

Universidad de Sancti Spíritus “José Martí Pérez”

Facultad de Ciencias Técnicas y Empresariales

Carrera de Ingeniería Informática



Trabajo de Diploma para optar por el título de Ingeniería Informática

Título: Aplicación web para la gestión de la información
departamento de posgrado de la Universidad de Sancti Spíritus
"José Martí Pérez" (Uniss).

Autor: José Alberto Pérez Castro.

Tutor(s): Ing. Geisy Yessica Griñán Guevara.

Sancti Spíritus, Cuba

2019

Pensamiento

“La tecnología hizo posible las grandes poblaciones; ahora las grandes poblaciones hacen que la tecnología sea indispensable”

José Krutch

Dedicatoria

A mis familiares y amigos.

Agradecimientos

Quisiera agradecer en primer lugar a todos los profesores que en mi transcurso por la universidad me han infundido los conocimientos y elementos necesarios para convertirme en un buen profesional. A mis compañeros de aula por los buenos momentos que hemos pasado y brindarme ayuda en los momentos que la e necesitado, en especial a José Couzo y Jorge Martínez. A mi tutora Geisy Yessica Griñán Guevara por su ayuda y guía en el proceso de creación de este trabajo de diploma. Pero el mayor de los agradecimientos es para mi familia en general, especialmente mi madre Idelsi, mi padre José Rafael y a mi esposa Lianet Quiñones, a todos ellos por su paciencia, buenos consejos y apoyo incondicional.

Resumen

Este trabajo se desarrolla en el departamento de posgrado de la Universidad de Sancti Spíritus “José Martí Pérez” (Uniss). El objetivo de la investigación es desarrollar una aplicación web que facilite la gestión de la información de los diferentes tipos de cursos de posgrado, así como la relación de estudiantes y profesores que intervienen en estos. Para ello se realizó una investigación dentro y fuera del departamento con el objetivo de conocer cómo se llevaba a cabo el proceso del negocio. Para su realización se siguieron las etapas planteadas por la metodología de desarrollo ágil *Extreme Programming* (XP) como herramienta CASE Visual Paradigm 8.0. Para implementar el sistema se utilizó el lenguaje de programación PHP 5, Symfony como *framework* web, MySQL para la persistencia de los datos y como herramienta de programación el PhpStorm. El principal aporte de este trabajo lo constituye una aplicación web para la gestión de la información en el departamento de posgrado de la Universidad de Sancti Spíritus “José Martí Pérez”.

Abstract

This work is developed in the postgraduate department of the University of Sancti Spíritus “José Martí Pérez”. The main goal is to design and implement a web application that facilitates the management of information on the different types of postgraduate courses and the relationship of students and professors involved in them. For this, an investigation was carried out inside and outside the department with the objective of knowing how the business process was carried out. For its realization, the stages proposed by the agile development methodology *Extreme Programming* (XP) as CASE Visual Paradigm 8.0 tool were followed. To implement the system, we used the PHP 5 programming language, Symfony as a web framework, MySQL for data persistence and PhpStorm as a programming tool. The main contribution of this research is a web application for the management of information in the postgraduate department of the University of Sancti Spíritus “José Martí Pérez”.

Índice general

Capítulo I: Fundamentación del empleo de herramientas informática para la gestión de la información en el departamento de posgrado de la Uniss.	6
1.1 Gestión de la información.....	6
1.2 Tecnologías informáticas para asistir la gestión de la información en el departamento de posgrado.	7
1.2.1 Aplicaciones web	8
1.2.2 Metodología de desarrollo de software	9
1.3 Lenguajes empleados para el desarrollo de software	10
1.4 Herramientas y Tecnologías utilizadas para el desarrollo de software	15
1.4.1 Framework web	16
1.4.2 Jet Brains PhpStorn	20
1.4.3 Sistema Gestor de Base de Datos.....	21
1.4.4 XAMPP	22
1.4.5 Visual Paradigm.....	22
Conclusiones Parciales	23
Capítulo II: Desarrollo de la aplicación web para la gestión de la información en el Departamento de Posgrado de la Uniss.....	25
2.1 Proceso de desarrollo del sistema utilizando la metodología XP.	25
2.2 Descripción de los procesos del negocio	25
2.2.1 Reglas del negocio	26
2.3 Lista de reserva del producto.	28
2.4 Proceso de desarrollo de la aplicación siguiendo la metodología XP.	31
2.4.1 Historias de usuario	31
2.4.2 Tareas de ingeniería	35
2.5 Diagrama Entidad-Relación de la Base de Datos	41
Figura 2 Diagrama Entidad-Relación de la base de datos	42
Conclusiones Parciales	43

Capítulo III: Construcción de la aplicación web propuesta para facilitar la gestión de la información en el Departamento de Posgrado de la UNISS.....	44
3.1 Tratamiento de excepciones y seguridad. Principio de interfaz de usuario y ayuda del sistema.	44
3.1.1 Tratamiento de excepciones.....	44
3.1.2 Seguridad	45
3.1.3 Interfaz de usuario.	46
3.4 Ayuda.....	47
3.2 Fase de pruebas según XP: Pruebas de aceptación	48
Anexo 1 Historias de Usuario	58
Anexo 2 Tareas de Ingeniería	72
Anexo 3 Pruebas de Aceptación	74

Índice de tablas

Tabla 1 Historias de Usuario	32
Tabla 2 Historia de usuario: Gestionar estudiantes.....	33
Tabla 3 Historia de Usuario: Gestionar profesores.....	33
Tabla 4 Historia de usuario: Gestionar ediciones de maestrías.	34
Tabla 5 Historia de usuario: Mostrar Base Nominalizada de Maestrías.	35
Tabla 6 Tareas de ingeniería.....	38
Tabla 7 Tarea de ingeniería: Diseño de la interfaz gestionar estudiantes.....	38
Tabla 8 Tarea de ingeniería: Implementación de operaciones CRUD para gestionar estudiantes.	39
Tabla 9 Tarea de ingeniería: Diseño de la interfaz gestionar profesores.	39
Tabla 10 Tarea de ingeniería: Implementación de operaciones CRUD para gestionar profesores.....	39
Tabla 11 Tarea de ingeniería: Diseño de la interfaz gestionar ediciones de maestrías.....	40
Tabla 12 Implementación de operaciones CRUD para gestionar ediciones de maestrías.....	40
Tabla 13 Diseño de la interfaz mostrar base nominalizada de maestrías.	40
Tabla 14 Prueba de aceptación: Gestionar estudiante.....	50
Tabla 15 Prueba de aceptación: Gestionar profesores	50
Tabla 16 Prueba de aceptación: Gestionar ediciones de maestrías.	51
Tabla 17 Prueba de aceptación: Mostrar Base Nominalizada de Maestrías	52

Índice de figuras

Figura 1 Proceso Posgrado.....	27
Figura 2 Diagrama Entidad Relación de la base de datos.	Error! Bookmark not defined.

Introducción

Las tecnologías de la información y comunicación (TIC) han provocado un gran impacto en el ámbito social, económico, político y cultural, es decir, el fenómeno de la Globalización, con el que no existen barreras espaciales para acceder a la información; y tienen como características la inmaterialidad, la instantaneidad, la interactividad y la posibilidad de información multimedia. (Rodríguez, 2005)

Se puede hacer mención, que las TIC son herramientas que permiten innovar conocimientos, los cuales nos permiten ser personas más preparadas y abiertas a las cuestiones que se presenten, pues son muy importantes en el mundo tan cambiante en el que vivimos, ya que estas nos permiten generar nuevos conocimientos que nos ayuden a mejorar y facilitarnos la vida, ante esta necesidad de actualización es conveniente estar dispuestos a adoptar los nuevos cambios. Por todo ello, es necesario que le demos el uso adecuado a estas herramientas, ya que si lo hacemos de esa forma obtendremos mayores beneficios de las mismas.

Cuba está consciente de que una sociedad para ser más eficaz, eficiente y competitiva debe aplicar la informatización en todas sus esferas y procesos. En este sentido, Cuba ha identificado desde muy temprano la conveniencia y necesidad de dominar e introducir en la práctica social las TICs y lograr una cultura digital como una de las características imprescindibles del hombre nuevo, lo que facilitaría a nuestra sociedad acercarse más hacia el objetivo de un desarrollo justo, equitativo, sostenible y alcanzable. (Carbó, 2017)

En Cuba las TIC están influyendo en los procesos de creación y cambio de las corrientes públicas. Algo tan habitual como la televisión, el móvil, la radio y el ordenador, están constantemente transmitiendo mensajes, intentando llevar a su terreno a los usuarios de estos medios, a través de mensajes de texto, correos electrónicos, blogs, y otros espacios dentro de la web.

Hoy en día, es innegable que las nuevas tecnologías de la información y comunicación permiten mayor contribución a través de las redes sociales;

favorecen el intercambio social, cultural, profesional, etcétera; permiten realizar educación a distancia a través de las plataformas virtuales que en un futuro serán precursoras entre la sociedad al generar diferentes entornos de comunicación y, sobre todo, la posibilidad de tener acceso a entornos virtuales de aprendizaje a través de la educación distancia. (Sánchez, Añorve, & Alarcón, 2017)

El estudio, análisis y evaluación del impacto que tienen las denominadas tecnologías de la información y comunicación (TIC) sobre la enseñanza y sobre la innovación pedagógica en las escuelas es un ámbito problemático al que se le está prestando una atención relevante en la investigación educativa de estos últimos años. (Sánchez et al., 2017)

La Educación Superior no se ha quedado al margen en el uso de las tecnologías. Durante los últimos diez años, las TIC han aumentado su presencia en la educación superior, ya que enriquecen el proceso de enseñanza-aprendizaje.

Los futuros profesionales requieren del uso de las TIC en su proceso de formación. De lo contrario, carecerán de las herramientas que exige el competitivo mundo globalizado. En este contexto, es necesario que los docentes estén debidamente capacitados en el uso de herramientas tecnológicas y apliquen nuevas metodologías de enseñanza que promuevan en el estudiante un aprendizaje activo.

La adopción de las TIC en los programas formativos de las universidades, así como en las escuelas de posgrado permite a los estudiantes acceder a nuevas fuentes de información y canales de comunicación para compartir trabajos, debatir ideas y complementar su aprendizaje. También refuerzan su curiosidad y capacidad para resolver problemas y tomar decisiones, con una actitud más activa de exploración que los lleva a la reflexión y el replanteamiento de opiniones. (Valle, 2018)

La Universidad de Sancti Spíritus “José Martí Pérez” (Uniss) no se ha quedado atrás en la incorporación y uso de los avances tecnológicos para su labor habitual. En el departamento de posgrado de la UNISS, resulta de gran beneficio la utilización de las mismas, ya que se maneja un gran cúmulo de

información de los estudiantes, profesores y las diferentes formas de Posgrado, así como las ediciones de cada uno de estos. Todos estos datos proporcionan varios documentos como:

- El Modelo de Estadística de Educación de Posgrado 1164-03.
- Base Nominalizada de las diferentes ediciones de cada una de las formas de Posgrado
- Base Nominalizada Maestrías y Especialidades (unificados).

Actualmente este trabajo se efectúa de forma manual, contemplando la redacción y copia, en muchos casos, de documentos elaborados tanto por los profesores como jefes de departamento y de los cuales un gran porcentaje es confeccionado de forma incorrecta o con errores. Para la creación y generación de la documentación se utiliza de forma mayoritaria Word o el Excel.

De forma similar, durante el estudio del proceso se han podido constatar las siguientes insuficiencias:

- La información de las diferentes actividades de posgrado no siempre se brinda con los datos requeridos.
- Demoras en la entrega de información por parte de los implicados.
- Dificultades en el llenado de los documentos por parte de los profesores.
- La generación de informes y estadísticas se realiza de forma manual, propiciando errores y demoras en la entrega de la misma.
- Aun cuando este proceso se realiza utilizando herramientas informáticas, las mismas no garantizan el flujo de trabajo requerido.

Todos estos antepuestos, dan lugar al siguiente **Problema de investigación**: ¿Cómo contribuir a la gestión de la información en el departamento de postgrado de la Uniss?

Para dar salida al problema planteado anteriormente se delimita como **Objetivo general** de la investigación: Desarrollar una aplicación web que contribuya a la gestión de la información en el departamento de postgrado de la Uniss.

Para dar cumplimiento al objetivo general se establecieron las siguientes **preguntas de investigación**:

1. ¿Cuáles son los fundamentos teóricos-metodológicos que sustentan la elaboración de una herramienta informática que facilite la gestión de la información en el departamento de postgrado de la Uniss?
2. ¿Cómo diseñar una aplicación web que facilite la gestión de la información en el departamento de postgrado de la Uniss?
3. ¿Cómo implementar una aplicación web que facilite la gestión de la información en el departamento de postgrado de la Uniss?

Para lograr el objetivo y dar respuesta a las preguntas de investigación se plantean las siguientes **tareas de investigación**:

1. Determinar los fundamentos teóricos-metodológicos que sustentan la elaboración de un software que facilite la gestión de la información en el departamento de postgrado de la Uniss.
2. Diseñar una aplicación web que facilite la gestión de la información en el departamento de postgrado de la Uniss.
3. Implementar una aplicación web que facilite la gestión de la información en el departamento de postgrado de la Uniss.

Distribución de la investigación:

El presente trabajo está estructurado en: introducción, tres capítulos, conclusiones generales, recomendaciones, bibliografía y anexos.

Capítulo I: Fundamentación del empleo de herramientas informáticas para la gestión de la información en el departamento de posgrado de la Uniss.

En este capítulo se exponen los fundamentos teóricos y metodológicos de la investigación. Además, se describen las características de las herramientas, tecnologías y lenguajes de programación a tomar en cuenta para la realización de este trabajo.

Capítulo II: Desarrollo de la aplicación web para la gestión de la información en el departamento de posgrado de la Uniss.

En este capítulo se desarrolla la aplicación teniendo en cuenta la metodología de desarrollo de software XP.

Capítulo III: Construcción de la aplicación web propuesta para facilitar la gestión de la información en el departamento de posgrado de la Uniss.

En este capítulo se describe la forma en que se realizará la implementación del sistema a través de la descripción del tratamiento de excepciones, seguridad, interfaz visual y ayuda de la aplicación web propuesta. Se definen las pruebas de aceptación necesarias para corroborar el correcto funcionamiento del sitio.

Capítulo I

Capítulo I: Fundamentación del empleo de herramientas informática para la gestión de la información en el departamento de posgrado de la Uniss.

