

UNIVERSIDAD DE MATANZAS
Facultad de Ciencias Técnicas
Departamento de Informática



Trabajo para optar por el Título de Ingeniero en Informática.

**Título: Sistema informático para la tramitación de documentos en la
Secretaría General de la Universidad de Matanzas.**

Autor: Keythy Sotolongo Alonso.

Tutor: Ing. Roider Rojas Sánchez.

Matanzas, Cuba

Octubre de 2018

Dedicatoria

Le dedico este trabajo de diploma a:

Mis padres por todo su apoyo.

Mi sobrinita por ser mi luz.

Mi hermana por ser mi guía en todo momento.

Mi cuñado por impulsarme a escoger este camino.

Agradecimientos

Agradezco a todas las personas que me han dedicado al menos un instante de su tiempo, en especial a:

 Mi familia por confiar siempre en mí y especialmente a mis padres.

 Mi hermana y mi cuñado por no dudar.

 Mi tutor por su colaboración y apoyo.

 Yeslaine y Henry, tan serviciales y nunca apurados.

 Todos los que me acogieron como familia y me ayudaron en los momentos más difíciles.

 La Secretaría General de la Universidad de Matanzas por la información ofrecida.

 Todos los expertos consultados que brindaron un pedacito de su preciado tiempo.

 Mis compañeros de aula por su apoyo incondicional.

Declaración de autoría

Yo, Keythy Sotolongo Alonso, declaro que soy la única autora de este trabajo y autorizo a la Universidad de Matanzas y al Centro de Investigación y Desarrollo de Software, a que hagan el uso que estimen pertinente de él.

Y para que así conste, firmo la presente a los 18 días del mes de junio del año 2018.

Firma del Autor

Firma del Tutor

Opinión del Tutor

DATOS PERSONALES DEL TUTOR

Nombre y apellidos: Roider Rojas Sánchez

Centro de trabajo: Empresa de Tecnologías y Telecomunicaciones XETID

Organismo al que pertenece: Unión de Industrias Militares

Cargo que ocupa: Especialista Superior en Ciencias Informáticas

Especialidad de la que es graduado: Ingeniería Informática

Categoría docente o investigativa: Profesor Instructor

Grado científico: Ing.

DATOS DE LA TESIS Y EL DIPLOMANTE

Nombre y apellidos: Keythy Sotolongo Alonso

Centro de estudio: Universidad de Matanzas Sede Camilo Cienfuegos

Título de la Tesis: Sistema informático para la tramitación de documentos en la Secretaría General de la Universidad de Matanzas.

OPINION SOBRE EL TRABAJO

El tutor de este trabajo considera que, durante su ejecución, el estudiante mostró las cualidades que a continuación se detallan:

Alta independencia demostrada en todo el desarrollo de su investigación en el tema. Supo adquirir con rapidez los principales elementos del conocimiento a que se enfrentó. El estudio de prácticamente toda la información sobre el tema, la creación y diseño de un sistema bien concebido y todos los recursos usados para el desarrollo de la investigación, corroboran la laboriosidad con que ha trabajado durante este último período. Enfrentó de manera responsable una tarea difícil y la ha llevado a cabo con gran profesionalidad, lo que se observa en la calidad de su trabajo final, tanto de la aplicación como del documento.

El trabajo cumple con los objetivos propuestos, aborda una temática de gran utilidad e importancia para el sector donde se va a explotar.

El documento fue escrito con gran rigor y cumpliendo las normas que rigen estos materiales. Posee una redacción y una estructura correcta.

En atención a lo anteriormente planteado, considero que el estudiante Keythy Sotolongo Alonso reúne los requisitos para el título de Ingeniero Informático y recomendamos que luego de atendidos los planteamientos del oponente, le sea otorgada la mejor calificación de este Tribunal.

Ing. Roider Rojas

Junio/2019

Resumen

La elevada complejidad que actualmente conlleva realizar los trámites en las secretarías de las universidades cubanas, hace necesaria la existencia de una herramienta informática que permita una manipulación más sencilla y eficiente de los mismos. Para ello se propone la elaboración de un sistema que contribuya a mejorar los procesos de solicitud y trámites de certificaciones de notas, cotejos de títulos y planes de estudio en la secretaría general de la Universidad de Matanzas. La metodología que se utiliza para la elaboración mismo es Prodesoft, mientras que se utiliza ProcessMaker como Sistema de Gestión de Procesos de Negocios, teniendo PHP para la lógica del negocio en la plataforma, HTML en las vistas, JavaScript para la validación de los componentes de HTML y SQL para el trabajo con los datos. Para su realización se llevó a cabo un análisis de la forma de realización actual a través de entrevistas en el área para el cual se elabora el sistema, así como un estudio del estado del arte acerca de este tema a nivel nacional y foráneo que permite a su vez elaborar el marco teórico referencial de la investigación y la construcción de una solución fiable para el problema que se presenta.

Abstract

The high complexity that currently involves carrying out procedures in the secretariats of Cuban universities, makes necessary the existence of a computer tool that allows a more simple and efficient handling of them. To this end, it is proposed to develop a system that contributes to improving the application processes and procedures for certifications of grades, comparisons of degrees and curricula in the General Secretariat of the University of Matanzas. The methodology that is used for the elaboration itself is Prodesoft, while ProcessMaker is used as a Business Process Management System, having PHP for the business logic in the platform, HTML in the views, JavaScript for the validation of the components of HTML and SQL for working with data. For its realization, an analysis of the current form of realization was carried out through interviews in the area for which the system is elaborated, as well as a study of the state of the art about this issue at a national and foreign level that allows turn elaborate the theoretical framework of research and the construction of a reliable solution for the problem that arises.

Índice:

<u>Declaración de autoría</u>	<u>II</u>
<u>Opinión del Tutor</u>	<u>III</u>
<u>Resumen</u>	<u>V</u>
<u>Abstract</u>	<u>VI</u>
<u>Introducción</u>	<u>1</u>
<u>Capítulo I: Marco teórico- referencial de la investigación</u>	<u>6</u>
<u>1.1 Conceptos asociados al dominio del problema</u>	<u>6</u>
<u>1.2 Descripción de la entidad</u>	<u>7</u>
<u>1.3 Antecedentes</u>	<u>9</u>
<u>1.4 Metodología de desarrollo</u>	<u>11</u>
<u>1.5 Herramientas y tecnologías</u>	<u>13</u>
<u>1.5.1 Lenguaje de modelado: UML y BPMN</u>	<u>13</u>
<u>1.5.2 Herramientas para el modelado: Visual Paradigm</u>	<u>14</u>
<u>1.5.3 Lenguajes de programación: PHP, CSS, JavaScript, HTML y JQuery ..</u>	<u>14</u>
<u>1.5.4 Lenguaje de consultas: SQL.....</u>	<u>17</u>
<u>1.5.5 Gestor de Base de Datos: PostgreSQL y MySQL.....</u>	<u>18</u>
<u>1.5.6 Framework: ProcessMaker, Zend Framework, Doctrine, ExtJs y Bootstrap.....</u>	<u>19</u>
<u>1.5.7 Servicios web: XML y WSDL.....</u>	<u>22</u>
<u>1.6 Conclusiones parciales</u>	<u>23</u>
<u>Capítulo II: Diseño y construcción de la Solución Propuesta</u>	<u>23</u>
<u>2.1 Introducción</u>	<u>23</u>
<u>2.2 Modelado del negocio</u>	<u>24</u>
<u>2.2.1 BPD</u>	<u>24</u>
<u>2.2.2 Reglas del negocio</u>	<u>25</u>
<u>2.2.3 Modelo Conceptual</u>	<u>26</u>
<u>2.2.4 Requisitos</u>	<u>27</u>
<u>2.2.4.1 Requisitos no funcionales.....</u>	<u>27</u>
<u>2.2.4.2 Requisitos funcionales.....</u>	<u>28</u>

<u>2.3 Especificación de requisitos funcionales.....</u>	<u>30</u>
<u>2.4 Implementación de la Arquitectura de Software</u>	<u>40</u>
<u>2.4.1 Arquitectura Cliente-Servidor.....</u>	<u>41</u>
<u>2.4.2 Arquitectura N-Capas.....</u>	<u>42</u>
<u>2.5 Diseño de la Base de Datos.....</u>	<u>42</u>
<u>2.6 Análisis de factibilidad.....</u>	<u>44</u>
<u>2.6.1 Costo</u>	<u>44</u>
<u>2.6.2 Beneficios tangibles e intangibles.....</u>	<u>46</u>
<u>2.6.3 Análisis de costo y beneficios</u>	<u>46</u>
<u>2.7 Patrones.....</u>	<u>46</u>
<u>2.8 Mecanismo de diseño</u>	<u>48</u>
<u>2.9 Diagramas de clases de diseño.....</u>	<u>49</u>
<u>2.10 Diagramas de secuencia.....</u>	<u>50</u>
<u>2.11 Seguridad.....</u>	<u>51</u>
<u>2.12 Conclusiones parciales</u>	<u>52</u>
<u>Capítulo III: Validación de la Solución Propuesta.</u>	<u>53</u>
<u>3.1 Introducción.....</u>	<u>53</u>
<u>3.2 Pruebas realizadas y resultados</u>	<u>53</u>
<u>3.2.1 Pruebas de aceptación</u>	<u>53</u>
<u>3.2.3 Pruebas de caja negra</u>	<u>59</u>
<u>3.2.4 Resultados de las pruebas.....</u>	<u>61</u>
<u>3.2 Conclusiones parciales.....</u>	<u>62</u>
<u>Conclusiones Generales</u>	<u>63</u>
<u>Recomendaciones</u>	<u>64</u>
<u>Referencias</u>	<u>65</u>
<u>Anexos.....</u>	<u>69</u>

Introducción

Con el paso de los años la informática ha ido ganando espacio en la vida de los hombres, hasta llegar a convertirse en indispensable para el desarrollo de las mismas. Innumerables son hoy las acciones que se pueden realizar con tan solo un clic, cuando hace solo décadas era necesario realizar complejos acciones para lograr el mismo objetivo. Actualmente en las universidades cubanas y muy específicamente en la universidad de Matanzas, existen varios procesos que continúan efectuándose manualmente, ocasionando atrasos y molestias para sus trabajadores y la población en general.

La Universidad “Es el centro encargado de la formación de especialistas fundamentalmente en los campos de las ciencias naturales y exactas, ciencias sociales y humanísticas y ciencias económicas”. (Estructura y titulaciones de educación superior, pág. 10). Lo que la sitúa como un eje fundamental de la sociedad cubana y mundial.

La secretaría general de la misma es uno de sus departamentos más importantes, por la amplia cantidad y variedad de documentos que en ella se realizan, tanto para los estudiantes como para los profesores y la sociedad en general. Las certificaciones de notas, cotejos de títulos y planes de estudio, son algunos de los que actualmente se realizan en ella y son una necesidad de todos los profesionales del país tanto para optar por un trabajo, como para cualquier trámite legal que pretendan realizar, dentro y fuera del territorio nacional, así como para las empresas que necesitan contratar personal y por tanto un paso indispensable es corroborar su formación académica, llegando a incurrirse en un delito, en caso de resultar falsa, dicha afirmación.

Realizar la búsqueda y certificación de dichos documentos tanto de forma manual en los archivos, como en una base de datos que ha realizado el propio personal de la secretaría, conlleva a un aumento de los tiempos necesarios para atender cada solicitud, disminuye la calidad de los servicios y aumenta los costos necesarios para

la realización de los mismos. Además, disminuye la eficiencia en el trabajo y administración del departamento y por tanto de la universidad.

Por estos motivos es indispensable flexibilizar, optimizar, transparentar y abaratar su realización con el objetivo principal de propiciar el aumento de los niveles de satisfacción de la población, lo que coincide en gran medida con los principios que sustentan el gobierno electrónico, y que actualmente se encuentra en los primeros lugares del desarrollo informático de Cuba.

En este momento, la Empresa de Tecnologías de la Información para la Defensa (XETID), conjuntamente con el Centro de Investigación y Desarrollo (CIDES), se encuentran inmersos en el desarrollo de un proyecto de este tipo, Bienestar, en la provincia de Matanzas, al cual se suma la presente investigación.

Como parte de la misma, se hace necesario desarrollar un sistema que permita la realización de lo anteriormente mencionado, obteniéndose como resultado el siguiente problema científico: ¿Cómo contribuir a mejorar la eficiencia en la solicitud y trámites de las certificaciones de notas, cotejos de títulos y planes de estudios en la secretaría general de la Universidad de Matanzas mediante su informatización?

El objeto de estudio es la solicitud y trámite de las certificaciones de notas, cotejos de títulos y planes de estudio en la secretaría general de la Universidad de Matanzas.

La hipótesis a la que se pudo llegar partiendo de lo anteriormente dicho es: si se desarrolla un sistema informático, entonces se puede disminuir el tiempo necesario para realizar las certificaciones de notas, cotejos de títulos y planes de estudio en la secretaría general de la Universidad de Matanzas.

Variables:

Independiente: Sistema informático

Dependiente: Solicitud y trámite de las certificaciones de notas, cotejos de títulos y planes de estudio en la Secretaría General de la universidad de Matanzas.

El objetivo general de la misma es desarrollar una aplicación web para informatizar la solicitud y trámite de certificaciones de notas, cotejo de títulos y planes de estudio en la secretaría general de la Universidad de Matanzas.

Para cumplir el objetivo general, se desarrollan los siguientes objetivos específicos:

- Realizar un estudio que permita conformar el marco teórico referencial de la investigación.
- Diseñar e implementar la aplicación según lo indicado en la metodología Prodesoft.
- Validar la aplicación realizada.

Para darle solución al problema que se plantea y cumplir los objetivos propuestos se utilizaron los métodos siguientes:

- Teóricos
 - Lógico- histórico: Se utilizó durante la revisión de la literatura existente sobre el tema para analizar los antecedentes y evolución de este tipo de sistemas informáticos tanto dentro como fuera del país.
 - Análisis y síntesis: Se consultó literatura especializada y documentos rectores de la teoría que se utiliza para realizar dichos trámites.
 - Hipotético- deductivo: Para el análisis y formulación de la hipótesis.
 - Modelación: Para modelar los procesos de negocio.
- Empíricos
 - Entrevistas: Fue un método que se utilizó con bastante frecuencia para levantar los requisitos y dar los primeros pasos en el entendimiento del problema.

Entre los aportes del trabajo se tienen:

- El práctico, al desarrollar una herramienta que permita la correcta realización de los trámites en la Secretaría de la Universidad de Matanzas.

Entre los resultados esperados con esta investigación, se encuentran: contar con una herramienta Web que permita de una forma sencilla, ahorrar tiempo y aumentar la confiabilidad en los trámites que se realizan en la secretaría. Además, con su implementación se logrará facilitar la actividad de los implicados y reducir los errores que estos puedan cometer en la manipulación de una información tan importante como esta, posibilitando una mayor organización y seguridad.

Atendiendo a lo planteado, la investigación queda estructurada en introducción, tres capítulos, conclusiones, recomendaciones y referencia bibliográficas según se observa:

- Introducción: Se caracteriza la situación problemática y se fundamenta el problema científico a resolver.
- Capítulo I: Marco teórico-referencial del tema a tratar, donde se plantean los conceptos fundamentales asociados con el tema. Además, se realiza un análisis de las tendencias tecnológicas que serán utilizadas, así como un estudio del estado del arte y de los antecedentes que enmarcan la problemática actual.
- Capítulo II: Diseño y construcción de la Solución Propuesta, donde se argumenta la solución propuesta al problema y se describe la implementación del software a través de la metodología Prodesoft.
- Capítulo III: Validación de la solución propuesta, donde se muestran las principales interfaces del prototipo inicial, se realizan pruebas y se realiza un análisis de los resultados obtenidos.
- Un apartado de conclusiones donde se verifica el cumplimiento de los objetivos trazados al inicio de la investigación.
- Las recomendaciones en la cual se plasman una serie de propuestas encaminadas a la continuidad de esta investigación.

- Y las referencias de la bibliografía citada.

Capítulo I: Marco teórico- referencial de la investigación

Para realizar una buena investigación siempre es necesario tener bien claro cuáles son las bases que la sustentan y le dan firmeza, por eso este capítulo estará dedicado a conocer mejor los detalles del sistema que se pretende realizar, partiendo de un estudio de sus antecedentes. Además, se mostrarán las tecnologías y herramientas que se utilizarán para el desarrollo del mismo, así como un acercamiento a la tecnología utilizada.

1.1 Conceptos asociados al dominio del problema

Gobierno electrónico:” El gobierno electrónico es la aplicación de las tecnologías de la información y la comunicación (TIC) al funcionamiento del sector público, con el objetivo de brindar mejores servicios al ciudadano e incrementar la eficiencia, la transparencia y la participación ciudadana”. (Organización de Estados Americanos, 2010), y según la página web de la Agencia de Gobierno Electrónico y Sociedad de la Información y del Conocimiento (Agesic) (2010) este debe asumirse como un proceso evolutivo que consta de cuatro etapas principales: presencia, interacción, transacción y transformación. En la primera el gobierno tiene presencia en internet pero solo a través de la divulgación de sus sitios, es decir, solo se pone información básica para que los usuarios la conozcan pero no existe interacción entre ellos y las empresas; en la segunda ya es posible la comunicación entre ambos a través del correo electrónico, envío de formularios y comentarios de opinión o de foros; al llegar a la tercera ya se tiene la posibilidad de iniciar, realizar el seguimiento y concluir cualquier trámite en línea y con la última los organismos deben modificar sus estructuras, su estilo de atención y el flujo de trámites para proveer sus servicios de forma electrónica, además de fomentarse y promoverse el uso de las tecnologías para que exista una participación activa de los ciudadanos en la formación, discusión y evaluación de las políticas públicas.

Cotejo de títulos: Según el artículo 40 de la Resolución 184/11 (Asan, 2012, pág. 14) “Cotejar un documento es la comparación del original con su fotocopia, teniendo a ambos a la vista, que es sinónimo de comparar, confrontar, contraponer y verificar”.

Certificaciones de notas o certificación de estudios terminados: “En esta certificación se relacionan, por curso escolar, las asignaturas establecidas en el plan del proceso docente cursado por el estudiante y las calificaciones obtenidas”. (Asan, 2012, pág. 73)

Plan de estudio: “El Plan de estudio es el documento estatal y obligatorio que contiene la planificación y organización (diseño de la carrera) y las características más generales del desarrollo de los estudios correspondientes al título. El total de horas de los mismos es de entre 4000 y 5000 horas y caracteriza la profesión y el modelo del profesional, el campo profesional, las esferas de actuación, así como los modos en que se desempeñará el egresado”. (Estructura y titulaciones de educación superior, pág. 18)

Emisión: Según el sitio web Definición de (Julián Pérez, 2009), “El termino emisión está relacionado con la acción y efecto de emitir (arrojar o echar algo hacia afuera, poner en circulación títulos o valores, manifestar una opinión o juicio, lanzar ondas hercianas para difundir información). Una emisión, por lo tanto, puede estar constituida por el conjunto de efectos públicos u otro tipo de valores que son puestos en circulación”

Secretario general: “El Secretario General de la universidad es el funcionario nombrado con facultades para dirigir la Secretaría General. Es el jefe de las oficinas o departamentos que por estructura aprobada están adscriptos a la Secretaría General. Se subordina directamente al Rector y establece adecuadas relaciones de trabajo con sus subordinados” (Asan, 2012)

Solicitante: Es la persona que se dirige a la Secretaría para solicitar cualquiera de los servicios que en ella se ofrecen, en este caso, cotejos de títulos, planes de estudio y certificaciones de notas.

