Windows:** La distribución de Windows 2000, 2003, XP, Vista y 7. Esta versión contiene: Apache, MySQL, PHP y Perl.
- **XAMPP para Mac OS X:** La distribución para Mac OS X contiene: Apache, MySQL, PHP y PEAR, SQLite y Perl.
- **XAMPP para Solari:** La versión para Solaris (desarrollada y probada con Solaris 8, probada con Solaris 9) contiene: Apache, MySQL, PHP y PEAR, Perl.

1.7.2 WAMP

¹⁷ Lenguaje de programación interpretado diseñado especialmente para el desarrollo de páginas web dinámicas.

Capítulo 1 Fundamentación Teórica

Wamp, es una computadora con *Windows* que dispone de un servidor Apache, un gestor de bases de datos MySQL y el lenguaje de programación PHP. Disponer de este servidor, permite instalar aplicaciones desde la red local, también serán accesibles desde *Internet*. Se puede instalar sus aplicaciones por separado, pero obteniendo su paquete se instalan y configuran automáticamente las mismas, estas son la base para poder instalar disímiles de aplicaciones web, entre las cuales se destacan:

Gestores de Contenidos orientados a sitios web: Joomla, Drupal, entre otros.

Gestores de Contenidos orientados a educación: Caroline, Moodle, Dokeos, entre otros.

Blogs: WordPress.

Las funcionalidades de Wamp son muy completas y fáciles de usar, a continuación se muestran sus características:

- Administrar sus servicios Apache y MySQL. Administrar la configuración de los servidores.
- Cambiar en línea / fuera de línea (dar acceso a todo el mundo o sólo localhost).
- Instalar y cambiar comunicados Apache, MySQL y PHP.
- Acceder a sus registros. Crear alias.
- Acceder a sus archivos de configuración. Cambiar el idioma del menú.

1.7.3 INTERNET INFORMATION SERVICE (IIS)

Es un servidor web y un conjunto de servicios para el sistema operativo *Microsoft Windows*. Servidor web flexible, seguro y manejable para acoger cualquier forma en la web. Los servicios de *Internet Information Service* (IIS) son los servicios de *software* que admiten la creación, configuración y administración de sitios web, además de otras funciones de *Internet*. Incluye el protocolo de transferencia de noticias a través de la red (NNTP), el protocolo de transferencia de archivos (FTP) y el protocolo simple de transferencia de correo (SMTP).

1.7.4 CONCLUSIONES

Existen disímiles servidores web, entre los cuales se encuentran IIS, Xampp y Wamp, los mismos presentan características particulares ya explicadas con anterioridad, que facilitan el desarrollo de

Capítulo 1 Fundamentación Teórica

los programadores. Para el desarrollo del portal de trámites, se hace uso del servidor web Xampp principalmente por ser un servidor multiplataforma.

1.8 METODOLOGÍA DE DESARROLLO

MSF¹⁸ for CMMI¹⁹, define cinco fases durante el ciclo de vida del proyecto que encapsula flujo de actividades y actividades, que cumple con el requerimiento del nivel tres de CMMI. Las fases son visión, planificación, desarrollo, estabilización e implantación. Cada fase concluye con un punto de control. Cada punto de control proporciona una oportunidad para autorizar el trabajo, continuar en el proyecto, o cancelar o suspender el proyecto.[29]

Es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.[30]

Fases de la metodología de desarrollo MSF for CMMI:

- **Visión:** obtener la visión y el alcance del proyecto, el cual debe estar compartido, comunicado, entendido y alineado con los objetivos del negocio. En esta fase el equipo y el cliente integran el proyecto y definen los requerimientos funcionales, sus alcances y restricciones, identifican los beneficios del proyecto y también los riesgos del proceso. La fase culmina con el hito Visión y Alcance aprobados. Dentro de los entregables de esta fase se encuentran: el Documento de Visión y Alcance y la Matriz de Identificación de riesgos.
- **Planificación:** el equipo del proyecto creará un borrador del plan maestro del proyecto, además de la especificación funcional del proyecto y un cronograma que identifica puntos de control específicos. Esta fase culmina con el hito Plan del Proyecto (Especificaciones y Cronograma) aprobado. Dentro de los entregables de esta fase se encuentran: Minuta de reunión de Kick-off¹⁵ del proyecto, Documento de Especificaciones, Cronograma del

¹⁸Framework de Soluciones de Microsoft (Microsoft Solution Framework (MSF), por sus siglas en inglés).

¹⁹ Integración de Modelos de Madurez de Capacidades (Capability Maturity Model Integration (CMMI), por sus siglas en inglés).

Capítulo 1 Fundamentación Teórica

proyecto (establecimiento de línea base) y Documentos de proceso de licitación aprobados (de acuerdo a la Norma Operativa vigente).

- **Desarrollo:** involucrar la serie de comunicadores internos o entregas parciales del producto, desarrollados por partes para medir su progreso y para asegurarse que todos sus módulos o partes están sincronizados y pueden integrarse. La fase culmina con el hito Alcance completo. Dentro de los entregables de esta fase se encuentran: Código fuente y ejecutables (resguardados y etiquetados en VSS), Actas de aceptación de entregas parciales, Plan de Pruebas, Manual de Instalación y Operación, Documento de Arquitectura, Actas de control de cambios aprobadas (que justifican el ajuste al alcance, tiempo y/o costo del proyecto, de existir).
- **Estabilización:** lleva a cabo pruebas en una solución completa. Las pruebas durante esta fase hacen hincapié en el uso y el funcionamiento en el medio ambiente realista. Una vez que una generación ha sido considerada lo suficientemente estable como para ser un candidato de liberación, la solución se instala en un grupo piloto. La fase de estabilización culmina en el hito de: Aceptación de Pruebas. Dentro de los entregables de esta fase se encuentran: Acta de Aceptación de Pruebas, Acta de Capacitación a Usuarios y Mesa de Servicios, Acta de Entrega (comité de proyectos).
- **Implantación:** en esta fase el equipo implanta la tecnología y los componentes utilizados por la solución, apoya el funcionamiento y la transición del proyecto, y obtiene la aprobación final del cliente. En ocasiones en esta fase se ejecutan planes piloto de implementación. La fase termina con el hito Cierre de la Entrega. Dentro de los entregables de esta fase se encuentran: Acta de Implantación, Encuesta de satisfacción del cliente y Acta de Cierre de proyecto.

Entre los diferentes roles definidos por esta metodología se encuentran el líder, jefe de desarrollo, jefe de producto, el arquitecto de *software*, desarrollador, analista, diseñadores, probador, integrador, documentador-capacitador, administrador de la calidad, entre otros. Los miembros del equipo de trabajo pueden desempeñar roles diferentes en las distintas fases del ciclo de vida del

Capítulo 1 Fundamentación Teórica

proyecto y son responsables de cumplir con las actividades y de generar la documentación. Esta metodología define que ningún rol es más importante que otro.

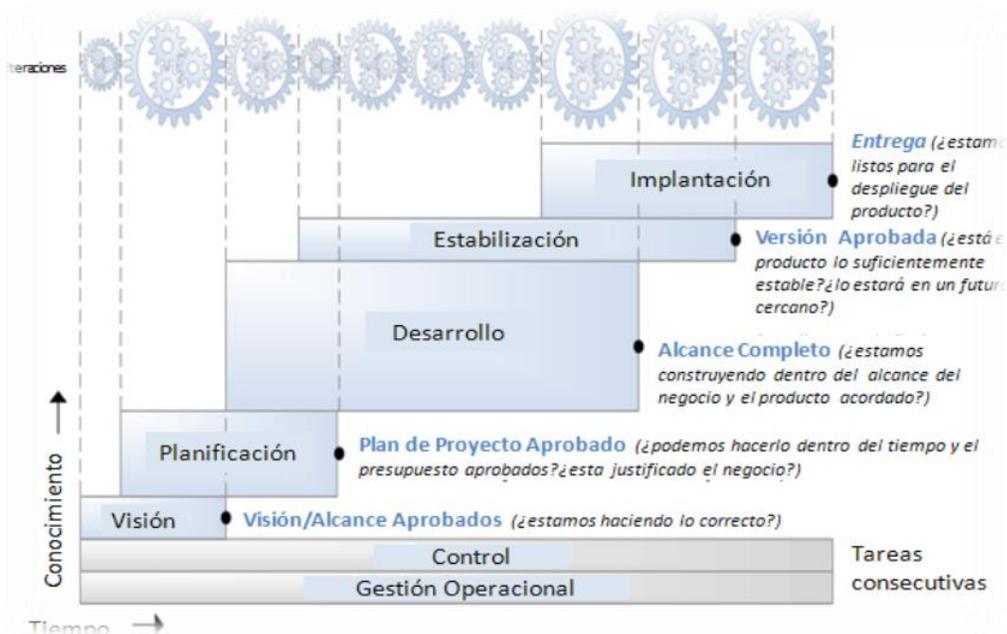


Ilustración 2 Fases de MSF for CMMI. Fuente: MSDN

1.9 LENGUAJES DE PROGRAMACIÓN

“Un lenguaje de programación es un idioma artificial diseñado para expresar computaciones que pueden ser llevadas a cabo por las computadoras”.[31] Existen diferentes lenguajes de programación para el desarrollo en la web, los mismo han surgido debido a la necesidad de cada plataforma. En el presente epígrafe se hace un análisis de los principales lenguajes de programación y tendencias que se están empleando en el mundo actual para el trabajo con las aplicaciones web. Se hace un análisis de los lenguajes más utilizados hoy en día para la programación de aplicaciones web tales como PHP, JavaScript y CSS.

1.9.1 PHP

PHP es un lenguaje orientado a objetos, lo que permite la reutilización de código y un rápido desarrollo. Posee una gran librería de funciones y mucha documentación, que se incrustan dentro

Capítulo 1 Fundamentación Teórica

del código HTML²⁰. Este lenguaje es gratuito y multiplataforma, teniendo compatibilidad con las bases de datos más comunes como MySQL, Oracle y PostgreSQL²¹. [32] Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida, pero puede ser utilizado desde una interfaz de líneas de comandos o en la creación de otros programas como aplicaciones de interfaz gráfica. Una de sus características más potentes es su soporte para gran cantidad de bases de datos y ofrece integración con varias bibliotecas externas. Permite a los desarrolladores la generación dinámica de páginas, la técnica de programación orientada a objetos, no requiere definición de tipos de variables y es un producto de código abierto, permitiendo que los fallos de funcionamiento se encuentren y se reparen rápidamente. [33]

1.9.2 JAVASCRIPT

Es un lenguaje de programación compatible con la mayoría de los navegadores modernos, por lo que es el lenguaje del lado del cliente más utilizado actualmente. Con JavaScript se puede crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones y ejecutarlas para realizar estos efectos e interactividades. Es un lenguaje con muchas posibilidades, usado para crear pequeños programas que son insertados en una página web, orientado a objetos muchos más complejos. Además, pone a disposición del programador todos los elementos que conforman la página web, para que este pueda acceder a ellos y modificarlos dinámicamente. Tiene algunas características, entre ellas se puede mencionar que es un lenguaje *open-source*²², es basado en acciones que poseen menos restricciones, gran parte de su programación está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas y cargas de páginas. [34]

1.9.3 CSS

CSS²³ son un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML, separando el contenido de la presentación. Es un simple mecanismo para agregar

²⁰Lenguaje de marcado de hipertexto: acrónimo de *Hypertext Markup Language*.

²¹Sistema de Gestión de Base de Datos orientado a objetos y libre.

²²Código abierto, distribuido y desarrollado libremente.

²³Hojas de estilo en cascada (*Cascading Sheet Style*, por sus siglas en inglés).

Capítulo 1 Fundamentación Teórica

estilo (fuentes, colores, espaciado) a los documentos. La información de estilo explica cómo se ha de mostrar una etiqueta determinada, esta puede ser adjuntada como un documento separado o en el mismo documento HTML. Se pueden definir hojas de estilo para diferentes navegadores y tipos de medios (impresos, auditivos, braille). Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario. Usando CSS se aminora la cantidad de código, permite separar los contenidos de la página y la información, sobre su aspecto; y posee alta velocidad de carga. Utilizando CSS, se pueden establecer los mismos estilos con menos esfuerzo y sin ensuciar el código HTML de los contenidos con etiquetas. Una de las principales características de CSS es su flexibilidad y las diferentes opciones que ofrece para realizar una misma tarea.[35]

1.9.4 C#

C# es un lenguaje orientado a objetos, elegante y con seguridad de tipos que permite a los desarrolladores crear una amplia gama de aplicaciones sólidas y seguras que se ejecutan en *.NET Framework*. Puede ser utilizado para crear aplicaciones cliente para *Windows* tradicionales, servicios web, componentes distribuidos, aplicaciones de base de datos, y muchas tareas más. Se utilizará este lenguaje para la implementación del servicio de intercambio entre el portal de trámites y el SUIN.[36]

1.9.5 LINQ

Es muy común que las aplicaciones utilicen datos en bases de datos SQL o documentos XML. Tradicionalmente, los programadores tenían que aprender un lenguaje de programación primario, como C#, y uno secundario, como SQL o XQuery. *Language-Integrated Query*²⁴(LINQ) incorpora capacidades de consulta al lenguaje C# propiamente dicho. Ahora, en lugar de aprender un lenguaje de consulta independiente, se puede utilizar los conocimientos de C#, junto con algunas palabras clave y conceptos adicionales, para consultar bases de datos SQL, conjuntos de datos ADO.NET, documentos XML y cualquier clase de colección .NET que implemente la interfaz *IEnumerable*.[37]

²⁴ Lenguaje de consultas integrado.

Capítulo 1 Fundamentación Teórica

1.10 LENGUAJE DE MODELADO: UML

UML²⁵ es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema *software* orientado a objetos. Los principales objetivos en el diseño de UML fueron: obtener un lenguaje simple y legible, pero suficientemente expresivo, que permitiese modelar aplicaciones en cualquier dominio y la generación automática del código, puesto que sería un lenguaje utilizado por las personas. UML provee beneficios para los ingenieros de *software* y las organizaciones al ayudarles a construir modelos rigurosos y trazables, que soporten el ciclo de vida de desarrollo de *software*. Aporta ventajas por ser un método de formal de modelado, las mismas se exponen a continuación:

- Mayor rigor en la especificación.
- Permite realizar una verificación y validación del modelo realizado.
- Se puede automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa, es decir a partir del código fuente generar modelos.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas; para el desarrollo del portal de trámites se utilizará el diagrama de componentes y el diagrama de despliegue.

1.11 HERRAMIENTAS Y TECNOLOGÍAS.

1.11.1 HERRAMIENTA DE MODELADO: ALTOVA UMODEL 2010

“UModel²⁶ 2010 incluye las más altas funcionalidades para potenciar a los usuarios con las más completas ventajas del desarrollo de software UML”. Dentro de las características que posee se encuentran, el soporte para los 13 tipos de diagramas UML, modelado de esquemas, diagramas de procesos de negocios, ingeniería inversa de código fuente y ficheros binarios, sincronizado de modelo y código a través de ingeniería, generación de documentación personalizable de proyecto, e integración con sistemas de control de versiones.[38]

²⁵ Lenguaje Unificado de Modelado (*Unified Modeling Language*, *UML* por sus siglas en inglés).

²⁶ UModel es un UML (*Unified Modeling Language*) herramienta de modelado de *software* de Altova.