El presente capítulo comprende la base de la fundamentación teórica y metodológica asociada al tema a efectuar. Se determinan los principales conceptos afines al dominio del problema. Ofrece un análisis de las normas y tecnologías actuales adecuadas para el desarrollo de la aplicación, además se figura la metodología a utilizar para el diseño, las herramientas y lenguajes de programación en las que se fundamenta la aplicación propuesta.

1.1 Gestión de la información

La Gestión de Procesos asegura que las actividades se piensen, diseñen y ejecuten en el marco de un proceso. Cuando los empleados reconocen que sus actividades individuales son parte de algo mayor, se encolumnan hacia metas comunes. Cuando un proceso tiene un diseño explícito del principio al fin, la gente puede realizarlo de manera coherente y los gerentes están en condiciones de mejorarlo en forma disciplinada se asegura que todos los procesos de una empresa estén bien diseñados, que los diseños se respeten y se mantengan actualizados. (Ángel, 2010)

En la actualidad ha surgido la necesidad de ir mejorando las técnicas de gestión que de alguna manera alinean los objetivos planteados por cada una de las áreas con los objetivos estratégicos del largo plazo de las organizaciones, ya que deben ser coherentes, cuantificables, medibles y auto evaluables frecuentemente, buscando que estos lineamientos no pierdan validez ni importancia y que las operaciones que se generan al interior de las organizaciones integren y reflejen situaciones que fortalezcan e impacten positivamente a la empresa.

Según (Ledo & Ana Bárbara Araña Pérez, 2017) la gestión de la información no es más que el proceso de organizar, evaluar, presentar, comparar los datos en un determinado contexto, controlando su calidad, de manera que esta sea

veraz, oportuna, significativa, exacta y útil y que esta información esté disponible en el momento que se le necesite. Ella se encamina al manejo de la información, documentos, metodologías, informes, publicaciones, soportes y flujos en función de los objetivos estratégicos de una organización.

Con los adelantos tecnológicos actuales en el mundo de la informática y las comunicaciones, es prácticamente imposible que una organización no haga uso de los sistemas de gestión de la información para el desarrollo de sus actividades cotidianas, pues constituyen un elemento fundamental para alcanzar un alto nivel de competitividad y posibilidades de desarrollo.

Asegurar la adquisición, el almacenamiento, procesamiento y control del caudal de información implica el uso de técnicas y tecnologías avanzadas que permitan facilitar su gestión.

1.2 Tecnologías informáticas para asistir la gestión de la información en el departamento de posgrado.

En el departamento de posgrado de la Uniss se maneja un gran conjunto de información de los estudiantes, profesores y las diferentes formas de Posgrado, así como las ediciones de cada uno de estos. Todos estos datos crean varios documentos llevados en Word o el Excel, como:

- El Modelo de Estadística de Educación de Posgrado 1164-03.
- Base Nominalizada de cada una de las formas de Posgrado
- Base Nominalizada Maestrías y Especialidades (unificados).

Las Bases Nominalizadas son listas de los estudiantes, con sus datos, que participan en cada una de las ediciones de las diferentes formas de posgrado. Los documentos normalmente son elaborados por el personal docente y administrativo.

Ya hoy en día, han ido surgiendo transformaciones en el acceso y gestión de la información; por lo que su uso se ha vuelto cada día más necesarios para las organizaciones debido a la constante actualización de sus recursos. Dado al progresivo desarrollo de las TIC, la información cuenta cada vez más con mejores y factibles herramientas para su gestión y con una buena elección de estas herramientas y tecnologías se logrará un producto final con calidad y eficiencia. Para esto se decide desarrollar una aplicación web.

1.2.1 Aplicaciones web

Inicialmente la web era simplemente una colección de páginas estáticas, documentos, etc., que podían consultarse o descargarse. El siguiente paso en su evolución fue la inclusión de un método para elaborar páginas dinámicas que permitiesen que lo mostrado fuese dinámico (generado o calculado a partir de los datos de la petición).

Dicho método fue conocido como CGI (*Common Gateway Interface*) y definía un mecanismo mediante el cual podíamos pasar información entre el servidor HTTP y programas externos. Los CGI siguen siendo muy utilizados, puesto que la mayoría de los servidores web los soportan debido a su sencillez. Además, nos proporcionan total libertad a la hora de escoger el lenguaje de programación para desarrollarlos. (Mateu, 2004)

El desarrollo de aplicaciones web involucra decisiones no triviales de diseño e implementación que inevitablemente influyen en todo el proceso de desarrollo, afectando la división de tareas. Los problemas involucrados, como el diseño del modelo del dominio y la construcción de la interfaz de usuario, tienen requerimientos disjuntos que deben ser tratados por separado. El alcance de la aplicación y el tipo de usuarios a los que estará dirigida son consideraciones tan importantes como las tecnologías elegidas para realizar la implementación. Así como las tecnologías pueden limitar la funcionalidad de la aplicación, decisiones de diseño equivocadas también pueden reducir su capacidad de extensión y reusabilidad. Es por ello que el uso de una metodología de diseño y de tecnologías que se adapten naturalmente a ésta, son de vital importancia para el desarrollo de aplicaciones complejas. (Mercerat & Silva, 2002)

El éxito espectacular de la web se basa en dos puntales fundamentales: el protocolo HTTP y el lenguaje HTML. Uno permite una implementación simple y sencilla de un sistema de comunicaciones que nos permite enviar cualquier tipo de ficheros de una forma fácil, simplificando el funcionamiento del servidor y permitiendo que servidores poco potentes atiendan miles de peticiones y reduzcan los costes de despliegue. El otro nos proporciona un mecanismo de composición de páginas enlazadas simple y fácil, altamente eficiente y de uso muy simple. (Mateu, 2004)

Las aplicaciones web son populares debido a lo práctico del navegador web, a la independencia del Sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. (Aponte, 2014)

1.2.2 Metodología de desarrollo de software

Las metodologías de desarrollo de software son indispensables para crear o actualizar software de calidad que cumpla con los requisitos de los usuarios; son una parte fundamental de la Ingeniería de software la cual denomina metodología a un conjunto de métodos coherentes y relacionados por unos principios comunes. (Chavez, Martín, Rodríguez, Murazzo, & Software, 2012)

La primera herramienta que surgió de la crisis del software fue el ciclo de vida del *software*, casi todas las metodologías la emplean, cambiando simplemente el énfasis, momentos de entrada de cada fase del ciclo, solapamientos en las mismas, entre otras variaciones que han ido surgiendo a medida que se encuentran fallas o virtudes de cada metodología, permitiendo una evolución de las mismas hasta llegar a la actualidad a un balance entre la forma, las herramientas, mecanismos, modelo y forma de implementación de dichas metodologías. (Corrales, 2015)

La mayoría de los sistemas cambian o evolucionan con el tiempo, esto implica que se deba mantener, corregir, diseñar, implementar y probar de nuevo dicho sistema (repetir el ciclo de vida para las nuevas funcionalidades, las que cambiaron o los errores descubiertos). El efecto claro sobre el desarrollo es que los tiempos de entrega se alargan respecto a los acordados previamente con el cliente. Mientras más se demore la entrega del mismo, es más probable que este haya cambiado, teniéndose que especificar nuevamente.

Con las metodologías ágiles lo que se pretende hacer es realizar entregas lo más rápido posible de manera incremental, grupos pequeños multidisciplinarios de trabajo, donde sea posible el empoderamiento, motivación, identificación rápida de los cambios y errores y contacto cercano con los clientes. Estas metodologías generalmente superponen la especificación, el diseño, el

desarrollo y las pruebas o emplean herramientas alternativas para cada una de estas etapas. (Corrales, 2015)

Algunas de las metodologías ágiles más usadas en la actualidad se describen a continuación.

- Metodología XP programación extrema La programación extrema XP es posiblemente el método ágil más conocido y ampliamente utilizado. El nombre de XP fue acuñado por Beck, debido a que el enfoque fue desarrollado utilizando las mejores prácticas del desarrollo iterativo y con la participación extrema de los usuarios y/o clientes de la aplicación. Para la descripción de los requerimientos se emplea generalmente historias de usuario, las cuales se desglosan posteriormente en un conjunto de tareas. La implementación se realiza generalmente en parejas, una de ellas pudiendo ser un usuario o un programador novel y desarrollan pruebas para las tareas previo a la escritura del código (metodología TDD). Solo hasta que todas las pruebas sobre las tareas se cumplen satisfactoriamente el código nuevo puede ser integrado dentro del sistema. Por lo general se realizan reuniones de entrega del producto previa a la implementación de la siguiente historia de usuario. El desarrollo incremental se realiza mediante entregas pequeñas y frecuentes del sistema, por lo general planeando las entregas de las historias de usuario.

- Metodología SCRUM aplica las mismas premisas conceptuales que XP, pero para resolver un problema ligeramente distinto como es el de desarrollo evolutivo de aplicaciones. SCRUM es una metodología ágil y flexible que sirve para gestionar el desarrollo de software, cuyo objetivo es maximizar el retorno de la inversión para el cliente. Se basa principalmente en construir la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación.

1.3 Lenguajes empleados para el desarrollo de software

Las aplicaciones web, tal y como las conocemos hoy en día, han permitido un flujo de comunicación global a una escala sin precedentes en la historia humana. Todo contenido puede ser compartido digitalmente con el menor esfuerzo, haciéndolo llegar casi de forma inmediata a cualquier otro punto.

Gracias a su carácter virtual, la información en la web puede ser buscada más fácil, eficiente y mucho más rápido que con cualquier otro medio físico. Esta plataforma ha permitido a los usuarios interactuar con muchos más grupos de personas dispersas en diferentes lugares.

Por estas razones se decidió crear una aplicación web para la gestión de la información en el departamento de posgrado de la Uniss, ya que con esta se logra mantener una comunicación eficiente, fluida y directa entre todas las personas que intervienen en los procesos de posgrados con el departamento para el intercambio de datos entre estos.

Para crear dicha aplicación fueron utilizados diferentes lenguajes, los cuales se mencionan a continuación.

PHP

Es un lenguaje de programación de páginas web del lado del servidor cuyos rasgos principales son la independencia de plataforma y su gratuidad. Para poder programar en PHP se requiere de un servidor capacitado para ello. Como el lenguaje de programación es multiplataforma, cualquiera de los principales servidores web nos servirá para ello. (M. A. Alvarez, 2005)

Sus siglas significan (PHP: *Hypertext Pre Processor*), es un lenguaje sencillo, de sintaxis cómoda y similar a la de otros lenguajes como Perl, C y C++. Es rápido, interpretado, orientado a objetos y multiplataforma. Para él se encuentra disponible una multitud de librerías. Es un lenguaje ideal tanto para aprender a desarrollar aplicaciones web como para desarrollar aplicaciones web complejas. Añade a todo eso la ventaja de que el intérprete de PHP, los diversos módulos y gran cantidad de librerías desarrolladas para PHP son de código libre, con lo que el programador de PHP dispone de innumerables herramientas libres para desarrollar aplicaciones. (Mateu, 2004)

Según (Bakken et al., 2002) existen tres campos en los que *scripts* escritos en PHP son usados.

- Scripts en la parte del servidor. Este es el campo más tradicional y el principal campo de trabajo. Se necesitan tres cosas para que esto funcione. El parseador PHP (CGI o módulo), un servidor web y un

navegador. Se necesita correr el servidor web con PHP instalado. El resultado del programa PHP se puede obtener a través del navegador, conectando con el servidor web. Consultar la sección Instrucciones de instalación para más información.

- Scripts en línea de comandos. Pueden crear un script PHP y correrlo sin ningún servidor web o navegador. Solamente necesitan el parseador PHP para usarlo de esta manera. Este tipo de uso es ideal para scripts ejecutados regularmente desde cron (en *nixó Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden ser usados para tareas simples de procesado de texto. Consultar la sección Usos de PHP en la línea de comandos para más información.
- Escribir aplicaciones gráficas clientes. PHP no es probablemente el mejor lenguaje para escribir aplicaciones gráficas, pero si sabéis bien PHP, y os gustaría utilizar algunas características avanzadas en programas clientes, podéis utilizar PHP-GTK para escribir dichos programas. Es también posible escribir aplicaciones independientes de una plataforma. PHP-GTK es una extensión de PHP, no disponible en la distribución principal.

Una de las característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos. Escribir un interfaz vía web para una base de datos es una tarea simple con PHP.

HTML5

Uno de los puntales en el éxito del WWW ha sido el lenguaje HTML (*Hyper Text Mark-up Language*). Se trata de un lenguaje de marcas (se utiliza insertando marcas en el interior del texto) que nos permite representar de forma rica el contenido y también referenciar otros recursos (imágenes, etc.), enlaces a otros documentos (la característica más destacada del WWW) y mostrar formularios para posteriormente procesarlos. (Mateu, 2004)

HTML5 no es una nueva versión del antiguo lenguaje de etiquetas, ni siquiera una mejora de esta ya antigua tecnología, sino un nuevo concepto para la construcción de sitios web y aplicaciones.

Propone estándares para cada aspecto de la web y también un propósito claro para cada una de las tecnologías involucradas. A partir de ahora, HTML provee los elementos estructurales, CSS se encuentra concentrado en cómo volver esa estructura utilizable y atractiva a la vista, y JavaScript tiene todo el poder necesario para proveer dinamismo y construir aplicaciones web completamente funcionales.

Provee básicamente tres características: estructura, estilo y funcionalidad. Nunca fue declarado oficialmente, pero incluso cuando algunas APIs (Interface de Programación de Aplicaciones) y la especificación de CSS3 por completo no son parte del mismo, HTML5 es considerado el producto de la combinación de HTML, CSS y JavaScript. Estas tecnologías son altamente dependientes y actúan como una sola unidad organizada bajo la especificación de HTML5.

Más allá de esta integración, la estructura sigue siendo parte esencial de un documento. La misma provee los elementos necesarios para ubicar contenido estático o dinámico, y es también una plataforma básica para aplicaciones. Con la variedad de dispositivos para acceder a Internet y la diversidad de interfaces disponibles para interactuar con la web, un aspecto básico como la estructura se vuelve parte vital del documento. Ahora la estructura debe proveer forma, organización y flexibilidad. (Gauchat, 2012)

CSS3

Hojas de Estilo en Cascada (*Cascading Style Sheets*), es un lenguaje que trabaja junto con HTML para proveer estilos visuales a los elementos del documento, como tamaño, color, fondo, bordes, etc.