1.2 Descripción de la entidad

La presente investigación se desarrolla para la Secretaría General de la Universidad de Matanzas, la cuál es más que la dependencia universitaria que realiza funciones ejecutivas como órgano de dirección de los procesos de registro general,

documentación y expedición de certificaciones del centro de Educación Superior, así como la orientación y control metodológico del trabajo que se desarrolla en las secretarías docentes de las facultades y filiales universitarias. Garantiza el registro, procesamiento, control, custodia y conservación de la documentación que le corresponda, relacionada con la educación de posgrado que desarrolla la institución. (Universidad de Granma, 2017) Se subordina directamente al Decano y “es un valioso elemento auxiliar en las funciones de dirección de la universidad, pues facilita la información necesaria para el análisis relacionado con el ingreso, el movimiento de la matrícula, los resultados académicos de los estudiantes, el egreso, y el posgrado, todo lo cual forma parte de la memoria histórica de la universidad”. (Universidad de Oriente, 2016). Su ubicación dentro del organigrama de la universidad se puede observar en Ilustración 1.

Su jefe es el Secretario Docente y a él su subordinan los técnicos o especialistas que por plantilla se aprueben. Esto se puede observar en el organigrama de la Secretaría en la Ilustración 2. En su trabajo diario mantiene una estrecha vinculación con todos los miembros del Consejo de Dirección y con el personal técnico y administrativo que labora en la filial a la que pertenece. (Asan, 2012).

Actualmente la mayoría de los trámites que se llevan a cabo en ella se ejecutan de forma manual trayendo como consecuencia retrasos en los mismos. Por igual motivo las personas tanto naturales como jurídicas, incluyendo las que no radican en la ciudad, pero estudiaron en esta universidad, deben dirigirse varias veces a la entidad con el objetivo de completar una única solicitud. Además, a pesar de que actualmente se cuenta con una base de datos digital para los graduados, esta no es lo suficientemente eficiente para permitir una fácil extracción y manipulación de los datos necesarios. La planificación de los días de recogida no se encuentra informatizada, realizándose según la apreciación del personal que trabaja en la secretaría por lo que muchas veces los trámites no se encuentran concluidos el día indicado.

Debido a la elevada importancia que actualmente presentan estos trámites para todos los graduados universitarios este sistema debe dotar a la Secretaría de una

herramienta informática capaz de garantizar que sus trámites se realicen a la altura de las exigencias de la sociedad moderna y de las necesidades de su motor impulsor, el hombre.

1.3 Antecedentes

Decir gobierno electrónico hoy en día, es sinónimo de decir desarrollo. En el mundo entero cada vez son más los ejemplos que existen de esta forma de gobierno. En América la lista de Gobierno electrónico está encabezada por Estados Unidos y Canadá, seguida por Uruguay, Chile y Argentina, Colombia, Costa Rica, Brasil, Barbados y Antigua y Barbuda, mientras que la lista de América Latina se puede observar en la Ilustración 3. Sin embargo, a nivel mundial, de América solamente Estados Unidos se encuentra entre los primeros puestos. (UCI, 2014). Los primeros lugares en el ranking los ocupan Dinamarca, Australia, Corea, Reino Unido, Suecia, Finlandia, Singapur, Nueva Zelanda, Francia de E-government Survey que presenta la organización de Naciones Unidas se puede observar que Uruguay se encuentra en la posición 34, mientras que Chile está en el puesto 42, Argentina en el 43 y Brasil en el 44. (Diario de la República, 2018) Aunque no se encuentra entre los primeros, España también se puede contar como uno de los países que ha estado avanzando hacia la informatización del gobierno en el continente europeo. (Savia, 2016)

A pesar que aún falta mucho camino por recorrer, la sociedad cubana desde hace algunos años viene avanzando en el tema, al respecto el Presidente de los Consejos de estado y de Ministros de la República de Cuba Miguel Díaz-Canel Bermúdez comentó en la clausura de la primera Conferencia Nacional de la UIC “El proceso de informatización de sociedad es una prioridad del Estado cubano” (Rodríguez, 2018). Hoy en día existen 13 provincias del país que ya iniciaron la implementación de su portal, las cuales son: Pinar del Río, Artemisa, Mayabeque, La Habana, el municipio especial Isla de la Juventud, Cienfuegos, Villa Clara, Sancti Spíritus, Camagüey, Granma, Santiago de Cuba y Guantánamo; de las cuales seis ya se encuentran en línea: Pinar del Río (Red Pinar) el cual se encuentra en la primera fase y posee secciones sobre la gestión del Poder Popular para impulsar el

desarrollo local, el impacto de los programas sociales, el quehacer de las organizaciones de masas y también avisos en tiempo real, carteleras deportivas y culturales, la programación de centros recreativos como cabarés y restaurantes, e información sobre todo tipo de trámites, entre otros contenidos (Rivas, 2018), la Isla de la Juventud el cual actualmente se encuentra en la primera fase y tiene entre otros fines emplear herramientas de gobierno electrónico para el ejercicio de los derechos ciudadanos, establecer contacto e-mail y la integración total entre agencias, el sector privado y el pueblo, ofreciendo servicios más personalizados mediante la interacción con cada entidad permitiendo emitir criterios y sugerencias que puedan ser respondidas por los directivos y de igual forma sus espacios estarán destinados a la promoción de eventos, talleres, concursos, avisos entre otros (Sánchez, 2017), Camagüey provincia cuyos proyectos comenzaron con un enfoque hacia la cultura y la recreación, pero actualmente entidades como Servicios Comunes, Planificación Física, Bufetes Colectivos y Trabajo y Seguridad Social se benefician del mismo (Redacción Central, 2018), Granma cuya plataforma brinda información sobre trámites y servicios que realizan y brindan empresas y organismos del territorio, así como posibilita a las personas poder expresar sus opiniones, interactuar en foro-debates, dar seguimiento a inquietudes o generar nuevos planteamientos y además se podrá consultar las fechas previstas para la realización de asambleas de rendición de cuentas y despachos con delegados del Poder Popular (Tasé, 2018), Las Tunas (Panorama Tunero) que también se encuentra en la primera fase, cuenta con información de las entidades de la provincia, de trámites y servicios que se ofrecen en las mismas, así como notas de actualidad, datos de gestiones, el pronóstico meteorológico y las carteleras tanto cultural como deportiva (Bello, 2018) y Santiago de Cuba, mientras que Ciego de Ávila y Holguín se encuentran creando las condiciones para sumarse a este proceso. Matanzas también se encuentra en este grupo, pues en ella ya se encuentra funcionando el portal del ciudadano, que se inauguró el 26 de noviembre de 2018 con el nombre Somos Matanceros (www.matanceros.gob.cu), el cual los sitúa en la primera fase de gobierno electrónico, permitiendo acceder a información actualizada de las distintas entidades, horarios, carteleras, noticias y requerimientos

para distintos trámites. (Rosario Zapata, 2018) Además, se encuentra abierta una sala institucional con el objetivo de coordinar, catalizar y supervisar el funcionamiento de este mecanismo de participación popular, que dentro de poco tiempo se encontrará habilitado mediante la plataforma Bienestar (Acosta, 2018), la cual recogerá 16 procesos, los cuales se irán informatizando 3 cada año y la llevarán a la segunda fase de gobierno electrónico y en la cual estarán integradas la presente investigación y otras similares que actualmente se están desarrollando y que no poseen ningún antecedente en la provincia como son: el proceso de planteamientos a los delegados, asuntos planteados, gestión de inspectores, los subsidios y la solicitud de consultas médicas en los hospitales provinciales (Rosario Zapata, 2018).

1.4 Metodología de desarrollo

El Proceso de Desarrollo y Gestión de Proyectos de Software es una metodología que tiene como objetivo la producción eficiente de un producto de software que satisfaga los requisitos de un especialista funcional con una planificación y una estimación de recursos predecibles. Esta combina un desarrollo iterativo e incremental, el cual no es más que aquel en el que las fases del ciclo de vida se descomponen en iteraciones; con uno basado en componentes que permite alcanzar un mayor nivel de reutilización de software, aún en contextos distintos a aquellos para los que fue diseñado y permite que las pruebas sean ejecutadas probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados.

Esta metodología está compuesta por cinco fases, las cuales no son más que divisiones dentro del mismo proyecto que suelen completarse de manera secuencial, pero en determinadas situaciones pueden superponerse, cada una de ellas, concluye con un hito, cuyo objetivo fundamental es evaluar y decidir el paso a la siguiente fase de desarrollo, es decir, son actividades de duración nula (0 horas) que se utilizan como puntos de control. Estas fases son: inicio, modelación, construcción, explotación experimental y despliegue.

En la de inicio se describen los objetivos y el alcance del proyecto, se identifican los involucrados y ejecutores (entidades involucradas), se estiman las actividades a realizar durante todo el ciclo de desarrollo del proyecto, se establece la estrategia a seguir para realizar la modelación del negocio y la captura de requisitos y de ser necesario se estiman los recursos materiales que deberán ser adquiridos.

En la fase de modelación se capturan las partes esenciales del sistema, donde se identifican los procesos de negocio fundamentales y se aceptan los requerimientos funcionales, obteniéndose la línea base de la arquitectura y una estrategia de construcción de la aplicación aprobada por los implicados en el proyecto.

La tercera fase o construcción es donde se aclaran los requisitos restantes y se completa el desarrollo del sistema sobre una base estable de la arquitectura. En ella todas las características, componentes, y requerimientos deben ser integrados, implementados, y probados en su totalidad, obteniéndose una versión liberada del producto.

Durante la Explotación Experimental se convierte la versión obtenida en la fase anterior en una solución estable, donde se eliminan los errores que surgen durante las pruebas y se obtiene una certificación funcional y de seguridad del producto.

En la última fase o Despliegue se instala y configura el sistema para un ambiente de producción real, se capacita al personal que usará la aplicación y se continúa dando soporte durante la explotación del sistema, culminando de ser preciso con transferencias tecnológicas.

Si se observa desde el punto de vista ingenieril cada fase desarrolla y evoluciona una parte del producto con un esfuerzo variable en recursos humanos y logísticos. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012).

Esta metodología se utiliza en la presente investigación por todas las ventajas que anteriormente se mencionaron, unido a que es la que utiliza la empresa XETID, a la cual pertenece el proyecto.

1.5 Herramientas y tecnologías

1.5.1 Lenguaje de modelado: UML y BPMN

El Lenguaje Unificado de Modelado (UML)

UML es un lenguaje que se utiliza para escribir los planos del software y permite visualizar, especificar, construir y documentar todos los artefactos que componen un sistema con gran cantidad de software. Permite modelar desde sistemas de información hasta aplicaciones distribuidas basadas en Web, pasando por sistemas empotrados de tiempo real. UML es independiente del proceso, aunque para que sea óptimo debe usarse en un proceso dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental. Ayuda a interpretar grandes sistemas mediante gráficos o texto obteniendo modelos explícitos que ayudan a la comunicación durante el desarrollo pues al ser estándar, los modelos podrán ser interpretados por personas que no participaron en su diseño sin ninguna ambigüedad. Este lenguaje se utilizó para el modelado del proceso (Raúl, 2000, pág. 35)

Modelo y Notación de Procesos de Negocio (BPMN)

Es una notación gráfica que describe la lógica de los pasos de un proceso de Negocio. Esta notación ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades. BPMN proporciona un lenguaje común para que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente. Este brinda a todos los analistas de negocio y desarrolladores, un lenguaje común para representar los distintos flujos de trabajo que tienen que modelar. La notación es sencilla y fácil de comprender, pero sobre todo da una capacidad de expresividad que hace al modelado de procesos una tarea simple. Por otro lado, BPMN es un estándar internacional de modelado de procesos aceptado por la comunidad, es independiente de cualquier metodología de modelado de procesos, crea un puente estandarizado para disminuir la brecha entre los procesos de negocio y la implementación de estos y permite modelar los procesos de una manera unificada y estandarizada permitiendo un entendimiento a todas las personas de una organización. Está planeada para dar soporte únicamente a aquellos procesos que

sean aplicables a procesos de negocios. (Soto, 2016) Es una notación que presenta grandes ventajas y que se utilizó para realizar todo el modelado del proceso.

1.5.2 Herramientas para el modelado: Visual Paradigm

Visual Paradigm

Es una herramienta de diseño UML y una herramienta CASE diseñado para ayudar al desarrollo de software, que soporta el ciclo de vida completo de del desarrollo de software: análisis y diseño orientado a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Además, soporta los estándares de modelado clave como son UML 2.4, SoaML, SysML, ERD, DFD, ArchiMate y BPMN, siendo este último el que se utilizará en la presente investigación, haciendo de esta herramienta una buena opción para el trabajo en la misma. (Picazo, 2013) Esta se tomó para realizar todos los diagramas que propone Prodesoft como parte del modelado del negocio.

1.5.3 Lenguajes de programación: PHP, CSS, JavaScript, HTML y JQuery

Preprocesador de Hipertexto (PHP)

Es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. Al nivel más básico, PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. Además, presenta soporte para una gran cantidad de bases de datos, lo que es una de sus más potentes características. En ella escribir una interfaz web para una base de datos resulta una tarea simple. Algunas de las bases de datos que soporta son: Adabas D, Ingres, Oracle (OCI7 and OCI8), dBase, InterBase, Empress, FrontBase, Solid, FilePro, mSQL, Sybase, IBM, DB2, MySQL, Velocis, Informix, ODBC, Unix dbm y PostgreSQL, la cual se utilizará en esta investigación. PHP también soporta el uso de otros servicios que usen protocolos como IMAP, SNMP, NNTP, POP3, HTTP y derivados, características que hacen de este un potente lenguaje y que lo reafirman

como el indicado para las actuales necesidades. (autores, 2001) Este lenguaje se utilizó para la programación de la lógica del negocio en los plugin.

Hojas de Estilo en Cascada (CSS)

Según el sitio web Lenguaje CSS (Manz) este es un lenguaje que los navegadores conocen y pueden entender. Su nombre parte de un concepto simple pero potente: aplicar estilos a uno o varios documentos. Se le conoce como estilos en cascadas porque se aplican de arriba hacia abajo y en el caso de ambigüedad, se siguen diferentes normas para resolverlas. La idea de este lenguaje es la de utilizar el concepto de separación de presentación y contenido, intentando que los documentos HTML incluyan solo información y datos, relativos al significado de la información a transmitir, y todos los aspectos relacionados con el estilo se encuentren en otro documento independiente. Entre sus ventajas se encuentran que las modificaciones que se le realicen a la presentación se hagan en un solo lugar, sin tener que tocar los documentos HTML, además se reduce la duplicación de estilos en diferentes lugares por lo que la información a transmitir es menor y es más fácil crear versiones con diferente presentación para tablets, smartphones u otros dispositivos, lo cual es de gran importancia en la actualidad, y sobre todo en la investigación que aquí se presenta, pues uno de sus objetivos, es que sea utilizada por la población, desde su casa en cualquiera dispositivo informático que disponga. Este lenguaje permitió darles el estilo a las vistas del proceso.

Lenguaje de Marcas de Hipertexto (HTML)

Según el sitio Computer Hoy (2014) , HTML “Hypertext Markup Lenguaje”, es un lenguaje de programación que describe el formato que tendrá el contenido de un documento. Este sirve para la elaboración de páginas web definiendo una estructura básica y un código para ellas. Su última versión permite la reproducción interna de videos, audios y juegos sin necesidad de utilizar programas adicionales. Entre sus ventajas se encuentran que es compatible con los navegadores web más populares de la actualidad como Internet Explorer, Mozilla Firefox, Safari y Google Chrome, además de funcionar correctamente con smartphones y tablets. Otra de las

ventajas que señala el sitio Hipertextual (2013) es que los usuarios pueden acceder a sitios web de manera offline. Además, incluye la geolocalización como uno de sus puntos fuertes. Este lenguaje fue el que se tomó para modelar el cuerpo de las interfaces.

JavaScript

Según el sitio web de comercio electrónico Ecommerce platforms (2018) JavaScript es un lenguaje que hace que las páginas web sean dinámicas e interactivas mediante scripts. Este lenguaje se utiliza principalmente para proporcionar que una experiencia más amigable para el usuario, incluyendo mejoras en la interfaz de usuario como menús, cuadros de diálogos, animaciones, gráficas 2D y 3D, mapas interactivos y reproductores de videos. Por otra parte, según el sitio Maestros del Web (2007) es de fácil utilización tanto para profesionales como para personas que se inician en el desarrollo web pues no requiere compilación, es decir, los navegadores son los encargados de interpretar los códigos del lenguaje por lo que necesita ningún framework para su comprensión. Su sintaxis es similar a la usada en Java y C y permite la utilización de programación orientada a objetos. Otra característica que aumenta su utilidad es que JavaScript es soportado por la mayoría de los navegadores actuales.

JQuery

Es una biblioteca de JavaScript rápida y concisa que simplifica el documento HTML, el manejo de eventos, la animación y las interacciones AJAX para el desarrollo web. JQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían gran cantidad de código, es decir, permite lograr grandes resultados en menos tiempo y espacio. Su característica principal es que permite cambiar el contenido una página web sin necesidad de recargarla mediante la manipulación del árbol DOM y peticiones AJAX. Otras características de esta biblioteca son que permite la selección de elementos, la manipulación de la hoja de estilos CSS, soporta extensiones y presenta utilidades como obtener información del navegador, operar con objetos y vectores. Además,

tiene como una gran ventaja que es compatible con los navegadores Mozilla Firefox 2.0+, Internet Explorer 6+, Safari 3+, Opera 10.6+ y Google Chrome 8+. (Villareal Fuentes, 2013) Tanto esta librería como el propio JavaScript fueron tomadas para realizar las validaciones de los campos en las vistas.

1.5.4 Lenguaje de consultas: SQL

Lenguaje Estructurado de Consultas SQL

Según la web de Microsoft (s.f) es un lenguaje de computación para trabajar con conjuntos de datos y las relaciones entre ellas. Este lenguaje no es difícil de leer y entender por cualquier usuario y al igual que otros lenguajes de computación es un estándar reconocido por ISO y ANSI. Se utiliza para describir conjuntos de datos que ayudan a responder preguntas y su sintaxis se basa en la del idioma inglés. SQL no solo manipula datos sino también crea y modifica el diseño de objetos en la base de datos lo que permita que sea un lenguaje intuitivo con un alto grado de utilidad y de facilidades para el programador, características que hacen reconocerlo como una buena elección para su utilización en la aplicación como lenguaje para realizar las consultas a la base de datos desde ProcessMaker.