Capítulo 1 Fundamentación Teórica

Para el modelado del modelo de dominio, el diagrama de despliegue y el diagrama de componentes, se hace uso de la herramienta Altova UModel²⁷ 2010, que es el punto de entrada para el desarrollo de un *software* exitoso. Permite crear e interpretar diseños mediante la potencia del estándar UML, dibujar su diseño de aplicación y generar código a partir de sus planos, o hacer ingeniería inversa de programas existentes a diagramas UML claros y precisos para abarcar rápidamente su arquitectura de *software*. Incluso podrá corregir el código generado o sus modelos y completar la ronda produciendo automáticamente nuevos diagramas o regenerando el código. UModel elimina el misterio de UML con ayuda contextual a la introducción de datos, sintaxis coloreada, estilos en cascada, elementos de diseño personalizables, múltiples vistas de composición, función deshacer/rehacer ilimitada, y muchas otras funciones de usabilidad. Su rica interfaz y usabilidad superior ayudan a disminuir la curva de aprendizaje de UML, permitiendo que desarrolladores, incluso los nuevos en el modelado de *software*, manejen rápidamente UML, potenciando su productividad y maximizando resultados.

1.11.2 HERRAMIENTA DE DESARROLLO: *MICROSOFT VISUAL STUDIO 2010*

Microsoft Visual Studio 2010, es un conjunto de herramientas que simplifica el desarrollo de aplicaciones para desarrolladores individuales o equipos, haciendo posible la creación de aplicaciones escalables y de alta calidad.[39] Proporciona un entorno integrado de herramientas e infraestructura de servidor que simplifica todo el proceso de desarrollo de aplicaciones. Ofrece resultados empresariales usando procesos productivos, predecibles y personalizables, y aumenta la transparencia y el seguimiento durante el ciclo de vida con análisis detallados. Aumenta la productividad del equipo usando características de colaboración avanzadas y herramientas de pruebas y depuración integradas para detectar y corregir errores de un modo rápido y sencillo, lo que permite crear soluciones de alta calidad y reducir el coste del desarrollo de soluciones. Presenta novedosas herramientas de pruebas que automatizan la realización de pruebas de la interfaz de usuario en aplicaciones basadas en web: pruebas manuales, pruebas de rendimiento de web, pruebas de carga, cobertura de código, generación de datos de pruebas.

1.11.3 *WINDOWS COMMUNICATION FOUNDATION*

²⁷Herramienta de modelado, basada en *UML* incluye ingeniería inversa.

Capítulo 1 Fundamentación Teórica

Windows Communication Foundation (WCF) es un marco de trabajo para la creación de aplicaciones orientadas a servicios. Con WCF, es posible enviar datos como mensajes asincrónicos de un extremo de servicio a otro. Un extremo de servicio puede formar parte de un servicio disponible continuamente hospedado por IIS, o puede ser un servicio hospedado en una aplicación. Un extremo puede ser un cliente de un servicio que solicita datos de un extremo de servicio. Los mensajes pueden ser tan simples como un carácter o una palabra que se envía como XML, o tan complejos como una secuencia de datos binarios. A continuación se indican unos cuantos escenarios de ejemplo:

- Un servicio seguro para procesar transacciones comerciales.
- Un servicio que proporciona datos actualizados de un trámite solicitado por una persona.
- Un servicio de chat que permite a dos personas comunicarse o intercambiar datos en tiempo real.
- Una aplicación de panel que sondea los datos de uno o varios servicios y los muestra en una presentación lógica.

Si bien era posible crear tales aplicaciones antes de que existiera WCF, con WCF el desarrollo de extremos resulta más sencillo que nunca. En resumen, WCF se ha diseñado para ofrecer un enfoque manejable para la creación de servicios web y clientes de servicios web.[37]

1.11.4 ADO.NET ENTITY FRAMEWORK

ADO.NET *Entity Framework* permite a los desarrolladores crear aplicaciones de acceso a datos programando con un modelo de aplicaciones conceptuales en lugar de programar directamente con un esquema de almacenamiento relacional. El objetivo es reducir la cantidad de código y el mantenimiento necesarios para las aplicaciones orientadas a datos. Las aplicaciones de *Entity Framework* ofrecen las siguientes ventajas:[40]

- Las aplicaciones pueden funcionar en términos de un modelo conceptual más centrado en la aplicación, que incluye tipos con herencia, miembros complejos y relaciones.
- Las aplicaciones están libres de dependencias de codificación rígida de un motor de datos o de un esquema de almacenamiento.

Capítulo 1 Fundamentación Teórica

- Las asignaciones entre el modelo conceptual y el esquema específico de almacenamiento pueden cambiar sin tener que cambiar el código de la aplicación.
- Los desarrolladores pueden trabajar con un modelo de objeto de aplicación coherente que se puede asignar a diversos esquemas de almacenamiento, posiblemente implementados en sistemas de administración de base de datos diferentes.
- Se pueden asignar varios modelos conceptuales a un único esquema de almacenamiento.
- La compatibilidad con Language Integrated Query (LINQ) proporciona validación de la sintaxis en el momento de la compilación para consultas en un modelo conceptual.

1.12 CONCLUSIONES PARCIALES

Luego del estudio realizado sobre los principales conceptos tratados en este capítulo, y el análisis de las soluciones existentes en el mundo referente a los portales de trámites, se pudo detectar características que no satisfacen las necesidades del gobierno cubano, lo cual conlleva a implementar un Portal de Trámites de la República de Cuba y un servicio web que permita el intercambio con el SUIN. Para la implementación del portal se utilizará un CMS, pues facilita el desarrollo y la gestión de contenidos. Para lograr una buena integración con el SGBD Oracle *Database* 11g, herramienta definida por el proyecto, se decide desarrollar el portal con el CMS Drupal en su versión 7.0. Se definió utilizar el servidor web Xampp principalmente por ser multiplataforma. Por último se caracteriza la metodología, lenguajes y herramientas de desarrollo definidas por el proyecto para el desarrollo del servicio web y el portal.

Capítulo 2 Análisis y Diseño

CAPÍTULO 2 ANÁLISIS Y DISEÑO

A partir de la selección realizada en el capítulo anterior se presenta la aplicación de las herramientas, tecnologías y metodología para el diseño del sistema. En el presente capítulo se realiza el levantamiento y especificación de los requisitos funcionales y no funcionales para el desarrollo del portal, conjuntamente con los roles que intervienen. Se describe la arquitectura a utilizar, los patrones de diseño para el desarrollo del portal y el modelo de datos usado.

2.1 PRESENTACIÓN de la Solución

La solución es un Portal de Trámites de la República de Cuba cuyo objetivo es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de trámites y servicios, donde se pueda conocer la información requerida para la realización de trámites. Además de cómo saber llegar a cada una de las oficinas donde se realizará el trámite, reservar su cita para presentarse en las oficinas evitando tener que hacer colas, debido a la aglomeración de personas en las mismas. El usuario va a poder conocer el estado en que se encuentra sus trámites de manera *online* a través del portal, además de poder realizar consultas, quejas y sugerencias.

La solución contará además con un servicio web que permitirá la comunicación del portal con el SUIN posibilitando conocer los estados de los trámites. A través de este servicio se podrá administrar los turnos de una oficina cuando se solicite un trámite determinado, además de conocer las solicitudes que tenga una oficina determinada.

2.2 MODELO DE DOMINIO

Para el desarrollo de la propuesta de solución se realizó un modelo de dominio con el objetivo de lograr una representación general de las entidades y conceptos que interceden en la creación del portal. Con la realización del modelo de dominio se pretende unificar el vocabulario entre los usuarios y los desarrolladores, para poder entender el contexto en que se emplaza el sistema.

Capítulo 2 Análisis y Diseño

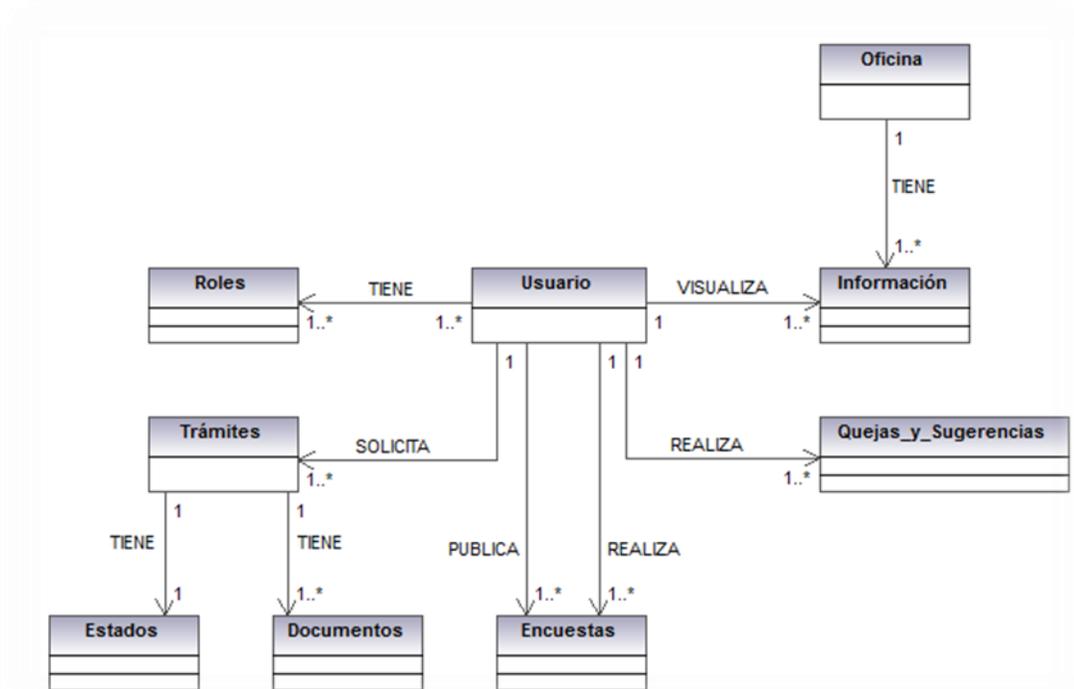


Ilustración 3 Modelo de Dominio

2.2.1 DESCRIPCIÓN DEL MODELO DE DOMINIO

A continuación se describen todos los conceptos y sus relaciones presentes en el modelo de dominio:

- **Usuario:** es quien interactúa de acuerdo al rol con las funcionalidades que brinda el portal.
- **Roles:** es una colección de permisos definidos para todo el sistema que puede asignar a usuarios específicos en contextos específicos.
- **Trámites:** gestión que realiza una o varias personas en función de resolver un asunto.
- **Estados:** entidad que resume la condición a la que pertenece un trámite. Ejemplo de estados por los que debe pasar un trámite (sin iniciar, cancelado, confección, terminado).
- **Documentos:** escrito que contiene información referente a cada trámite que se desea realizar.
- **Encuestas:** es un conjunto de preguntas que responden a una información específica.[41]

Capítulo 2 Análisis y Diseño

- **Quejas y Sugerencias:** manifestación o propuesta documentada de un usuario o grupo de usuarios con el fin de hacer constar una situación anómala derivada del servicio brindado o de un procedimiento inadecuado con el objetivo de que quede subsanada y de mejorar la situación.[42]
- **Oficina:** área de trabajo o espacio físico, donde se ejecutan los diferentes trámites para la identificación y registro de la población. Cada oficina tiene un nombre que lo identifica.
- **Información:** es la información referente a cada una de las oficinas.

2.3 REQUERIMIENTOS DEL SISTEMA

2.3.1 DESCRIPCIÓN DE LOS ROLES

El sistema propuesto dispondrá de los siguientes roles de usuarios:

Rol	Descripción
Usuario Anónimo	<ul style="list-style-type: none">• Es la persona que navega por el sistema sin haberse registrado aún, interactúa con éste sin privilegios y tiene la posibilidad de visualizar las diferentes opciones que brinda el mismo.• Se emplea para los usuarios que no tienen cuenta o no se han autenticado en el sistema. No puede ser eliminado del sistema.
Usuario Registrado	<ul style="list-style-type: none">• Todos los usuarios deben autenticarse para acceder a cualquier opción que así lo requiera, exceptuando las de la administración.• Se asigna automáticamente a todos los usuarios que tengan una cuenta y han iniciado una sesión. Por tanto todos los usuarios registrados tendrán este rol y podrán tener asignados otros roles adicionales. No puede ser eliminado del sistema.
Administrador	<ul style="list-style-type: none">• Son aquellos usuarios que tienen permiso para administrar

Capítulo 2 Análisis y Diseño

	<p>distintas funcionalidades del portal.</p> <ul style="list-style-type: none">• Tiene asignado todos los permisos disponibles, tanto de los módulos del núcleo, como de los módulos adicionales que se instalen en el portal. Puede editarse e incluso eliminarse, aunque no se recomienda.• Define los permisos necesarios para poder gestionar todos los contenidos del portal.
Gestor de Encuesta	<ul style="list-style-type: none">• Define los permisos para la realización de encuestas.• Gestiona las encuestas.
Escritor	<ul style="list-style-type: none">• Define permisos para poder escribir artículos en el portal, pero no para publicar el contenido.• Los contenidos enviados por este rol tendrán que ser validados y publicados por el Editor.
Recepcionista	<ul style="list-style-type: none">• Consultar los reportes generados.

Tabla 1 Descripción de roles

2.3.2 DEFINICIÓN DE REQUISITOS FUNCIONALES

Los requisitos funcionales de un sistema describen la funcionalidad que se espera que éste proporcione. Son entendidos como capacidades que debe exhibir un sistema con el fin de resolver un problema.[43] A continuación se muestran los requisitos funcionales del portal y del servicio web.

RF 1	Registrar Usuario.
RF 2	Autenticar Usuario.
RF 3	Obtener requisitos para la realización de trámites.
RF 4	Solicitar trámites.
RF 5	Consultar el estado de los trámites.
RF 6	Mostrar información de las oficinas.
RF 7	Cancelar la Solicitud del trámite.

Capítulo 2 Análisis y Diseño

RF 8	Publicar encuesta.
RF 9	Publicar quejas y sugerencias.
RF10	Gestionar información del portal.
RF11	Gestionar Usuario.
RF12	Generar Reporte.

Tabla 2 Definición de los Requisitos Funcionales del Portal de Trámites

RF 1	Solicitar trámites.
RF 2	Obtener el estado de un trámite solicitado.
RF 3	Administrar los turnos para una oficina.
RF 4	Obtener las solicitudes de una oficina para una fecha determinada.
RF 5	Obtener las solicitudes de una persona determinada.

Tabla 3 Definición de los Requisitos Funcionales del servicio web ExchangePortalService

2.3.3 ESPECIFICACIÓN DE REQUISITOS FUNCIONALES

El objetivo de la especificación de requisitos es definir de forma precisa y completa las funcionalidades y restricciones del sistema. A continuación se expone un resumen de la especificación de los requisitos funcionales del portal y del servicio web. **Ver Anexo I.**

RFP 4 Solicitar trámites.

Propósito	Iniciar trámites de los usuarios.	
Roles	Usuario registrado.	
Pre-condiciones	El usuario tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	USERS	Usuario Contraseña

Capítulo 2 Análisis y Diseño

	DRESERVACION_CITAS	idEnvio carne_ident nombre apellidos provincia oficina documento tramite idUser usuario idSolicitud
Descripción	<p>4.1 Seleccionar la opción “Reservar Cita”, en el menú “Servicios para usuario”.</p> <p>4.2 Mostrar los datos si se selecciona la opción “Reservar Cita”.</p> <ul style="list-style-type: none">a) Carné de Identidad.b) Nombre.c) Apellido.d) Provincia.e) Documentos.f) Trámites. <p>4.3 Introducir datos.</p> <p>4.4 Validar datos.</p> <p>4.5 Reservar Cita luego de haber introducido los datos.</p> <p>4.6 Mostrar mensaje de confirmación.</p>	
Validaciones	Validar que los datos introducidos estén correctos.	
Post-condiciones	Se realizó la solicitud.	