Oficialmente nada tiene que ver con HTML5. Este lenguaje es, de hecho, un complemento desarrollado para superar las limitaciones y reducir la complejidad de HTML. Al comienzo, los atributos dentro de las etiquetas HTML proveían estilos para cada elemento, pero a medida que el lenguaje se desarrolló, la escritura de códigos se volvió más compleja y HTML por sí mismo no pudo más satisfacer las demandas de diseñadores. En consecuencia, CSS fue adoptado como la forma de separar la estructura de la presentación. Desde entonces ha crecido y ganado importancia, pero siempre desarrollado en paralelo, enfocado en las necesidades de los diseñadores y apartado del proceso de evolución de HTML.

En un intento por reducir el uso de código JavaScript y para estandarizar funciones populares, CSS3 no solo cubre diseño y estilos web sino también forma y movimiento. La especificación de CSS3 es presentada en módulos que permiten a la tecnología proveer una especificación estándar por cada aspecto involucrado en la presentación visual del documento. Desde esquinas redondeadas y sombras hasta transformaciones y reposicionamiento de los elementos ya presentados en pantalla, cada posible efecto aplicado previamente utilizando JavaScript fue cubierto. Este nivel de cambio convierte CSS3 en una tecnología prácticamente inédita comparada con versiones anteriores. (Gauchat, 2012)

El modo de funcionamiento de las CSS consiste en definir, mediante una sintaxis especial, la forma de presentación que le aplicaremos a:

- Un web entero, de modo que se puede definir la forma de todo el web de una sola vez.
- Un documento HTML o página, se puede definir la forma, en un pequeño trozo de código en la cabecera, a toda la página.
- Una porción del documento, aplicando estilos visibles en un trozo de la página.
- Una etiqueta en concreto, llegando incluso a poder definir varios estilos diferentes para una sola etiqueta. Esto es muy importante ya que ofrece potencia en nuestra programación. Podemos definir, por ejemplo, varios tipos de párrafos: en rojo, en azul, con márgenes, sin ellos, etc.

CSS permite aplicar al documento formato de modo mucho más exacto. Si antes el HTML se nos quedaba corto para maquetar las páginas y teníamos que utilizar trucos para conseguir nuestros efectos, ahora tenemos muchas más herramientas que nos permiten definir esta forma:

- Podemos definir la distancia entre líneas del documento.
- Se puede aplicar indentado a las primeras líneas del párrafo.
- Podemos colocar elementos en la página con mayor precisión, y sin lugar a errores.
- Definir la visibilidad de los elementos, márgenes, subrayados, tachados.

(M. A. Alvarez et al., 2015)

JavaScript

JavaScript es un lenguaje de programación que surgió con el objetivo inicial de programar ciertos comportamientos sobre las páginas web, respondiendo a la interacción del usuario y la realización de automatismos sencillos. En ese contexto podríamos decir que nació como un "lenguaje de *scripting*" del lado del cliente, sin embargo, hoy JavaScript es mucho más. Las necesidades de las aplicaciones web modernas y el HTML5 han provocado que el uso de JavaScript que encontramos hoy haya llegado a unos niveles de complejidad y prestaciones tan grandes como otros lenguajes de primer nivel.

Pero, además, en los últimos años JavaScript se está convirtiendo también en el lenguaje "integrador". Lo encontramos en muchos ámbitos, ya no solo en Internet y la Web, también es nativo en sistemas operativos para ordenadores y dispositivos, del lado del servidor y del cliente. Aquella visión de JavaScript "utilizado para crear pequeños programitas encargados de realizar acciones dentro del ámbito de una página web" se ha quedado muy pequeña.

En el contexto de un sitio web, con JavaScript puedes hacer todo tipo de acciones e interacción. Antes se utilizaba para validar formularios, mostrar cajas de diálogo y poco más. Hoy es el motor de las aplicaciones más conocidas en el ámbito de Internet: Google, Facebook, Twitter, Outlook... absolutamente todas las aplicaciones que disfrutas en tu día a día en la Web tienen su núcleo realizado en toneladas de JavaScript. La Web 2.0 se basa en el uso de JavaScript para implementar aplicaciones enriquecidas que son capaces de realizar todo tipo de efectos, interfaces de usuario y comunicación asíncrona con el servidor por medio de Ajax. (Gauchat, 2012)

1.4 Herramientas y Tecnologías utilizadas para el desarrollo de software

El proceso de desarrollo de aplicaciones de software habitualmente comprende distintas etapas. En los programas de software se emplean varios lenguajes y tecnologías diferentes, con las herramientas típicamente establecidas para tecnologías específicas. El desarrollo de software puede ser una actividad compleja y larga, por lo que las herramientas adecuadas pueden reducir el estrés y aumentar el desempeño tanto de desarrolladores como de las aplicaciones resultantes.

1.4.1 Framework web

Actualmente podemos encontrar diversas herramientas informáticas para el desarrollo de aplicaciones web, sobre todo *frameworks* que integran algunas de estas creando así nuevas formas de desarrollo, todas encaminadas a mejorar y optimizar el proceso de creación.

El concepto *framework* se emplea en muchos ámbitos del desarrollo de sistemas *software*, no solo en el ámbito de aplicaciones Web. Podemos encontrar *Frameworks* para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrírse nos.

En general, con el término *framework*, nos estamos refiriendo a una estructura *software* compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Los objetivos principales que persigue son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Por tanto, podemos definirlo como un conjunto de componentes (por ejemplo, clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web. (Gutiérrez, 2015)

Agrega funcionalidad extendida a un lenguaje de programación, automatiza muchos de los patrones de programación para orientarlos a un determinado propósito, proporcionando una estructura al código, mejorándolo y haciéndolo más entendible y sostenible, y permite separar en capas la aplicación. En general, según (Martínez, Camacho, & Biancha, 2010) se divide la aplicación en tres capas:

- La lógica de presentación que administra las interacciones entre el usuario y el *software*.
- La Lógica de datos que permite el acceso a un agente de almacenamiento persistente u otros.

- La lógica de dominio o de negocio, que manipula los modelos de datos de acuerdo a los comandos recibidos desde la presentación.

Existen varios tipos de *frameworks* Web: orientados a la interfaz de usuario, como *Java Server Faces*, orientados a aplicaciones de publicación de documentos, como *Cocoon*, orientados a la parte de control de eventos, como *Struts* y algunos que incluyen varios elementos como *Tapestry*. La mayoría se encargan de ofrecer una capa de controladores de acuerdo con el patrón MVC o con el modelo 2 de *Servlets* y JSP, ofreciendo mecanismos para facilitar la integración con otras herramientas para la implementación de las capas de negocio y presentación. (Gutiérrez, 2015)

Poseen características que satisfacen en su gran mayoría a todos los programadores web según el estilo de desarrollo que deseen. Ahora bien, existen *frameworks* con todo tipo de características como la seguridad, robustez, facilidades de uso. Según (F. Sierra, Acosta, & Ariza, 2013) entre los *frameworks* web más usados están:

- **Laravel**

Es un *framework* de Código abierto para desarrollar aplicaciones web y servicios web con PHP 5. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011 y actualmente está en continuo desarrollo. Este *framework* usa el paradigma Orientado a objetos, permite el uso del patrón MVC, ORM. Gran parte de Laravel está formado por dependencias, especialmente de Symfony, esto implica que el desarrollo de Laravel dependa del desarrollo de sus dependencias. Laravel, propone en el desarrollo usar '*Routes with closures*', en lugar de un MVC tradicional con el objetivo de hacer el código más claro. Aun así, permite el uso de MVC tradicional.

- **CodeIgniter**

Es un *framework* de aplicaciones web de código abierto para ayudar a desarrollar programas en PHP. El objetivo de la aplicación es ayudar a los desarrolladores de proyectos de código, a desarrollar más rápido que escribir código desde cero. Esto se logra ofreciendo un amplio conjunto de bibliotecas

para tareas comúnmente necesarias, así como una interfaz sencilla y la estructura lógica de acceso a estas bibliotecas. Se basa en el patrón de desarrollo Modelo-Vista-Controlador. Contiene una serie de librerías que sirven para el desarrollo de aplicaciones web y además propone una manera de desarrollarlas que debemos seguir para obtener provecho de la aplicación. Esto es, marca una manera específica de codificar las páginas web y clasificar sus diferentes *scripts*, que sirve para que el código esté organizado y sea más fácil de crear y mantener.

- **Yii**

El proyecto Yii comenzó el 1 de enero de 2008, con el fin de solucionar algunos problemas con el PRADO *framework*. Por ejemplo, PRADO es lento manejando páginas complejas, tiene una curva de aprendizaje muy pronunciada y tiene varios controles que dificultan la personalización, mientras que Yii es mucho más fácil y eficiente. En octubre de 2008 después de 10 meses de trabajo en privado, la primera versión alfa de Yii fue lanzada. El 3 de diciembre del mismo año, Yii 1.0 fue formalmente presentado. Un poco más fácil de aprender que otros framework, cuenta con foro y IRC que proporciona ayuda al usuario. Software libre, patrón de diseño modelo vista controlador, integración con JQuery.

- **Zend Framework**

Implementa el patrón MVC, es 100% orientado a objetos y sus componentes tienen un bajo acoplamiento por lo que los puedes usar en forma independiente. Un punto importante es que nos brinda un estándar de codificación que deberíamos seguir en nuestros proyectos. A su vez, cuenta con soporte para internalización y localización de aplicaciones (construir sitios multidioma, convertir formatos de fechas, monedas, etc. según la región. Algo importantísimo para crear aplicaciones con un enfoque global y llegar de la mejor manera a la mayor cantidad de gente posible). Facilita el *setup* de nuestro proyecto brindándonos herramientas para crear la estructura de directorios, clases, etc. por línea de comandos, integración con php Unit por medio de Zend Test para facilitar el *testing* de nuestra aplicación. Posee adaptadores para gran cantidad de tipos de bases de datos diferentes; brinda

componentes para la autenticación y autorización de usuarios, envío de mails, cache en varios formatos, creación de *web services*, etc. Es simple, si tienes algo que hacer, seguramente ya lo tuvo que hacer alguien antes. Así que aprovecha eso para invertir menos tiempo en el desarrollo y hacer uso de componentes ya testeados.

Dentro de los *frameworks* más utilizados para el desarrollo web en los últimos años se encuentra Symfony.

- **Symfony**

Es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja.

Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Symfony está desarrollado completamente en PHP 5.3. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows. (F. Sierra et al., 2013)

Reutiliza conceptos y desarrollos exitosos de terceros y los integra como librerías para ser utilizados por nosotros. Entre ellos encontramos que integra plenamente uno de los frameworks ORM más importantes dentro de los existentes para PHP llamado Doctrine, el cual es el encargado de la comunicación con la base de datos, permitiendo un control casi total de los datos ya que la mayoría de las secuencias SQL no son generadas por el programador sino por el mismo Doctrine.

Otro ejemplo de esto es la inclusión del framework Twig, un poderoso motor de plantillas que nos da libertad de separar el código PHP del HTML permitiendo

una amplia gama de posibilidades y por sobre todo un extraordinario orden para nuestro proyecto.

Gracias al lenguaje YAML, competidor del XML, tenemos una gran cantidad de configuración totalmente separada del código permitiendo claridad como lo iremos viendo en los demás capítulos. Cabe mencionar que en caso de no querer trabajar con YAML también podemos usar estos archivos de configuración con XML o PHP.

Contamos con las instrucciones de consola denominadas *tasks* (tareas), que permiten ejecutar comandos en la terminal diciéndole a Symfony que nos genere lo necesario para lo que le estamos pidiendo, como por ejemplo podría ser la generación completa de los programas necesarios para crear ABMs, tarea que suele ser muy tediosa para los programadores ya que siempre implica mucho código para realizar la misma idea para diferentes tablas.

Otra de las funcionalidades más interesantes, es que contiene un subframework para trabajar con formularios. Con esto, creamos una clase orientada a objetos que representa al formulario HTML y una vez hecho esto simplemente lo mostramos y ejecutamos.

Contamos con un gran número de librerías, herramientas y helpers que nos ayudan a desarrollar una aplicación mucho más rápido que haciéndolo de la manera tradicional, ya que muchos de los problemas a los que nos enfrentamos ya fueron pensados y solucionados por otras personas por lo tanto ¡¡¡dediquémonos a los nuevos problemas que puedan surgir!!! (M. Alvarez & Ardissonne, 2011)

1.4.2 Jet Brains PhpStorm

La herramienta IDE que se utiliza para la realización de este trabajo fue PhpStorm, ya que es adecuada para trabajar con Symfony y otros framework. Este IDE, de forma general, se destaca por las refactorizaciones, completado de código sobre la marcha de análisis de código y la productividad de codificación. Ofrece un editor de código rico e inteligente para PHP con resaltado de sintaxis, configuración de formato de código extendido, sobre la marcha de la comprobación de errores, y la finalización de código inteligente. Finaliza clases, métodos, nombres de variables y palabras clave de PHP,

además de los nombres de uso común para los campos y las variables en función de su tipo. Proporciona sugerencias de finalización de código apropiadas. Permite usar el motor de plantillas Smart y ofrece funciones de productividad para trabajar con archivos tpl. (Rodríguez, 2018)

1.4.3 Sistema Gestor de Base de Datos

Un Sistema Gestor de Bases de Datos es una colección de programas que permite a los usuarios crear y mantener una base de datos; o sea, es un sistema software de propósito general que facilita los procesos de definición, construcción y manipulación de la base de datos para distintas aplicaciones.

Desde que se comenzó a usar el modelo de bases de datos relacionales, en 1970, ha ido sufriendo una serie de transformaciones hasta convertirse, hoy en día, en el modelo más utilizado para administrar bases de datos.

Este modelo se basa fundamentalmente en establecer relaciones o vínculos entre los datos, imaginando una tabla aparte por cada relación existente con sus propios registros y atributos. (Marín, 2010)

A continuación, se describe el SGBD seleccionado para el desarrollo de este trabajo.

MySQL

Según (Carrascal-reyes & Arciniegas-herrera, 2018) MySQL es un manejador de bases de datos principalmente utilizado en las aplicaciones WEB en conjunto con PHP. Con su última versión MYSQL 5.0 anexa nuevas características importantes que lo hacen más competitivo con manejadores como SQL Server, Sybase y Oracle. Entre sus nuevas características tenemos:

- Uso de transacciones ACID (*Atomic, Consistent Isolated, Durable*): Para construir aplicaciones más seguras mediante *commit, rollback, crash recovery* y bloqueo por registro.
- *Store Procedures*: Para mejorar la programación.
- *Triggers*: Para mejorar las reglas del negocio.
- *Vistas*: Para que la información sensible sea más segura.
- *Information Schema*: Para un fácil acceso a los metadatos.