Lenguaje de Consulta de Doctrine DQL

DQL es el lenguaje de consulta propio de Doctrine, el cual según el sitio web Marco de Desarrollo de la Junta de Andalucía (s.f), proporciona potentes capacidades de consulta sobre el modelo de objetos. Imaginando todos los objetos dispersos en algún almacenamiento (como un objeto de base). Al escribir consultas DQL, la consulta recoge un cierto subconjunto de los objetos del almacenamiento. Tiene instrucciones del tipo SELECT, UPDATE y DELETE que se asignan a sus correspondientes sentencias SQL. INSERT no se permite en DQL, porque las entidades y sus relaciones tienen que ser introducidos en el contexto de persistencia a través del método `persist` () del EntityManager para garantizar la coherencia de su modelo de objetos. En la actual investigación DQL se implementó como parte de los plugin para el trabajo con la base de datos.

1.5.5 Gestor de Base de Datos: PostgreSQL y MySQL

PostgreSQL

Es un gestor de bases de datos orientadas a objetos muy conocido y usado en entornos de software libre porque cumple los estándares SQL92 y SQL99, y también por el conjunto de funcionalidades avanzadas que soporta, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales. Entre las características principales que permiten que esta sea una herramienta importante a tener en cuenta para la ejecución de cualquier proyecto se pueden citar que la API de acceso al SGBD se encuentra disponible en C, C++, Java, Perl, PHP, Python y TCL, entre otros; cuenta con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario; su administración se basa en usuarios y privilegios; sus opciones de conectividad abarcan TCP/IP, sockets Unix y sockets NT; los mensajes de error pueden estar en español y hacer ordenaciones correctas con palabras acentuadas o con la letra 'ñ'; es altamente confiable en cuanto a estabilidad se refiere; puede extenderse con librerías externas para soportar encriptación y búsquedas por similitud fonética entre otros; permite un control de concurrencia multi-versión, lo que mejora sensiblemente las operaciones de bloqueo y transacciones en sistemas multi-usuario; contiene soporte para vistas, claves foráneas, integridad referencial, disparadores, procedimientos almacenados, subconsultas y casi todos los tipos y operadores soportados en SQL92 y SQL99 y por último permite la implementación de algunas extensiones de orientación a objetos. Además, puede funcionar en múltiples plataformas basadas en Unix y, a partir de la versión 8.0 también en Windows de forma nativa. (Marc Gibert Ginestá, s.f) Este se tomó como gestor de base de datos de la aplicación.

MySQL

Según el sitio web Espestudio (2005), MySQL es un sistema de administración de bases de datos que se encuentra entre los más extendidos dentro del campo. Esta aplicación utiliza múltiples tablas para almacenar y organizar la información, fue

escrita en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java. Además, es un sistema open source, es decir, su utilización es gratuita y se puede modificar en su totalidad, pudiendo descargarse su código fuente. Siendo estas características las que lo han convertido en una de las herramientas más utilizadas por los programadores orientados a internet y lo que lo lleva a utilizarse como gestor de la base de datos que implementa ProcessMaker por defecto.

1.5.6 Framework: ProcessMaker, Zend Framework, Doctrine, ExtJs y Bootstrap

ProcessMaker

“ProcessMaker es una solución de software de flujos de trabajo, de código abierto simple y rentable. También conocido como Gestor de procesos empresariales (BPM), ProcessMaker ayuda a las organizaciones de todos los tamaños para diseñar fácilmente, automatizar e implementar procesos de negocio.

La caja de herramientas ProcessMaker permite a los usuarios de negocio crear formas y mapas de flujos de trabajo completamente funcionales. El software está completamente basado en web, lo que facilita la coordinación del flujo de trabajo entre los usuarios, departamentos y organizaciones. Como una aplicación de gran alcance, ProcessMaker puede interconectarse con sistemas que incluyen la gestión de documentos, ERP, CRM y aplicaciones de inteligencia empresarial.

ProcessMaker es ligero, extremadamente eficiente, e implica los gastos generales más bajos de cualquier BPM en la industria. Los clientes empresariales de ProcessMaker disfrutan de un pleno apoyo, la suite BPM es de calidad superior con los beneficios añadidos de código abierto. (Negocios y Soluciones Informáticas, s.f). Este fue el framework que utiliza la empresa para implementar los procesos de negocios que conllevan trámites, por la amplia cantidad de ventajas que presenta el mismo.

Zend Framework

presenta código abierto para el desarrollo de aplicaciones y servicios web con PHP. Es una implementación que usa código 100% orientado a objetos. La estructura de los componentes de Zend Framework es algo único; cada componente está construido con una baja dependencia de otros componentes. Esta arquitectura débilmente acoplada permite a los desarrolladores utilizar los componentes por separado. Zend Framework hace énfasis fundamentalmente en la calidad del código, a través de una batería de pruebas unitarias, utilizando PHP Unit, cubriendo alrededor del 85% del código escrito para el framework. Como objetivo, Zend Framework persigue una meta muy clara: Simplicidad, ante todo. Busca tener una Interfaz de Programación de Aplicaciones (API) muy fácil de aprender, de manera que como desarrolladores podamos comenzar a escribir aplicaciones rápidamente. Esta simplicidad se ve representada en tres puntos:

- Simplicidad en el uso. Menos código y más simple de leer.
- Simplicidad representa código más estable y con menos probabilidad de error.
- Simplicidad para mantener luego el código. (Zend Framework, 2009).

Este se utilizó como framework para la implementación de la lógica de negocio en los plugin.

Doctrine

Es un ORM o (Object Relation Mapper) es una técnica de programación que nos permite convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, es decir, las tablas de nuestra base de datos pasan a ser clases y los registros objetos que podemos manejar con facilidad.

Utilizar un ORM tiene una serie de ventajas que nos facilitan enormemente tareas comunes y de mantenimiento:

- Reutilización: La principal ventaja que aporta un ORM es la reutilización permitiendo llamar a los métodos de un objeto de datos desde distintas partes de la aplicación e incluso desde diferentes aplicaciones.
- Encapsulación: La capa ORM encapsula la lógica de los datos pudiendo hacer cambios que afectan a toda la aplicación únicamente modificando una función.
- Portabilidad: Utilizar una capa de abstracción nos permite cambiar en mitad de un proyecto de una base de datos MySQL a una Oracle sin ningún tipo de complicación. Esto es debido a que no utilizamos una sintaxis MySQL, Oracle o SQLite para acceder a nuestro modelo, sino una sintaxis propia del ORM utilizado que es capaz de traducir a diferentes tipos de bases de datos.
- Seguridad: Suelen implementar mecanismos de seguridad que protegen nuestra aplicación de los ataques más comunes como SQL Injections.
- Mantenimiento del código: Gracias a la correcta ordenación de la capa de datos, modificar y mantener nuestro código es una tarea sencilla. (tecnoretals, s.f)

ExtJs

Es una librería JavaScript que está basado completamente a la programación orientada a objeto. Cada objeto contiene lo típico: propiedades, métodos y eventos. Basa toda su funcionalidad en JavaScript a través de librerías. Así, en tiempo de ejecución carga y crea todos los objetos HTML a través del uso intenso de DOM. Los datos son obtenidos con AJAX a través de XML. Una de las grandes ventajas de utilizar ExtJS es que permite crear aplicaciones complejas utilizando componentes predefinidos, así como un manejador de diseños similar al que provee Java Swing. Existe un balance entre Cliente–Servidor. La carga de procesamiento se distribuye, permitiendo que el servidor al tener menor carga, pueda manejar más clientes al mismo tiempo. ('Cutter' Blades, s.f)

Bootstrap

Es un framework desarrollado y liberado por Twitter que tiene como objetivo facilitar el diseño web. Sus webs, son adaptables a cualquier dispositivo y tamaño de pantalla y siempre se van a ver bien. Además, permite que la carga sea rápida y que la navegación sea fluida e intuitiva. Permite utilizar elementos de HTML y CSS. Se integra bien con las librerías de JavaScript y permite maquetar por columnas que son muy configurables y fáciles. (Punto Abierto, 2016) Este framework se utilizó para la elaboración de las vistas, a través de la utilización de HTML y CSS, que son lenguajes que este implementa.

1.5.7 Servicios web: XML y WSDL

Extensible Markup Language (XML)

XML es un lenguaje que permite jerarquizar, estructurar la información y como describir los contenidos, dentro del mismo documento, así como su reutilización. Este puede utilizarse sin tener que amoldarse a las reglas específicas de un fabricante y sus archivos válidos, también pueden utilizarse fuera de la web. Entre los principales usos de XML se pueden citar, el uso aplicado a los sitios web, que permite separar contenido y presentación, y que los mismos datos se pueden mostrar de varias formas distintas sin demasiado esfuerzo. Otro de sus usos es la comunicación entre aplicaciones por representar los datos de forma muy simple, fácil de transmitir por la red y estándar. Además, se utiliza para la configuración de programas. XML posee una estructura más abierta y extensible ya que no precisa versiones para que pueda funcionar en navegadores que se desarrollen en el futuro, lo que trae consigo grandes ventajas para la actualización y seguimiento de la aplicación. (Lapuente, 2018)

Lenguaje de Descripción de Servicios (WSDL)

Es una notación XML para describir un servicio web. Una definición WSDL indica a un cliente cómo componer una solicitud de servicio web y describe la interfaz que proporciona el proveedor del servicio web. Una definición WSDL se divide en varias secciones que especifican la interfaz lógica y los detalles físicos de un servicio web.

Los detalles físicos incluyen información de puntos finales, como el número de puerto HTTP y la información de enlaces que especifica cómo se representa la carga útil SOAP y qué transporte utiliza. El editor WSDL da soporte a una vista de diseño gráfico para que pueda desplazarse por desde el WSDL hasta sus raíces de mensajes asociadas. (IBM, 2017)

1.6 Conclusiones parciales

Después de haber realizado un análisis de los conceptos asociados al dominio del problema, una descripción de la entidad en la que se realizó la investigación, así como los antecedentes, la metodología y las herramientas utilizadas, se arribó a las siguientes conclusiones:

- No existe ningún sistema que resuelva las necesidades detectadas en el objeto de estudio.
- No existe ninguna experiencia anterior de gobierno electrónico en la provincia de Matanzas, aunque actualmente se están dando los primeros pasos en este sentido.
- En el mundo entero, y específicamente en Cuba se avanza cada día más hacia el desarrollo de sistemas para el gobierno electrónico.
- La combinación de herramientas, tecnologías y la metodología de desarrollo de software establecida por la XETID, es la apropiada para la realización del sistema ya que resuelve la situación problémica planteada en la investigación.
- El análisis del estado del arte permitió una mejor comprensión del objeto de estudio, estableciendo las bases para las siguientes fases de la investigación.

Capítulo II: Diseño y construcción de la Solución Propuesta

2.1 Introducción

En este capítulo se detallan los elementos esenciales del sistema a realizar, a través de su modelo conceptual. Además, se describen las características que deberá

cumplir el sistema según lo especificado en la metodología de desarrollo de software Prodesoft, correspondientes a la fase de modelación y se aclaran los requisitos funcionales y no funcionales que debe tener el mismo.

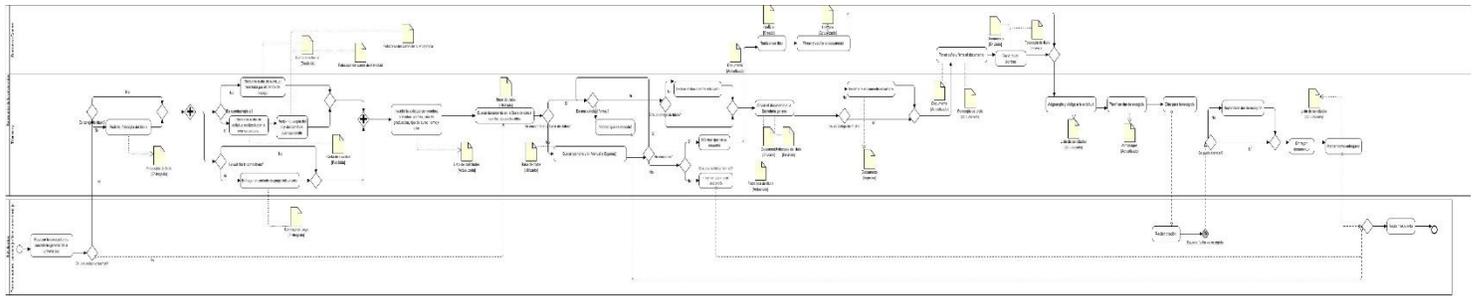
2.2 Modelado del negocio

2.2.1 BPD

BPD “es un diagrama diseñado para representar gráficamente la secuencia de todas las actividades que ocurren durante un proceso, basado en la técnica de “Flow Chart”, incluye además toda la información que se considera necesaria para el análisis. Está diseñado para ser usado por los analistas de procesos, quienes diseñan, controlan y gestionan los procesos. Dentro de un Diagrama de Procesos de negocio BPD se utilizan un conjunto de elementos gráficos, que se encuentran agrupados en categorías. Igualmente, sus componentes mapean las dimensiones Qué, Cómo, Cuándo, Dónde y Por Qué”. (Oscars, s.f)

El proceso modelado en BPD comienza cuando una persona llega a la Secretaría con el objetivo de solicitar un cotejo de título, certificación de notas o plan de estudio. En caso de que sea un cotejo de título, el interesado debe entregar la fotocopia del título junto a una carta de solicitud, y si no, se entregan todos los documentos menos la fotocopia. Con esos datos las técnicas inscriben el caso y buscan el documento en la base de datos y en el archivo físico. Si encuentran el documento lo imprimen, y la secretaria lo firma y acuña. En caso contrario la secretaria elabora un documento oficial como negativa a la solicitud. Luego se planifica el día de recogida y se le notifica al interesado. Ese día el interesado se dirige a la secretaría, se le entrega el documento y finaliza el proceso o en caso contrario si ese día no puede ir por algún motivo se replanifica el día de recogida.

Este proceso se puede observar de forma más detallada en la ilustración siguiente:



2.2.2 Reglas del negocio

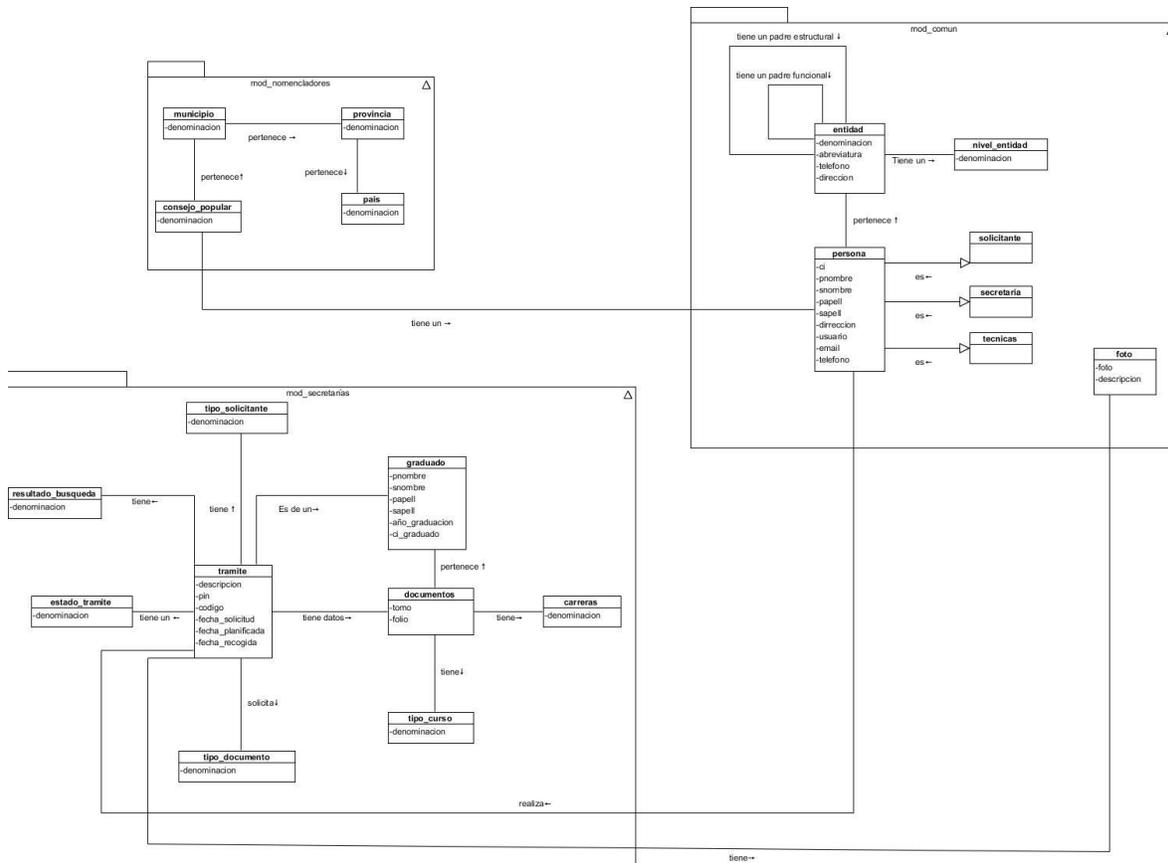
Las reglas del negocio rigen la actividad del mismo, describiendo las políticas, normas, operaciones, definiciones y restricciones presentes en una organización y que son de vital importancia para alcanzar sus objetivos (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012, pág. 46). En el caso de esta investigación se localizaron las siguientes reglas:

1. Los trámites solo se realizan de forma presencial.
2. Para realizar un trámite es obligatorio presentar una carta de solicitud del centro de trabajo o si el solicitante es cuentapropista una carta realizada por él que debe incluir una fotocopia del carné de cuentapropista y una del carné de identidad.
3. Si el solicitante es una consultoría jurídica, debe entregar una copia del contrato de pago que se le realizó.
4. El documento final solo puede ser firmado y acuñado por el Secretario General de la Universidad.
5. Si el documento solicitado no existe es deber del Secretario General realizar un documento legal que es antecedente de un proceso judicial.
6. Las técnicas de la Secretaría son las encargadas de realizar todo el trámite hasta su entregar el documento solicitado.
7. Todos los trámites deben ser incluidos en el sistema, aunque su respuesta sea negativa.

8. Los cuentapropistas son considerados por la Secretaría, personas naturales.
9. Si en la búsqueda realizada inicialmente en la Base de Datos no aparece el documento hay que realizar una nueva búsqueda en el archivo físico o en el software Sigenus.
10. La carta de solicitud que se adjunta debe ser una copia de la original, no realizada en ningún editor de texto.
11. La Consultoría Jurídica solamente solicita el trámite de cotejos de títulos.
12. Para la realización de los trámites se tiene un total de hasta 30 días hábiles a partir del día en que se realiza la solicitud.

2.2.3 Modelo Conceptual

“El Modelo conceptual, es el modelo que explica los conceptos más significativos en el dominio del problema, teniendo como cualidad esencial representar elementos del mundo real. Un concepto en el Modelo de Dominio o Conceptual es un elemento lógico o físico que ayuda a entender el problema, es parte del lenguaje utilizado por el cliente y generalmente se nombra como sustantivo. Los conceptos pueden o no tener atributos que lo caractericen en el mundo real”. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012, pág. 47). El Modelo Conceptual elaborado para la presente investigación se observa en la ilustración que se presenta a continuación.