Capítulo 2 Análisis y Diseño

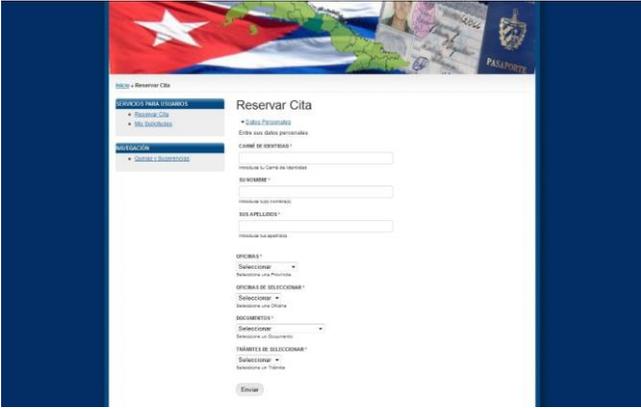
<p>Prototipo</p>	 <p>Seleccionar la opción “Reservar Cita”.</p>  <p>Mostrar los datos si se selecciona la opción “Reservar Cita”</p>  <p>Mostrar mensaje de confirmación.</p>
------------------	---

Tabla 4 Descripción del requisito del portal “Solicitar Trámites”

RFSW1 Solicitar trámites.

Capítulo 2 Análisis y Diseño

Propósito	Que una persona pueda solicitar un trámite.	
Pre-condiciones	-	
Conceptos tratados	Concepto	Atributos
	Persona	Número de identidad Nombre(s) Apellidos Solicitudes
	Solicitud	Identificador Identificador del tipo proceso Identificador del tipo trámite Oficina Fecha de registro Fecha inicio del trámite Propuesta de fecha fin del trámite Estados Turno
Descripción	<p>1.1 Entrar parámetros para la solicitud.</p> <p>a) Número de identidad.</p> <p>b) Nombre(s).</p> <p>c) Apellidos.</p> <p>d) Solicitudes.</p> <p> i. Oficina.</p> <p> ii. Identificador del tipo proceso.</p> <p> iii. Identificador del tipo trámite.</p> <p>1.2 Lanzar una excepción si los datos validados no son correctos.</p> <p>1.3 Ir al requisito funcional “<i>Administrar turnos</i>”, si los datos validados son correctos.</p> <p>1.4 Capturar, almacenar en un log y lanzar, si devuelve una excepción los pasos 1.1 y 1.2.</p> <p>1.5 Ir a los pasos 1.1 y 1.2, si no devuelve una excepción:</p>	

Capítulo 2 Análisis y Diseño

	<p>1.6 Devolver una persona con sus solicitudes y por cada una solicitud devolver los datos entrados y los que se generan.</p> <ul style="list-style-type: none">a) Número de identidad.b) Nombre(s).c) Apellidos.d) Solicitudes.<ul style="list-style-type: none">i. Identificador.ii. Identificador del tipo proceso.iii. Identificador del tipo trámite.iv. Oficina.v. Fecha registro.vi. Fecha de inicio trámite.vii. Propuesta de fecha fin del trámite.viii. Estados.ix. Turno.
Validaciones	<ol style="list-style-type: none">1. El número de identidad de la persona no puede ser nulo y debe tener 11 dígitos.2. Se debe entrar al menos una solicitud.3. La oficina de las solicitudes no pueden ser nula.4. El identificador del tipo de trámite y del tipo de proceso de las solicitudes no puede ser nulo.5. El número de identidad de la persona debe estar registrado en la base de datos del SUIN.
Post-condiciones	Devuelve los datos de la persona así como los trámites que solicitaron con los valores generados (turno y fecha para trámite).
Prototipo	No procede.

2.3.4 ESPECIFICACIÓN DE REQUISITOS NO FUNCIONALES

Capítulo 2 Análisis y Diseño

Los requisitos no funcionales son propiedades o cualidades que debe tener el producto, estas propiedades son las características que hacen al sistema atractivo, usable, rápido o confiable. En la definición de los requisitos no funcionales se tiene en cuenta aspectos como la usabilidad, la disponibilidad, la fiabilidad, la eficiencia, la interfaz, los requisitos de licencia, los requisitos legales de derecho de autor y el soporte. A continuación se muestra los requisitos establecidos por las políticas del proyecto.

Usabilidad

RnF1. El portal de trámites podrá ser utilizado por cualquier usuario, contando con los conocimientos básicos relativos al uso de una computadora.

RnF2. El portal será publicado en idioma español, aunque estará preparado para soportar el multilinguaje.

RnF3. El portal de trámites poseerá estructura y diseño homogéneos en todas sus pantallas, que facilite la navegación:

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

Disponibilidad

RnF4. No se realizarán mantenimientos preventivos en horarios donde se encuentre funcionando el portal, y deberán ejecutarse en un horario estipulado, para no afectar la disponibilidad del portal.

RnF5. El portal de trámites debe estar disponible las 24 horas durante los 7 días de la semana.

Fiabilidad

RnF6. Las fallas del portal se dividirán en dos categorías:

- **Simple:** la solución y la actualización se realizarán en línea en un período inferior a 10 minutos.
- **Complejas:** la solución y actualización se realizarán en un tiempo que se definirá posterior a una evaluación detallada.

RnF7. Solo podrán acceder a las funcionalidades del portal, los usuarios que posean los permisos suficientes.

Capítulo 2 Análisis y Diseño

RnF8. Se garantizará la consistencia de los datos, se realizarán comprobaciones y validaciones automáticas en todos los casos posibles.

Interfaz

RnF9. Todas las interfaces de usuario que se definan para el sistema respetarán los patrones de diseño establecidos.

RnF10. La plantilla del portal contendrán claro y bien estructurados los datos, y al mismo tiempo permitirán la interpretación correcta e inequívoca de la información.

RnF11. Especificar el diseño de la interfaz de usuario que buscará la ejecución de acciones de una manera rápida, minimizando los pasos a dar en cada proceso.

RnF12. Mostrar todos los textos y mensajes en pantalla en idioma castellano.

RnF13. Diseñar su funcionamiento de modo que sea intuitivo y requiera de información mínima.

Seguridad

RnF14. Chequear que el usuario este autenticado antes de realizar alguna acción sobre el portal.

RnF15. La información manejada por el portal estará protegida de acceso no autorizado y divulgación.

RnF16. Garantizar que las funcionalidades del portal se muestren de acuerdo al tipo de usuario que se encuentre activo.

Soporte

RnF17. Debe poseer documentos técnicos.

RnF18. Debe poseer el manual de usuario.

RnF19. Sistema multiplataforma.

Software

RnF20. Estaciones de trabajo (PC Cliente).

RnF21. Sistema operativo: Multiplataforma.

RnF22. Navegador web: Internet Explorer 7.0 o superior, Mozilla Firefox 8.0 o superior, Chrome 16.0 o superior.

Capítulo 2 Análisis y Diseño

RnF23. Servidor de aplicaciones: Xampp 1.7.7.

RnF24. Versión de PHP 5.3.8.

RnF25. Sistema Gestor de Bases de Datos: Oracle *Database* 11g.

2.4 ARQUITECTURA DE LA SOLUCIÓN

Drupal es un sistema modular. El sistema incluye un conjunto de herramientas y pautas que se deben seguir para desarrollar e integrar nuevas funcionalidades a través de módulos adicionales. Sus principales componentes de arquitectura son: núcleo, entidades, tipos de contenido, módulos, bloques, menús, temas, usuarios y permisos.[24]

Los **módulos** operan sobre los **nodos** y le dan funcionalidad a Drupal permitiendo incrementar su funcionalidad o adaptarlas a la necesidad de cada sitio web, es sobre esta capa que se aplica el módulo implementado. La capa anterior, la constituyen **bloques** y **menús**, a través de estos se puede estructurar y organizar el contenido, además de acceder o mostrar la información generada por los módulos a partir de la información almacenada en los nodos. La siguiente capa es la de **control de usuarios y permisos**. Drupal dispone de un registro de usuarios y roles que permiten especificar qué tareas pueden realizar de las que ofrecen los módulos instalados en el sitio, así como el contenido que se puede crear, al que se puede acceder o mostrar, el que puede visualizar cada usuario o tipo de usuario; es lo mismo que decir que esta capa controla las operaciones (visualización, edición, creación, etc.), que un usuario puede realizar sobre elementos de las capas inferiores. La última capa, es la capa que establece la apariencia o estilo de la información a mostrar al usuario. Esta separación entre información y aspecto permite cambiar la apariencia de un sitio web sin modificar el contenido.[44]

Capítulo 2 Análisis y Diseño

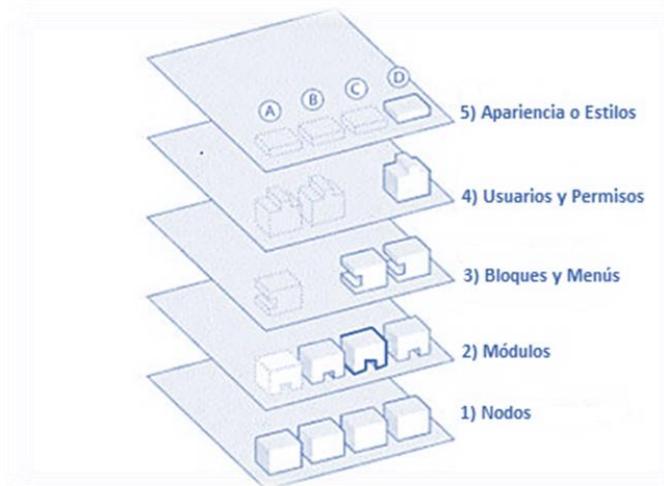


Ilustración 4 Arquitectura de Drupal

2.5 PATRONES DE DISEÑO EN DRUPAL

Los patrones de diseño, son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de *software* y otros ámbitos referentes al diseño de interacción o interfaces. Son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos, pueden incrementar o disminuir la capacidad de un diseño o una implementación, separar mejor los conceptos y simplificar la descripción. Se debe tener en cuenta que hacer uso de patrones de diseño disminuye en muchos casos el rendimiento de las aplicaciones y por ello se debe tener un equilibrio entre la flexibilidad y el rendimiento dentro del sistema, esto significa que no es razonable utilizar patrones hasta que no se tenga un entendimiento y dominio de la solución.[45]

Para llevar a cabo un buen diseño de *software* se ha definido una serie de patrones, estos constituyen un conjunto de principios generales y expresiones que ayudan a desarrollar mejor el sistema.

Los patrones de diseño que utiliza la arquitectura de Drupal son:[46]

- **Singleton** (instancia única): está diseñado para restringir la creación de objetos pertenecientes a una clase, mandando a que se pueda crear una única instancia. Su

Capítulo 2 Análisis y Diseño

intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella.[47]

- **Decorator:** Drupal hace amplio uso de este patrón, este responde a la necesidad de añadir dinámicamente funcionalidad a un objeto; esto permite no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera. Su intención es proporcionar una forma flexible de introducir o eliminar funcionalidad de un componente sin modificar su apariencia externa o su función.[47]
- **Observer** (observador): es generalizado en Drupal, define una dependencia del tipo uno-a-muchos entre objetos, de manera que cuando uno de los objetos cambia su estado, el observador se encarga de notificar este cambio a todos los otros dependientes. Su intención es proporcionar a los componentes una forma flexible de enviar mensajes de difusión a los receptores interesados.[47]
- **Bridge** (puente): la capa de abstracción de bases de datos de Drupal se aplica de una forma similar al patrón de diseño Bridge. Desvincula una abstracción de su implementación, de manera que ambas puedan variar de forma independiente. Su intención es disociar una abstracción de su aplicación, de manera que los dos pueden variar independientemente y publicar una interfaz en la jerarquía de herencia, e insertar en su propia aplicación herencia jerarquía.[47]
- **Chain of Responsibility** (cadena de responsabilidad): permite establecer una cadena de objetos receptores a través de los cuales se pasa una petición formulada por un objeto emisor. Cualquiera de los objetos receptores puede responder a la petición en función de un criterio establecido. Evita acoplar el emisor de una petición a su receptor, al dar a más de un objeto la posibilidad de responder a la petición. Crea una cadena con los objetos receptores y pasa la petición a través de la cadena hasta que esta sea tratada por algún objeto. Su intención es establecer una cadena en un sistema, para que un mensaje pueda ser manejado en el nivel en el que se recibe en primer lugar, o ser redirigido a un objeto que pueda manejarlo.[47]

Capítulo 2 Análisis y Diseño

- **Command:** permite solicitar una operación a un objeto sin conocer realmente el contenido de esta operación, ni el receptor real de la misma. Encapsular un mensaje como un objeto, con lo que se gestiona colas o registros de mensajes y deshacer operaciones. Ofrece una interfaz común que permita invocar las acciones de forma uniforme y extender el sistema con nuevas acciones de forma más sencilla.[47]

2.6 DIAGRAMA DE CLASES

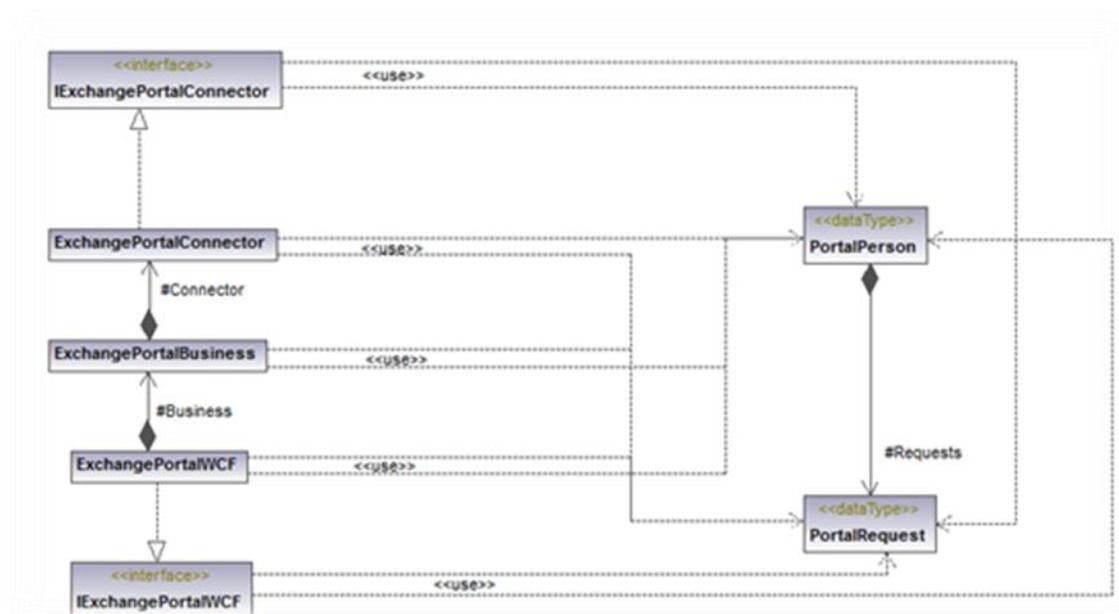


Ilustración 5 Diagrama de Clases

- **ExchangePortalWCF:** Clase que implementa de la interfaz **IExchangePortalWCF** para realizar la función de exponer el servicio con WCF. Esta consume las funcionalidades del servicio de la clase **ExchangePortalBusiness** brindándole transparencia a la presentación del servicio.
- **ExchangePortalBusiness:** Se encarga de implementar la lógica de negocio del servicio o en caso de ser un servicio de acceso a datos realiza función de mediador entre las capas Acceso a Datos y Presentación.