- Transacciones Distribuidas (XA): Para soportar transacciones entre múltiples ambientes de bases de datos.

1.4.4 XAMPP

XAMPP es el entorno más popular de desarrollo con PHP, es una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl. El paquete de instalación de XAMPP ha sido diseñado para ser increíblemente fácil de instalar y usar. (Tupe & Cisneros, 2016)

Es un servidor de plataforma libre, es un software que integra un servidor web Apache, intérpretes de lenguaje de scripts PHP, un servidor de base de datos MySQL, un servidor de FTP FileZilla, MySQL, entre otros módulos.

Permite instalar de forma sencilla Apache en tu propio ordenador, sin importar tu sistema operativo (Linux, Windows, Mac o Solaris). Además, es gratuito.

Es una herramienta de desarrollo que permite probar el trabajo (páginas web o programación, por ejemplo) en el ordenador sin necesidad de tener que ir a Internet.

Xampp consta de paquetes básicos como son:

- Apache.
- MySQL.
- PHP y Perl.
- ProFTPD.
- OpenSSL (para soporte a la capa de sockets segura)

(Cepeda, 2013)

1.4.5 Visual Paradigm

Visual Paradigm for UML es una herramienta CASE que soporta el modelado mediante UML y proporciona asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software. (Villareal, 2014).

Se integra con las siguientes herramientas Java:(C. D. Sierra, 2007)

- Eclipse/IBM WebSphere
- JBuilder
- NetBeans IDE
- Oracle JDeveloper

- BEA Weblogic

Visual Paradigm For UML proporciona el código y compatibilidad hasta con 10 lenguajes. Es apoyado por un conjunto de idiomas tanto en la generación del código como en la ingeniería inversa, por mencionar algunos ejemplos los cuales tiene la capacidad de soporte podríamos hablar de Java, C++, COBRA IDL, PHP, XML, Schema, Ada y Python. Además, apoya la generación del código C #, VB .NET, Object Definition Lenguaje (ODL), Flash ActionScript, Delphi, Perl, C – Objetivo y Ruby. De igual forma, la ingeniería inversa también apoya clase Java .NET .DLL y .exe, JDBC y archivos de mapeo de Hipertexto. (C. D. Sierra, 2007)

Las ventajas que proporciona Visual Paradigm for UML son: (Villareal, 2014).

- Facilita el modelado de UML, ya que proporciona herramientas específicas para ello. Esto también permite la estandarización de la documentación, ya que la misma se ajusta al estándar soportado por la herramienta.
- Controla que el modelado con UML sea correcto.
- Al disponer de un repositorio común, es posible visualizar el mismo elemento en varios diagramas, evitando duplicidades.
- Permite integrarse con otras aplicaciones, como herramientas ofimáticas, lo cual aumenta la productividad.
- Permite el trabajo en grupo, proporcionando herramientas de compartición de trabajo.
- Facilita la reutilización, ya que se dispone de una herramienta centralizada donde se encuentran los modelos utilizados para otros proyectos.
- Permite generar código de forma automática, reduciendo los tiempos de desarrollo y evitando errores en la codificación del software.
- Permite generar diversos informes, a partir de la información introducida en la herramienta.

Conclusiones Parciales

En este capítulo se realizó un estudio de los fundamentos teóricos y metodológicos para desarrollar una aplicación web para el proceso de gestión

de la información en el departamento de posgrado, evidenciando el impacto positivo del uso de las tecnologías.

Además, se realizó un análisis de los antecedentes históricos de algunas herramientas informáticas utilizadas para la gestión de la información, demostrando la necesidad de crear una herramienta acorde a las necesidades.

Se realizó un análisis de varios lenguajes, herramientas y tecnologías y se concluye utilizar el framework Symfony junto a la herramienta IDE de programación PhpStorm, XAMPP, como herramienta de modelado Visual Paradigm, MySQL como sistema gestor de base de datos, para la implementación los lenguajes PHP, HTML5, JavaScript, CSS y como metodología de desarrollo XP.

Capítulo II

Capítulo II: Desarrollo de la aplicación web para la gestión de la información en el Departamento de Posgrado de la Uniss

La Ingeniería del Software es una disciplina cuyo principal objetivo es ofrecer métodos, técnicas y herramientas que permitan la producción de software de calidad. Para lograr esto, existen varias metodologías, que guían el trabajo en las diferentes etapas del ciclo de vida de un software.

En el presente capítulo se aborda el proceso de desarrollo de la aplicación web para la gestión de la información en el Departamento de Posgrado de la Uniss, guiado por la metodología XP. Se elaboran las tareas de ingeniería y las pruebas de aceptación ayudando a esclarecer el proceso a automatizar.

2.1 Proceso de desarrollo del sistema utilizando la metodología XP.

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Esta metodología se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck, el padre de XP, describe la filosofía de XP en sin cubrir los detalles técnicos y de implementación de las prácticas. (C. Elizabeth, Bombón, Pedro, & Granda, 2012)

2.2 Descripción de los procesos del negocio

La Uniss tiene como misión la formación integral y continua de profesionales, el aporte de resultados científico-técnicos y culturales relevantes y la extensión de su acción con pertinencia en el territorio de espiritano y el país, con el objetivo

de contribuir a la satisfacción de las necesidades sociales, a la elevación de la calidad de vida de la población y al perfeccionamiento del sistema educativo, acorde con el modelo cubano. En ella se desarrollan tres procesos sustantivos fundamentales: el proceso formativo de pregrado y posgrado, el proceso investigativo y el proceso extensionista.

El proceso de formación de posgrado está conformado por dos vertientes de trabajo: la Superación Profesional y la Formación Académica. Las formas organizativas principales de la Superación Profesional son: el curso, el entrenamiento y el diplomado, y la Formación Académica incluye la maestría, especialidad y el doctorado.

El Departamento de Posgrado perteneciente a la Uniss el proceso de gestión del posgrado se realiza a través del Vicedecano de Investigación y Posgrado y Relaciones Internacionales y los Jefes de Departamentos, que realizan las coordinaciones con los capacitadores de las empresas del territorio para determinar las necesidades de superación y en función de estas conformar el plan de posgrado del año.

Los profesores encargados de impartir estos cursos de posgrado, entregan a la secretaria la documentación correspondiente, que es revisada por el técnico de posgrado encargado de esta función; luego de la culminación de su impartición los estudiantes reciben un certificado que acredita el posgrado vencido.

La gestión de la actividad de posgrado es un proceso que se encuentra bien definido por las normativas vigentes y comprende un conjunto de documentos para garantizar el control y evaluación de dicho proceso.

2.2.1 Reglas del negocio

Las reglas del negocio ofrecen una poderosa herramienta para simplificar el proceso del negocio a analistas y desarrolladores. Estas regulan y describen las principales políticas que deben cumplirse para el adecuado funcionamiento del negocio.

A continuación, se presentan las que fueron identificadas:

- La documentación derivada de la superación profesional (diplomados, cursos, entrenamientos, talleres, conferencias, entre otras) debe

archivarse en las secretarías generales de cada facultad y en las de las filiales municipales.

- Toda la documentación de las maestrías, especialidades y doctorados (expedientes del programa y de las ediciones según lo norma las modificaciones aprobadas el 1 de abril y el reglamento de secretarías) debe archivarse en la secretaria general de la institución.
- Controlar en cada facultad, filial o departamento central, que gestione un programa de formación académica las fechas de inicio y terminación de cada edición y las fechas de la obtención de la acreditación de los programas.
- Las solicitudes de las necesidades de superación o capacitación que demanden los organismos, empresas o entidades deberán ser entregadas en un documento oficial con cuño y firma del jefe o director del órgano demandante.
- Los responsables de la actividad de posgrado en cada área deben entregar antes de los días 20 de cada mes y para el mes siguiente las necesidades de transporte, alojamiento y hospedaje de profesores de otras instituciones que vienen a impartir posgrado, o son tribunales en las defensas de doctorado para poder realizar los trámites y asegurar dicha logística.
- Los informes estadísticos parciales y finales sobre la educación de postgrado por las áreas (facultades filiales y áreas centrales) deben entregarse a la dirección de posgrado de la Uniss antes del 15 del mes de junio y antes del 15 del mes de diciembre.
- Todas las áreas docentes de la institución pueden gestionar la presencia de profesionales de alto prestigio académico de otras instituciones y universidades para impartir posgrado.
- En el plan de postgrado las acciones que se planifiquen deben contar con: modalidad que se asume, título del programa, objetivo general, fecha de inicio y terminación, total de créditos, días del mes en que se imparte, participantes y profesor responsable.

A continuación, se muestra un diagrama donde se muestra cómo es que se realiza el proceso de posgrado.

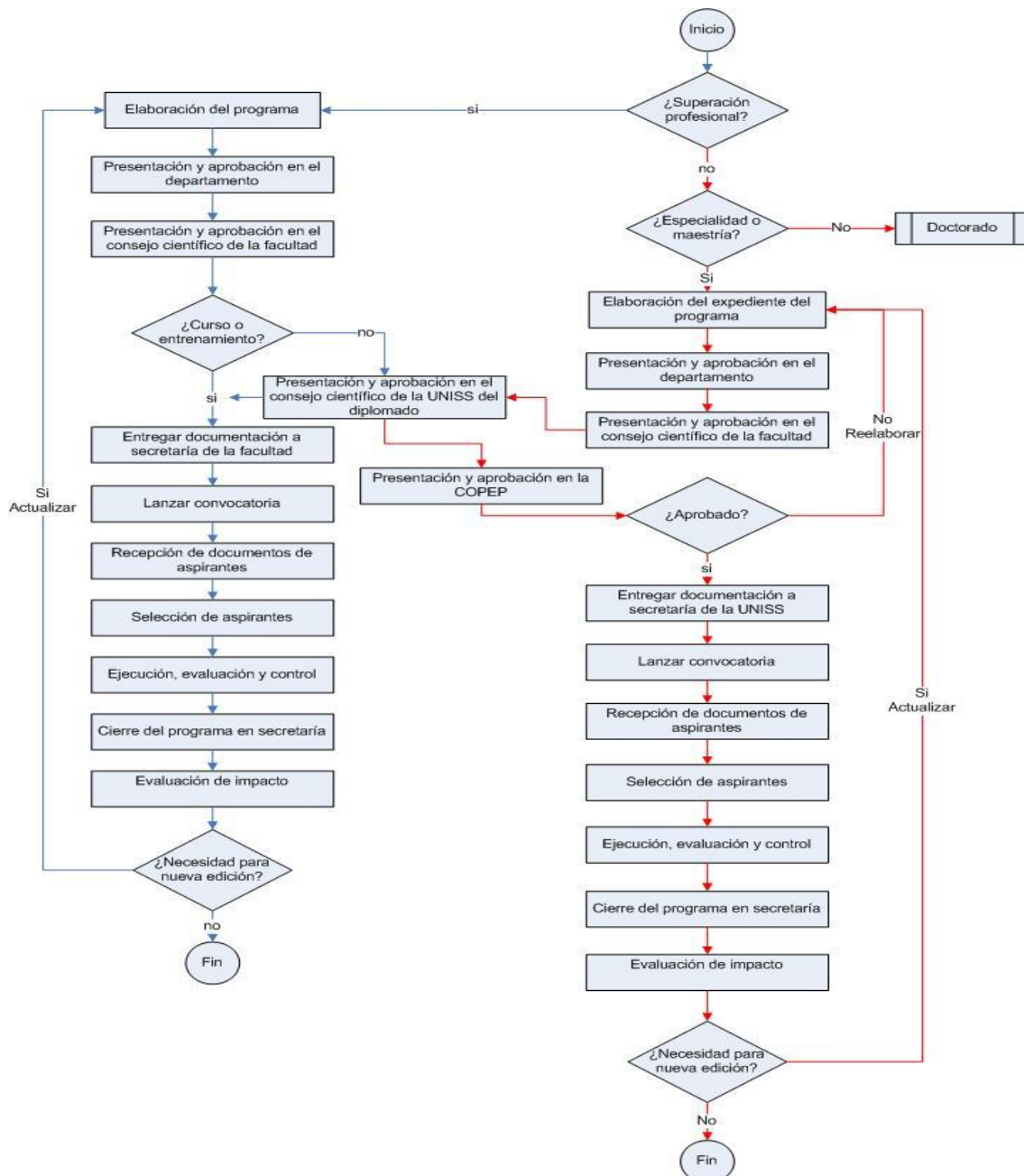


Figura 1 Proceso de Posgrado

2.3 Lista de reserva del producto.

La Lista de Reserva del Producto es un documento que define el trabajo que se va a efectuar en el proyecto, en relación con varios aspectos que se muestran a continuación:

Apariencia o Interfaz externa:

- El Software tendrá una interfaz sencilla para el usuario, de esa forma se facilita la interacción con el mismo.

- La interfaz estará diseñada de modo tal que el usuario pueda ir de punto a punto del sistema con facilidad, estando visibles todas las opciones del producto.

Seguridad:

- La información del software sólo podrá ser modificada por especialistas en la materia. El sistema utiliza la política de impedir accesos no autorizados que pudieran introducir errores en la información original.
- Debe garantizarse la conectividad e integridad de los datos almacenados a través de la red, usando el protocolo de comunicación HTTPS y el SGBD respectivamente.
- El producto estará orientado para ser usado por tres tipos de usuarios, el especialista que puede hacer uso de la mayoría de las funcionalidades de la aplicación, con excepción de la gestión de usuario, la cual solo puede ser utilizada por el administrador del sistema; y, por otro lado, el usuario consultor, el cual solo puede visualizar los reportes referentes a los trabajadores.

Usabilidad:

- La aplicación web podrá ser usada por usuarios que no tengan experiencia en el uso de la computadora, sólo necesitarían un ligero entrenamiento sobre el funcionamiento de los principales elementos en el ambiente de los Sistemas Operativos Windows o Linux (uso del mouse, manejo de menús, botones, cuadros de texto, entre otros.)

Rendimiento:

- No se requerirá una capacidad de procesamiento alta, pues la aplicación no ejecutará algoritmos complejos.