2.2.4 Requisitos

2.2.4.1 Requisitos no funcionales

“Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos los requisitos no funcionales son fundamentales en el éxito del producto. Normalmente están vinculados a requisitos funcionales, es decir, una vez se conozca lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser”. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012, pág. 49) Para la presente investigación se seleccionaron los siguientes requisitos no funcionales:

Requisitos de usabilidad

- El sistema debe ser fácil de utilizar por cualquier persona que tenga un conocimiento básico de trabajo en la web o de computación.

- El sistema debe presentar una opción de ayuda sobre las principales funcionalidades que presenta.
- El sistema debe implementarse lo más parecido posible a como se realiza el proceso en la actualidad para lograr una mejor comprensión y adaptación al mismo.

Requisitos de confiabilidad

- Deben establecerse mecanismos que aseguren el reinicio del sistema ante diferentes fallos de forma rápida y eficiente.
- Debe existir sistemas de respaldo eléctrico en los locales donde se encuentren los servidores.

Requisitos de seguridad y privacidad

- El usuario debe autenticarse para acceder al sistema, dependiendo del nivel de acceso se presentarán las interfaces para cada usuario.
- La información existente en el sistema será protegida contra actos ilícitos, de igual manera el origen y fuente de los datos.

Requisitos de ayudas y documentación en línea

- El sistema deberá contar con una ayuda integrada con el objetivo de facilitar al usuario su utilización.

Requisitos de diseño

- Para el diseño e implementación del sistema se debe utilizar el Marco de Trabajo ProcessMaker que presenta grandes ventajas para la elaboración de aplicaciones web para la realización de trámites.

2.2.4.2 Requisitos funcionales

“Los requisitos funcionales definen las condiciones o capacidades que el sistema será capaz de realizar. Estos describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Estos requisitos, al tiempo que avanza el proyecto de software, se convierten en los algoritmos, la lógica y gran parte del código del sistema”. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012, pág. 49) Para la presente investigación se reconocieron los siguientes requisitos funcionales:

1. Realizar solicitud por el solicitante.
2. Realizar solicitud por las técnicas.
3. Validar la solicitud.
4. Mostrar los resultados de la búsqueda.
5. Elaborar un documento legal por la secretaria.
6. Generar documento.
7. Planificar fecha de recogida.
8. Controlar recogida.
9. Replanificar fecha de recogida.
10. Insertar carrera.
11. Modificar carrera.
12. Eliminar de carrera.
13. Insertar tipo de curso.
14. Modificar tipo de curso.
15. Eliminar tipo de curso.
16. Insertar tipo de solicitante.
17. Modificar tipo de solicitante.

18. Eliminar tipo de solicitante.
19. Insertar estado de la solicitud.
20. Modificar estado de la solicitud.

Notificaciones y alertas:

1. Notificar Pin y Código de la solicitud al solicitante.
2. Notificar a la Secretaria General en caso que no se encuentre el documento.
3. Notificar fecha de recogida al solicitante.
4. Notificar respuesta del trámite.
5. Notificar nueva fecha de recogida.

Reportes

1. Reporte General.
2. Reporte por Tipo de Documento.
3. Reporte por Tipo de Solicitud.

2.3 Especificación de requisitos funcionales

A continuación se presentan los diagramas de actividades de los principales requisitos funcionales.

Diagrama de actividades del requisito funcional Realizar solicitud por el interesado

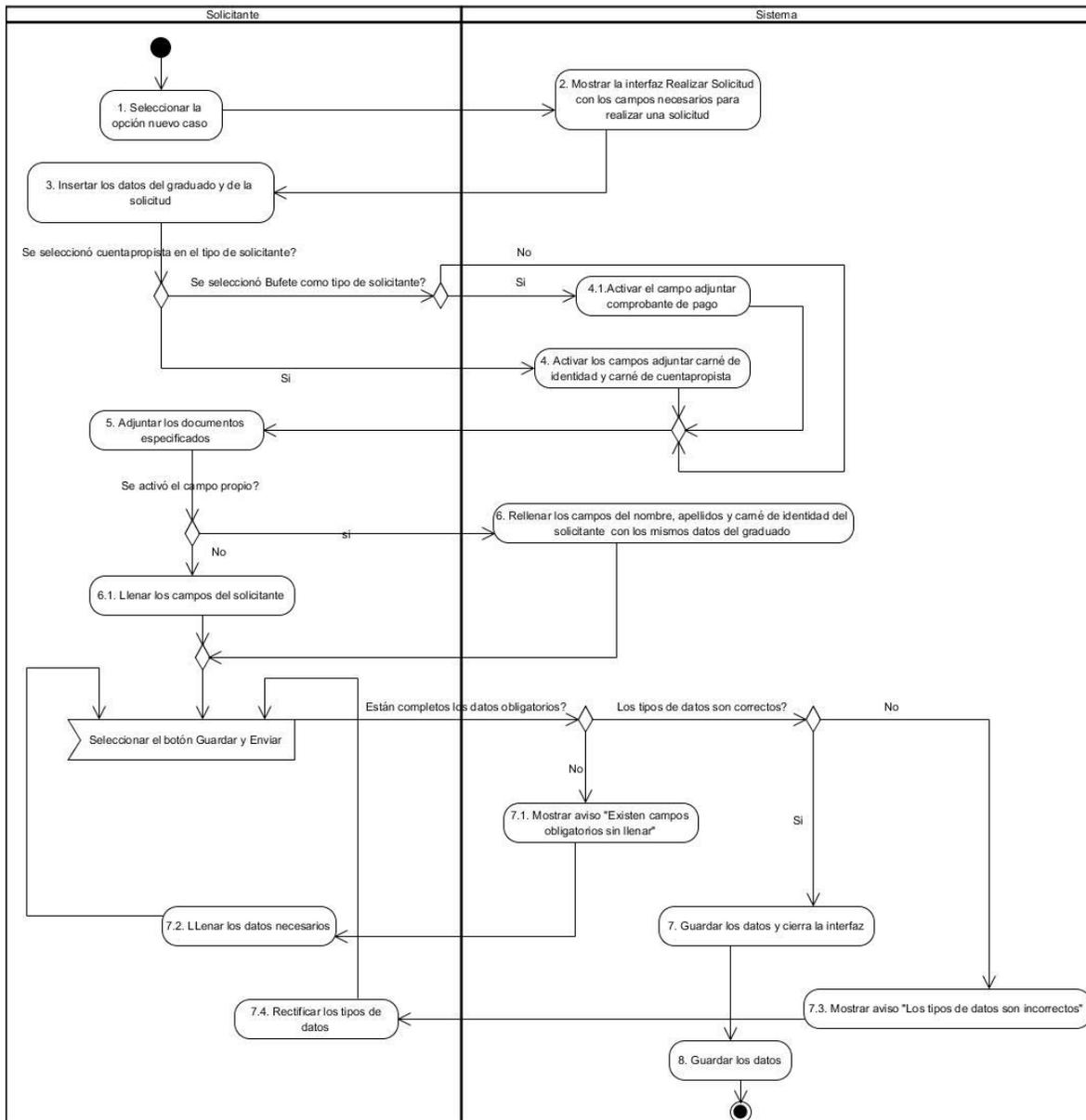


Diagrama de actividades del requisito funcional Realizar solicitud por las técnicas.

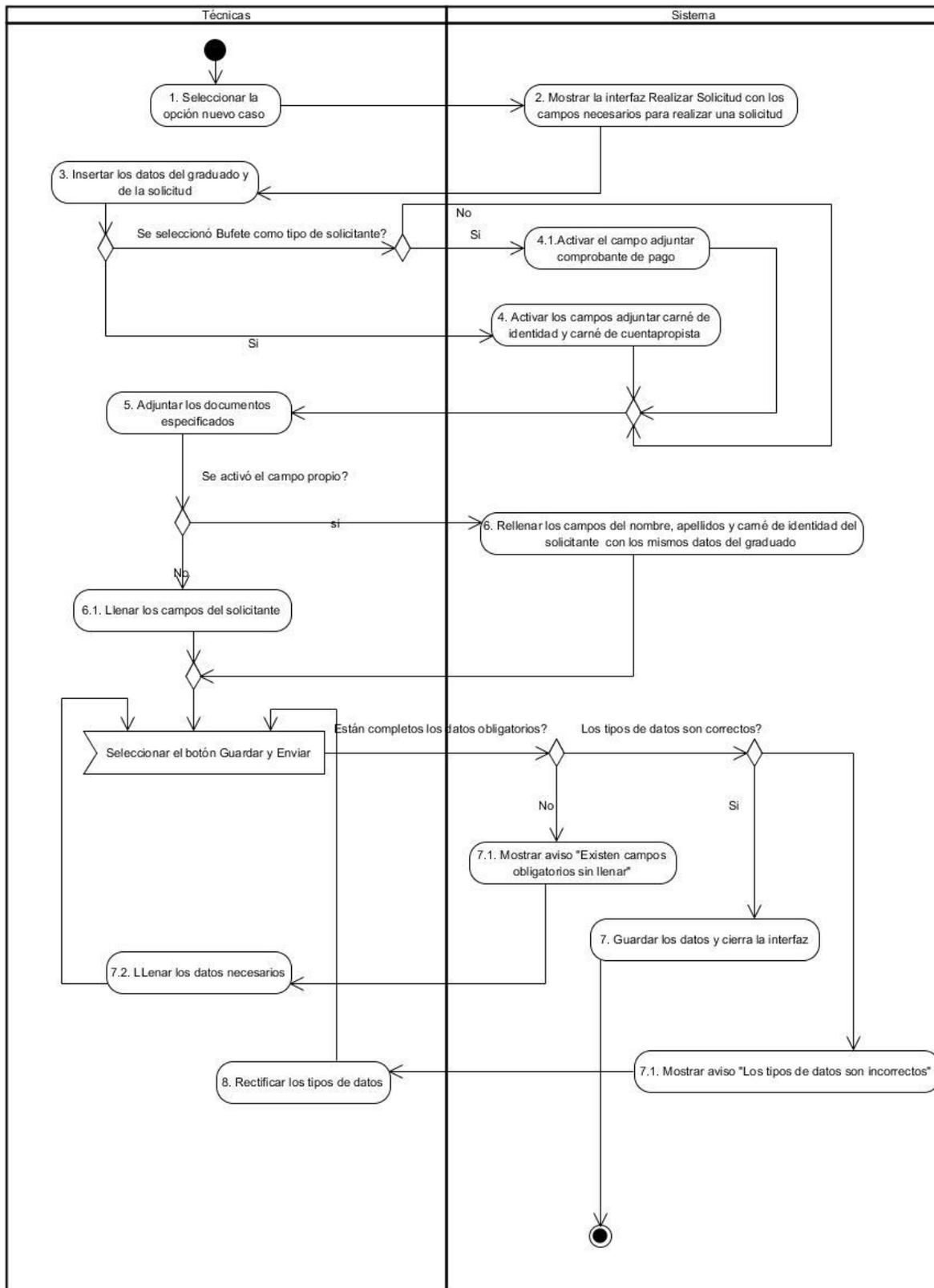


Diagrama de actividades del requisito funcional Validar la solicitud

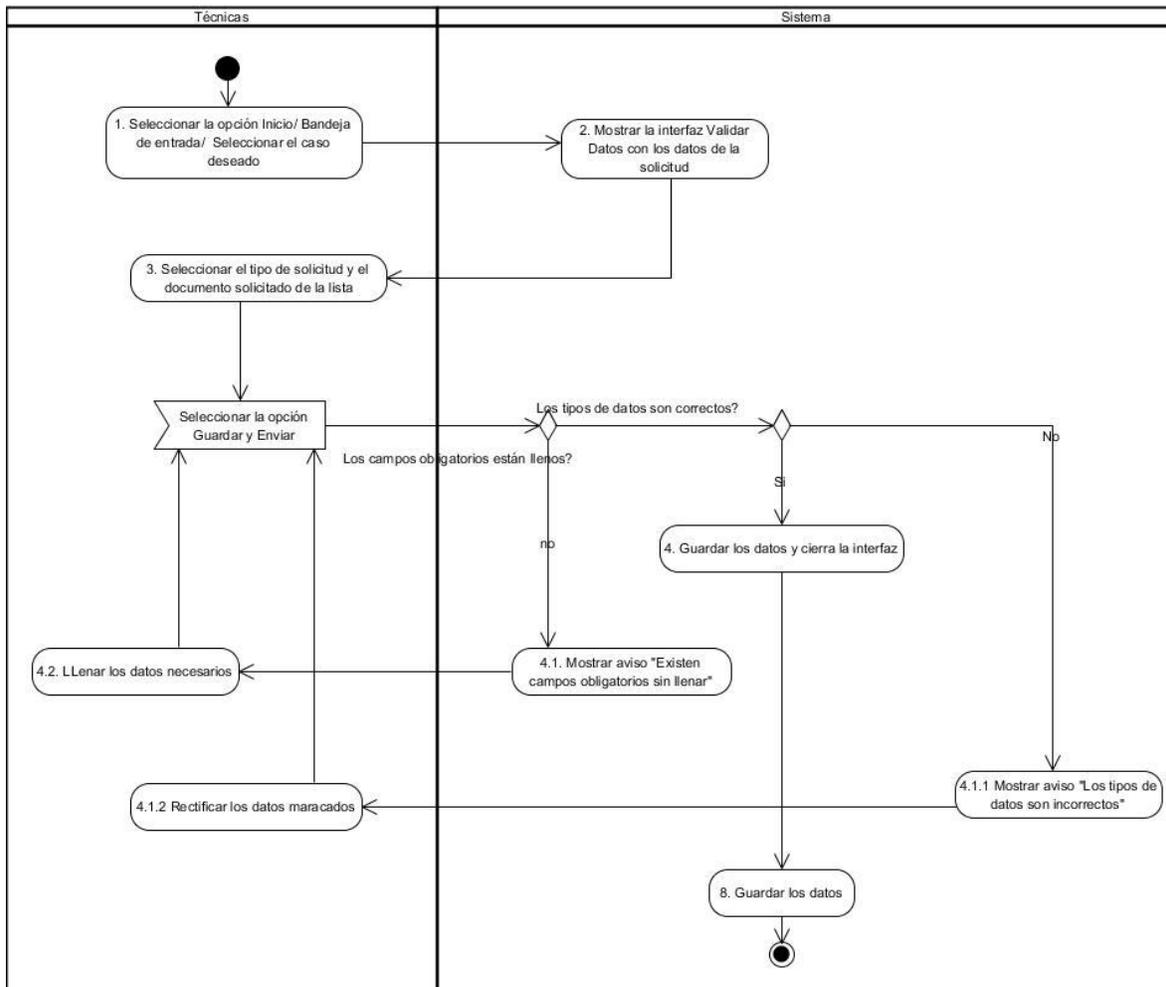


Diagrama de actividades del requisito funcional Mostrar los resultados de la búsqueda.

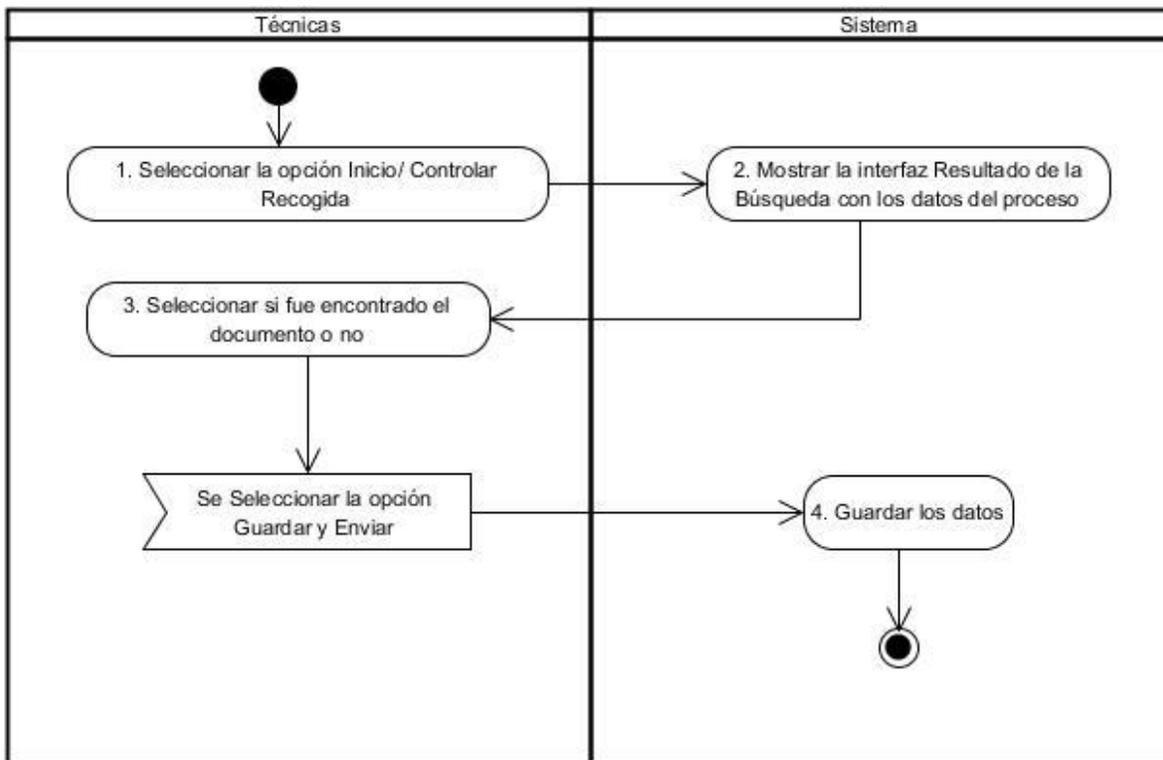


Diagrama de actividades del requisito funcional Elaborar un documento legal por la secretaria