Capítulo 2 Análisis y Diseño

- **ExchangePortalConnector:** Implementa de **IExchangePortalConnector** y es el encargado de interactuar con el modelo y las entidades mapeadas por el *Entity Framework*, realizando así todo el trabajo de acceso a datos del servicio. Este desarrolla las funcionalidades del servicio.
- **PortalPerson:** Clase que representa el tipo de dato persona. Contiene todas las propiedades de una persona necesarias para poder solicitar un trámite. Dentro de las propiedades se encuentra una lista de **PortalRequest**.
- **PortalRequest:** Clase que representa el tipo de dato solicitud. Contiene todas las propiedades de una solicitud de un trámite.

2.7 MODELO DE DATOS

En la rama de la ciencia informática un modelo de datos es un lenguaje usado para la descripción de una base de datos, permitiendo la representación de las estructuras, las restricciones de integridad y las operaciones de manipulación de los datos.[48]

Se propone un modelo de datos con el objetivo de validar los requisitos del cliente, en donde queda claro la estructura y el diseño de la base de datos a utilizar en el almacenamiento de datos del portal.

A continuación se presentan en el modelo de datos las nuevas entidades que fueron creadas para dar solución al problema que se modela.

Capítulo 2 Análisis y Diseño

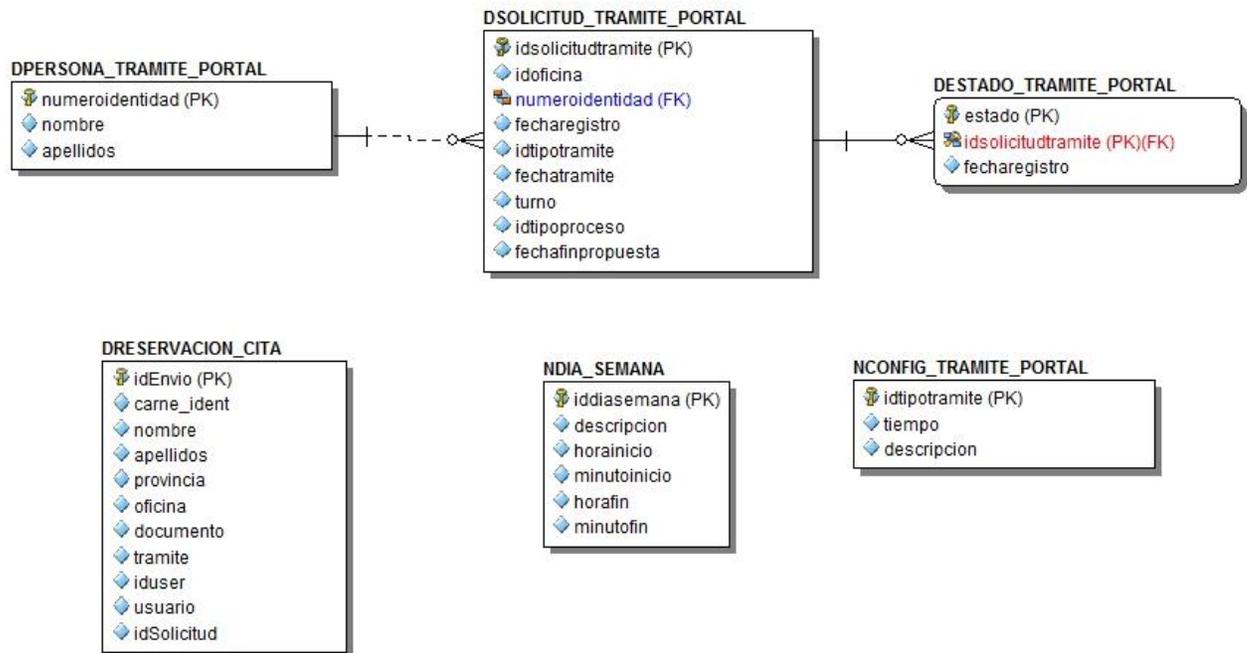


Ilustración 6 Modelo de Datos

2.7.1 DESCRIPCIÓN DE ENTIDADES

En la siguiente tabla se realiza una descripción de la entidad DESTADO_TRAMITE_PORTAL del modelo de datos perteneciente a la aplicación. **Ver Anexo II** para las restantes entidades.

Nombre: DESTADO_TRAMITE_PORTAL		
Descripción: se almacenan los estados de los trámites que se encuentran solicitados.		
Atributo	Tipo	Descripción
IDSOLICITUDTRAMITE	RAW(16)	Identificador del trámite solicitado.
ESTADO	VARCHAR2(100)	Estado que tiene el trámite.
FECHAREGISTRO	DATE	Fecha en que se registró dicho estado.

Tabla 5 Descripción de la Entidad DESTADO_TRAMITE_PORTAL

Capítulo 2 Análisis y Diseño

2.8 CONCLUSIONES PARCIALES

Con la realización de la propuesta de solución, se pudo tener una mejor comprensión y entendimiento de la aplicación a desarrollar previa a su implementación, tratando de evitar que se desarrollen funcionalidades innecesarias. Con el desarrollo del modelo de dominio se obtuvo una representación general de las entidades y conceptos que interceden en la creación del portal, se pudo unificar el vocabulario entre los usuarios y los desarrolladores para deducir el contexto en que se emplaza el sistema. Con la descripción de los requisitos funcionales y no funcionales se permitió determinar las principales funcionalidades que brindará el portal para darle cumplimiento a los principales objetivos propuestos. La utilización de los patrones de diseño en el desarrollo de la aplicación ayuda a gran medida en la reducción del tiempo de desarrollo, así como brindarle al portal transparencia, elegancia y un buen diseño.

Capítulo 3 Implementación y Pruebas

CAPÍTULO 3 IMPLEMENTACIÓN Y PRUEBAS

En el presente capítulo se describe la implementación del sistema a través de los estándares de codificación, el análisis del tratamiento de errores, el diagrama de despliegue y de componentes. Se realizan las pruebas unitarias al código del servicio web, las pruebas de aceptación al portal y las pruebas de caja negra con la utilización de diseños de casos de pruebas.

3.1 ESTÁNDARES DE CODIFICACIÓN

La legibilidad del código fuente incide directamente en como un programador comprende el sistema, su mantenimiento es la facilidad con que el *software* puede modificarse para añadirle nuevas características y depurar errores o mejorar el rendimiento. Un aspecto del desarrollo de *software* en la que todos los desarrolladores influyen es en la técnica de codificación, asegurando utilizar el mejor método para que un equipo de programadores mantenga la calidad del código. Este método es establecer los estándares de codificación, sobre el que se efectuará diariamente revisiones del código.

Usar técnicas de codificación consistentes y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad de la aplicación y para obtener un buen rendimiento.[49]

Con el objetivo de facilitar el mantenimiento del *software* se empleó los siguientes estándares de codificación que utiliza el CMS Drupal:

- **Indentación:** consiste en insertar espacios en blanco o tabuladores en determinadas líneas de código para facilitar su comprensión. En programación se utiliza la indentación para anidar elementos. En Drupal debemos indentar con 2 espacios, nunca con tabuladores. Además, no se debe dejar espacios en blanco al final de cada línea. En el siguiente ejemplo se muestra un fragmento de código con las indentaciones realizadas, de 2 espacios cada una (→→), y los saltos de línea o Enter (↵) al final de cada línea.[23]

Capítulo 3 Implementación y Pruebas

```
477 function hook_views_data_alter(&$amp;data) {
478     // This example alters the title of the node:nid field in the Views UI.
479     =>&$data['node']['nid']['title'] = t('Node-Nid');
480
481     // This example adds an example field to the users table.
482     =>&$data['users']['example_field'] = array(
483         =>&&&'title' => t('Example field'),
484         =>&&&'help' => t('Some example content that references a user'),
485         =>&&&'handler' => 'hook_handlers_field_example_field',
486     );
}
```

Ilustración 7 Estándar de Codificación: Indentación

- **Etiquetas de apertura y cierre de PHP:** Cuando se escribe en PHP, siempre se debe utilizar las etiquetas `<?php` y `?>`, y en ningún caso la versión corta `<?` y `?>`. En general se omite la etiqueta de cierre de PHP (`?>`) al final de los archivos `.module` y `.inc`. Esta convención evita que se puedan quedar olvidados espacios no deseados al final del archivo, que serían identificados como salida HTML y podrían provocar un error muy típico. Por tanto, la etiqueta de cierre final del archivo (`?>`) es opcional en Drupal. No hay que confundir esto con el uso normal del lenguaje PHP en archivos que también contienen HTML.[23]

```
93 <?php if ($display_submitted): ?>
94     <div class="submitted">
95         <?php print $submitted; ?>
96     </div>
97 <?php endif; ?>
```

Ilustración 8 Estándar de Codificación: Etiquetas de apertura y cierre de PHP

- **Uso de punto y coma (;) en código PHP:** Aunque PHP permite escribir líneas de código individuales sin el terminador de línea (;), como por ejemplo `<?php print $title ?>`. En Drupal es siempre obligatorio: `<?php print $title; ?>`.[23]

```
<div class="submitted">
    <?php print $submitted; ?>
</div>
```

Ilustración 9 Estándar de Codificación: Punto y Coma

- **Uso de comillas:** Se pueden usar tanto las comillas simples ('cadenas') como las comillas dobles ("cadena") para delimitar las cadenas de caracteres. Las comillas dobles son necesarias si se desean incluir variables dentro de las cadenas de texto. Por ejemplo

Capítulo 3 Implementación y Pruebas

"<h1>\${title}</h1>". También se recomienda el uso de las comillas dobles cuando el texto puede incluir alguna comilla simple [23].

```
<div id="block-<?php print $block->module . '-' . $block->delta; ??" class="<?php print $blockhide; ?"><?php print $classes; ?"><?php print $attributes; ?>>
  <?php print render($title_prefix); ?>
  <?php if (!empty($block->subject)); ?>
    <div class="block-title">
      <<?php print $blocktag; ?> class="title"><?php print $title_attributes; ?><?php print $block->subject ?></<?php print $blocktag; ?>
    </div>
  <?php endif; ?>
  <?php print render($title_suffix); ?>
  <div class="content">
    <?php print $content; ?>
  </div>
</div> <!-- /block -->
```

Ilustración 10 Estándar de Codificación: Comillas

- **Estructuras de control:** Para utilizar estas estructuras hay que tener en cuenta cuatro normas, las cuales definen el procedimiento a seguir. La primera norma es que debe haber un espacio entre el comando que define la estructura (if, while, for, etc.) y el paréntesis de apertura. La segunda norma es que la llave de apertura “{” se situara en la misma línea que la definición de la estructura, separada por un espacio. La tercera es que se recomienda usar siempre las llaves “{}” aun en los casos en que no sea obligatorio su uso. La cuarta y última norma expresa que las estructuras **else** y **elseif** se escribirán en la línea siguiente al cierre de la sentencia anterior.[23]

```
if (isset($query_on_top)) {
    variable_set('views_ui_show_sql_query', TRUE);
    if ($query_on_top) {
        variable_set('views_ui_show_sql_query_where', 'above');
    }
    else {
        variable_set('views_ui_show_sql_query_where', 'below');
    }
    variable_del('views_ui_query_on_top');
}
```

Ilustración 11 Estándar de Codificación: Estructuras de Control

- **Funciones:** Los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por guion bajo. Se debe incluir siempre como prefijo el nombre del módulo, tema, etc., para evitar duplicidad de funciones. En su declaración, después del nombre de la

Capítulo 3 Implementación y Pruebas

función, el paréntesis de inicio de los argumentos debe ir sin espacio. Cada argumento debe ir separado por un espacio, después de la coma del argumento anterior.[23]

```
$field = field_info_instance('node', 'taxonomy_forums', $node->type);
```

Ilustración 12 Estándar de Codificación: Funciones

- **Arrays** (arreglo): Los valores dentro de un array se deben separar por un espacio (después de la coma que los separa). El operador => debe separarse por un espacio a ambos lados. Cuando la línea de declaración del array supera los 80 caracteres, cada elemento se debe escribir en una única línea, indentándolo una vez (2 espacios). En este último caso, la coma de separación del último elemento también se escribirá, aunque no existan más elementos. De esta forma se evitan errores al añadir nuevos elementos al vector.[23]

```
array('data' => t('Topic'), 'field' => 'f.title'),  
array('data' => t('Replies'), 'field' => 'f.comment_count'),  
array('data' => t('Last reply'), 'field' => 'f.last_comment_timestamp'),
```

Ilustración 13 Estándar de Codificación: Arrays

3.2 TRATAMIENTO DE ERRORES

Las excepciones son parte del diseño cuando aún la gran mayoría de ellas son identificadas en el proceso de implementación, no obstante es necesario definir con anterioridad un mecanismo efectivo para su tratamiento. En el caso de la aplicación desarrollada es necesario llevar una traza de todas aquellas excepciones que se proyecten, y en muchos casos es necesario que el usuario que está interactuando con la aplicación en ese momento especifique alguna información que tribute a la futura validación o corrección de la excepción lanzada. Para el tratamiento de excepciones se definirá una interfaz base que permita dejar una traza clara de todas las excepciones lanzadas dentro del sistema. Se evitan, minimizan y tratan los posibles errores, para garantizar la integridad y confiabilidad de la información que en este se registra y muestra.[50]

Para garantizar una mayor integridad y confiabilidad en los datos que utiliza el sistema, se adoptan varias estrategias para el tratamiento de errores:

Capítulo 3 Implementación y Pruebas

- Los mensajes de error que emitirá el sistema se mostrarán en un lenguaje de fácil comprensión para los usuarios.
- Cada vez que se introduzca información errónea en un formulario o se dejen campos vacíos, se mostrará un mensaje en color rojo debajo del campo señalado con el error.

```
function _consultar_estado($idSol) {  
    try {  
        $service="http://commonservices.ident.cu/PortalExchangeService/ExchangePortalService.svc?wsdl";  
        $client = new SoapClient($service);  
  
        $param = new stdClass();  
        $param->requestId = $idSol;  
  
        $query = $client->GetStatus($param);  
        $result = $query->GetStatusResult;  
  
        return $result->string;  
    }  
    catch (Exception $e1) {  
        return null;  
    }  
}
```

Ilustración 14 Tratamiento de errores: Consultar Estado del Trámite

3.3 IMPLEMENTACIÓN

En la fase de implementación, es donde se desarrolla el sistema en términos de componentes, los cuales pueden ser ejecutables, ficheros de código fuente, scripts, librerías, etc. En este flujo se desarrolla la arquitectura y el sistema como un todo, definiendo y aplicando patrones y estándares para la codificación para alcanzar una recomendable organización del código. En el diagrama de despliegue se muestra la situación física de los componentes lógicos desarrollados, en el diagrama de componentes se describe la organización y dependencia de cada uno de los componentes implementados representándose la distribución de los nodos físicos en los que funcionara la aplicación.

3.3.1 DIAGRAMA DE COMPONENTES

Capítulo 3 Implementación y Pruebas

Un diagrama de componente describe los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo el código fuente, binario y ejecutable. Representan todos los tipos de elementos de *software* que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, etc.[51]

Permite visualizar con más facilidad la estructura general del sistema y el comportamiento del servicio que estos componentes proporcionan y utilizan a través de las interfaces, mostrando los elementos de diseño del *software*.