Integridad:

- La información manejada estará protegida contra la corrupción y los estados inconsistentes, pues los mecanismos de validación y el administrador del sistema se encargarán de que los datos entrados sean confiables, de calidad y salvado para los casos de errores.
- **Disponibilidad:** Los usuarios tienen garantizado el acceso a la información sin ningún inconveniente y al mismo tiempo.

- **Confiabilidad:** Garantizar un tratamiento adecuado de las excepciones y la validación de las entradas del usuario. La información manejada por el sistema será protegida contra la corrupción y estados inconsistentes.

Software:

Del lado del Servidor:

- Se requerirá una computadora que cuente con un servidor web (se recomienda Apache, ya que es el que ha sido utilizado para el desarrollo) con soporte para PHP. Además de un gestor de base de datos (MySQL). El sistema operativo puede ser Windows o Linux. Todo lo anterior para una eficiencia óptima.

Del lado del cliente:

- Por parte del cliente se requerirá un navegador que interprete las funciones básicas de JavaScript, recomendado Mozilla Firefox 25 o superior.

Requerimientos de hardware:

- Cliente: Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo un procesador Pentium III, 128 Mb de memoria RAM. Estas máquinas deben estar conectadas en red con el servidor.
- Administrador: Una máquina que funcione como servidor, la cual debe tener como mínimo un Procesador Pentium III con 1 GHz de frecuencia o superior, 512 Mb de Memoria RAM y 40 GB de Disco Duro.

Restricciones en el diseño y la implementación:

- La interfaz del sistema debe ser a través de una página web, con determinadas características de acuerdo con el tipo de usuario que acceda al sistema.
- Utilización del lenguaje de programación PHP.
- Empleo de herramientas de diseño, programación y documentación como Symfony.
- Empleo de la metodología de desarrollo de software XP.

2.4 Proceso de desarrollo de la aplicación siguiendo la metodología XP.

La metodología XP divide el proceso de desarrollo de software en cuatro fases:

- Planificación.
- Diseño.
- Codificación.
- Pruebas.

Y entre sus principales artefactos se encuentran:

- Historias de Usuario.
- Tareas de Ingeniería.
- Pruebas de Aceptación.

2.4.1 Historias de usuario

Las historias de usuario son descripciones, siempre muy cortas y esquemáticas, que resumen la necesidad concreta de un usuario al utilizar un producto o servicio, así como la solución que la satisface.

Como muchas otras herramientas Ágiles, las historias de usuario surgieron como una respuesta orientada al sector de desarrollo de software, aunque con el tiempo se están aplicando a otros tipos de negocio.

Su función principal es identificar problemas percibidos, proponer soluciones y estimar el esfuerzo que requieren implementar las ideas propuestas. (Delía, Cáseres, Ramón, Thomas, & Bertone, 2017)

Iteración	Número Historia	Historias de Usuario.
1	1	Gestionar Organismos
	2	Gestionar Países
	3	Gestionar Ramas de Programas
2	4	Gestionar Estudiantes
	5	Gestionar Profesores
3	6	Gestionar Maestrías
	7	Gestionar Especialidades
	8	Gestionar Doctorados
	9	Gestionar Diplomados
	10	Gestionar Cursos

	11	Gestionar Entrenamientos
4	12	Gestionar Ediciones de Maestrías
	13	Gestionar Ediciones de Especialidades
	14	Gestionar Ediciones de Doctorados
	15	Gestionar Ediciones de Diplomados
	16	Gestionar Ediciones de Cursos
	17	Gestionar Ediciones de Entrenamientos
5	18	Mostrar Base Nominalizada de Maestrías
	19	Mostrar Base Nominalizada de Especialidades
	20	Mostrar Base Nominalizada de Doctorados
6	21	Mostrar Base Nominalizada de Diplomados
	22	Mostrar Base Nominalizada de Cursos
	23	Mostrar Base Nominalizada de Entrenamientos
	24	Mostrar Base Nominalizada de Maestrías y Especialidades
7	25	Autenticar Usuario
	26	Registrar Usuario

Tabla 1 Historias de Usuario

A continuación, se presentan algunas de las historias de usuario más importantes para el cliente, realizadas durante el desarrollo del proyecto. Las demás se pueden consultar en el Anexo 1.

Historia de Usuario	
Número: 4	
Nombre de Historia de Usuario: Gestionar Estudiantes	
Iteración Asignada: 2	
Prioridad en negocio: alta	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: alta	Puntos estimados: 0.6
	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	

Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los estudiantes.
Observaciones: <ul style="list-style-type: none"> • La información de los estudiantes solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado.

Tabla 2 Historia de usuario: Gestionar estudiantes

Historia de Usuario	
Número: 5	
Nombre de Historia de Usuario: Gestionar Profesores	
Iteración Asignada: 2	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6
	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los profesores.	
Observaciones: <ul style="list-style-type: none"> • La información de los profesores solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Tabla 3 Historia de Usuario: Gestionar profesores.

Historia de Usuario	
Número: 12	
Nombre de Historia de Usuario: Gestionar Ediciones de Maestrías.	
Iteración Asignada: 4	
Prioridad en negocio: alta	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: alta	Puntos estimados: 0.8
	Puntos reales: 0.8
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite insertar, modificar, eliminar y observar los datos de las ediciones de maestrías.	
Observaciones: Las ediciones se diferencian por las fechas de inicio y defensa	

Tabla 4 Historia de usuario: Gestionar ediciones de maestrías.

Historia de Usuario	
Número: 18	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Maestrías	
Iteración Asignada: 5	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de las diferentes maestrías.	
Observaciones:	

Tabla 5 Historia de usuario: Mostrar Base Nominalizada de Maestrías.

2.4.2 Tareas de ingeniería

Una vez realizadas las historias de usuario, estas se desglosan en tareas más pequeñas denominadas tareas de ingeniería.

Las tareas de la ingeniería son escritas por el equipo de desarrollo, a partir de las historias de usuario elaboradas por el cliente. Cada tarea describe a cada Historia de usuario, dando un detalle más profundo de las mismas para realizar la implementación, estimando un tiempo más cercano a la realidad para realizar cada una de ellas.

Según el criterio de (C. S. B. Elizabeth, 2018) Una historia de usuario se descompone en varias tareas de ingeniería.

Las tareas de ingeniería se vinculan más al desarrollador ya que permite tener un acercamiento con el código.

Una tarea de ingeniería se compone de lo siguiente:

- Identificador.
- Relación con la historia.
- Tipo de tarea.
- Responsable.

A continuación, se muestra en una tabla las tareas de ingeniería necesarias para la realización del sistema propuesto.

HU	Tarea de ingeniería	
1	A-1	Diseño de la interfaz Gestionar Organismos.
	A-2	Implementación de operaciones CRUD para Gestionar Organismos.
2	B-1	Diseño de la interfaz Gestionar Países.
	B-2	Implementación de operaciones CRUD para Gestionar Países.
3	C-1	Diseño de la interfaz Gestionar Ramas de Programas.
	C-2	Implementación de operaciones CRUD para Gestionar Ramas de Programas.

4	D-1	Diseño de la interfaz Gestionar Estudiantes.
	D-2	Implementación de operaciones CRUD para Gestionar Estudiantes.
5	E-1	Diseño de la interfaz Gestionar Profesores.
	E-2	Implementación de operaciones CRUD para Gestionar Profesores.
6	F-1	Diseño de la interfaz Gestionar Maestrías.
	F-2	Implementación de operaciones CRUD para Gestionar Maestrías.
7	G-1	Diseño de la interfaz Gestionar Especialidades.
	G-2	Implementación de operaciones CRUD para Gestionar Especialidades.
8	H-1	Diseño de la interfaz Gestionar Doctorados.
	H-2	Implementación de operaciones CRUD para Gestionar Doctorados.
9	I-1	Diseño de la interfaz Gestionar Diplomados.
	I-2	Implementación de operaciones CRUD para Gestionar Diplomados.
10	J-1	Diseño de la interfaz de Gestionar Cursos.
	J-2	Implementación de operaciones CRUD para Gestionar Cursos.
11	K-1	Diseño de la interfaz Gestionar Entrenamientos.
	K-2	Implementación de operaciones CRUD para Gestionar Entrenamientos.
12	L-1	Diseño de la interfaz Gestionar Ediciones de Maestrías.
	L-2	Implementación de operaciones CRUD para Gestionar Ediciones de Maestrías
13	M-1	Diseño de la interfaz Gestionar Ediciones de Especialidades.
	M-2	Implementación de operaciones CRUD para Gestionar Ediciones de Especialidades.
14	N-1	Diseño de la interfaz Gestionar Ediciones de Doctorados.
	N-2	Implementación de operaciones CRUD para Gestionar Ediciones de Doctorados

15	Ñ-1	Diseño de la interfaz Gestionar Ediciones de Diplomados.
	Ñ-2	Implementación de operaciones CRUD para Gestionar Ediciones de Diplomados
16	O-1	Diseño de la interfaz Gestionar Ediciones de Cursos.
	O-2	Implementación de operaciones CRUD para Gestionar Ediciones de Cursos.
17	P-1	Diseño de la interfaz Gestionar Ediciones de Entrenamientos
	P-2	Implementación de operaciones CRUD para Gestionar Ediciones de Entrenamientos.
18	Q-1	Diseño de la interfaz Mostrar Base Nominalizada de Maestrías.
	Q-2	Implementación de operaciones para Mostrar Base Nominalizada de Maestrías.
19	R-1	Diseño de la interfaz Mostrar Base Nominalizada de Especialidades.
	R-2	Implementación de operaciones para Mostrar Base Nominalizada de Especialidades.
20	S-1	Diseño de la interfaz Mostrar Base Nominalizada de Doctorados.
	S-2	Implementación de operaciones para Mostrar Base Nominalizada de Doctorados.
21	T-1	Diseño de la interfaz Mostrar Base Nominalizada de Diplomados.
	T-2	Implementación de operaciones para Mostrar Base Nominalizada de Diplomados.
22	U-1	Diseño de la interfaz Mostrar Base Nominalizada de Cursos.
	U-2	Implementación de operaciones para Mostrar Base Nominalizada de Cursos.
23	V-1	Diseño de la interfaz Mostrar Base Nominalizada de Entrenamientos.
	V-2	Implementación de operaciones para Mostrar Base Nominalizada de Entrenamientos.
24	W-1	Diseño de la interfaz Mostrar Base Nominalizada de Maestrías y

		Especialidades.
	W-2	Implementación de operaciones para Mostrar Base Nominalizada de Maestrías y Especialidades.
25	X-1	Diseño de la interfaz Autenticar Usuario.
	X-2	Implementación de operaciones CRUD para Autenticar Usuario.
26	Y-1	Diseño de la interfaz Gestionar Usuario.
	Y-2	Implementación de operaciones CRUD para Gestionar Usuario.

Tabla 6 Tareas de ingeniería.

A continuación, están las tareas de ingeniería correspondientes a las historias de usuario representadas anteriormente, las demás se pueden consultar en el Anexo 2.

Tarea de ingeniería	
Número Tarea: D-1	Número Historia de Usuario: 4
Nombre Tarea: Diseño de la interfaz Gestionar Estudiantes.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita gestionar la información lo más eficiente posible.	

Tabla 7 Tarea de ingeniería: Diseño de la interfaz gestionar estudiantes.

Tarea de ingeniería	
Número Tarea: D-2	Número Historia de Usuario: 4
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Estudiantes.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: El usuario accede al sistema para insertar, modificar o eliminar la Información del estudiante. Para insertar, deben registrarse sus datos pertinentes; de estar todo correcto, se ejecuta la acción, en caso contrario se emite un mensaje de error. Para modificar los datos primeramente se selecciona, y luego se actualiza el campo deseado. Para eliminar se debe	

seleccionar lo que se desea borrar, posterior a esto se escoge la opción de eliminar.

Tabla 8 Tarea de ingeniería: Implementación de operaciones CRUD para gestionar estudiantes.

Tarea de ingeniería	
Número Tarea: E-1	Número Historia de Usuario: 5
Nombre Tarea: Diseño de la interfaz Gestionar Profesores.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita gestionar la información lo más eficiente posible.	

Tabla 9 Tarea de ingeniería: Diseño de la interfaz gestionar profesores.

Tarea de ingeniería	
Número Tarea: E-2	Número Historia de Usuario: 5
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Profesores.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: El usuario accede al sistema para insertar, modificar o eliminar la Información del profesor. Para insertar, deben registrarse sus datos pertinentes; de estar todo correcto, se ejecuta la acción, en caso contrario se emite un mensaje de error. Para modificar los datos primeramente se selecciona, y luego se actualiza el campo deseado. Para eliminar se debe seleccionar lo que se desea borrar, posterior a esto se escoge la opción de eliminar.	

Tabla 10 Tarea de ingeniería: Implementación de operaciones CRUD para gestionar profesores.

Tarea de ingeniería	
Número Tarea: L-1	Número Historia de Usuario: 12
Nombre Tarea: Diseño de la interfaz Gestionar Ediciones de Maestrías	

Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita gestionar la información lo más eficiente posible.	

Tabla 11 Tarea de ingeniería: Diseño de la interfaz gestionar ediciones de maestrías.

Tarea de ingeniería	
Número Tarea: L-2	Número Historia de Usuario: 12
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Ediciones de Maestrías.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Jose Alberto Pérez Castro	
Descripción: El usuario accede al sistema para insertar, modificar o eliminar la Información de la edición. Para insertar, deben registrarse sus datos pertinentes; de estar todo correcto, se ejecuta la acción, en caso contrario se emite un mensaje de error. Para modificar los datos primeramente se selecciona, y luego se actualiza el campo deseado. Para eliminar se debe seleccionar lo que se desea borrar, posterior a esto se escoge la opción de eliminar.	

Tabla 12 Implementación de operaciones CRUD para gestionar ediciones de maestrías.

Tarea de ingeniería	
Número Tarea: Q-1	Número Historia de Usuario: 18
Nombre Tarea: Diseño de la interfaz Mostrar Base Nominalizada de Maestrías.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita obtener la información lo más eficiente posible.	

Tabla 13 Diseño de la interfaz mostrar base nominalizada de maestrías.

Tarea de ingeniería	
Número Tarea: Q-2	Número Historia de Usuario: 12
Nombre Tarea: Implementación del reporte Mostrar Base Nominalizada de Maestrías.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Realizar la implementación para obtener los datos de la Base Nominalizada de Maestrías.	

Tabla 14 Implementación del reporte mostrar base nominalizada de maestrías.