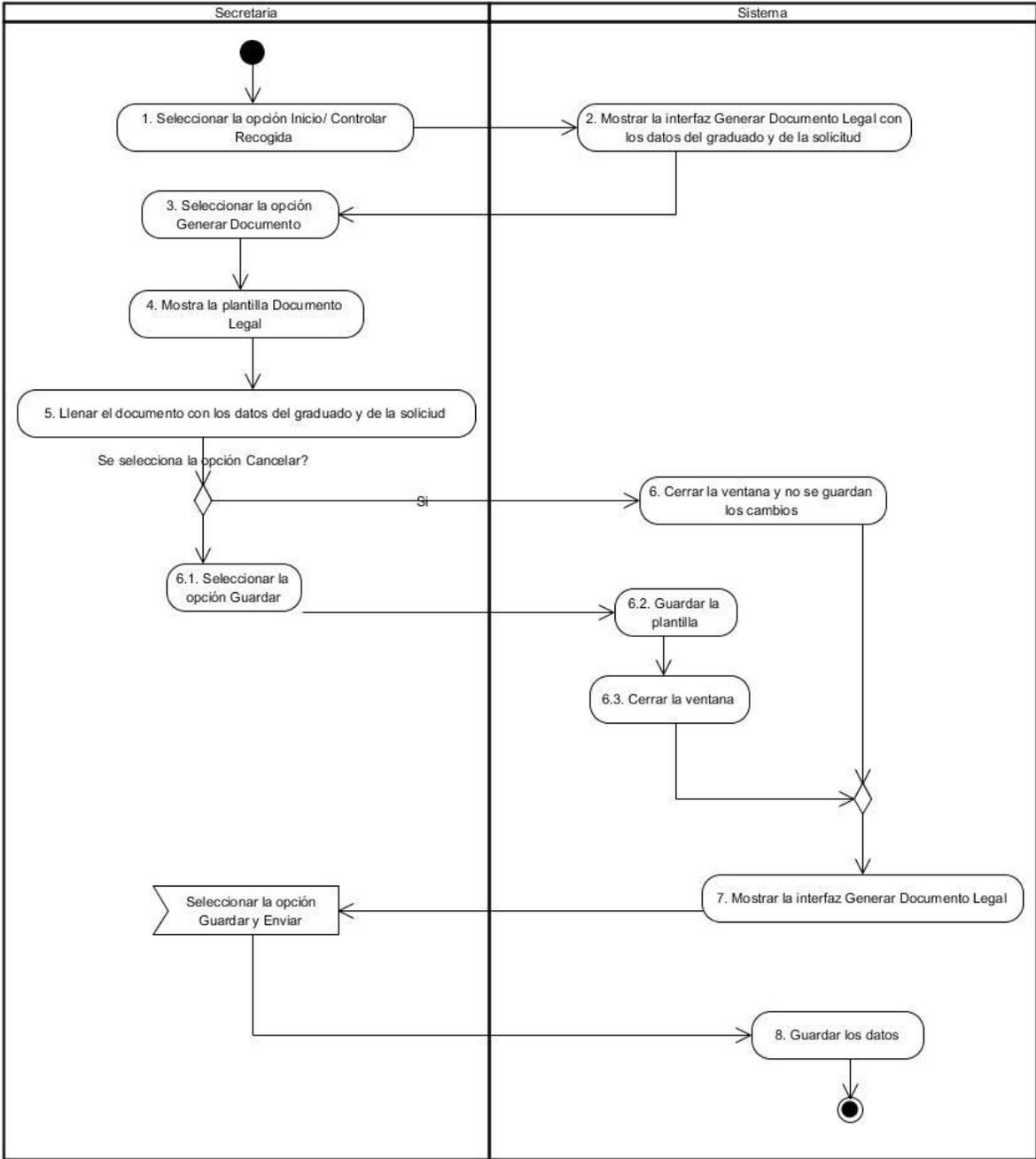


Diagrama de actividades del requisito funcional Generar documento

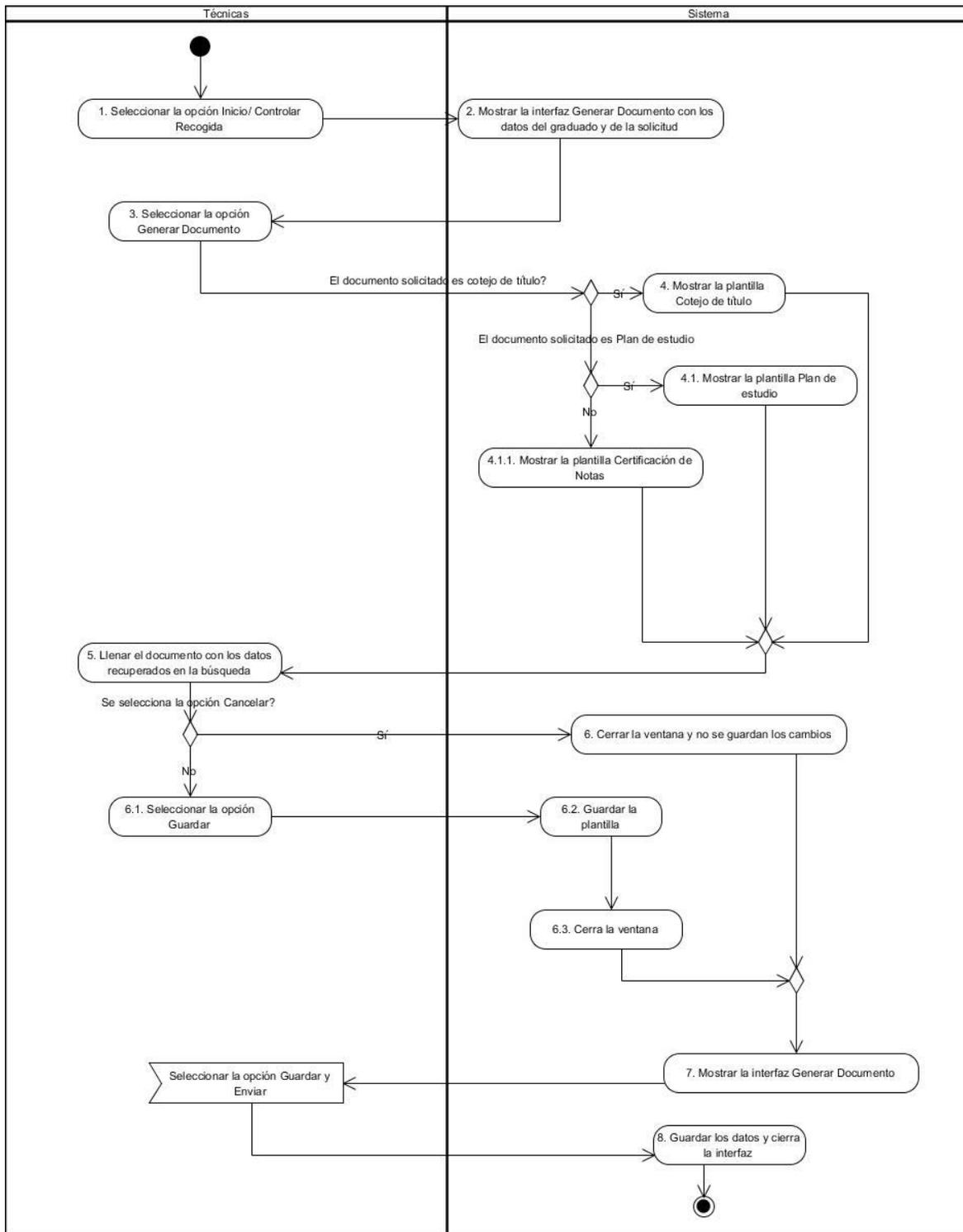


Diagrama de actividades del requisito funcional Planificar fecha de recogida

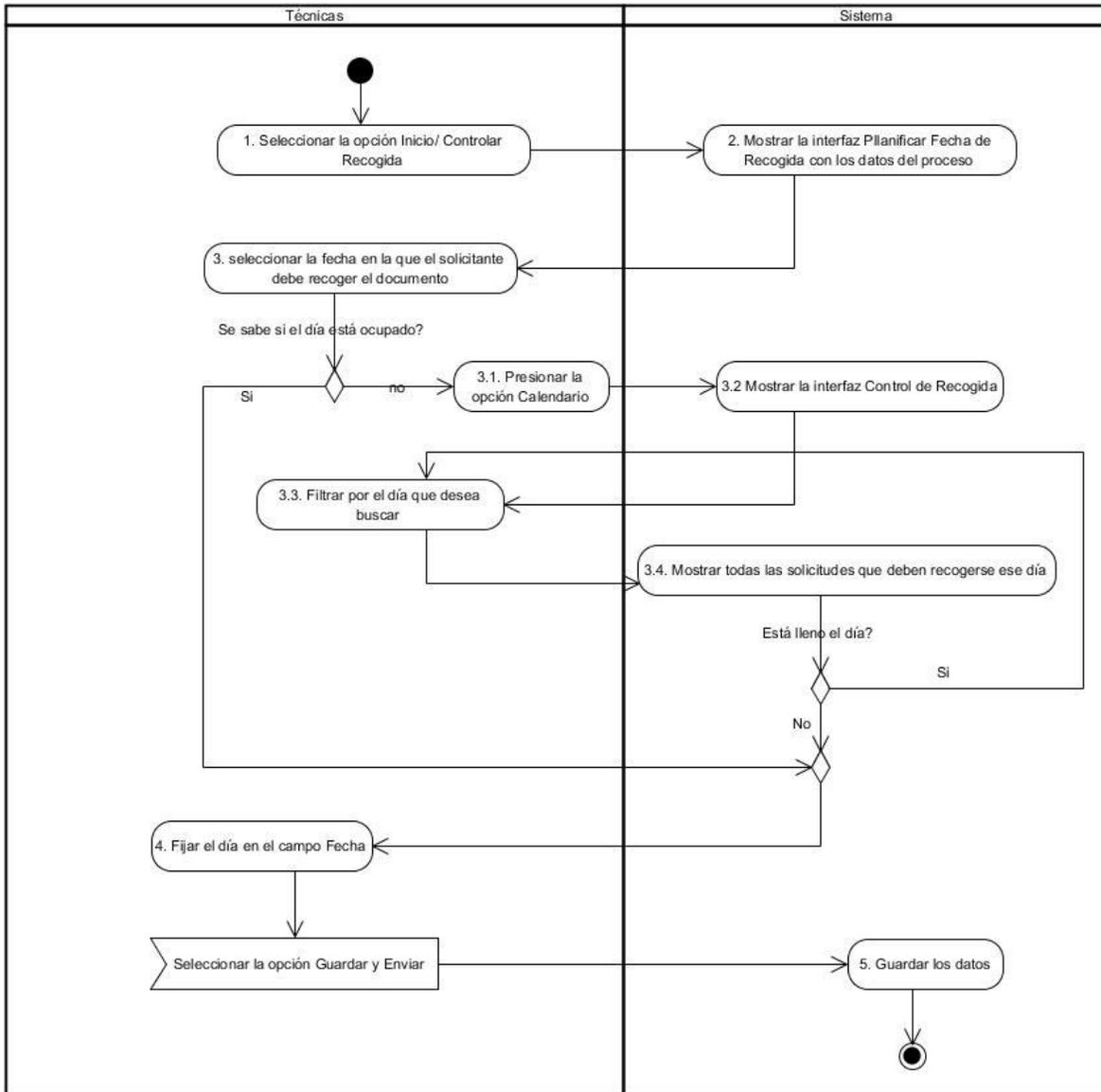


Diagrama de actividades del requisito funcional Replanificar fecha de recogida

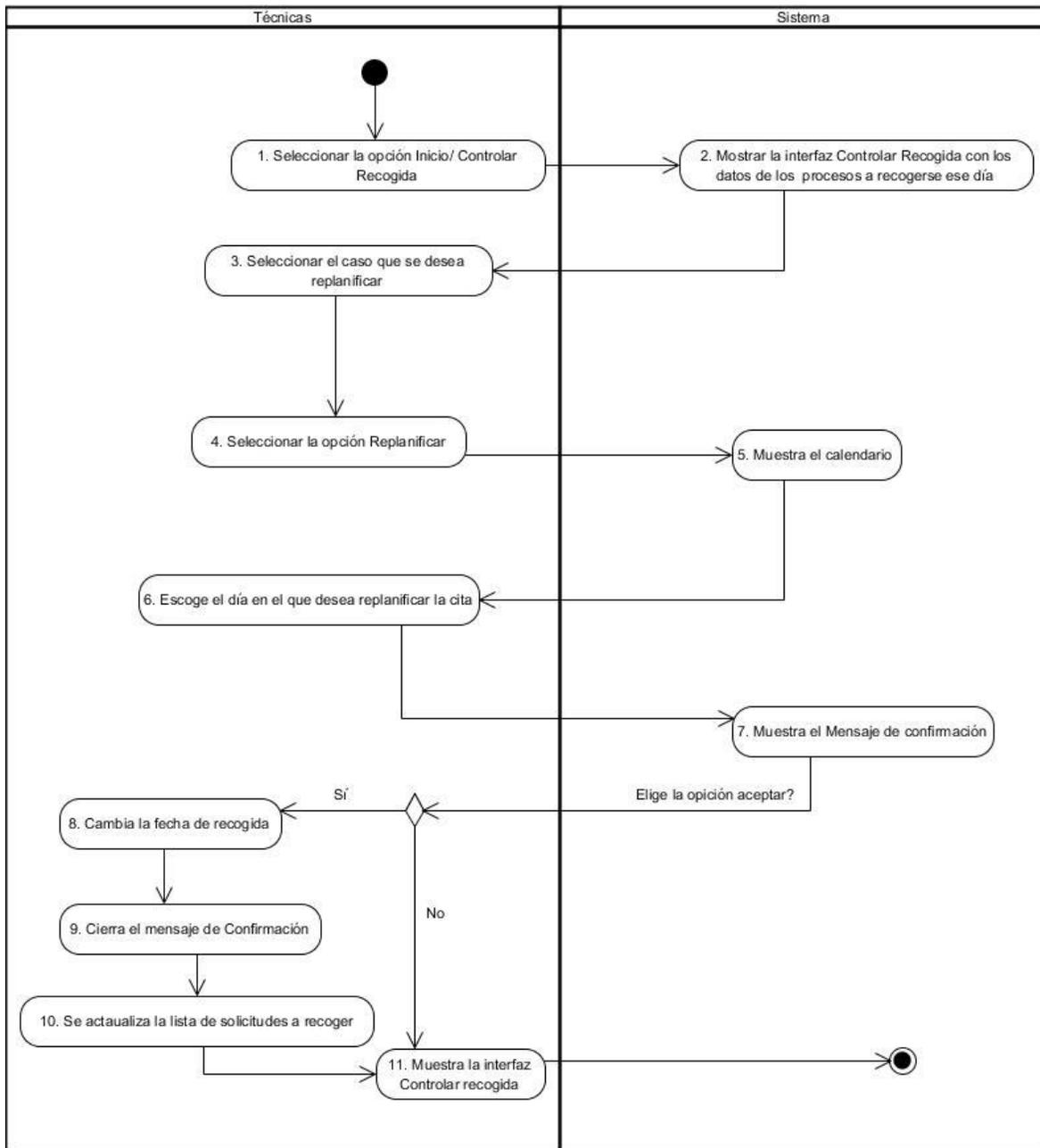
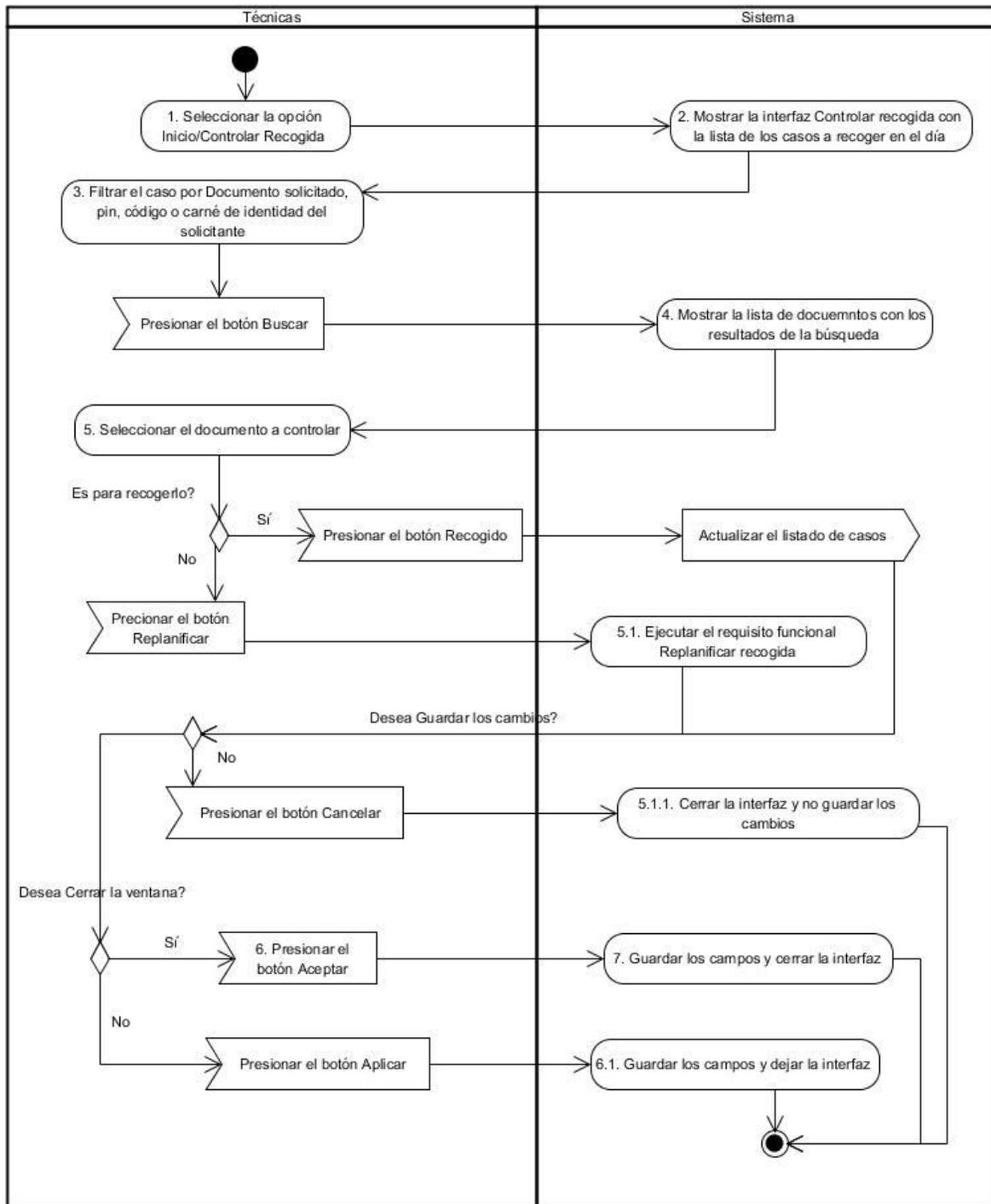


Diagrama de actividades del requisito funcional Controlar recogida



2.4 Implementación de la Arquitectura de Software

Según Prodesoft (2012, pág. 53) "La Arquitectura de Software constituye un puente entre el requisito y el código, ocupando el lugar que en los modelos antiguos se

reservaba para el diseño” o también se puede definir como “la organización fundamental de un sistema encarnado en sus componentes, las relaciones entre ellos, el ambiente y los principios que orientan su diseño y evolución.”

2.4.1 Arquitectura Cliente-Servidor

Para la realización de la presente investigación se utilizó una arquitectura Cliente-Servidor la cual es una aplicación distribuida en la cual las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Las aplicaciones Clientes realizan peticiones a una o varias aplicaciones Servidores, que deben encontrarse en ejecución para atender dichas demandas. (Marini, 2012). Esta arquitectura fue la escogida para trabajar con ProcessMaker, por sus muchas ventajas, entre las que se encuentran la posibilidad de utilizar máquinas considerablemente más baratas que las requeridas por una solución centralizada, basada en sistemas grandes. Además, se pueden utilizar componentes, tanto de hardware como de software, de varios fabricantes, lo cual contribuye considerablemente a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones. Además, facilita la integración entre sistemas diferentes y comparte información permitiendo, por ejemplo, que las máquinas ya existentes puedan ser utilizadas, pero con interfaces más amigables al usuario. Al favorecer el uso de interfaces gráficas interactivas, los sistemas construidos bajo este esquema tienen mayor interacción y es más intuitiva para con el usuario. Una ventaja adicional del uso del esquema Cliente/Servidor es que es más rápido el mantenimiento y el desarrollo de aplicaciones, pues se pueden emplear las 7 herramientas existentes (por ejemplo, los servidores de SQL o las herramientas de más bajo nivel como los sockets o el RPC). Por otra parte, contribuye, además, a proporcionar, a los diferentes departamentos de una organización, soluciones locales, pero permitiendo la integración de la información relevante a nivel global. (Paszniuk, 2013). Aunque presenta desventajas, estas son mucho menores que las ventajas por lo que es una buena solución utilizarla para el fin propuesto.