El diagrama de componente hace parte de la vista física de un sistema, la cual modela la estructura de implementación de la aplicación por sí misma, su organización en componentes y su despliegue en nodos de ejecución. Esta vista proporciona la oportunidad de establecer correspondencias entre las clases y los componentes de implementación y nodos. La vista de implementación se representa con los diagramas de componentes.

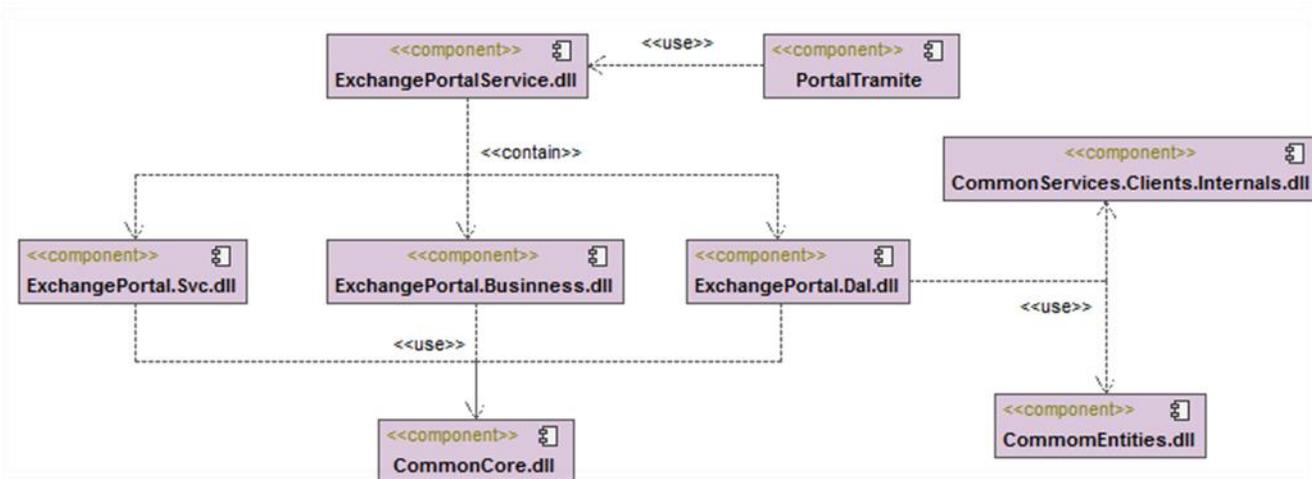


Ilustración 15 Diagrama de Componentes

Descripción de los componentes:

Como se ilustra en el diagrama de componentes el componente principal es el **PortalTramite**, donde se incluyen los módulos, temas y contenidos del portal. Este componente interactúa con el servicio web a través del componente principal del mismo **ExchangePortalService**. Este último

Capítulo 3 Implementación y Pruebas

contiene tres componentes principales: **ExchangePortal.Svc** que incluye todo lo referente a la configuración y a la exposición del servicio, **ExchangePortal.Business** que engloba toda la lógica del servicio y **ExchangePortal.Dal** que abarca el trabajo de acceso a datos. Los dos primeros componentes usan el CommonCore para el trabajo de seguimiento de errores y la identificación de instancias.

3.4 DIAGRAMA DE DESPLIEGUE

El diagrama de despliegue se utiliza para modelar el *hardware* usado en las implementaciones de sistemas y las relaciones entre sus componentes. Los elementos usados por este tipo de diagrama son nodos, componentes y asociaciones.[52] Además muestran la disposición física de los distintos nodos que entran en la composición de un sistema y el reparto de los programas ejecutables sobre estos nodos.[53]

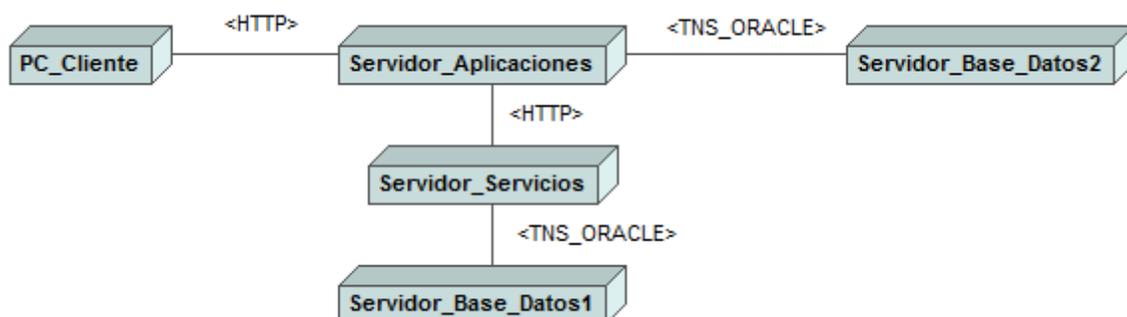


Ilustración 16 Diagrama de Despliegue

A continuación se describen cada uno de los nodos presentes en el diagrama anterior y la comunicación entre ellos:

- **PC_Cliente:** Representa las computadoras clientes que se conectan al servidor de aplicaciones nacional, las mismas se comunican con el servidor a través del protocolo HTTP.
- **Servidor_Aplicaciones:** Representa el servidor donde se encuentra desarrollado el portal de trámites, el mismo accede al servidor de base de datos para el manejo de la información mediante el protocolo TNS_ORACLE.

Capítulo 3 Implementación y Pruebas

- **Servidor_Servicios:** Representa el servidor donde se encuentra desarrollado el servicio que permite la comunicación con el SUIN, el mismo accede al servidor de base de datos para el manejo de información mediante el protocolo TNS_ORACLE.
- **Servidor_Base_Datos1:** Es donde se almacena todos los datos pertenecientes al servicio, es decir, donde se encuentra la base de datos.
- **Servidor_Base_Datos2:** Es donde se almacena todos los datos pertenecientes al portal de trámites, es decir, donde se encuentra la base de datos.

3.5 PRUEBAS

La realización de pruebas al *software* es de importancia en la calidad del mismo y representa una revisión final de las especificaciones, del diseño y de la codificación. Es generalmente considerada costosa y molesta por lo que la mejor forma de hacerlas sería en etapas tempranas y a menudo. Para la mayoría de las compañías la meta debería ser hacer el mejor trabajo de prueba posible, y entre más se pruebe el *software* mayor cantidad de errores serán encontrados.[54] Las pruebas constituyen una actividad en la cual un sistema es ejecutado bajo condiciones específicas, se observan los resultados y además se realiza una evaluación del componente. El objetivo de esta actividad es verificar los requerimientos de *software* que han sido satisfechos a través de distintos tipos de pruebas.

3.5.1 PRUEBAS UNITARIAS

Las pruebas unitarias representan una forma de probar una fracción de código, por lo que es de gran importancia para los desarrolladores y el equipo de pruebas. En estas se prueban rutas lógicas del *software* y la colaboración entre componentes, al proporcionar casos de pruebas que ejerciten conjuntos específicos de condiciones.[55]

Para la realización de las pruebas unitarias se utilizó la herramienta *Visual Studio 2010*. Estas pruebas se utilizan para ejecutar código fuente llamando directamente a los métodos de una clase, pasándole los parámetros apropiados. Los métodos de pruebas unitarias residen en clases *Test*, que se almacenan en archivos de código fuente. Esta prueba se le realizó al servicio que comunica el portal de trámites con el SUIN.

Capítulo 3 Implementación y Pruebas

Con la realización de las pruebas unitarias se comprueba la correcta ejecución de un segmento de código de la aplicación, examinando el estado del programa en varios puntos, para determinar si el estado real coincide con el esperado.[56]

```
/// <summary>
///A test for RequestLegalAction
///</summary>
[TestMethod()]
public void RequestLegalActionTest()
{
    ExchangePortalWCF target = new ExchangePortalWCF();
    PortalPerson portalPerson = new PortalPerson
    {
        IdentityNumber = "87072610331",
        LastNames = "Silva Fernández",
        Name = "Yanet",
        Requests = new List<PortalRequest>
        {
            new PortalRequest
            {
                OfficeId = "1",
                LegalActionId = 1
            }
        }
    };

    PortalPerson actual;
    try
    {
        actual = target.RequestLegalAction(portalPerson);
    }
    catch (Exception)
    {
        actual = null;
    }
    Assert.IsNotNull(actual);
    //Assert.Inconclusive("Verify the correctness of this test method.");
}
```

Ilustración 17 Prueba unitaria de la funcionalidad Solicitar Trámites del servicio web

Capítulo 3 Implementación y Pruebas

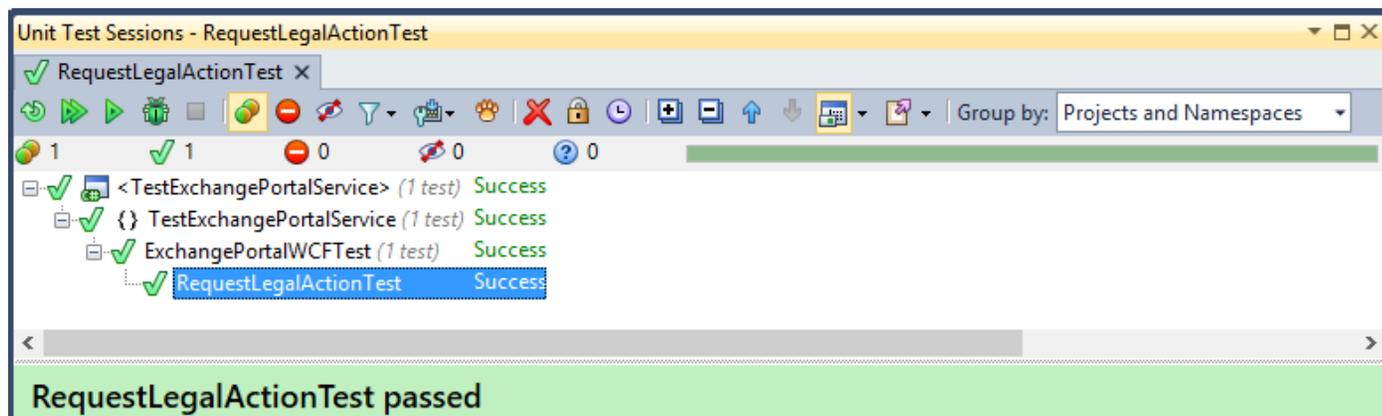


Ilustración 18 Resultado de la prueba unitaria de la funcionalidad Solicitar Trámites

3.5.2 PRUEBAS DE ACEPTACIÓN

Prueba de aceptación, es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el *software* está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el *software* fue construido. **Ver Anexo III**

3.5.3 PRUEBAS DE CAJA NEGRA

Estas pruebas principalmente se centran en las funcionalidades que se espera de un sistema, es decir, intentan encontrar casos en que el sistema no se atiende a su especificación. Por tanto se denomina pruebas funcionales, y el probador se limita a abastecer los datos como entrada y estudiar a la salida, sin preocuparse de lo que pueda estar haciendo el sistema por dentro.[57], [58] Se enfocan directamente en el exterior del módulo, sin importar el código; una prueba caja negra examina algún aspecto funcional de un sistema que tiene poca relación con la estructura lógica interna del *software*. Estas pruebas se centran en los requisitos fundamentales del *software* y permite obtener entradas que prueben estos requisitos. Con este tipo de pruebas se intenta encontrar:[59]

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a la base de datos externa.

Capítulo 3 Implementación y Pruebas

- Errores de rendimiento.
- Errores de inicialización y terminación.

3.5.3.1 DISEÑO DE CASOS DE PRUEBAS

Un caso de prueba es un conjunto de entradas, condiciones de ejecución y resultados esperados desarrollados para un objetivo en particular.[60] Cada técnica de prueba proporciona unos criterios desiguales para generar casos o datos de prueba. El objetivo es diseñar pruebas que tengan la mayor probabilidad de encontrar el mayor número de errores con la mínima cantidad de esfuerzo y de tiempo. Cada caso de prueba va acompañado del resultado que ha de producir el *software* al ejecutarlo para detectar un posible fallo en el programa. Tiene el propósito de comprobar todos los requisitos de la aplicación, donde el analista determinará si el requisito de la aplicación es parcial o completamente satisfactorio.

A continuación se muestra el diseño de caso de pruebas para Solicitar Trámites. Para ver los casos de pruebas restantes.

Escenario	Descripción	Variante 1	Variante 2	Variante 3	Variante 4	Variante 5	Variante 6	Variante 7	Respuesta del sistema	Flujo central
EC 4.1	Permite reservar una cita.								Se muestra los datos personales que se llenan para reservar una cita.	4.1 Selección de la opción "Reservar Cita".
EC 4.2	Permite	Carn	Nom	Apellido	Ofici	Oficina	Docum	Trám	Se	4.2 Mostra

Capítulo 3 Implementación y Pruebas

Mostrar los datos si se selecciona la opción "Reservar Cita".	llenar los datos para reservar una cita.	é de identidad.	bre.	dos.	nas.	s de seleccionar.	entos.	ites.	muestran los campos que debe llenar para reservar una cita y enviarla.	r los datos si se selecciona la opción "Reservar Cita". a) Carné de Identidad. b) Nombre. c) Apellido. d) Provincia. e) Documentos. f) Trámites.
		I	V	V	N/A	N/A	N/A	N/A		4.3 Introducir datos. 4.4 Validar datos.
EC 4.3 Reservar cita luego de haber introducido	Permite reservar la cita.								Se reserva la cita y se muestra un mensaje de solicitud registrada	4.5 Reservar Cita luego de haber introducido los datos. 4.6 Mostrar mensaje de confirmación.

Capítulo 3 Implementación y Pruebas

los datos.									satisfactoriamente.	
------------	--	--	--	--	--	--	--	--	---------------------	--

Tabla 6 Diseño de caso de pruebas "Solicitar Trámites"

3.5.4 RESULTADO DE LAS PRUEBAS

Se realizaron tres iteraciones de pruebas donde se detectaron varias no conformidades, las cuales fueron mitigadas durante la etapa de desarrollo. En la primera iteración se detectaron un total de 15 no conformidades. Posteriormente en la segunda iteración se rectificaron los errores encontrados en la pasada iteración, aunque se detectaron 6 no conformidades nuevas. Por último se realizó una tercera iteración mostrando resultados satisfactorios. La Ilustración muestra cómo se fueron aminorando las no conformidades detectadas, con el objetivo de eliminar los errores encontrados y garantizar la calidad de la aplicación.



Ilustración 19 Resultados de las No Conformidades por Iteraciones.

3.6 BENEFICIOS DEL PORTAL

Capítulo 3 Implementación y Pruebas

El desarrollo del Portal de Trámites para la República de Cuba contribuye a un Gobierno en Línea permitiendo:

- Tener conocimiento sobre los documentos que se pueden solicitar, así como la información de los trámites asociados.
- Solicitar trámites, consultar el estado de cada uno y cancelarlo.
- Realizar quejas y sugerencias en línea.

El desarrollo del servicio web permite:

- Comunicar al portal con el Sistema Único de Identificación Nacional posibilitando la consulta de los estados de los trámites solicitados a través del portal.
- Administrar los turnos de las oficinas.