2.5 Diagrama Entidad-Relación de la Base de Datos

El modelo entidad-relación es el modelo conceptual más utilizado para el diseño de bases de datos. Fue introducido por Peter Chen en 1976. Está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

El modelo entidad-relación se basa en una percepción del mundo real que consiste en un conjunto de objetos básicos llamados entidades y de relaciones entre estos objetos. Se desarrolló para facilitar el diseño de bases de datos, permitiendo especificar un esquema empresarial. Este esquema representa la estructura lógica general de la base de datos. (Mercerat & Silva, 2002)

En este caso, se decide almacenar los datos de la aplicación en una única base de datos estructurada por 14 tablas relacionadas entre sí. La cual se muestra a continuación:

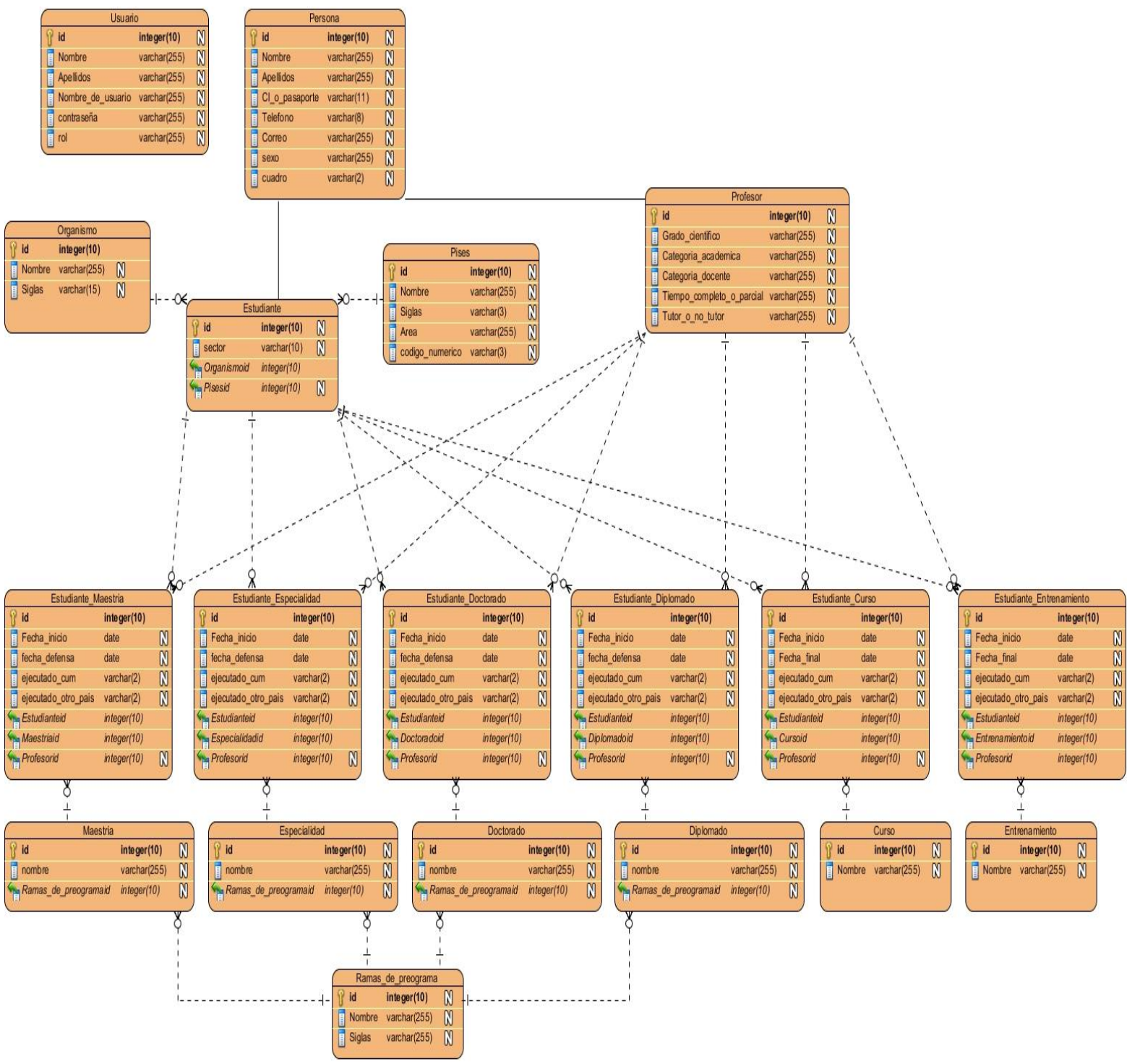


Figura 2 Diagrama Entidad-Relación de la base de datos

Conclusiones Parciales

En este capítulo deja claro cómo funciona el negocio. Se utilizó la metodología de desarrollo ágil XP para guiar los procesos de diseño e implementación de la aplicación web. La misma permitió identificar las historias de usuario y la lista de reserva del producto, así como las tareas de ingeniería correspondientes a dichas historias de usuario. Además, se elaboró el modelo físico de la base de datos.

Capítulo III

Capítulo III: Construcción de la aplicación web propuesta para facilitar la gestión de la información en el Departamento de Posgrado de la UNISS.

Durante el desarrollo de este capítulo se especifica a través de la aplicación de la metodología ágil XP el proceso de desarrollo de la aplicación web, se especifican temas de seguridad, diseño de interfaz y tratamiento de errores. Además, se puntualizan las pruebas de aceptación artefactos planteados por XP para la fase de prueba.

3.1 Tratamiento de excepciones y seguridad. Principio de interfaz de usuario y ayuda del sistema.

Cuando se va a construir una aplicación para cualquier tipo de cliente hay que tener muy en cuenta varios elementos para que el producto final quede con la calidad requerida, entre los principales elementos a tener en cuenta se encuentran: las reglas para el diseño de interfaces, el tratamiento de las excepciones, la seguridad requerida por el usuario y la concepción de la ayuda. A continuación, se expondrá de qué manera se trató cada uno de estos elementos en la aplicación web en cuestión.

3.1.1 Tratamiento de excepciones

Durante la fase de implementación de la aplicación web se tuvo entre las prioridades, evitar la ocurrencia de excepciones y errores indeseados. Para esto se aprovecharon las potencialidades del framework Symfony y la biblioteca Parsley, mediante los formularios de JavaScript, Parsley ayuda a proporcionar comentarios a sus usuarios sobre el envío de formularios antes de enviarlos a su servidor, ahorra ancho de banda, carga de servidor y tiempo a los usuarios. O sea, se trata de una librería ligera y rica en funciones.

Además, se procuró realizar la validación de la información gestionada, de modo tal que la aplicación no permitiera a los usuarios entrar información en un formato que no fuera válido o dejar campos indispensables vacíos, evitando así la pérdida de tiempo y la generación de excepciones. En los casos donde la

operación anterior no pudiera realizarse se procuró elaborar los mensajes de error siguiendo las directrices siguientes:

1. No culpar al usuario del problema ocurrido.
2. Escribir los mensajes de error en el lenguaje del usuario.
3. Explicarle en el mensaje cuál ha sido el error y cómo recuperarse.
4. Utilizar siempre el mismo color y la misma forma para lograr una consecuencia entre los errores y lograr una detección casi intuitiva del error.

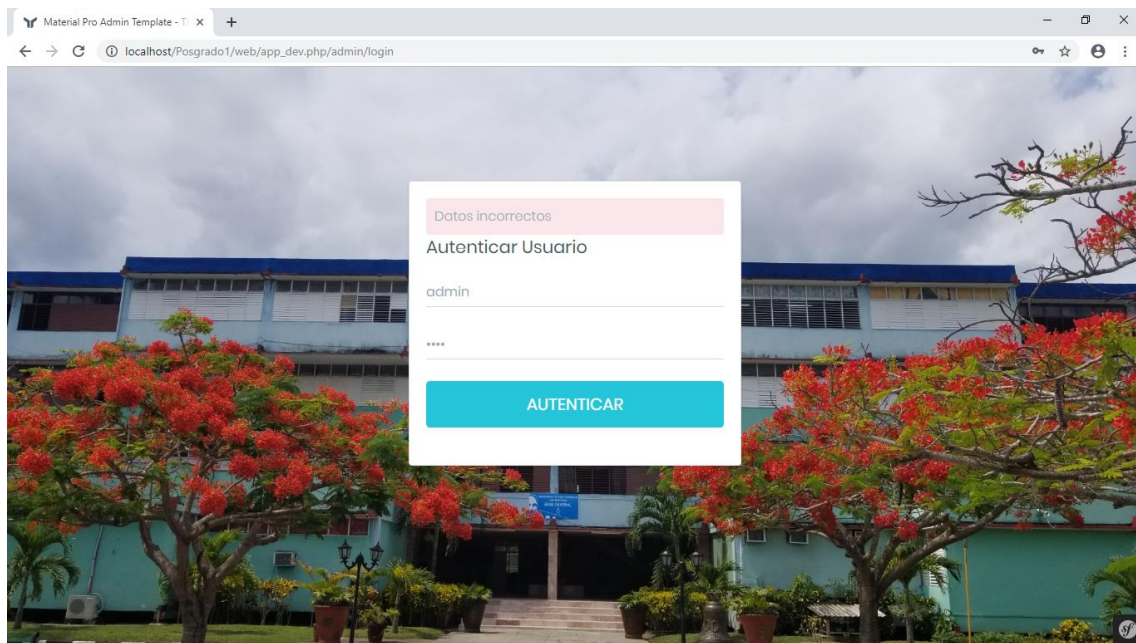


Figura 2 Error al entrar contraseña.

3.1.2 Seguridad

La seguridad del software, al decir de (Lienys & Legra, 2018), es una actividad de garantía de calidad del software que se centra en la identificación y evaluación de los riesgos potenciales que pueden producir un impacto negativo en el software y hacer que falle el sistema completo. Si se pueden identificar pronto los riesgos en el proceso de ingeniería del software podrán especificarse las características del diseño del software que permitan eliminar o controlar los riesgos potenciales.

La seguridad del sistema se gestionó, a través de la autenticación de usuarios, garantizando que solo si se registra el nombre y contraseña de manera correcta tenga acceso a las opciones de la aplicación. Para lograr esto, en la base de datos hay una tabla para los usuarios donde se guardan sus datos y

contraseña. La gestión de los usuarios de la aplicación solo la puede realizar el administrador del sistema, de esta forma y con la asignación de los roles correspondientes; los usuarios podrán acceder a modificar, eliminar o insertar una información solo si tienen permiso.

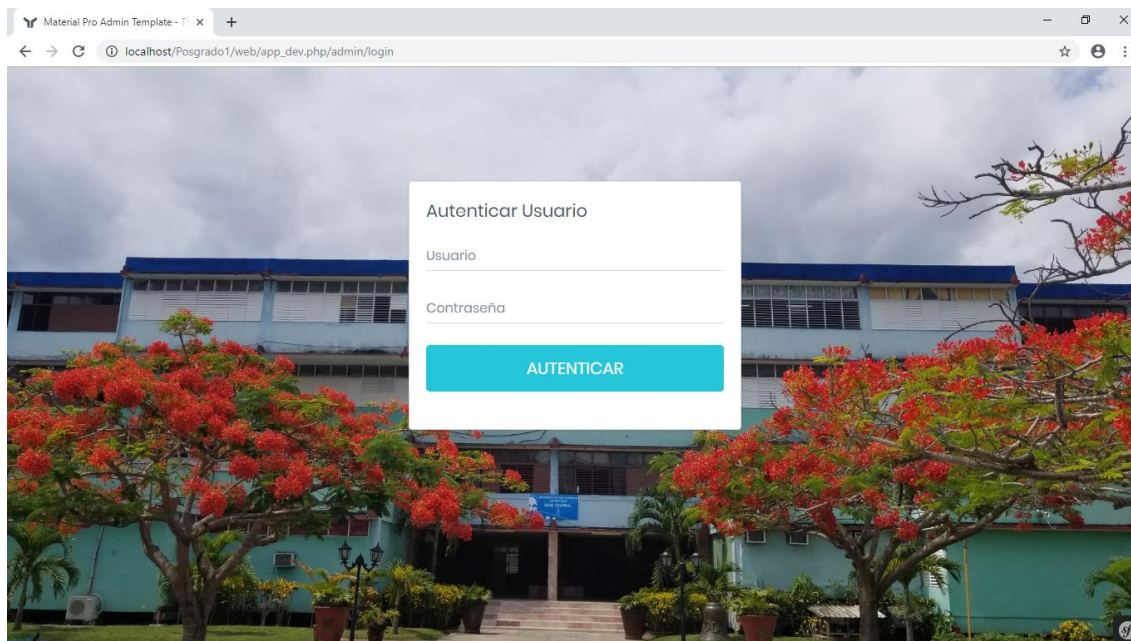


Figura 3 Interfaz para autenticarse y acceder al sistema.

3.1.3 Interfaz de usuario.

La interfaz se concibe con la mayor sencillez posible, de manera que el trabajo con el sistema sea fácil y ameno. Se utiliza una letra legible y un lenguaje conocido por el usuario, sin emplear términos informáticos. Para su diseño se siguieron las tres reglas de oro planteadas por (Gombao, 2017)

1. Dar el control al usuario.
 - ✓ Definir los modos de interacción de manera que no obligue a que el usuario realice acciones innecesarias y no deseadas.
 - ✓ Tener en consideración una interacción flexible.
 - ✓ Diseñar la interacción directa con los objetos que aparecen en la pantalla.
2. Reducir la carga de memoria del usuario.
 - ✓ Reducir la demanda de memoria a corto plazo.
 - ✓ Establecer valores por defecto útiles.
 - ✓ El formato visual de la interfaz se deberá basar en una metáfora del mundo real.

3. Construir interfaces consecuentes.

- ✓ Permitir que el usuario realice una tarea en el contexto adecuado.
- ✓ Mantener la consistencia en toda la familia de aplicaciones.