2.4.2 Arquitectura N-Capas

Para la realización del proyecto fue necesario la implementación de plugin, los cuales no son más que elementos externos que ayudan a potenciar el desarrollo de un proceso, es decir son funcionalidades que ProcessMaker no permite implementar y que son necesarias para el desarrollo de los procesos. Estas no interfieren en el funcionamiento del sistema porque se pueden añadir o quitar y este va a continuar funcionando sin ningún problema. Estos, utilizan un patrón arquitectónico Modelo Vista controlador, el cual se convierte en una arquitectura N-Capas, al añadirsele nuevas capas, entre las que se encuentran la capa de acceso a datos y la de configuración, como se puede observar en la Ilustración 4, en la estructura de carpetas del mismo.

La arquitectura N-Capas organiza el sistema en capas, cada una de las cuales proporciona un conjunto de servicios. Esta soporta el desarrollo incremental de sistemas, es decir, a medida, q se desarrolla una capa, algunos de los servicios de esta capa están disponibles para los usuarios. Además soporta bien los cambios y es portable. Por otro lado, si una capa cambia sus interfaces o se le añaden nuevas funcionalidades, solamente se ve afectada la capa adyacente. Siendo otra de sus ventajas el hecho de que facilita la implementación multiplataforma de sus aplicaciones. (Sommerville, 2005)

2.5 Diseño de la Base de Datos

“Teniendo como entrada el Modelo conceptual, la Especificación de la arquitectura de sistema y la Especificación de los requisitos de software se diseñan las tablas, sus atributos y relaciones agrupados por componentes ya sea por paquetes o colores delimitando cada uno de ellos, obteniendo como resultado el Modelo de datos”. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012)

Este último es el que define cómo se modela la estructura lógica de una base de datos. Estos son entidades fundamentales para introducir la abstracción en una base de datos. Además, define cómo los datos se conectan entre sí y cómo se procesan y almacenan dentro del sistema. (Tecnologías de Información, s.f)

2.6 Análisis de factibilidad

Para el análisis del costo y beneficios del sistema se utilizó la metodología de Prodesoft donde desde el inicio se estima de forma empírica la duración de la implementación de cada uno de los requisitos, basado en la experiencia del programador en el trabajo con el lenguaje de programación, el entorno de desarrollo, el conocimiento sobre el tema de investigación y las técnicas de programación necesarias para resolver el problema. Para esto es necesario conocer el tiempo de desarrollo de cada requisito y la cantidad de trabajadores que participan para estimar si resulta beneficioso su desarrollo.

2.6.1 Costo

Para la estimación del costo del software se empleó la fórmula propuesta por la metodología, para lo cual primeramente se tiene en cuenta los requisitos funcionales a desarrollar, así como su prioridad, complejidad y tiempo estimado de desarrollo, lo que se detalla en la siguiente tabla, en la cual se tiene en cuenta que el tiempo total estimado para la realización del proyecto es de 6 meses, que serían 24 semanas:

No	Nombre del requisito	Prioridad	Complejidad	Tiempo de desarrollo (semanas)
1	Realizar solicitud por el solicitante.	Alta	Media	2
2	Realizar solicitud por las técnicas	Alta	Media	2
3	Validar la solicitud.	Alta	Baja	1
4	Mostrar los resultados de la búsqueda.	Alta	Baja	1

5	Elaborar un documento legal por la secretaria.	Alta	Media	2
6	Generar documento.	Alta	Media	2
7	Planificar fecha de recogida	Alta	Media	2
8	Controlar recogida	Alta	Alta	3
9	Replanificar fecha de recogida.	Alta	Media	2
10	Gestionar carrera.	Media	Baja	1
11	Gestionar tipo de curso.	Media	Baja	1
12	Gestionar tipo de solicitante.	Media	Baja	1
13	Gestionar estado de la solicitud.	Media	Baja	1
14	Realizar Reportes	Media	Alta	3

Luego se determina que:

Costo diario de un trabajador= Tarifa horaria (MT) * Trabajador * Tiempo diario.

= \$18.00 * 1 trabajador * 8 horas.

= \$144.00.

Tomando en cuenta que un trabajador de la entidad trabaja 20 días al mes

= \$144.00. *20

= \$2 880.00.

Costo mensual de un trabajador = \$2 880.00.

Teniendo en cuenta un tiempo estimado de aproximadamente 6 meses para el desarrollo del proyecto se determina:

Costo total= \$2 880.00* 6 meses.

= \$ 17,280.00

El costo de desarrollo del sistema fue \$17 280.00 aproximadamente, lo que en materia económica constituye una cifra moderada de dinero con respecto a los beneficios que se evidencian a continuación.

2.6.2 Beneficios tangibles e intangibles

El sistema permite una optimización de los tiempos necesarios para la realización de la solicitud y trámite de los cotejos de títulos, certificaciones de notas y plan de estudios en la Universidad de Matanzas, aumenta en la calidad de los servicios y reduce los costos. Además, mejora la eficiencia en el trabajo y administración del departamento y por tanto de la universidad. Cada trabajador responde por la efectividad del proceso con el objetivo de propiciar el aumento de los niveles de satisfacción de la población, permitiendo así una fácil extracción y manipulación de los datos necesarios para su desarrollo. La planificación de los días de recogida se encuentra automatizada, logrando que todos los trámites se encuentren concluidos el día indicado y la seguridad y protección de los datos se corresponde con el nivel requerido por el cliente y cumple sus expectativas.

2.6.3 Análisis de costo y beneficios

Anteriormente se calculó el costo del sistema y se analizaron los beneficios que representa el mismo, lo que permite concluir que el sistema es positivo para la gestión de trámites en la secretaría puesto que permite a la misma su flexibilización y agilización aumentando así su acercamiento a la sociedad y la satisfacción de esta última. Todo lo que evidencia que la solución propuesta sustenta su desarrollo e implementación y es factible la ejecución del mismo.

2.7 Patrones

Los patrones de diseño, según Craig Larman (1999, pág. 190) son una pareja de problema/solución con un nombre y que es aplicable a otros contextos, con una

sugerencia sobre la manera de usarlo en situaciones nuevas. Los patrones no se proponen descubrir ni expresar nuevos principios de la ingeniería del software. Todo lo contrario: intentan codificar el conocimiento, las expresiones y los principios ya existentes: cuanto más trillados y generalizados, mucho mejor.

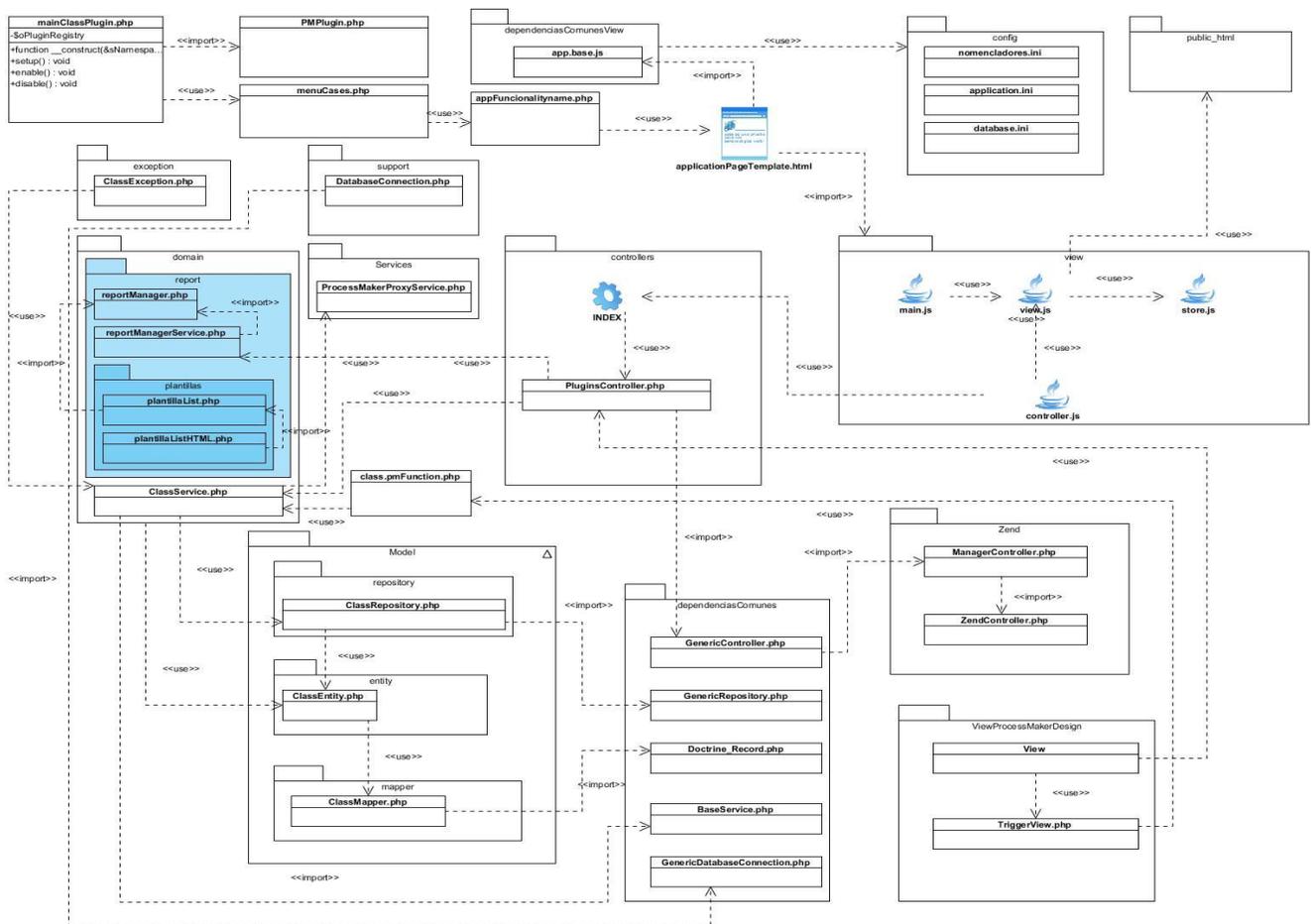
Para la implementación de la presente investigación se utilizaron como patrones, los siguientes:

- **Singleton:** Este patrón se encuentra dentro de los patrones de tipo creación, que son aquellos que abstraen el proceso de creación de instancias de los clientes que lo utilizan. Gracias al patrón Singleton, se puede hacer un sistema independiente de la forma en la que se crean las instancias, además, se hace independiente de la forma en la que se componen y representan las mismas. Esto es debido a que oculta, a los clientes, el proceso de creación y asociación de las instancias. El patrón Singleton es aplicable en cuando, únicamente debe de existir una instancia de una clase y debe ser accesible desde un punto conocido. La instancia única debe ser extensible por los clientes mediante herencia sin necesidad de modificar su código (Time of Software, 2016). Un ejemplo de este patrón se puede observar en la Ilustración 5.
- **Modelo Vista Controlador:** Es un patrón de diseño arquitectónico de software, que sirve para clasificar la información, la lógica del sistema y la interfaz que se le presenta al usuario. En este tipo de arquitectura existe un sistema central o controlador que gestiona las entradas y la salida del sistema, uno o varios modelos que se encargan de buscar los datos e información necesaria y una interfaz que muestra los resultados al usuario final. Es muy usado en el desarrollo web porque al tener que interactuar varios lenguajes para crear un sitio es muy fácil generar confusión entre cada componente si estos no son separados de la forma adecuada. Este patrón permite modificar cada uno de sus componentes si necesidad de afectar a los demás. (García, 2017). Un ejemplo de este, se puede observar en la Ilustración 6.

- **Constructor:** Como Patrón de diseño, el Constructor, es clasificado como creacional y es usado para permitir la creación de una variedad de objetos complejos desde un objeto fuente (Producto). El objeto fuente se compone de una variedad de partes que contribuyen individualmente a la creación de cada objeto complejo a través de un conjunto de llamadas a interfaces comunes de la clase ConstructorAbstracto. Tiene como ventajas que el algoritmo para la creación de objetos complejos sea independiente de las partes que construyen el objeto y cómo son ensambladas y que el proceso de construcción pueda tener diferentes representaciones para el objeto que está construido. (Andalucía, s.f) Un ejemplo de la estructura del mismo se puede observar en la Ilustración 7.

2.8 Mecanismo de diseño

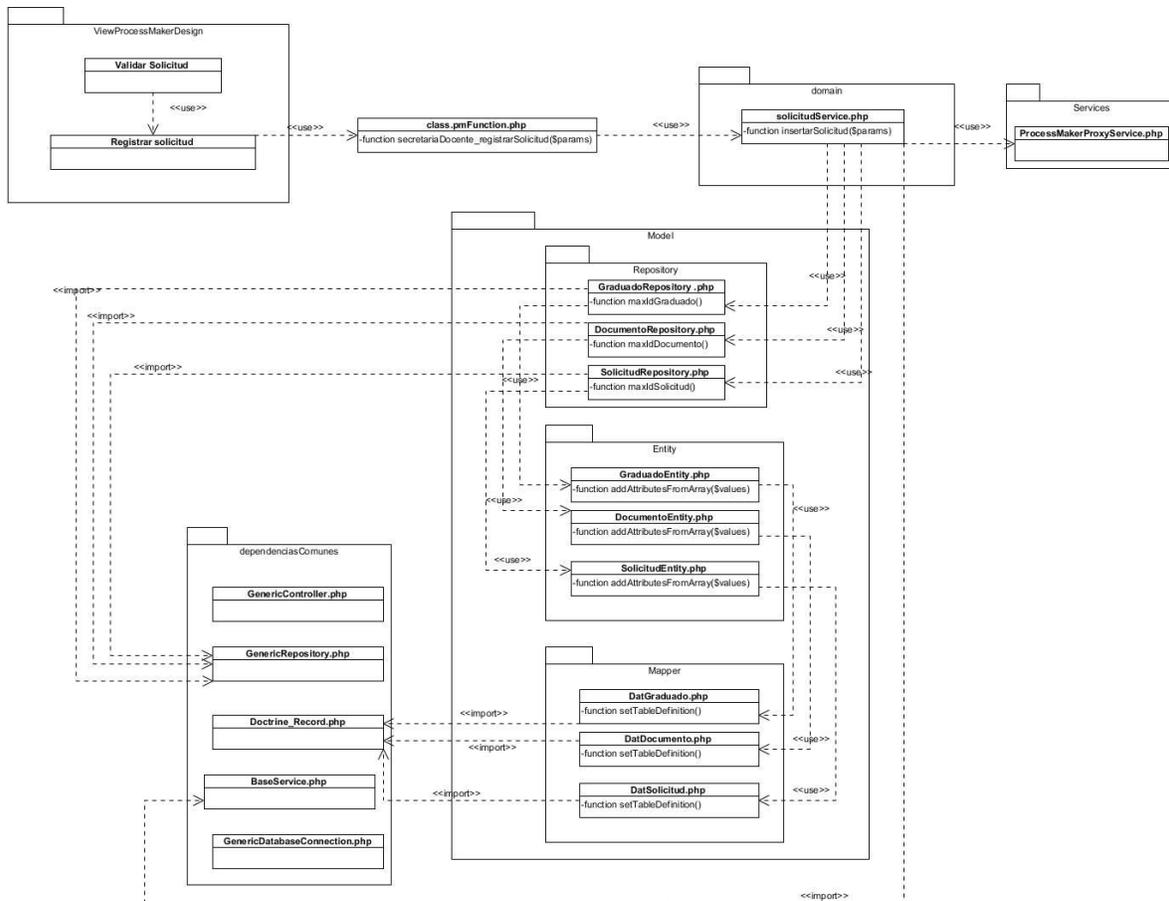
El mecanismo de diseño resume cómo construir el diseño de las clases que se pueden implementar en el software, se visualizan las relaciones entre ellas y se muestra gráficamente la interacción de los objetos para comunicarse entre sí. Su propósito es representar los diagramas de paquetes, así como los diagramas de clases y la descripción de las mismas. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012, pág. 73)



2.9 Diagramas de clases de diseño

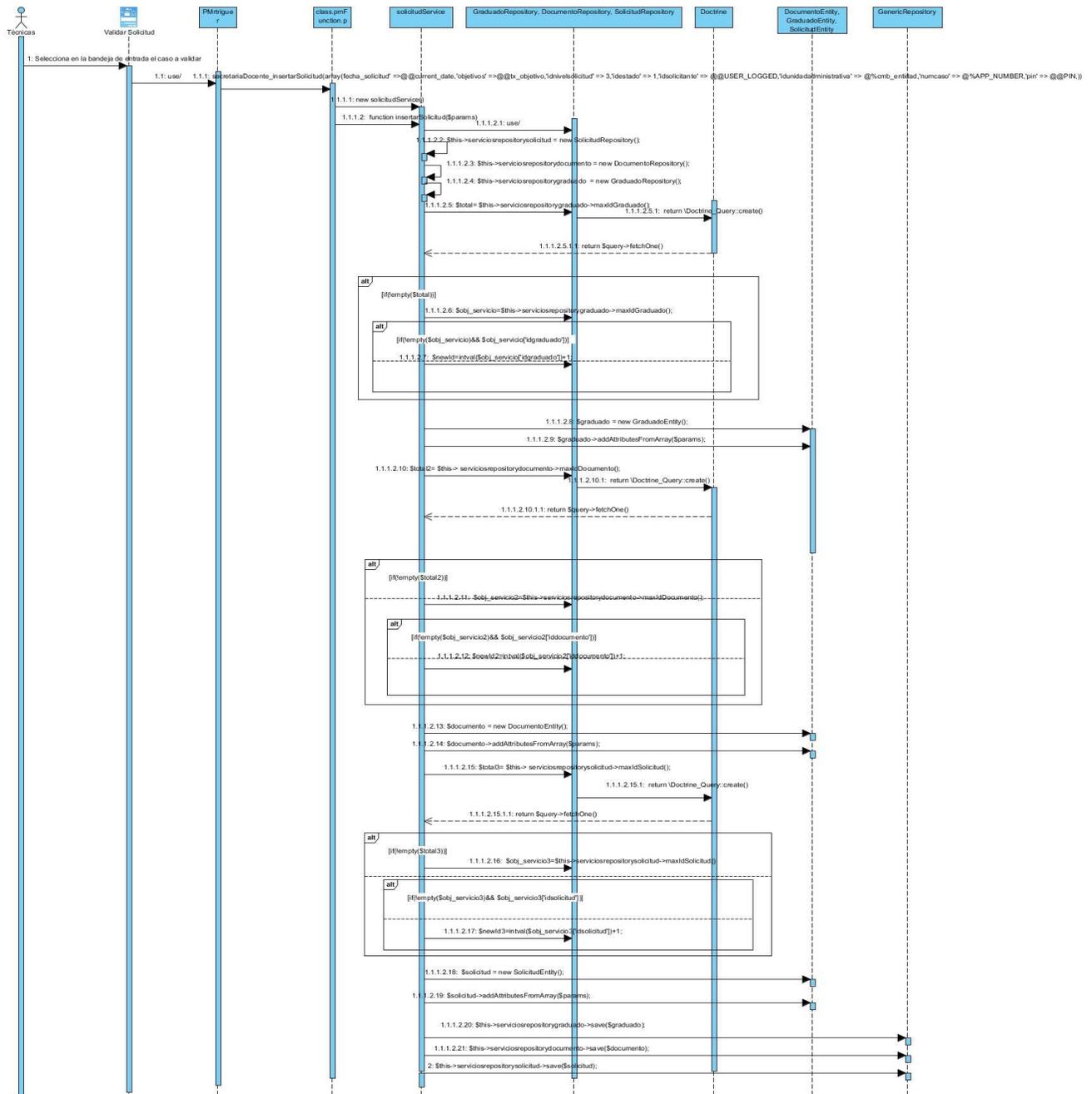
Un Diagrama de Clases de Diseño muestra la especificación para las clases software de una aplicación. Incluye las clases, asociaciones y atributos; las interfaces, con sus operaciones y constantes; los métodos y las dependencias.