3.7 CONCLUSIONES PARCIALES

En esta etapa de implementación se llega a la solución final del portal, describiéndose cada uno de los componentes, siguiendo la arquitectura definida, lo que posibilitó la obtención de un sistema que garantizará la solicitud, el control y supervisión de los trámites que el ciudadano realice. El uso de los distintos estándares de codificación aportó legibilidad y limpieza en el código generado. Se realizó un tratamiento de errores, utilizados por el equipo de desarrollo del proyecto para el desarrollo de la aplicación, logrando la calidad requerida. La aplicación de las pruebas permitió exponer los errores presentes en la implementación, las cuales ayudaron a darles cumplimiento a los principales objetivos trazados y brindando a la aplicación un significativo nivel de calidad. Se verificó la conformidad con los requisitos durante la realización de las pruebas.

Conclusiones

CONCLUSIONES

- El estudio de portales de trámites arrojó la necesidad de implementar un portal para la República de Cuba que se adaptara a los trámites que se llevan en el país y permitiera la comunicación con el SUIN.
- La mejor opción para el desarrollo del portal es la utilización del CMS Drupal en su versión 7 principalmente por contener un módulo que permite la comunicación con el SGBD Oracle y para el hospedaje se definió el servidor web XAMPP por ser multiplataforma.
- Como mecanismo de comunicación entre el portal y el SUIN se desarrolló un servicio web, teniendo en cuenta con ello la arquitectura que utiliza este sistema.
- El desarrollo del servicio web se ajustó a la metodología, arquitectura y herramientas seleccionadas por el proyecto.
- Para un mejor entendimiento de la problemática se realizó el modelo del dominio especificándose los principales conceptos que interceden en la creación de la solución.
- A partir de este modelo de dominio se determinaron los requisitos funcionales con los que debe cumplir el portal.
- La validación de la solución mediante las pruebas realizadas garantizan la calidad del portal y del servicio web.
- Con el desarrollo del portal y del servicio web se logra el control de flujo de personas en las Oficinas de Carné de Identidad, así como la obtención del estado de opinión de los ciudadanos.

Recomendaciones

RECOMENDACIONES

Los objetivos de este trabajo han sido logrados de manera satisfactoria, teniendo en cuenta que se cumplieron todos los requerimientos planteados. Como resultado del proceso de investigación y realización del portal han surgido ideas que serían recomendables tener en cuenta para una futura versión del mismo, para ello se recomienda:

- Continuar el desarrollo de este sistema, adicionándole nuevas funcionalidades e integrándole servicios que sean necesarios para el país.
- Restringir la solicitud de trámites en donde los usuarios puedan solicitar solamente trámites para ellos o para menores.
- Optimizar el algoritmo de administración de turnos de acuerdo a un estudio que se realice previamente de los trámites en cada oficina.
- Integrar el portal con el Módulo para la Evaluación de la Satisfacción del Cliente del Sistema Único de Identificación Nacional.

Referencia Bibliográfica

REFERENCIA BIBLIOGRÁFICA

1. Salinas Ibáñez, J., Revista de Universidad y Sociedad del Conocimiento. 2004.
2. Nielsen, J., Usabilidad. Diseño de páginas web, ed. P. Hall. 2000.
3. Gayo Avello, D., Desarrollo del portal web de la E.U. de Ingeniería Técnica en Informática de Oviedo. 2001.
4. García-Gasco Bretón, P., EVALUACION DE PORTALES EDUCATIVOS UN ESTUDIO CONCRETO: ORIENTAREDES. 2005, Valencia.
5. García Gómez, J.C., PORTAL: DEFINICIÓN, EVOLUCIÓN Y CLASIFICACIÓN¹, ed. C.L.p.d.i.F.D.U.d.M. 2, Murcia.
6. Dr. Brito Febles, O., La Técnica Criminalística., ed. F.d.D.U.d.L. Habana. 2001.
7. Servicio administrativo de identificación, migración y extranjería (SAIME). Available from: www.saime.gob.ve/.
8. PORTAL DE SERVICIOS AL CIUDADANO Y EMPRESAS. . Available from: www.serviciosalciudadano.gob.pe.
9. Portal ciudadano del Gobierno del Estado de México. Available from: www.edomex.gob.mx/edomex.
10. Portal de trámite del Gobierno de Santa Fe. Available from: www.santafe.gov.ar/tramites/.
11. Ing. Silva Fernández, Y.y.I.T.M., Daniel R, Tesis CommonCore: Paquete de servicios comunes para el Sistema Único de Identificación Nacional de Cuba. 2011.
12. Casalet, M.y.L.R., Felipe, Tecnología: Concepto, Problemas y Perspectivas. en coedición con el centro de investigaciones interdisciplinarias, ed. s.a.d.c.v. siglo xxi editores. 1998, Mexico.
13. Colomina Pardo, O., Tema 1: HTTP y aplicaciones web. 2011.
14. Martínez Usero, J.Á.y.L.N., Pablo, La producción de contenidos web, ed. E. UOC. 2007, Rambla del Poblenou, Barcelona.
15. Pastor Sánchez, J.A., Drupal 7: Web Semántica al alcance de todos. 2011.
16. Luján, S., Programación de aplicaciones web., ed. E.C. Universitario. 2002, España.
17. Lafosse, J., Expert IT Struts 2 - El framework de desarrollo de aplicaciones Java EE. 2010, Barcelona.
18. Shepherd, G., Microsoft® ASP.NET 4 Step by Step.
19. Gil García, G., El gran libro de Joomla! 1.6. Vol. Volumen 212 de Manual Imprescindible.
20. Harwani, B. and D.B. Moyano, Joomla! 2010: Grupo Anaya Comercial.
21. White Steve, W.A., Manual del usuario Joomla 1.0.x. . 2006, Barcelona.
22. Tomlinson, T., Drupal 7. 2012: Grupo Anaya Comercial.

Referencia Bibliográfica

23. Gil Rodríguez, F., Experto en Drupal 7, Nivel Avanzado, Curso de creacion y gestion de portales web con Drupal 7, ed. C.-F. S.L. Vol. Nivel Avanzado. 2011.
24. Fernández, M.M., Ejemplos de uso de Drupal para el desarrollo de aplicaciones web. Vol. Vol 1. 2008.
25. Rull, L., R. Poveda, and R. Valdivia, WordPress para Dummies. 2012: Grupo Planeta.
26. Pons Capote, O., Introducción a las bases de datos: el modelo relacional. 2005.
27. Ill, C., Oracle Oracle database 11g Press Kit, CALIDAD Y SEGURIDAD A NIVEL DE FILAS EN BBDD 2009, Madrid.
28. Lázaro Laporta, J.y.M.A., Marcel, Fundamentos de telemática.
29. Paz Loboguerrero, A.F., Metodología Agil para equipos pequeños usando plataformas Microsoft. 2011.
30. Villarroel Gonzalez, L.R.M.Y., Cesar Alejandro., APLICACIÓN DE LA METODOLOGÍA MSF V4.0 A LA DEFINICIÓN E IMPLEMENTACIÓN DE ARQUITECTURAS ORIENTADAS A OBJETOS EN VISUAL STUDIO .NET 2005, CASO PRÁCTICO G5 SHARING FILES. 2008.
31. Zukowski, J., Programación, Java ed. E.F. Varela. Vol. 1.4 V-I-II-III. 2007.
32. Henst. Introducción al PHP. 2006; Available from: <http://www.maestrosdelweb.com/editorial/phpintro/>, <http://www.php.net/manual/es/faq.general.php>.
33. Afergan, B., Programación en Web, ed. P.-H. Hispanoamericana. 1998, Mexico.
34. Garbage. Available from: http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgbd.php.
35. 2009; Available from: http://www.librosweb.es/css/capitulo_1/como_incluir_css_en_un_documento_xhtml.html, http://www.librosweb.es/css/capitulo_1/soporte_de_css_en_los_navegadores.html.
36. (C), M. 2013; Available from: <http://msdn.microsoft.com/es-es/library/z1zx9t92%28v=vs.80%29.aspx>.
37. Microsoft. 2013; Available from: <http://msdn.microsoft.com/es-es/library/bb655883%28v=vs.90%29.aspx>.
38. Edicion 2010 de la Empresa de Altova Umodel 2010; Available from: [http:// es.download.cnet.com/ALTOVA-UMODEL-enterprise-edition/3000-23834-3121015.html#ixzz2LuCyRxeZ](http://es.download.cnet.com/ALTOVA-UMODEL-enterprise-edition/3000-23834-3121015.html#ixzz2LuCyRxeZ).
39. Blog de Soma en español. Nuevas ofertas para Visual Studio 2010. 2010; Available from: <http://blogs.msdn.com/b/somaespanol/archive/2010/03/02/nuevas-ofertas-para-visual-studio-2010.aspx>.
40. Microsoft, A. 2013; Available from: <http://msdn.microsoft.com/es-es/library/bb399572%28v=vs.100%29.aspx>.

Referencia Bibliográfica

41. Campaña Anguisasaca, R.M., ANALISIS, DISEÑO E IMPLEMENTACION DE UN MODULO DE GENERACION, RECOLECCION Y EXAMINACION DE ENCUESTAS QUE SE INTEGRARA AL SISTEMA DE ADMINISTRACION DE CONTENIDOS. 2010.
42. Quejas y Sugerencias. 2006.
43. Thayer, R.H.y.D., Merlin, IEEE Software Requirement Engineering Second Edition ed. 1997, New York
44. Buytaert, D. Drupal Community. Drupal Api. 2013; Available from: <http://api.drupal.org/api/drupal/includes%21module.inc/group/hooks/7>.
45. Tramullas, J., Drupal para bibliotecas y archivos, ed. [Zaragoza] : Grupo de investigación sobre Gestión de Recursos de Información en las Organizaciones (Universidad de Zaragoza) - Fundación Zaragoza Ciudad del Conocimiento. 2010.
46. Gamma, E.H., R.; Johnson, R.; Vlissides, J., Patrones de Diseño Elementos de software orientado a elementos reutilizables., ed. P.E.S.M. 200263, Madrid.
47. Morales Morales, A. 2007; Available from: <http://jitcode.blogspot.com/2010/06/poo-y-patrones-de-diseno-en-drupal.html>.
48. Date, C.J., Introducción a los Sistemas de Bases de Datos. 2008.
49. Deitel, H.M., Deitel, Paul J Cómo programar? 2004, Mexico.
50. Seco, J.A.; Available from: <http://www.clikear.com/manuales/csharp/index.aspx>.
51. Gracia Mejía, J.F., Diseno e implementacion de un sistima de posicion de los dedos se la mano con aplicaciones en el tratamiento de trastornos de procesamiento auditivo-visual.
52. Jacobson, I., El proceso unificado de desarrollo de Software. 2000.
53. Marca Huallpara, H.M.y.Q.L., Nancy Susana.; Available from: http://www.google.com/cu/url?sa=t&source=web&cd=9&ved=0CEUQFjAI&url=http%3A%2F%2Fvirtual.usal.esiana.edu.bo%2Fweb%2Fpractica%2Farchiv%2Fdespliegue.doc&rct=j&q=diagrama%20de%20despliegue&ei=6_TZTfmpGcautwer45jpDg&usg=AFQjCNGu7s__JDdYCEwEmbKynxa2XI2Q2Q&cad=r.
54. Alonso Amo, F.y.M.N., Loic., Introducción a la Ingeniería del Software: Modelos de desarrollo de programas. 2005, Madrid.
55. Tipos de Pruebas de Software. 2004.
56. Joskowicz, J., Reglas y Prácticas en eXtreme Programming. 2011.
57. Pressman, R., Ingeniería del Software: Un Enfoque Practico. 2006.
58. López, C., Pruebas de Caja Negra: Una Experiencia Real en Laboratorio. 2000.

Referencia Bibliográfica

59. Técnicas Cuantitativas para la Gestión en la Ingeniería del Software. 2007.
60. Sommerville, I., Ingeniería del Software. 6ª Edición. 2002.

Glosario de Términos

GLOSARIO DE TÉRMINOS

- **Base de Datos (Database):** Una base de datos es una colección organizada de información. Ésta contiene una colección de registros que puede buscar, ordenar y analizar rápidamente.
- **CMS:** Un Sistema de gestión de contenidos (Content Management System en inglés, abreviado CMS) es un programa que permite crear una estructura de soporte (framework) para la creación y administración de contenidos por parte de los participantes principalmente en páginas web.
- **CSS:** Hoja de Estilo en Cascada del inglés (Cascading Style Sheets). Se utiliza para definir el estilo dentro de las plantillas de diseño.
- **HTML:** HyperText Markup Language (lenguaje de marcas hipertextuales), diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.
- **Http:** Protocolo de Transmisión Hipertexto. Protocolo de comunicaciones utilizado por los programas clientes y servidores de WWW para intercambiar archivos (texto, gráfica, imágenes, sonido, video y otros archivos multimedia).
- **Internet:** Método de interconexión de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP y garantiza que redes físicas heterogéneas funcionen como una red (lógica) única.
- **PHP:** Lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas.
- **Software:** Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.
- **UML:** “Unified Modeling Language” Lenguaje gráfico que brinda un vocabulario y reglas para especificar, construir, visualizar y documentar los artefactos de un sistema utilizando el enfoque orientado a objetos.

Anexos

ANEXOS

Anexo I. ESPECIFICACIÓN DE REQUISITOS FUNCIONALES

RF1 Registrar usuario.

Propósito	Registrar los usuarios al sistema.	
Roles	Usuario no registrado.	
Pre-condiciones	El usuario no puede estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	USERS	Uid Name Pass Mail
Descripción	<ol style="list-style-type: none">1.1 Seleccionar la opción "Crear nueva cuenta", en el formulario "Inicio de sesión".1.2 Mostrar datos del registro:<ol style="list-style-type: none">a) Nombre de usuario.b) Dirección de Correo Electrónico.1.3 Introducir datos de registro.<ol style="list-style-type: none">a) Nombre de usuario.b) Dirección de Correo Electrónico.1.4 Guardar datos si se selecciona la opción "Crear nueva cuenta".1.5 Mostrar mensaje de confirmación.	
Validaciones	Validar que los datos introducidos sean válidos.	
Post-condiciones	Se registró el usuario.	

Anexos

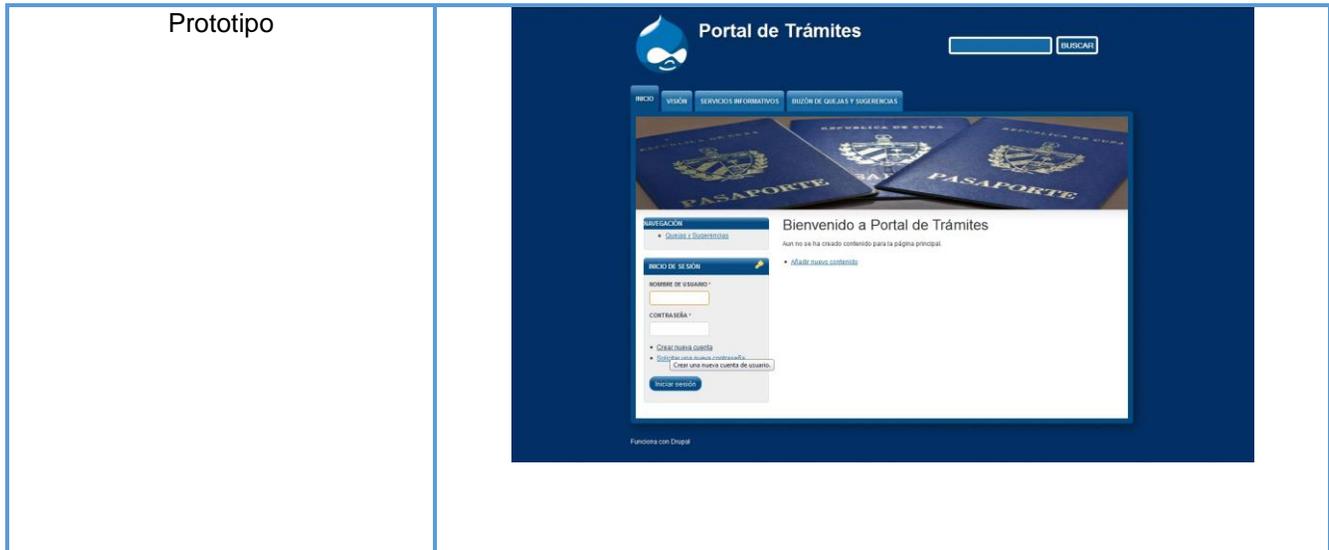


Tabla 7 Descripción del requisito del portal “Registrar Usuario”

RF2 Autenticar usuario.