A continuación, se evidencia lo planteado anteriormente con una pantalla de la aplicación:

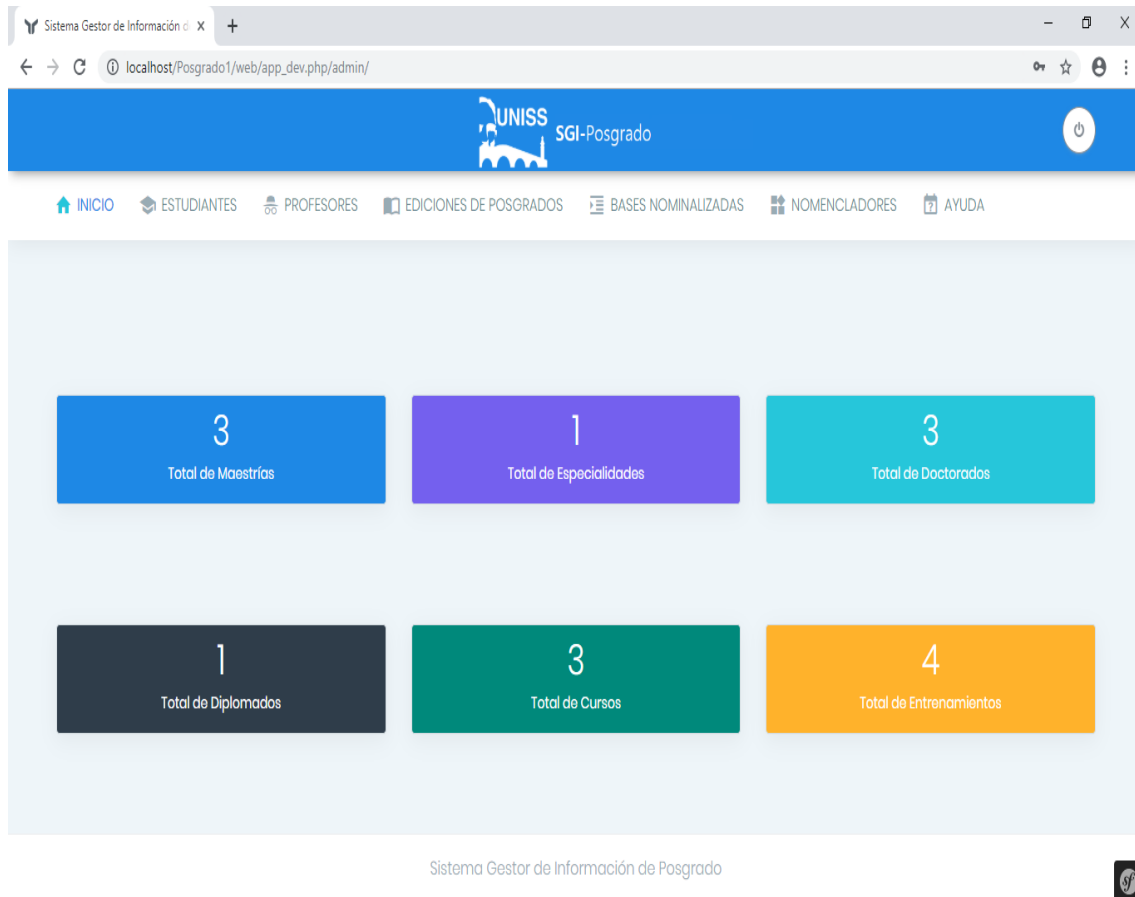


Figura 4 Interfaz principal del sistema.

3.4 Ayuda

La aplicación cuenta con una ayuda, de interés para todos los usuarios del sistema, explica el funcionamiento y las bondades del mismo, así como la vía para acceder a cada una de las funcionalidades de la aplicación, haciéndole más fácil el trabajo del usuario.

Para la realización de dicha ayuda se hace uso de la aplicación Help&Manual5, una aplicación independiente de autoría de ayudas y manuales para programas, equipada con un editor visual (WYSIWYG). Help & Manual permite crear todo tipo de ficheros de ayuda estándar, incluidos HTML HELP Winhelp

(.hlp), ayuda basada en el navegador, manuales de usuario en PDF, ficheros ejecutables y manuales para imprimir.(San & Rosabal, 2011)

Esta ayuda cuenta con una interfaz principal que le brinda al usuario información sobre el software y las ventajas que este le brinda.

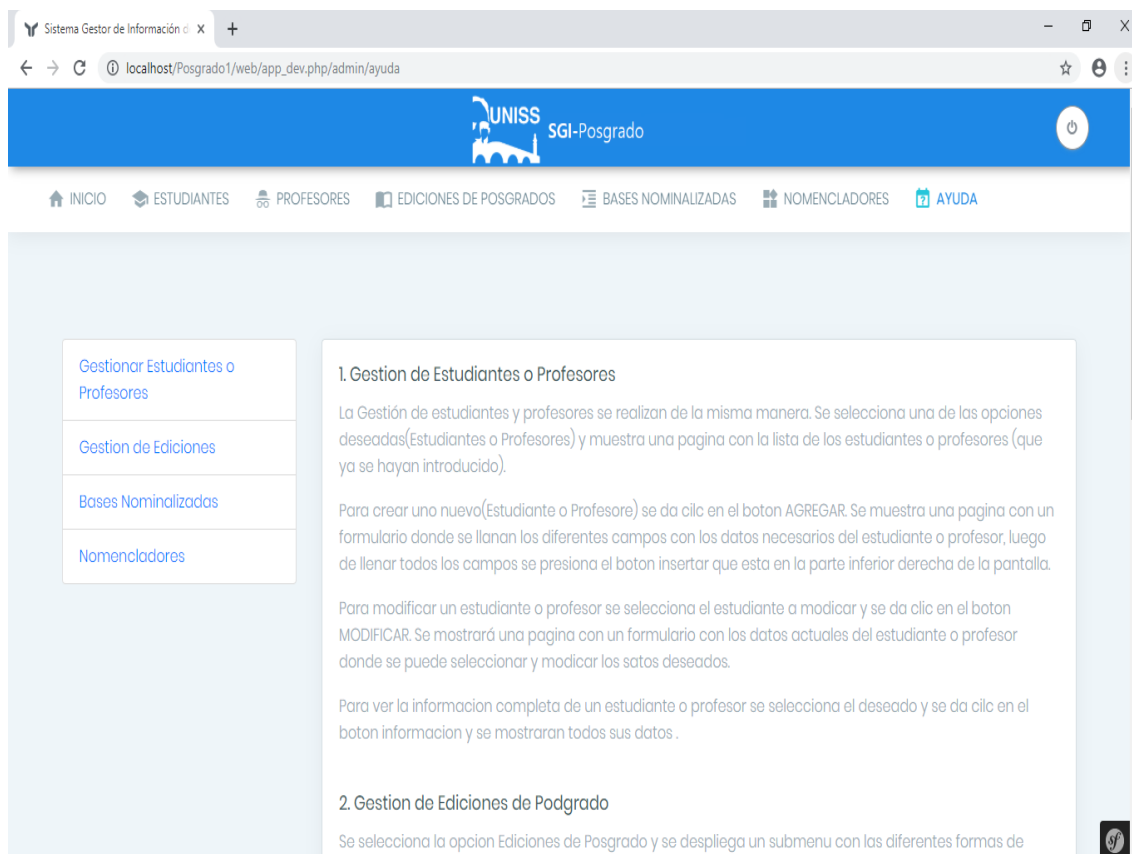


Figura 4 Interfaz Ayuda.

Desde esta interfaz se le brinda al usuario un acceso directo que le permite llegar hasta otras pantallas que le brindan información que puede ser de su interés, como la respuesta a las siguientes preguntas:

3.2 Fase de pruebas según XP: Pruebas de aceptación

La producción de código está dirigida por las pruebas unitarias. Las pruebas unitarias son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema. Los clientes escriben las pruebas funcionales para cada historia de usuario que deba validarse. En este contexto de desarrollo evolutivo y de énfasis en pruebas constantes, la automatización para apoyar esta actividad es crucial. (C. Elizabeth et al., 2012)

Las pruebas de aceptación o también llamadas pruebas funcionales son supervisadas por el cliente, basándose en los requerimientos tomados de las historias de usuario. En todas las iteraciones cada una de las historias de usuarios seleccionadas por el cliente para implementar en la iteración deberá pasar una o varias pruebas de aceptación.

La prueba de aceptación utilizada es la de caja negra que representa el resultado esperado de determinada acción, para que una historia de usuario se considere finalizada deberá pasar exitosamente las pruebas de aceptación correspondiente.

A continuación, se muestran algunas de las pruebas realizadas al sistema las demás se pueden consultar en el Anexo 4.

Caso de Prueba de Aceptación	
Código: P-4	Nombre Historia de Usuario: Gestionar Estudiante
Nombre: Comprobación de los datos de la gestión del estudiante.	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los datos del estudiante.	
Condiciones de Ejecución: Se debe contar con un ordenador con conexión a la aplicación web. El usuario debe haberse autenticado previamente y tener permisos para realizar esa función.	
Entradas/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Solicitar en el menú la pestaña estudiante. 2. Se muestra la interfaz para gestionar estudiante. 3. Se elige una de las opciones: <ol style="list-style-type: none"> a. Insertar (Se llenarán todos los datos que indica la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) b. Modificar (Se selecciona y se cambian los datos deseados. Se verifica que los datos fueron modificados correctamente) c. Eliminar (Se selecciona los datos a eliminar. Se verifica que fue eliminado correctamente) 	

Resultado Esperado: Gestionar estudiante efectuada correctamente.
Evaluación de la Prueba: Satisfactoria.

Tabla 14 Prueba de aceptación: Gestionar estudiante.

Caso de Prueba de Aceptación	
Código: P-5	Nombre Historia de Usuario: Gestionar Profesores.
Nombre: Comprobación de los datos en la gestión de profesores.	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los datos de los profesores.	
Condiciones de Ejecución: Se debe contar con un ordenador con conexión a la aplicación web. El usuario debe haberse autenticado previamente y tener permisos para realizar esa función.	
Entradas/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Solicitar en el menú la pestaña profesores. 2. Se elige una de las opciones: <ol style="list-style-type: none"> a. Insertar (Se llenarán todos los datos que indica la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) b. Modificar (Se selecciona y se cambian los datos deseados. Se verifica que los datos fueron modificados correctamente) c. Eliminar (Se selecciona los datos a eliminar. Se verifica que fue eliminado correctamente) 	
Resultado Esperado: Gestionar profesores efectuadas correctamente.	
Evaluación de la Prueba: Satisfactoria.	

Tabla 15 Prueba de aceptación: Gestionar profesores

Caso de Prueba de Aceptación	
Código: P- 12	Nombre Historia de Usuario: Gestionar Ediciones de

	Maestrías.
Nombre: Comprobación de los datos en la gestión de ediciones de maestrías.	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los datos de las ediciones de maestrías.	
Condiciones de Ejecución: Se debe contar con un ordenador con conexión a la aplicación web. El usuario debe haberse autenticado previamente y tener permisos para realizar esa función.	
Entradas/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Solicitar en el menú la pestaña ediciones de posgrados. 2. Se elige la opción maestrías. 3. Se elige una de las opciones: <ol style="list-style-type: none"> a. Insertar (Se llenarán todos los datos que indica la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) b. Modificar (Se selecciona y se cambian los datos deseados. Se verifica que los datos fueron modificados correctamente) c. Eliminar (Se selecciona los datos a eliminar. Se verifica que fue eliminado correctamente) 	
Resultado Esperado: Gestionar ediciones de maestrías efectuada correctamente.	
Evaluación de la Prueba: Satisfactoria.	

Tabla 16 Prueba de aceptación: Gestionar ediciones de maestrías.

Caso de Prueba de Aceptación	
Código: P- 18	Nombre Historia de Usuario: Mostrar Base Nominalizada de Maestrías.
Nombre: Probar que el sistema muestre reporte con los datos de la base nominalizada de maestrías.	
Descripción: Se comprobará que la aplicación sea capaz de mostrar los datos	

de la base nominalizada de maestrías.
Condiciones de Ejecución: Para esta acción es necesario que se encuentre los datos necesarios correspondientes a la base nominalizada de maestrías.
Entradas/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se ejecutará el sistema informático 2. Se selecciona la opción bases nominalizadas 3. Se selecciona la opción BN. de maestrías 4. Se ejecutará la acción escogida por el usuario
Resultado Esperado: Mostrar los datos de la la base nominalizada de maestrías correctamente.
Evaluación de la Prueba: Satisfactoria.

Tabla 17 Prueba de aceptación: Mostrar Base Nominalizada de Maestrías

Conclusiones Parciales

En este capítulo se realizó el proceso de implementación de la aplicación web teniendo en cuenta lo planteado por la metodología de desarrollo ágil XP. Se siguieron los principios de diseño de interfaz de usuario, seguridad y tratamiento de excepciones. Finalmente, se elaboran las pruebas de aceptación con el cliente a partir de los requerimientos de las historias de usuario.

Conclusiones generales

- Se determinaron las metodologías y herramientas adecuadas que permiten el desarrollo de una aplicación web para contribuir al proceso de gestión de la información. Se utilizó la metodología ágil XP, Symfony como Framework web, MySQL como sistema gestor de base de datos, el Visual Paradigm y UML como herramienta y lenguaje de modelado respectivamente y PHP Storm como herramienta de programación.
- Se diseñó la propuesta para facilitar el proceso de gestión de la información, la cual es capaz de brindarle al cliente la posibilidad de futuras mejoras y transformaciones.
- Se implementó la aplicación web utilizando el lenguaje de programación PHP con funcionalidades que se ajustan a las necesidades del cliente y que brinda al usuario la posibilidad de interactuar de manera dinámica con los datos.

Recomendaciones

Al concluir esta investigación se hacen las siguientes recomendaciones:

- Agregarle un reporte que genere el Modelo de Estadística de Educación de Posgrado 1164-03, el cual lo que hace es recopilar la información referida a los totales de ediciones de cada una de las formas de posgrado dividiéndolas en Superación Profesional y Formación Académica, y dentro de estas los totales de estudiantes que participan junto algunos de sus datos.
- Diferenciar los usuarios con roles por departamentos para que solo puedan introducir información de sus departamentos.

Bibliografía

Alvarez, M. A. (2005). Taller de PHP.

Alvarez, M. A., Elgarte, F., Campaña, F., Serviweb, Correa, L. A., Martos, J. J. C., & Gutierrez, M. (2015). Manual de CSS, hojas de estilo, 1–68.

Alvarez, M., & Ardisson, J. (2011). GUÍA SYMFONY 2.

Ángel, M. (2010). Redalyc.LA GESTIÓN POR PROCESOS: UN ENFOQUE DE GESTIÓN EFICIENTE.

Aponte, Á. M. V. (2014). GUÍA COMPARATIVA DE FRAMEWORKS PARA LOS LENGUAJES HTML 5, CSS Y JAVASCRIPT PARA EL DESARROLLO DE APLICACIONES WEB.