A diferencia del Modelo Conceptual, un Diagrama de Clases de Diseño muestra definiciones de entidades software más que conceptos del mundo real. En la siguiente figura se muestra un ejemplo de Diagrama de Clases de Diseño del requisito funcional Realizar Solicitud, el resto de los diagramas se realizó de forma similar. (Universidad Nacional Abierta y a Distancia, s.f)



2.10 Diagramas de secuencia

Los diagramas de secuencia son aquellos diagramas que permiten mostrar la interacción entre los usuarios, la interfaz y las instancias de los objetos en el sistema. Ellos proveen un mapa secuencial del paso de los mensajes entre los objetos a lo largo del tiempo. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012) A continuación se muestran el diagrama de secuencia correspondiente a el requisito funcional Validar Solicitud, para el resto de los requisitos funcionales se realizaron los diagramas de secuencia de una forma similar.



2.11 Seguridad

El sistema deberá implementar diversas políticas de seguridad para garantizar la confidencialidad, integridad y disponibilidad de la información que maneje. Para ello se utilizará una seguridad RBAC o Control de acceso basado en roles, el cual permite administrar quién tiene acceso a cada recurso y que pueden hacer con esos recursos, es decir, separa las tareas y otorga solo la cantidad de accesos que los

usuarios necesitan para realizar sus trabajos (Microsoft Azure, 2019). Además, tendrá en cuenta el desarrollo de una interfaz fácil de usar, sencilla y amigable; así como una ayuda general, que guíe al usuario durante su trabajo en el sistema. Por otra parte, el framework utilizado para el acceso a los datos, en este caso Doctrine, al igual que la mayoría de los ORM, presenta su propia capa de seguridad para la protección contra los ataques más comunes, como son las inyecciones SQL.

2.12 Conclusiones parciales

Después de haber realizado todos los diagramas especificados por Prodesoft y realizado el software bajo esas especificaciones, se arribó a las siguientes conclusiones:

- La modelación de todos los procesos que intervinieron durante el desarrollo del sistema proporcionó una visión más completa del producto deseado.
- La realización del modelado del proceso permitió un mayor entendimiento del negocio, facilitando la posterior programación del mismo.
- Los diagramas de clases permitieron conocer los requisitos con un nivel de profundidad más amplio, aclarando todos los detalles necesarios para una buena implementación de los mismos.
- Los diagramas de secuencia permitieron el proceso de implementación del sistema de una forma más clara.

Capítulo III: Validación de la Solución Propuesta.

3.1 Introducción

Una parte importante del desarrollo de todo sistema, es probar si cumple con todos los requisitos tanto funcionales como no funcionales que se definieron para él, así como la eficiencia en el código, la seguridad y la satisfacción del cliente con el mismo. Una forma eficiente de realizarlo es mediante la aplicación de diferentes pruebas, siendo específicamente esto lo que se expone en el presente capítulo.

3.2 Pruebas realizadas y resultados

Las pruebas del software son la actividad más común de control de la calidad realizada en los proyectos para asegurar el correcto funcionamiento del software. Tienen como objetivos la verificación de la correcta implementación de los requisitos explícitamente establecidos, la adecuada integración de los componentes que conforman el sistema y la ejecución de casos de prueba que permitan detectar el mayor número de No conformidades y corregirlas antes de la entrega del software al cliente. Estas reducen la probabilidad de que aparezcan defectos ocultos en el software, pero incluso si no se encuentra ningún defecto, nunca será una garantía de su corrección. (Proceso de Desarrollo y Gestión de Proyectos de Software, 2012)

Sus objetivos principales son:

- Probar si el software no hace lo que debería hacer.
- Probar si el software hace lo que no debería hacer.

3.2.1 Pruebas de aceptación

Las pruebas de aceptación son las últimas pruebas realizadas donde el cliente prueba el software y verifica que cumpla con sus expectativas. Estas pruebas generalmente son funcionales y se basan en los requisitos definidos por el cliente y deben hacerse antes de la salida a producción. Esta termina de definir el nivel de calidad de la aplicación y le permite conocer al equipo qué tan bien supo interpretar los requerimientos del usuario. (Wordpress.com, 2017)

Estas pruebas se realizaron para cada uno de los requisitos funcionales del software y a continuación se presentan algunos de las realizadas para los requisitos

funcionales Realizar Solicitud, Validar Solicitud y Controlar recogida, el resto se realizó de una forma similar.

Estas son elaboradas por el propio equipo de desarrollo y sus resultados son guardados en tablas que cuentan con los campos que se exponen a continuación:

Código caso de prueba: Identificador de la prueba realizada. En el caso de las presentes se usa RF- seguido del número del requisito funcional al que pertenece, seguida de -P y un número consecutivo.

RF: Nombre del requisito funcional al que hace referencia la prueba.

Nombre del caso de Prueba: Nombre que tendrá el caso de prueba a realizar.

Responsable: Nombre de la persona que realiza la prueba.

Descripción: Contiene una breve descripción de la prueba realizada.

Condiciones de Ejecución: Las condiciones necesarias para que se pueda realizar la prueba.

Entradas / Pasos de Ejecución: Serie de pasos enumerados para lograr realizar la prueba.

Resultado esperado: Breve descripción del resultado que se espera obtener con la prueba realizada.

Evaluación de la prueba: acorde al resultado de la prueba realizada se emitirá una evaluación sobre la misma. Esta evaluación tendrá uno de los tres resultados que a continuación se describen:

- I. Satisfactoria: cuando el resultado de la prueba es exactamente el esperado por el usuario.
- II. Parcialmente satisfactoria: cuando el resultado no es completamente el esperado por el cliente o usuario de la aplicación y muestra resultados erróneos o fuera de contexto.

III. Insatisfactoria: cuando el resultado de la prueba realizada genera un error de codificación en la aplicación o muestra como resultado elementos no deseados o fuera de contexto, trayendo como consecuencia que la funcionalidad requerida por el cliente no tenga resultado, lo que invalida también el RF.

Prueba de Aceptación	
Código caso de prueba: RF-2-P1	RF: Realizar Solicitud por las técnicas
Nombre del caso de Prueba: Test cambiar documentos a entregar	
Responsable: Keythy Sotolongo	
Descripción: Se accederá al combo donde se selecciona el tipo de solicitante y los campos de adjuntar documentos debe variar	
Condiciones de Ejecución: El usuario técnicas debe estar logueado para esta operación	
Entrada/Pasos de Ejecución:	
<ul style="list-style-type: none"> - Seleccionar el tipo de solicitante 	
Resultados Esperados: Que se muestre solamente los documentos que debe adjuntar cada tipo de solicitante	
Evaluación: Satisfactoria.	

Prueba de Aceptación	
Código caso de prueba: RF-2-P2	RF: Realizar solicitud por las técnicas
Nombre del caso de Prueba: Test cambiar año de graduación	
Responsable: Keythy Sotolongo	
Descripción: Se accederá al campo donde se selecciona el año de graduación de la persona sobre la cual se está haciendo la solicitud	
Condiciones de Ejecución: El usuario técnicas debe estar logueado para esta operación	
Entrada/Pasos de Ejecución:	

- Seleccionar el año de graduación
Resultados Esperados: Que se permita seleccionar solo años menores al año actual si el mes actual es anterior a agosto o permita seleccionar hasta el año actual si el mes es agosto o posterior
Evaluación: Parcialmente Satisfactoria.

Prueba de Aceptación	
Código caso de prueba: RF-3-P1	RF: Validar solicitud
Nombre del caso de Prueba: Adicionar solicitud	
Responsable: Keythy Sotolongo	
Descripción: Se accederá a los combos donde se seleccionan el tipo de solicitud y el documento solicitado	
Condiciones de Ejecución: El usuario técnicas debe estar logueado para esta operación	
Entrada/Pasos de Ejecución:	
<ul style="list-style-type: none"> - Seleccionar el tipo de solicitud y el documento solicitado - Seleccionar la opción guardar y enviar 	
Resultados Esperados: Que la solicitud quede guardada en la base de datos y pase al siguiente paso del proceso	
Evaluación: Satisfactoria.	

Prueba de Aceptación	
Código caso de prueba: RF-8-P1	RF: Controlar recogida

Nombre del caso de Prueba: Realizar Recogida	
Responsable: Keythy Sotolongo	
Descripción: Se buscará el caso a recoger teniendo en cuenta el pin y el código y se marcará como recogido	
Condiciones de Ejecución: El usuario técnicas debe estar logueado para esta operación	
Entrada/Pasos de Ejecución:	
<ul style="list-style-type: none"> - Buscar el caso deseado teniendo en cuenta el pin y el código - Seleccionar la opción recogido - Dar click en el botón aceptar - Dar click en el botón cancelar 	
Resultados Esperados:	
<ul style="list-style-type: none"> - Que se muestre en el listado solo el caso buscado. - Que se muestre un cartel de confirmación donde se pregunte si está seguro de marcar el caso como recogido. - Que se cierre el cartel de confirmación. - Que se marque como recogido el caso y se muestre la lista de los casos pendientes donde se verifique que este ya no está pendiente. 	
Evaluación: Satisfactoria.	

Prueba de Aceptación	
Código caso de prueba: RF-8-P2	RF: Controlar recogida
Nombre del caso de Prueba: Replanificar recogida	

Responsable: Keythy Sotolongo
Descripción: Se buscará el caso a replanificar teniendo en cuenta el pin y el código y se cambiará el día de recogida
Condiciones de Ejecución: El usuario técnicas debe estar logueado para esta operación
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> - Buscar el caso deseado teniendo en cuenta el pin y el código - Seleccionar la opción replanificar - Dar click en el botón aceptar - Dar click en el botón cancelar
Resultados Esperados: <ul style="list-style-type: none"> - Que se muestre en el listado solo el caso buscado. - Que se muestre la nueva ventana para cambiar la fecha. - Que se cierre la ventana. - Que se replanifique el caso y se muestre la lista de los casos pendientes donde se verifique que este ya no está en ella.
Evaluación: Satisfactoria.

3.2.3 Pruebas de caja negra

Las pruebas de caja negra son una técnica en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna del código, detalles de implementación o escenarios de ejecución internos en el software. En ellas, el enfoque está dado solamente en las entradas y salidas del sistema, sin preocuparse en la estructura interna del programa. Para ellos se basa en los requerimientos de software y especificaciones funcionales. (Morrillo, 2017) Estas se enfocan en buscar:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Se realizó un caso de prueba para cada requisito implementado. A continuación se muestra el caso de prueba realizado al requisito Realizar Solicitud, que describe cada uno de los escenarios que pueden existir ante las posibles acciones realizadas por el usuario. De manera similar se realizaron los casos de prueba a los requisitos restantes.

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Realizar Solicitud	Se adiciona una solicitud, con los datos de la solicitud, del graduado y del solicitante	EP 1.1: Adicionar una nueva solicitud introduciendo datos válidos.	<ul style="list-style-type: none"> - Se insertan todos los datos. - Se presiona el botón Guardar y Enviar.

		<p>EP 1.2: Adicionar una nueva solicitud dejando campos requeridos en blanco.</p>	<ul style="list-style-type: none"> - Se introducen los datos dejando campos requeridos en blanco. - Se presiona el botón Guardar y Enviar. - Se muestra el mensaje de error en los campos obligatorios
		<p>EP 1.3: Adicionar una nueva solicitud introduciendo datos inválidos.</p>	<ul style="list-style-type: none"> - Se intenta introducir datos inválidos. - Se presiona el botón Guardar y Enviar. - Se muestra el mensaje de campos erróneos en cada uno de ellos

3.2.4 Resultados de las pruebas

Después de haber realizado el proceso de pruebas de aceptación, donde estuvieron presentes tanto el encargado como el cliente se obtuvieron resultados satisfactorios, pues se detectaron algunos errores, que fueron solucionados, pues su ocurrencia impedía el total cumplimiento de los requisitos funcionales definidos por el cliente.

Además, después de haberse realizado las pruebas de caja negra a cada uno de los requisitos se obtuvo como resultados un total de 11 no conformidades en la primera iteración, las cuales fueron resueltas, logrando que en la segunda iteración no existiera ninguna. Todo esto llevo a que el software obtenido tenga todas sus funcionalidades de acuerdo a las especificaciones del cliente y que además cumpla con los requerimientos de rendimiento.

3.2 Conclusiones parciales

Después de haberse realizados las pruebas utilizando las técnicas anteriormente especificadas, se llegaron a las siguientes conclusiones:

- Se demostró el buen funcionamiento del software y el cumplimiento de los requerimientos del cliente.
- El cliente confirmó que la aplicación agiliza y automatiza en gran medida el proceso de Solicitud y trámite de los cotejos de títulos, certificaciones de notas y planes de estudio en la universidad de Matanzas, siendo de esta manera mucho más fácil para las técnicas y la secretaria realizar sus funciones.
- Se confirmó que se requieren varias iteraciones de pruebas de caja negra para garantizar que el producto final cumpla con las funcionalidades exigidas por el cliente, aunque en este caso solo fueron necesario 2 iteraciones.

Conclusiones Generales

Después de la culminación de la presente investigación se pudo realizar un exhaustivo estudio en uno de los procesos fundamentales que se realizan en la Secretaría General de la Universidad de Matanzas, el cual es, la solicitud y trámite de los cotejos de títulos, certificaciones de notas y planes de estudio. Lográndose así dar respuesta a muchas de las insuficiencias que existían en el mismo. Además de cumplirse el objetivo general planteado para ella.

A lo largo de toda la investigación se llevaron a cabo un conjunto de tareas, arrojando los siguientes resultados:

- Se describió el estado del arte y se confeccionó un marco teórico referencial después de haberse realizado un estudio acerca de las herramientas y tecnologías a utilizar, determinando así sus eficiencias y deficiencias y mostrando la carencia de un software capaz de realizar este proceso de forma eficiente.
- Se realizó el modelado del negocio teniendo en cuenta los pasos que describe Prodesoft para su realización, entre ellos las reglas del negocio, la especificación de los requisitos funcionales, el diseño de la base de datos y los diagramas de secuencia, logrando una mayor comprensión del negocio.
- Se implementó el sistema cumpliendo con las políticas y los estándares de desarrollo definidos por Prodesoft y se realizaron pruebas de validación que permitieron obtener una versión estable del sistema.

El sistema ha sido avalado por especialistas de la Secretaría General, mostrando un alto grado de aceptación, teniendo en cuenta que cumple con todos los requerimientos exigidos por ellos y se adapta a las características del gobierno electrónico. La aplicación de este trabajo tiene gran importancia en el país ya que constituye un paso más al desarrollo informático y una herramienta de gran ayuda para la realización de este proceso.

Recomendaciones

Considerando que se han cumplido correctamente los objetivos trazados para el presente trabajo de diploma, se recomienda:

- Continuar con el estudio del proceso obtener mayores conocimientos acerca del proceso e identificar soluciones comunes para las entidades.
- Continuar realizando pruebas de calidad al componente.
- Implantar la aplicación en la Secretaría para así validar de forma experimental la verdadera utilidad de la misma.
- Agregar funcionalidades al sistema que permitan una mayor usabilidad del mismo, como la realización de varias solicitudes por una misma persona.
- Agregar nuevos reportes que sean de interés para los usuarios que interactúen con la aplicación

Referencias

Acosta, J. V. (12 de octubre de 2018). *Agencia Cubana de Noticias*. Obtenido de Agencia Cubana de Noticias.

Agesic. (14 de junio de 2010). *Agesic*. Obtenido de Gobierno electrónico: https://www.agesic.gub.uy/innovaportal/v/163/1/agesic/gobierno_electronico_.html

Andalucía, J. d. (s.f). *Marco de desarrollo de la junta de Andalucía*. Obtenido de Marco de desarrollo de la junta de Andalucía: <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/185>

Asan, J. V. (2012). *Resolución No. 184*. La Habana.

autores, C. d. (2001). *Manual de PHP*. Free Software Foundation.

Bello, I. O. (21 de octubre de 2018). *Periodico 26*. Recuperado el 24 de octubre de 2018, de Presentado oficialmente el Portal del ciudadano de Las Tunas: <http://www.periodico26.cu/index.php/es/especiales/reportaje-multimedia/item/12083-en-construcción-portal-del-ciudadano-de-las-tunas>

Computer Hoy. (24 de agosto de 2014). *Computer Hoy*. Obtenido de Computer Hoy: <https://www.google.com/amp/s/computerhoy.com/noticias/software/que-es-html5-todo-que-necesitas-saber-16425%3famp>

'Cutter' Blades, S. R. (s.f). *Learning Ext JS*.

Diario de la República. (23 de julio de 2018). *Uruguay es líder en América Latina con el gobierno digital más avanzado*. Obtenido de Uruguay es líder en América Latina con el gobierno digital más avanzado: <https://www.republica.com.uy/uruguay-es-lider-en-america-latina-con-el-gobierno-digital-mas-avanzado-id667325/>

ecommerce platforms. (2018). *Qué es JavaScript?* Obtenido de Qué es JavaScript?: <https://ecommerce-platforms.com/es/glossary/javascript>

Espestudio. (16 de agosto de 2005). *Espestudio*. Obtenido de Espestudio: <https://www.espestudio.com/noticias/que-es-mysql>

(s.f.). *Estructura y titulaciones de educación superior*. La Habana.