Propósito	Autenticar usuario.	
Roles	Usuario registrado.	
Pre-condiciones	El usuario tiene que estar registrado en el sistema.	
Entidades tratadas	Entidad	Atributos
	USERS	Uid Name Pass Mail
Descripción	2.1 Ingresar datos, en el formulario “Inicio de sesión”: a) Usuario. b) Contraseña. 2.2 Iniciar sesión si se selecciona la opción “Iniciar sesión”.	
Validaciones	Validar que los datos introducidos sean válidos.	
Post-condiciones	Se autenticó el usuario.	

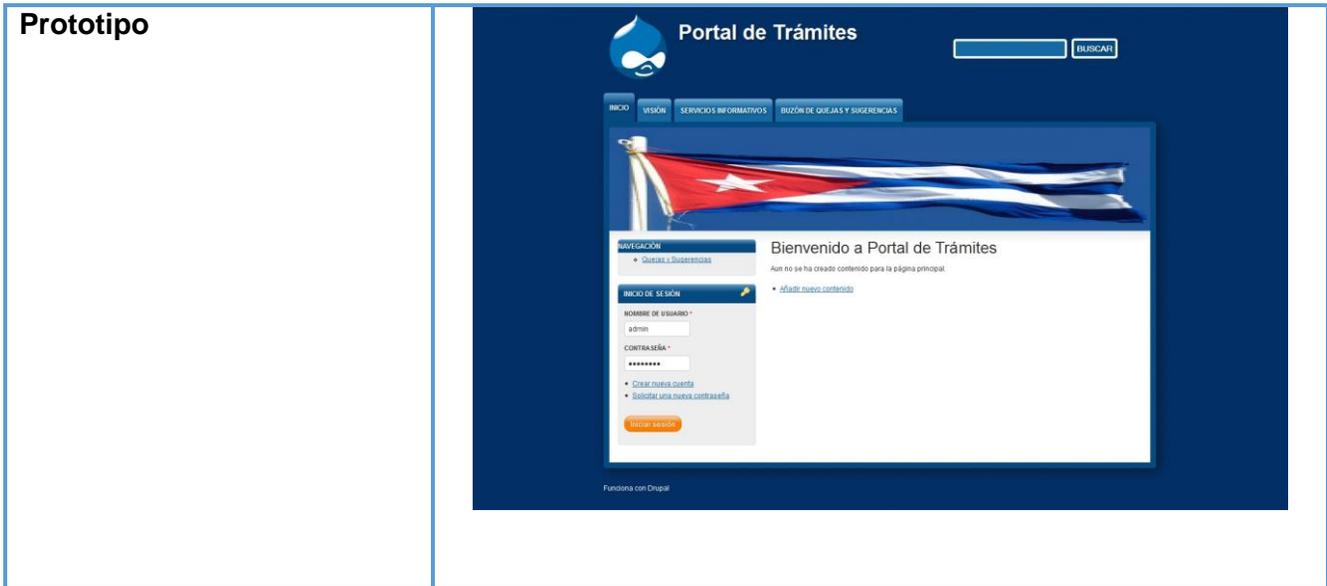


Tabla 8 Descripción del requisito del portal “Autenticar Usuario”

RF 3 Obtener requisitos para la realización de trámites.

Propósito	Conocer la información requerida para el trámite.	
Roles	Usuario.	
Pre-condiciones	El usuario no tiene que estar autenticado.	
Entidades tratadas	Entidad	Atributos
	NODE	Nid Type Tittle Uid Status
	FIELD_DATA_BODY	Entity_type Bundle Entity_id Body_value
Descripción	3.1 Mostrar el listado de documentos de tramitación si selecciona la opción “Servicios informativos”, en el menú principal.	

Anexos

	<ul style="list-style-type: none"> a) Carné de Identidad. b) Tarjeta de Menor. c) Pasaporte. d) Licencia de Conducción. e) Carné Transitorio. f) Registro de Vehículos. g) Portador de Arma de fuego. <p>3.2 Mostrar información de los trámites del documento seleccionado.</p> <ul style="list-style-type: none"> a) Cambio de Dirección. b) Arribo. c) Subsanación de error. d) Desmovilización FAR. e) Salida prisión. f) Entrada al país. g) Perdida. h) Deterioro.
Validaciones	-
Post-condiciones	-
Prototipo	

Tabla 9 Descripción del requisito del portal “Obtener requisitos para la realización de trámites”

RF 5 Consultar el estado de los trámites.

Anexos

Propósito	Consultar el estado en que se encuentra los trámites de un usuario.	
Roles	Usuario registrado.	
Pre-condiciones	El usuario tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	users	Usuario Contraseña
Descripción	<p>5.1 Seleccionar opción “Consultar estado”, en el menú “Servicios para usuarios”.</p> <p>5.2 Mostrar el listado de solicitudes de los trámites con los estados correspondientes.</p> <ul style="list-style-type: none"> a) Carné de Identidad. b) Nombre. c) Apellidos. d) Documento. e) Trámite. f) Estado. 	
Validaciones	-	
Post-condiciones	-	
Prototipo		

Tabla 10 Descripción del requisito del portal “Consultar el estado de los trámites”

Anexos

RF 6 Mostrar información de las oficinas.

Propósito	Conocer información general de las oficinas.	
Roles	Usuario.	
Pre-condiciones	El usuario no tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	NODE	Nid Type Tittle Uid Status
	FIELD_DATA_BODY	Entity_type Bundle Entity_id Body_value
Descripción	6.1 Mostrar listado de provincias si se selecciona la opción "Servicios Informativos". a) Pinar del Río. b) Artemisa. c) La Habana. d) Mayabeque. e) Matanzas. f) Cienfuegos. g) Villa Clara. h) Sancti-Spíritus. i) Ciego de Ávila. j) Camagüey. k) Las Tunas. l) Holguín. m) Granma. n) Santiago de Cuba. o) Guantánamo. p) Isla de la Juventud.	

Anexos

	<p>6.2 Mostrar información de las oficinas según la provincia seleccionada.</p> <ul style="list-style-type: none"> a) Número. b) Unidades del CIRP. c) Calle Principal. d) Número de Entre Calle. e) Entre Calle 1. f) Entre Calle 2. g) Teléfono.
Validaciones	-
Post-condiciones	-
Prototipo	 <p>The screenshot shows the 'Portal de Trámites' interface. It features a top navigation bar with 'INICIO', 'VISION', 'SERVICIOS INFORMATIVOS', and 'TRUZO DE GUIAS Y SUGERENCIAS'. A central menu lists various services under 'SERVICIOS PARA' and 'DOCUMENTOS'. A dropdown menu is open, showing a list of provinces including Pinar del Río, Matanzas, Cienfuegos, Villa Clara, Sancti Spiritus, and others. The background shows a blurred view of the main portal content.</p>

Tabla 11 Descripción del requisito del portal “Mostrar información de las oficinas”

RF 7 Cancelar la solicitud del trámite.

Propósito	Cancelar la solicitud de un trámite.	
Roles	Usuario registrado.	
Pre-condiciones	El usuario tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	users	Usuario Contraseña
Descripción	7.1 Seleccionar la opción de “Cancelar reservación”, en el menú	

	<p>“Servicios para usuarios”.</p> <p>7.2 Mostrar listado de solicitud de trámite realizada por el usuario.</p> <p>a) Listado de Solicitudes.</p> <ol style="list-style-type: none"> i. Identificador del Envío. ii. Carné de Identidad. iii. Nombre. iv. Apellidos. v. Provincia. vi. Oficina. vii. Documento. viii. Trámite. ix. Usuario. <p>b) Seleccionar Identificador de Usuario.</p> <p>c) Aceptar.</p> <p>7.3 Seleccionar identificador de la reservación.</p> <p>a) Seleccionar el Identificador de la reservación.</p> <p>7.4 Cancelar la solicitud del trámite si se selecciona la opción “Aceptar”.</p>
Validaciones	-
Post-condiciones	-
Prototipo	 <p>The screenshot shows the 'Portal de Trámites' website. At the top, there is a logo on the left and a search bar with a 'BUSCAR' button on the right. Below the search bar, there are several menu items: 'Inicio', 'Servicios Informativos', and 'Búsqueda de Guías y Subservicios'. The main content area features a banner with three blue passport covers labeled 'PASAPORTE'. Below the banner, there is a 'Bienvenido a Portal de Trámites' message. To the left of the message, there are two columns of links: 'Servicios Inicio y Usuarios' (with links for 'Inicio', 'Mi Subiduría', 'Cancelar Estado', and 'Cancelar Reservación') and 'Servicios' (with links for 'Inicio' and 'Subiduría y Documentos'). A small note at the bottom of the message says 'Aun no se ha creado contenido para la página principal.' and there is a link for 'Añadir nuevo contenido'. A footer note mentions 'Formulario para que los usuarios cancelen sus reservaciones.'</p>

Tabla 12 Descripción del requisito del portal “Cancelar solicitud del trámite”

Anexos

RF 8 Publicar encuesta.

Propósito	Publicar las encuestas para los usuarios del portal.	
Roles	Gestor de Encuestas.	
Pre-condiciones	El usuario tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	POLL	Nid Active
	POLL_CHOICE	Chid Nid Chtext Chvotes Weight
	POLL_VOTE	Chid Nid Uid Hostname
Descripción	<p>8.1 Seleccionar la opción "Añadir Contenido".</p> <p>8.2 Mostrar campos para crear la encuesta.</p> <ul style="list-style-type: none">a) Pregunta.b) Opción.c) Más opciones.d) Estado de la encuesta.e) Duración de la encuesta.f) Guardar.g) Vista Previa <p>8.3 Introducir datos.</p> <ul style="list-style-type: none">a) Pregunta.b) Opción.c) Más opciones.d) Estado de la encuesta.e) Duración de la encuesta.	

Anexos

	<p>f) Guardar.</p> <p>g) Vista Previa.</p> <p>8.4 Guardar la encuesta si se selecciona la opción “Guardar”.</p> <p>8.5 Mostrar vista previa de la encuesta si se selecciona la opción “Vista Previa”.</p>
Validaciones	-
Post-condiciones	-
Prototipo	

Tabla 13 Descripción del requisito del portal “Publicar encuesta”

RF 9 Publicar quejas y sugerencias.

Propósito	Publicar quejas y sugerencias.	
Roles	Usuario.	
Pre-condiciones	El usuario no tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	NODE	<p>Nid</p> <p>Type</p> <p>Title</p> <p>Uid</p> <p>Status</p>
	FIELD_DATA_FIELD_COM ENTARIO	<p>ENTITY_TYPE</p> <p>Bundle</p>

Anexos

		Deleted Entity_ID Revision_ID Language Delta Field_Comentario_value Field_Comentario_Format
	FIELD_DATA_FIELD_T_TU LO	ENTITY_TYPE Bundle Deleted Entity_ID Revision_ID Language Delta Field_T_tulo_value Field_t_tulo_Format
Descripción	<p>9.1 Seleccionar la opción “Quejas y sugerencias”.</p> <p>9.2 Mostrar formulario de quejas y sugerencias si se selecciona la opción “Quejas y sugerencias”.</p> <p style="padding-left: 40px;">a) Nombre.</p> <p style="padding-left: 40px;">b) Título.</p> <p style="padding-left: 40px;">c) Comentario.</p> <p>9.3 Introducir datos.</p> <p style="padding-left: 40px;">a) Nombre.</p> <p style="padding-left: 40px;">b) Título.</p> <p style="padding-left: 40px;">c) Comentario.</p> <p>9.4 Enviar datos si se selecciona la opción “Guardar”.</p> <p>9.5 Mostrar vista previa si se selecciona la opción “Vista previa”.</p> <p>9.6 Seleccionar la opción “Buzón de Quejas y Sugerencias” para ver las quejas y sugerencias aprobadas por los responsables del portal.</p> <p>9.7 Mostrar quejas y sugerencias aprobadas si se selecciona la opción “Buzón de Quejas y Sugerencias”.</p>	
Validaciones	Verificar que los campos estén llenos.	

Anexos

Post-condiciones	-
Prototipo	

Tabla 14 Descripción del requisito del portal “Publicar quejas y sugerencias”

RF 10 Gestionar información del portal.

Propósito	Gestionar la información de los trámites.	
Roles	Administrador.	
Pre-condiciones	El usuario tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	NODE	Nid Type Tittle Uid Status
	FIELD_DATA_BODY	Entity_type Bundle Entity_id Body_value
Descripción	<i>Crear</i> 10.1 Seleccionar la opción “Añadir Contenido”.	

10.2 Mostrar lista de contenidos si se selecciona la opción “Añadir Contenido”.

a) Artículo.

b) Página Básica.

10.3 Mostrar formulario para crear artículo si se selecciona la opción “Artículo” o “Página Básica”.

a) Título.

b) Tags.

c) Cuerpo.

d) Imagen.

e) Opciones de menú.

i. Información de la revisión.

ii. Opciones de la URL.

iii. Opciones de comentarios.

iv. Información de autoría.

v. Opciones de publicación.

f) Guardar.

g) Vista Previa.

10.4 Llenar campo.

10.5 Guardar cambios si se selecciona la opción “Guardar”.

10.6 Mostrar vista previa si se selecciona la opción “Vista previa”.

Editar y Eliminar

10.1 Seleccionar artículo.

10.2 Mostrar el artículo seleccionado.

10.3 Mostrar formulario para editar artículo o página básicas si se selecciona la opción “Editar”.

h) Título.

i) Tags.

j) Cuerpo.

k) Imagen.

l) Opciones de menú.

i. Información de la revisión.

ii. Opciones de la URL.