García, M. (05 de octubre de 2017). *Coding or not*. Obtenido de Coding or not: <https://codingornot.com/mvc-modelo-vista-controlador-que-es-y-para-que-sirve>

Hipertextual. (28 de mayo de 2013). *Hipertextual*. Obtenido de Hipertextual: <https://hipertextual.com/archivo/2013/05/entendiendo-html5-guia-para-principiantes/>

- IBM. (17 de julio de 2017). *IBM*. Obtenido de IBM:
https://www.ibm.com/support/knowledgecenter/es/SSEPGG_8.2.0/com.ibm.db2.ii.doc/opt/c0007799.htm
- Julián Pérez, A. G. (2009). *Definición de*. Obtenido de Definición de: <https://definicion.de/emision/>
- Junta de Andalucía. (s.f). *Marco de Desarrollo de la Junta de andalucía*. Obtenido de Marco de Desarrollo de la Junta de andalucía:
<http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/260>
- Lapiente, M. J. (29 de julio de 2018). *Hipertexto*. Obtenido de Hipertexto:
<http://www.hipertexto.info/documentos/xml.htm>
- Larman, C. (1999). *UML y Patrones, introducción al análisis y diseño orientado a objetos*. México: Prentice Hall.
- Manz. (s.f.). *Lenguaje Css*. Obtenido de ¿Qué es CSS?:
<https://lenguajecss.com/p/css/introduccion/que-es-css>
- Marc Gibert Ginestá, O. P. (s.f). *Bases de datos en PstgreSQL*. Barcelona: Universidad Abierta de Cataluña.
- Marini, E. (2012). *El Modelo Cliente/Servidor*.
- Microsoft Azure. (12 de Marzo de 2019). *¿Qué es el control de acceso basado en rol (RBAC) para los recursos de Azure?* Obtenido de ¿Qué es el control de acceso basado en rol (RBAC) para los recursos de Azure?: <https://docs.microsoft.com/es-es/azure/role-based-access-control/overview>
- Microsoft. (s.f). *Microsoft*. Obtenido de Acces SQL: conceptos básicos, vocabulario y sintaxis: <https://support.office.com/es-es/article/acces-sql-conceptos-básicos-vocabulario-y-sintaxis-444d0303cde1-424e-9a74-e8dc3e460671>
- Morrillo, R. A. (20 de febrero de 2017). *Pruebas de caja negra*. Obtenido de Pruebas de caja negra: <http://www.pmoinformatica.com/2017/02/pruebas-de-caja-negra-ejemplos.html?=1>
- Negocios y Soluciones Informáticas. (s.f). *Negocios y Soluciones Informáticas*. Obtenido de Negocios y Soluciones Informáticas:
<http://www.nsisoluciones.com/index.php/features/ibpms2/process-make>
- Organización de Estados Americanos. (2010). *Organización de Estados Americanos*. Recuperado el 1 de octubre de 2018, de Programa de e-gobierno:
<http://portal.oas.org/Portal/Sector/SAP/DptodeModernizacióndelEstadoyGobernabilidad/NPA/SobreProgramadeeGobierno/tabid/811/Default.aspx>

- Oscars, Ó. S. (s.f). *Introducción a BPMN*. Obtenido de Introducción a BPMN: <https://bpmn-bayard.blogspot.com/2011/03/12-que-es-el-bpd.html>
- Paszniuk, R. (19 de julio de 2013). *Arquitectura Cliente-Servidor*. Obtenido de Arquitectura Cliente-Servidor: <https://www.programacion.com.py/varios/arquitectura-cliente-servidor>
- Picazo, M. (17 de noviembre de 2013). *Herramientas Automatizadas*. Obtenido de Visual Paradigm UML: http://picazomariana.blogspot.com/2013/11/blog-post_16.html?m=1
- (2012). *Proceso de Desarrollo y Gestión de Proyectos de Software*. La Habana.
- Raúl, A. (2000). *Diseño Orientado a Objetos con UML*. Madrid: EIDOS Consultoría y Documentación.
- Redacción Central. (29 de mayo de 2018). *Radio Reloj*. Recuperado el 2018 de octubre de 15, de Radio Reloj: <http://www.radioreloj.cu/es/noticias-radio-reloj/sociedad/desarrollan-camaguey-proyecto-gobierno-electronico/>
- Rivas, R. S. (16 de febrero de 2018). *Granma*. Recuperado el 15 de octubre de 2018, de Granma: <http://www.granma.cu/cuba/2018-02-16/pinar-continua-dando-pasos-hacia-el-gobierno-electronico-16-02-2018-00-02-17>
- Rodríguez, S. A. (4 de octubre de 2018). *Cómo contribuye la UIC a la informatización de la sociedad? Granma(237)*, pág. 3.
- Rosario Zapata, A. (6 de diciembre de 2018). *Gobierno electrónico, puertas hacia el futuro. Girón, 1398*, pág. 5.
- Sánchez, Y. (14 de septiembre de 2017). *Victoria, diario digital de la Isla de la Juventud*. Recuperado el 24 de octubre de 2018, de Victoria, diario digital de la Isla de la Juventud: <http://www.periodicovictoria.cu/construyen-portal-del-ciudadano-en-la-isla-de-la-juventud/>
- Savia. (22 de agosto de 2016). *El sexto país mejor valorado en Gobierno Electrónico: España*. Obtenido de El sexto país mejor valorado en Gobierno Electrónico: España.
- Sommerville, I. (2005). *Ingeniería de Software, Séptima edición*. Madrid: Pearson Educación.
- Soto, D. (16 de septiembre de 2016). *¿Qué es BPMN y para qué sirve?* Obtenido de ¿Qué es BPMN y para qué sirve?: <https://nextech.pe/que-es-bpmn-y-para-que-sirve/>
- Tasé, E. R. (27 de agosto de 2018). *Periódico La Demajagua, Diario digital de la provincia de Granma*. Recuperado el 15 de octubre de 2018, de Periódico La Demajagua.

Diario digital de la provincia de Granma: <http://lademaiagua.cu/trabaja-desoft-granma-consolidar-gobierno-electronico/>

Tecnologías de Información. (s.f). *Modelos de datos: Modelo Conceptual, Físico y Lógico*.
Obtenido de Modelos de datos: Modelo Conceptual, Físico y Lógico:
<https://www.tecnoloias-informacion.com/modelos-datos.html>

tecnoretas. (s.f). *tecnoretas*. Obtenido de
<http://www.tecnoretas.com/programacion/que-es-doctrine-orm/>.

Time of Software. (14 de diciembre de 2016). *Time of Software*. Obtenido de Time of Software: <http://timeofsoftware.com/patron-singleton/>

UCI. (4 de septiembre de 2014). *Uruguay, Chile y Argentina encabezan a Latinoamérica en Gobierno electrónico*. Obtenido de Uruguay, Chile y Argentina encabezan a Latinoamérica en Gobierno electrónico:
<http://www.cubadebate.cu/noticias/2014/09/04/uruguay-chile-y-argentina-encabezan-a-latinoamerica-en-gobierno-electronico/>

Universidad de Granma. (24 de noviembre de 2017). *Universidad de Granma*.
Recuperado el 10 de octubre de 2018, de Universidad de Granma:
<http://www.udg.co.cu/index.php/institucion/secretaria-general>

Universidad de Oriente. (13 de septiembre de 2016). *Universidad de Oriente*. Recuperado el 10 de octubre de 2018, de Universidad de Oriente:
<https://www.uo.edu.cu/secretaria-general>

Universidad Nacional Abierta y a Distancia. (s.f). *Diagrama de Clases de Diseño*.
Obtenido de Diagrama de Clases de Diseño:
http://stadium.unad.edu.co/ovas/10596_9836/diagrama_de_clases_de_diseo.html

Valdés, D. P. (3 de julio de 2007). *Maestros del web*. Obtenido de Maestros del web:
<http://www.maestrosdelweb.com/que-es-javascript/>

Villareal Fuentes, C. A. (2013). *¿Qué es jQuery?* México.

Wordpress.com. (23 de agosto de 2017). *Los Andes Training*. Obtenido de Los Andes Training: <https://losandestraining.com/2017/08/23/que-son-las-pruebas-de-aceptacion/>

Zend Framework. (2009). *¿Qué es Zend Framework?* . Obtenido de
<http://spanish.zendfw.com/que-es-zend-framework/>

Anexos

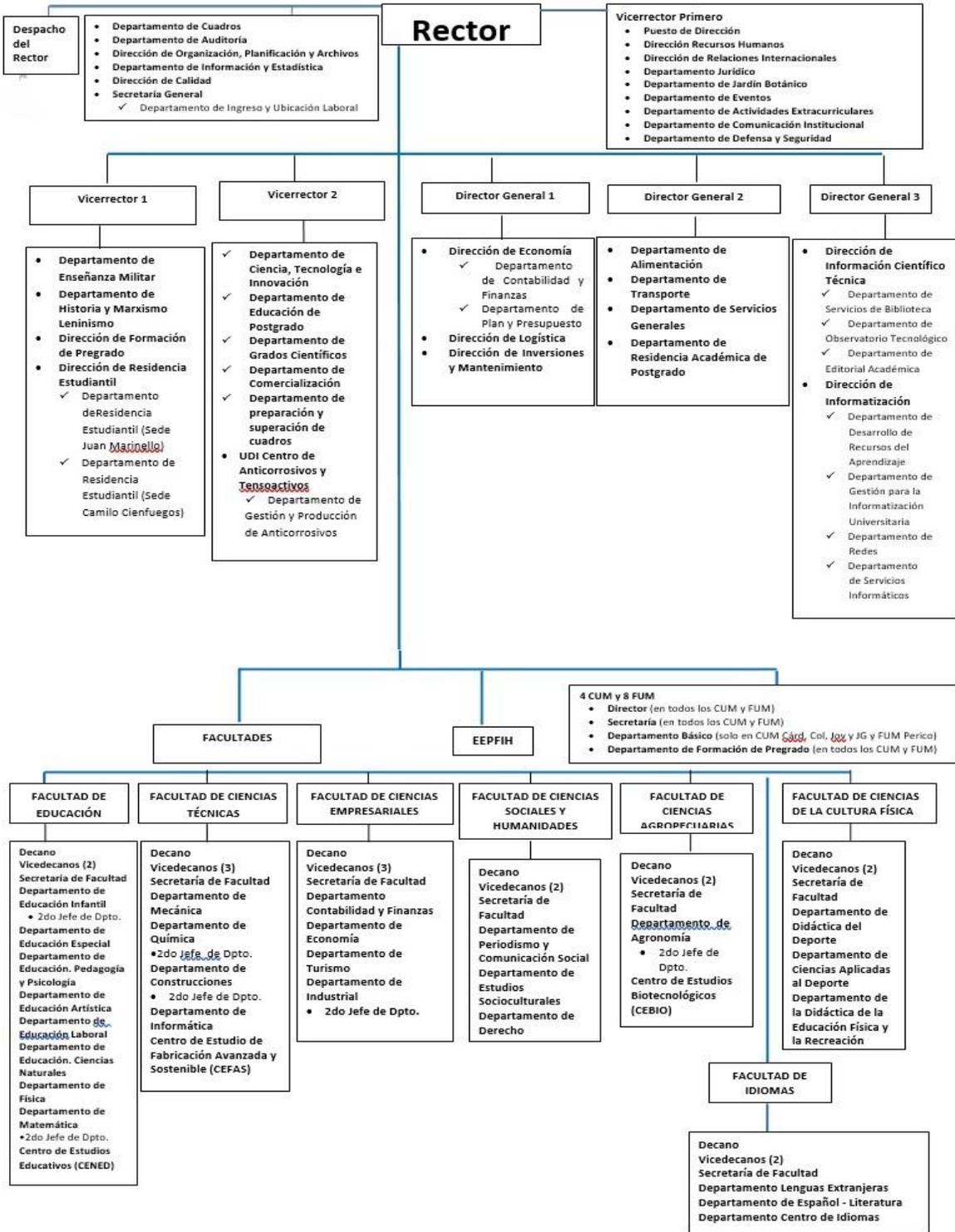


Ilustración 1

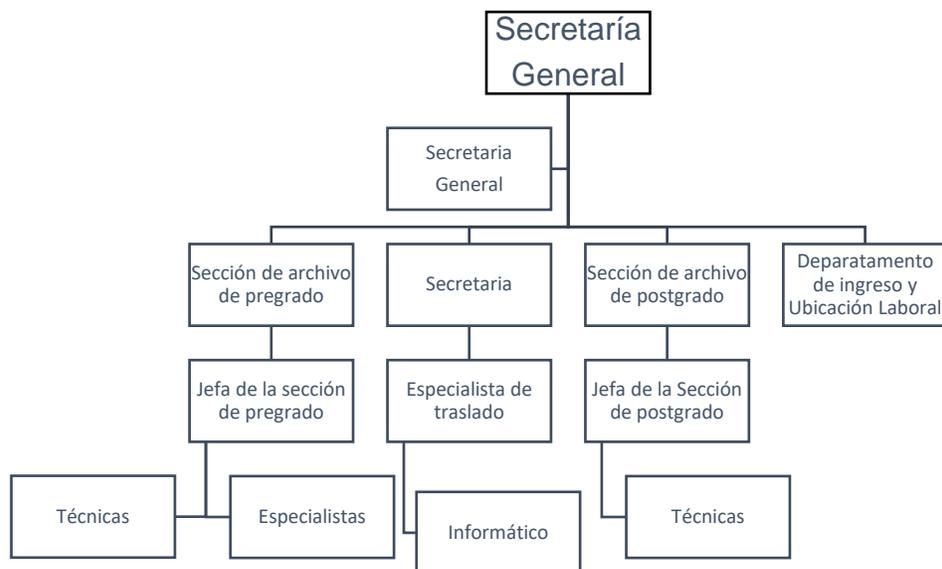


Ilustración 2

América Latina en la Encuesta de Gobierno Electrónico 2018

Publicado por [Policy & Law](#)



América Latina en la Encuesta de Gobierno Electrónico 2018 Valor de cada país en el Índice de Desarrollo de E-Gobierno (EGDI) 2014, 2016 y 2018

Muy alto EGDI – más de 0.75
Alto EGDI – entre 0.50 a 0.75
Medio EGDI – entre 0.25 a 0.50
Bajo EGDI – entre 0.0 a 0.25

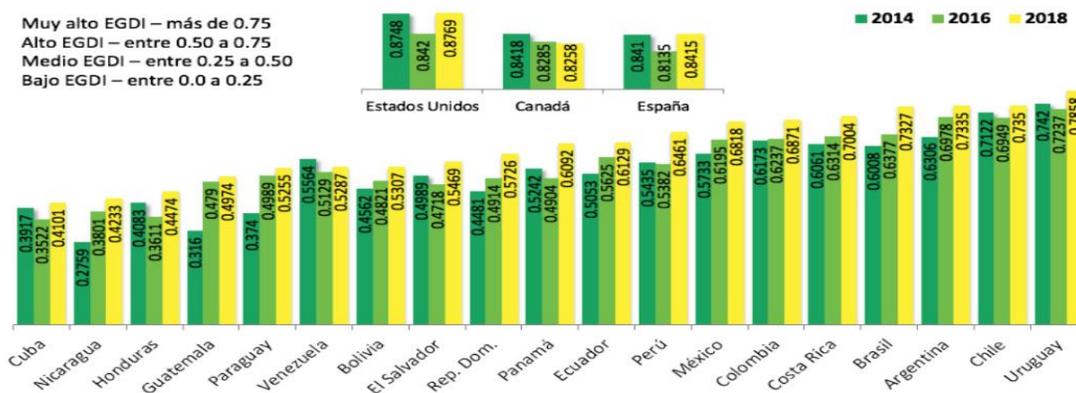


Ilustración 3

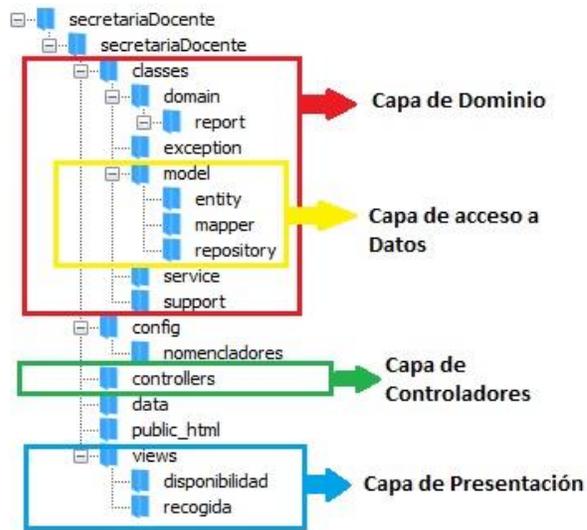


Ilustración 4

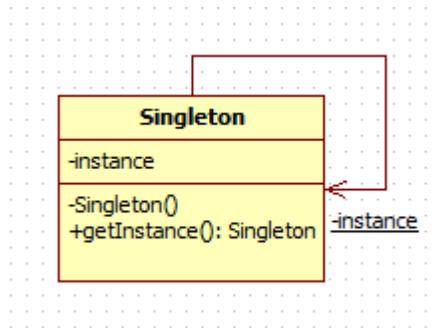


Ilustración 5

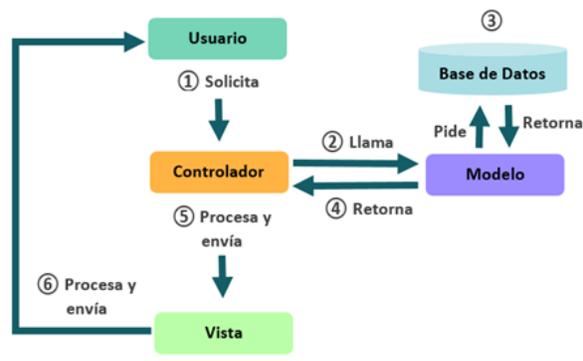


Ilustración 6

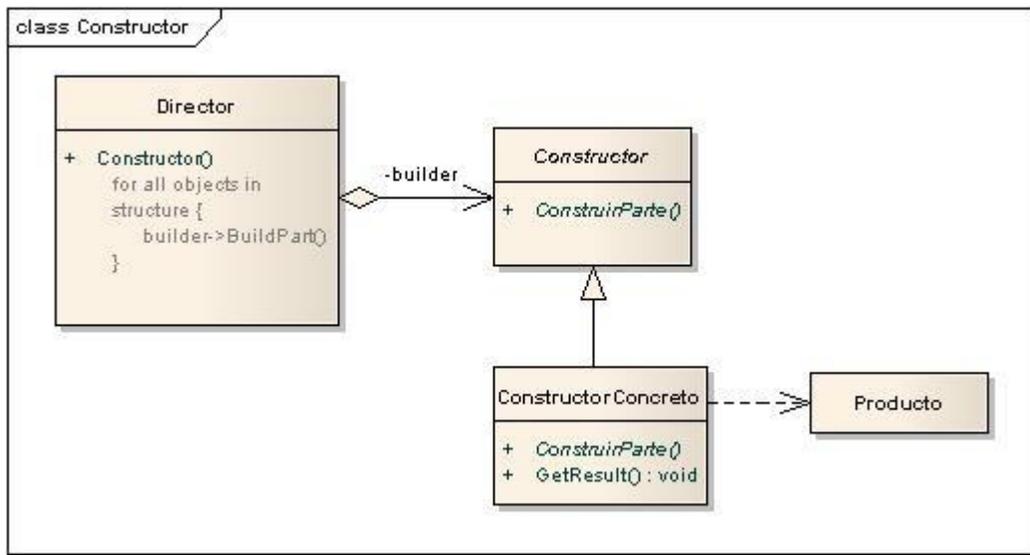


Ilustración 7