	<ul style="list-style-type: none"> iii. Opciones de comentarios. iv. Información de autoría. v. Opciones de publicación. <p>m) Guardar</p> <p>n) Vista previa</p> <p>o) Eliminar</p> <p>10.4 Actualizar información.</p> <p>10.5 Guardar cambios si se selecciona la opción “Guardar”.</p> <p>10.6 Mostrar vista previa si se selecciona la opción “Vista previa”.</p> <p>10.7 Eliminar si se selecciona la opción “Eliminar”.</p>
Validaciones	-
Post-condiciones	-
Prototipo	<p style="text-align: center;"><i>Crear</i></p>  <p style="text-align: center;"><i>Editar y Eliminar</i></p>

Anexos

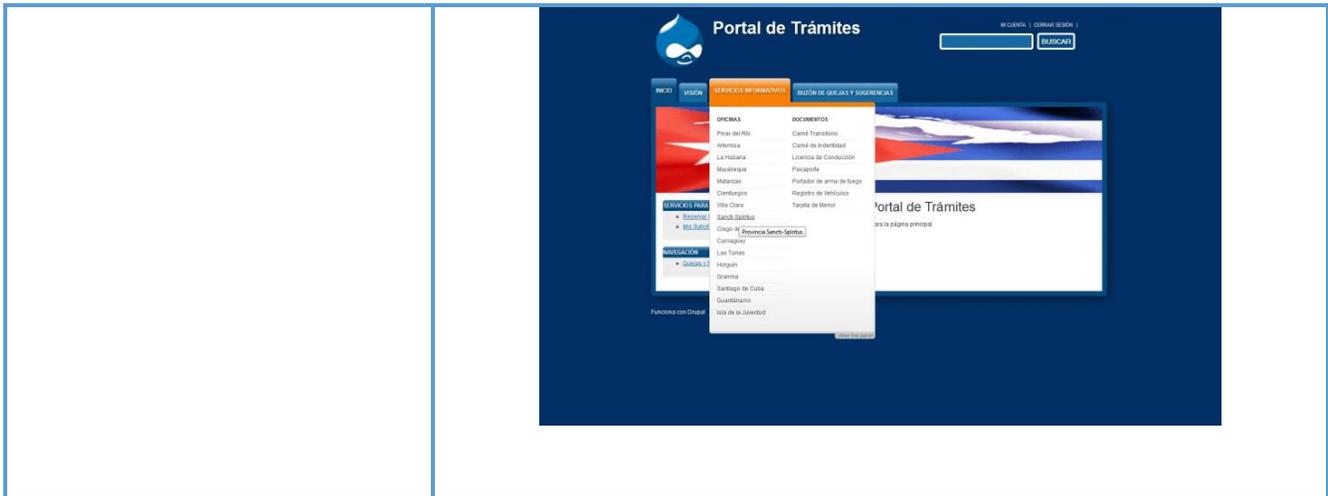


Tabla 15 Descripción del requisito del portal “Gestionar información del portal”

RF 11 Gestionar usuario.

Propósito	Gestionar usuario.	
Roles	Administrador.	
Pre-condiciones	El usuario tiene que estar autenticado en el sistema.	
Entidades tratadas	Entidad	Atributos
	USERS	Uid Name Pass Mail
	ROLE	Rid Name Weight
	USERS_ROLES	Uid Rid
Descripción	Crear 11.1 Seleccionar la opción “Persona”. 11.2 Mostrar Administración de Usuarios.	

	<p>11.3 Mostrar formulario para crear usuario si se selecciona la opción “Añadir usuario”.</p> <ul style="list-style-type: none">a) Nombre de usuario.b) Dirección de correo electrónico.c) Contraseña.d) Confirmar contraseña.e) Estado.f) Roles.g) Idioma.h) Crear nueva cuenta. <p>11.4 Llenar campos.</p> <p>11.5 Crear cuenta si se selecciona la opción “Crear nueva cuenta”.</p> <p>11.6 Mostrar mensaje de confirmación.</p> <p style="text-align: center;"><i>Editar y Eliminar</i></p> <p>11.1 Seleccionar la opción “Persona”.</p> <p>11.2 Mostar Administración de Usuarios.</p> <p>11.3 Mostrar formulario para editar usuario si se selecciona la opción “Editar”.</p> <ul style="list-style-type: none">a) Nombre de usuario.b) Dirección de correo electrónico.c) Contraseña.d) Confirmar contraseña.e) Estado.f) Roles.g) Idioma.h) Imagen.i) Guardar.j) Cancelar Cuenta. <p>11.4 Actualizar datos</p> <p>11.5 Actualizar información si se selecciona la opción “Guardar”.</p> <p>11.6 Eliminar cuenta si se selecciona la opción “Cancelar Cuenta”.</p>
--	--

Anexos

Validaciones	-
Post-condiciones	-
Prototipo	

Tabla 16 Descripción del requisito del portal “Gestionar Usuario”

RF 12 Generar reporte.

Propósito	Consultar reportes del sistema.	
Roles	Administrador, Recepcionista.	
Pre-condiciones	El usuario tiene que estar registrado en el sistema.	
Entidades tratadas	Entidad	Atributos
	DRESERVACION_CITAS	idEnvio carne_ident nombre apellidos provincia oficina documento tramite idUser usuario idSolicitud
Descripción	<i>Perona por Día.</i>	

Anexos

	<p>12.1 Seleccionar la opción “Perona por Día” en el menú “Reportes”.</p> <p>12.2 Mostrar los campos a llenar.</p> <p>a) Provincias.</p> <p>b) Oficinas.</p> <p>c) Aceptar.</p> <p>12.3 Mostrar listado de personas si se selecciona la opción “Aceptar”.</p> <p style="text-align: center;"><i>Solicitudes por oficinas.</i></p> <p>12.1 Seleccionar la opción “Solicitudes por oficinas” en el menú “Reportes”.</p> <p>12.2 Mostrar listado de solicitudes por oficinas.</p> <p style="text-align: center;"><i>Trámite más solicitado.</i></p> <p>12.1 Seleccionar la opción “Trámite más solicitado” en el menú “Reportes”.</p> <p>12.2 Mostrar listado del trámite más solicitado.</p>
Validaciones	En el reporte Personas por Día que se seleccione los campos.
Post-condiciones	-
Prototipo	

Tabla 17 Descripción del requisito del portal “Generar reporte”

RFSW 2 Obtener el estado de un trámite.

Propósito	Obtener los estados que tenga un trámite solicitado.
Pre-condiciones	-

Anexos

Conceptos tratados	Concepto	Atributos
	Solicitud	Identificador Identificador del tipo proceso Identificador del tipo trámite Oficina Fecha de registro Fecha inicio del trámite Propuesta de fecha fin del trámite Estados Turno
Descripción	2.1 Entrar parámetros para obtener los estados. a) Identificador de la solicitud. 2.2 Lanzar una excepción si los datos validados no son correcto. 2.3 Buscar el estado de la solicitud en el esquema de base de datos del portal. 2.4 Devolver el estado, si es finalizado o cancelado. 2.5 Buscar el estado en el esquema de base datos del SUIN, si el estado no es ni finalizado ni cancelado. 2.6 Devolver los estados del trámite.	
Validaciones	Debe existir el identificador de la solicitud en la base de datos.	
Post-condiciones	Se devuelve los estados que tenga un trámite.	
Prototipo	No procede.	

Tabla 18 Descripción del requisito del servicio web “Obtener el estado de un trámite”

RFSW 3 Administrar turno.

Anexos

Propósito	Administrar los turnos para una oficina determinada.	
Pre-condiciones	Se seleccionara como fecha inicio del trámite como mínimo tres días después de realizar la solicitud, en dependencia de si existe o no solicitudes para esa oficina.	
Conceptos tratados	Concepto	Atributos
	Solicitud	Identificador Identificador del tipo proceso Identificador del tipo trámite Oficina Fecha de registro Fecha inicio del trámite Propuesta de fecha fin del trámite Estados Turno
	Config_Tramite	Identificador del tipo trámite Descripción del tipo de trámite Tiempo aproximado en minutos que dura el trámite
Config_Jornada_Laboral	Día de la semana Hora inicio de la jornada laboral Minuto inicio de la jornada laboral Hora fin de la jornada laboral	

Anexos

		Minuto fin de la jornada laboral
Descripción	<p>3.1 Entrar parámetros para administrar el turno.</p> <p>a) Oficina.</p> <p>b) Identificador del tipo de trámite.</p> <p>3.2 Lanzar una excepción si los datos validados no son correcto.</p> <p>3.3 Validar los datos si son correctos:</p> <p>1.1 Seleccionar que la fecha de un día laborable y el primer turno, de no existir turnos para la oficina introducida la fecha del trámite será la de fecha de registro más tres días.</p> <p>1.2 Escoger el último turno, la fecha del trámite de la próxima solicitud, de existir turnos.</p> <p>1.3 Devolver los datos del turno.</p> <p>a) Fecha de inicio del trámite (incluida la hora).</p> <p>b) Propuesta de la fecha fin del trámite (incluida la hora).</p> <p>c) Turno.</p>	
Validaciones	<p>1. Debe existir configuración para el tipo de trámite que se pasa por parámetro.</p> <p>2. Debe existir configuración para el día de la semana que se seleccione para que sea el trámite.</p>	
Post-condiciones	Se devuelve la fecha inició del trámite, el turno y la fecha fin propuesta que sería la fecha de inicio más el tiempo de demora del trámite.	
Prototipo	No procede.	

Tabla 19 Descripción del requisito del servicio web “Administrar turno”

RFSW 4 Obtener las solicitudes de una oficina.

Propósito	Obtener las solicitudes de una oficina para una fecha determinada.	
Pre-condiciones	-	
Conceptos tratados	Concepto	Atributos

Anexos

	Persona	Número de identidad Nombre(s) Apellidos Solicitudes
	Solicitud	Identificador Identificador del tipo proceso Identificador del tipo trámite Oficina Fecha de registro Fecha inicio del trámite Propuesta de fecha fin del trámite Estados Turno
Descripción	4.1 Entrar parámetros para obtener las solicitudes. a) Oficina. b) Fecha inicio del trámite. 4.2 Obtener los datos de las solicitudes que tenga una oficina determinada para una fecha en específico. a) Número de identidad. b) Nombre(s). c) Apellidos. d) Solicitudes. i. Identificador.	

Anexos

	<ul style="list-style-type: none"> ii. Identificador del tipo proceso. iii. Identificador del tipo trámite. iv. Oficina. v. Fecha registro. vi. Fecha de inicio trámite. vii. Propuesta de fecha fin del trámite. viii. Estados. ix. Turno.
Validaciones	-
Post-condiciones	Se devuelve las solicitudes de trámites para una oficina y una fecha determinada.
Prototipo	No procede.

Tabla 20 Descripción del requisito del servicio web “Obtener las solicitudes de una oficina”

RFSW 5 Obtener las solicitudes de una persona determinada.

Propósito	Obtener las solicitudes de una persona dado el número de identidad.	
Pre-condiciones	-	
Conceptos tratados	Concepto	Atributos
	Persona	<ul style="list-style-type: none"> Número de identidad Nombre(s) Apellidos Solicitudes
	Solicitud	<ul style="list-style-type: none"> Identificador Identificador del tipo proceso Identificador del tipo trámite

Anexos

	<p>Oficina</p> <p>Fecha de registro</p> <p>Fecha inicio del trámite</p> <p>Propuesta de fecha fin del trámite</p> <p>Estados</p> <p>Turno</p>
Descripción	<p>5.1 Entrar parámetros para obtener las solicitudes.</p> <p>a) Número de identidad.</p> <p>5.2 Obtener las solicitudes que tenga asociado una persona.</p> <p>a) Identificador.</p> <p>b) Identificador del tipo proceso.</p> <p>c) Identificador del tipo trámite.</p> <p>d) Oficina.</p> <p>e) Fecha registro.</p> <p>f) Fecha de inicio trámite.</p> <p>g) Propuesta de fecha fin del trámite.</p> <p>h) Estados.</p> <p>i) Turno.</p>
Validaciones	-
Post-condiciones	Se devuelve las solicitudes de trámites para una oficina y una fecha determinada.
Prototipo	No procede.

Tabla 21 Descripción del requisito del servicio web “Obtener las solicitudes de una persona determinada”

Anexos

Anexo II. DESCRIPCIÓN DE ENTIDADES MODELO DE DATOS

Nombre: DPERSONA_TRAMITE_PORTAL		
Descripción: se almacenan los datos de las personas que se registran en el portal.		
Atributo	Tipo	Descripción
NUMEROIDENTIDAD	VARCHAR2(11)	Carné de identidad de la persona.
NOMBRE	VARCHAR2(100)	Nombre de la persona.
APELLIDOS	VARCHAR2(100)	Apellidos de la persona.

Tabla 22 Descripción de la Entidad DPERSONA_TRAMITE_PORTAL

Nombre: DRESERVACION_CITAS		
Descripción: se almacena la información de todas las reservaciones.		
Atributo	Tipo	Descripción
IDENVIO	NUMBER	Identificador del envío.
CARNE_IDENT	VARCHAR2(11)	Carné de Identidad de la persona.
NOMBRE	VARCHAR2(50)	Nombre(s) de la persona.
APELLIDOS	VARCHAR2(50)	Apellidos de la persona.
PROVINCIA	VARCHAR2(50)	Provincia donde se realizará el trámite.
OFICINA	VARCHAR2(50)	Oficina donde se realizará el trámite.
DOCUMENTO	VARCHAR2(50)	Documento del que se quiere realizar un trámite.
TRAMITE	VARCHAR2(50)	Trámite que se quiere realizar.
IDUSER	NUMBER	Identificador del usuario que realizó el envío.
USUARIO	VARCHAR2(50)	Usuario de la persona.
IDSOLICITUD	VARCHAR2(36)	Identificador de la solicitud.

Tabla 23 Descripción de la Entidad DRESERVACION_CITAS

Anexos

Nombre: DSOLICITUD_TRAMITE_PORTAL		
Descripción: se almacena la información referente a una solicitud.		
Atributo	Tipo	Descripción
IDOFICINA	VARCHAR2(4)	Identificador de la oficina.
IDSOLICITUDTRAMITE	RAW(16)	Identificador de la solicitud del trámite.
NUMEROIDENTIDAD	VARCHAR2(11)	Número de identidad de la persona.
FECHAREGISTRO	DATE	Fecha de registro de la solicitud.
IDTIPOTRAMITE	NUMBER	Identificador del tipo de trámite.
FECHATRAMITE	DATE	Fecha de realización del trámite.
TURNO	NUMBER	Turno en la cola para realizar el trámite.
IDTIPOPROCESO	NUMBER	Identificador del tipo de proceso.
FECHAFINPROPUESTA	DATE	Fecha fin del trámite que se propone.

Tabla 24 Descripción de la Entidad DSOLICITUD_TRAMITE_PORTAL

Nombre: NCONFIG_TRAMITE_PORTAL		
Descripción:		
Atributo	Tipo	Descripción
IDTIPOTRAMITE	NUMBER	Identificador del trámite.
TIEMPO	NUMBER	Tiempo para la realización del trámite.
DESCRIPCION	VARCHAR2(100)	Descripción del trámite.

Tabla 25 Descripción de la Entidad NCONFIG_TRAMITE_PORTAL

Nombre: NDIA_SEMANA		
Descripción:		
Atributo	Tipo	Descripción
IDDIASEMANA	NUMBER	Identificador del día de la semana.

Anexos

DESCRIPCION	VARCHAR2(50)	Descripción del día de la semana.
HORAINICIO	NUMBER	Hora de inicio de la oficina.
MINUNICIO	NUMBER	Minuto de inicio de la oficina.
HORAFIN	NUMBER	Hora de cerrar la oficina.
MINFIN	NUMBER	Minuto de cerrar la oficina.

Tabla 26 Descripción de la Entidad NDIA_SEMANA

Anexo III. PRUEBA DE ACEPTACIÓN

UCI Universidad de las Ciencias Informáticas

"Aniversario 55 de la Revolución"
Mayo 22 de 2013.

ACTA DE ACEPTACIÓN DEL PRODUCTO:
Portal de Límites de la República de Cuba.

Por este medio se certifica que el producto realizado por los estudiantes con motivo de su trabajo de diploma cumple con los requisitos establecidos y aprobados por el proyecto identidad Cuba.

Han sido entregados para trabajos futuros los principales artefactos resultantes de la investigación: software, especificación de requerimientos y documento de tesis.

Otras observaciones:

Para que así conste firman:


Jefe de Proyecto
Msc. Ing. Erick de la Vega
García


Jefe de Desarrollo
Ing. Reynier Blanco Zambrano


Jefe de Producto
Msc. Ing. Liudnet Caballero
Duboy

Ilustración 20 Prueba de Aceptación