Bakken, S. S., Aulbach, A., Schmid, E., Winstead, J., Wilson, L. T., Lerdorf, R., ... Ahto, J. (2002). Manual de PHP.

Carbó, Y. M. (2017). Cuba y el impacto de las TIC en la informatización de la sociedad - Monografías. Retrieved from <http://www.monografias.com/trabajos24/informatizacion-cuba/informatizacion-cuba.shtml>

Carrascal-reyes, M. C., & Arciniegas-herrera, J. L. (2018). Características de MySQL.

Cepeda, R. (2013). Cepeda, R.

Chavez, S., Martín, A., Rodríguez, N., Murazzo, M., & Software, E. (2012). Metodología AGIL para el desarrollo SaaS, 577–581.

Corrales, E. H. H. (2015). ASISTENTE WEB PARA LA GENERACIÓN Y PERSONALIZACIÓN DE CRUDEL EN EL FRAMEWORK PHP CORE, 105.

Delía, L., Cáseres, G., Ramón, H., Thomas, P., & Bertone, R. (2017). Qué son las historias de usuario y su función en Agilidad.

- Elizabeth, C., Bombón, S., Pedro, I. N. G., & Granda, D. (2012). *Ingeniería De Software Metodología XP*.
- Elizabeth, C. S. B. (2018). Universidad técnica del norte.
- Gauchat, J. D. (2012). *El gran libro de HTML5, CSS3 y Javascript*.
- Gombao, F. martinez. (2017). c.
- Gutiérrez, J. J. (2015). ¿ Qué es un framework web ?, 1–4.
- Ledo, M. J. V., & Ana Bárbara Araña Pérez. (2017). Gestión de la información y el conocimiento Gestión de la información y el conocimiento Information management and knowledge, (August).
- Lienys, I., & Legra, L. (2018). SOFTWARE LEARNING MANAGEMENT SYSTEM FOR SOFTWARE ENGINEERING, 3.
- Marín, R. (2010). Los gestores de bases de datos (SGBD) más usados. Retrieved from <https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>
- Martínez, G. V., Camacho, G. D. S., & Biancha, D. A. G. (2010). DISEÑO DE FRAMEWORK WEB PARA EL DESARROLLO DINÁMICO DE APLICACIONES.
- Mateu, C. (2004). *Desarrollo de aplicaciones web*.
- Mercerat, B., & Silva, D. A. (2002). Construyendo aplicaciones web con una metodología de diseño orientada a objetos, 115.
- Rodríguez, C. R. B. (2018). Revista Márgenes . Vol.6, No.3, Septiembre-Diciembre , 2018. RNPS: 2460, 6(3).
- Rodriguez, N. (2005). las-tic-en-la-actualidad. Retrieved from slideshare/las-tic-en-la-actualidad.htm
- San, A., & Rosabal, J. (2011). Cuadernos de Educación y Desarrollo, (Cv), 1–11.
- Sánchez, M. del R. G., Añorve, J. R., & Alarcón, G. G. (2017). Las Tic en la educación superior , innovaciones y retos, 6. doi:10.23913/ricsh.v6i12.135

Sierra, C. D. (2007). Visual Paradigm For Uml.

Sierra, F., Acosta, J., & Ariza, J. (2013). Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web, 1–13.

Tupe, C., & Cisneros, J. (2016). XAMPP Installers and Downloads for Apache Friends. Retrieved from <http://www.apachefriends.org/download.html>.

Valle, Y. (2018). Sin las TIC, la educación superior y de posgrado no está completa _ Educación _ Actualidad _ ESAN.

Anexos

Anexo 1 Historias de Usuario

Historia de Usuario	
Número: 1	
Nombre de Historia de Usuario: Gestionar Organismos	
Iteración Asignada: 1	
Prioridad en negocio: medio	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: bajo	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los organismos.	
Observaciones: <ul style="list-style-type: none">• La información de los organismos solo puede ser consultada por el personal autorizado.• La información solo puede ser modificados por el personal autorizado.	

Historia de Usuario	
Número: 2	
Nombre de Historia de Usuario: Gestionar Países	
Iteración Asignada: 1	
Prioridad en negocio: medio	Usuario: Secretaria de Posgrado

Riesgo en Desarrollo: bajo	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los países.	
Observaciones: <ul style="list-style-type: none"> • La información de los países solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 3	
Nombre de Historia de Usuario: Gestionar Ramas de Programas	
Iteración Asignada: 1	
Prioridad en negocio: medio	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: bajo	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con las ramas de programas.	
Observaciones: <ul style="list-style-type: none"> • La información de las ramas de programas solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 6	
Nombre de Historia de Usuario: Gestionar Maestrías	
Iteración Asignada: 3	
Prioridad en negocio: alta	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6
	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con las maestrías.	
Observaciones: <ul style="list-style-type: none"> • La información de las maestrías solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 7	
Nombre de Historia de Usuario: Gestionar Especialidades	
Iteración Asignada: 3	
Prioridad en negocio: alta	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6

	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con las especialidades.	
Observaciones:	
<ul style="list-style-type: none"> • La información de las especialidades solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 8	
Nombre de Historia de Usuario: Gestionar Doctorados	
Iteración Asignada: 3	
Prioridad en negocio: alta	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6
	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los doctorados.	
Observaciones:	
<ul style="list-style-type: none"> • La información de los doctorados solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 9	
Nombre de Historia de Usuario: Gestionar Diplomados	
Iteración Asignada: 3	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6
	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los diplomados.	
Observaciones: <ul style="list-style-type: none"> • La información de los diplomados solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 10	
Nombre de Historia de Usuario: Gestionar Cursos	
Iteración Asignada: 3	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6

	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los cursos.	
Observaciones:	
<ul style="list-style-type: none"> • La información de los cursos solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 11	
Nombre de Historia de Usuario: Gestionar Entrenamientos	
Iteración Asignada: 3	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: media	Puntos estimados: 0.6
	Puntos reales: 0.6
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los datos relacionados con los entrenamientos.	
Observaciones:	
<ul style="list-style-type: none"> • La información de los entrenamientos solo puede ser consultada por el personal autorizado. • La información solo puede ser modificados por el personal autorizado. 	

Historia de Usuario	
Número: 14	
Nombre de Historia de Usuario: Gestionar Ediciones de Doctorados.	
Iteración Asignada: 4	
Prioridad en negocio: alta	Usuario: Secretaria de Posgrado
Riesgo en Desarrollo: alta	Puntos estimados: 0.8
	Puntos reales: 0.8
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite insertar, modificar, eliminar y observar los datos de las ediciones de doctorados.	
Observaciones: Las ediciones se diferencian por las fechas de inicio y final	

Historia de Usuario	
Número: 15	
Nombre de Historia de Usuario: Gestionar Ediciones de Diplomados.	
Iteración Asignada: 4	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: alta	Puntos estimados: 0.8
	Puntos reales: 0.8
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite insertar, modificar, eliminar y observar los datos de las ediciones de diplomados.	
Observaciones: Las ediciones se diferencian por las fechas de inicio y final	

Historia de Usuario	
Número: 16	
Nombre de Historia de Usuario: Gestionar Ediciones de Cursos.	
Iteración Asignada: 4	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: alta	Puntos estimados: 0.8
	Puntos reales: 0.8
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite insertar, modificar, eliminar y observar los datos de las ediciones de cursos.	
Observaciones: Las ediciones se diferencian por las fechas de inicio y final	

Historia de Usuario	
Número: 17	
Nombre de Historia de Usuario: Gestionar Ediciones de Entrenamientos.	
Iteración Asignada: 4	
Prioridad en negocio: alta	Usuario: Vicedecano de investigación y posgrado
Riesgo en Desarrollo: alta	Puntos estimados: 0.8
	Puntos reales: 0.8
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite insertar, modificar, eliminar y observar los datos de las ediciones de entrenamientos.	
Observaciones: Las ediciones se diferencian por las fechas de inicio y final	

Historia de Usuario	
Número: 19	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Especialidades	
Iteración Asignada: 5	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de las diferentes especialidades.	
Observaciones:	

Historia de Usuario	
Número: 20	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Doctorados	
Iteración Asignada: 5	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de los diferentes doctorados.	
Observaciones:	

Historia de Usuario	
Número: 21	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Diplomados	
Iteración Asignada: 6	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de los diferentes diplomados.	
Observaciones:	

Historia de Usuario	
Número: 22	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Cursos	
Iteración Asignada: 6	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de los diferentes cursos.	
Observaciones:	

Historia de Usuario	
Número: 23	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Entrenamientos	
Iteración Asignada: 6	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de los diferentes entrenamientos.	
Observaciones:	

Historia de Usuario	
Número: 24	
Nombre de Historia de Usuario: Mostrar Base Nominalizada de Maestrías y Especialidades	
Iteración Asignada: 6	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite conocer los estudiantes de todas las ediciones de las diferentes maestrías y especialidades.	
Observaciones:	

Historia de Usuario	
Número: 25	
Nombre de Historia de Usuario: Autenticar Usuario	
Iteración Asignada: 7	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Permite la entrada de los usuarios al sistema.	
Observaciones:	

Historia de Usuario	
Número: 26	
Nombre de Historia de Usuario: Gestionar Usuario	
Iteración Asignada: 7	
Prioridad en negocio: alta	Usuario: Todos
Riesgo en Desarrollo: medio	Puntos estimados: 0.4
	Puntos reales: 0.4
Programador responsable: José Alberto Pérez Castro	
Descripción: Llevaremos el alta, baja, modificación y observación de los usuarios que podrán acceder al sistema.	
Observaciones:	

Anexo 2 Tareas de Ingeniería

Tarea de ingeniería	
Número Tarea: D-1	Número Historia de Usuario: 4
Nombre Tarea: Diseño de la interfaz Gestionar Estudiantes.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita gestionar la información lo más eficiente posible.	
Tarea de ingeniería	
Número Tarea: D-2	Número Historia de Usuario: 4
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Estudiantes.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: El usuario accede al sistema para insertar, modificar o eliminar la Información del estudiante. Para insertar, deben registrarse sus datos pertinentes; de estar todo correcto, se ejecuta la acción, en caso contrario se emite un mensaje de error. Para modificar los datos primeramente se selecciona, y luego se actualiza el campo deseado. Para eliminar se debe seleccionar lo que se desea borrar, posterior a esto se escoge la opción de eliminar.	
Tarea de ingeniería	
Número Tarea: E-1	Número Historia de Usuario: 5
Nombre Tarea: Diseño de la interfaz Gestionar Profesores.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita gestionar la información lo más eficiente posible.	
Tarea de ingeniería	

Número Tarea: E-2	Número Historia de Usuario: 5
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Profesores.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Jose Alberto Pérez Castro	
Descripción: El usuario accede al sistema para insertar, modificar o eliminar la Información del profesor. Para insertar, deben registrarse sus datos pertinentes; de estar todo correcto, se ejecuta la acción, en caso contrario se emite un mensaje de error. Para modificar los datos primeramente se selecciona, y luego se actualiza el campo deseado. Para eliminar se debe seleccionar lo que se desea borrar, posterior a esto se escoge la opción de eliminar.	
Tarea de ingeniería	
Número Tarea: L-1	Número Historia de Usuario: 12
Nombre Tarea: Diseño de la interfaz Gestionar Ediciones de Maestrías	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita gestionar la información lo más eficiente posible.	
Tarea de ingeniería	
Número Tarea: L-2	Número Historia de Usuario: 12
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Ediciones de Maestrías.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Jose Alberto Pérez Castro	
Descripción: El usuario accede al sistema para insertar, modificar o eliminar la Información de la edición. Para insertar, deben registrarse sus datos pertinentes; de estar todo correcto, se ejecuta la acción, en caso contrario se emite un mensaje de error. Para modificar los datos primeramente se selecciona, y luego se actualiza el campo deseado. Para eliminar se debe seleccionar lo que se desea borrar, posterior a esto se escoge la opción de eliminar.	

Tarea de ingeniería	
Número Tarea: Q-1	Número Historia de Usuario: 18
Nombre Tarea: Diseño de la interfaz Mostrar Base Nominalizada de Maestrías.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Crear una interfaz amigable, de fácil manejo para usuarios con conocimientos básicos de informática, que permita obtener la información lo más eficiente posible.	
Tarea de ingeniería	
Número Tarea: Q-2	Número Historia de Usuario: 12
Nombre Tarea: Implementación del reporte Mostrar Base Nominalizada de Maestrías.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Jose Alberto Pérez Castro	
Descripción: Realizar la implementación para obtener los datos de la Base Nominalizada de Maestrías.	

Anexo 3 Pruebas de Aceptación

Caso de Prueba de Aceptación

Código: P-4	Nombre Historia de Usuario: Gestionar Estudiante
Nombre: Comprobación de los datos de la gestión del estudiante.	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los datos del estudiante.	
Condiciones de Ejecución: Se debe contar con un ordenador con conexión a la aplicación web. El usuario debe haberse autenticado previamente y tener permisos para realizar esa función.	
Entradas/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Solicitar en el menú la pestaña estudiante. 2. Se muestra la interfaz para gestionar estudiante. 3. Se elige una de las opciones: <ol style="list-style-type: none"> a. Insertar (Se llenarán todos los datos que indica la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) b. Modificar (Se selecciona y se cambian los datos deseados. Se verifica que los datos fueron modificados correctamente) c. Eliminar (Se selecciona los datos a eliminar. Se verifica que fue eliminado correctamente) 	
Resultado Esperado: Gestionar estudiante efectuada correctamente.	
Evaluación de la Prueba: Satisfactoria.	
Caso de Prueba de Aceptación	
Código: P-5	Nombre Historia de Usuario: Gestionar Profesores.
Nombre: Comprobación de los datos en la gestión de profesores.	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los datos de los profesores.	
Condiciones de Ejecución: Se debe contar con un ordenador con conexión a la aplicación web. El usuario debe haberse autenticado previamente y tener	

permisos para realizar esa función.

Entradas/Pasos de Ejecución:

1. Solicitar en el menú la pestaña profesores.
2. Se elige una de las opciones:
 - a. Insertar (Se llenarán todos los datos que indica la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - b. Modificar (Se selecciona y se cambian los datos deseados. Se verifica que los datos fueron modificados correctamente)
 - c. Eliminar (Se selecciona los datos a eliminar. Se verifica que fue eliminado correctamente)

Resultado Esperado: Gestionar profesores efectuadas correctamente.

Evaluación de la Prueba: Satisfactoria.

Caso de Prueba de Aceptación

Código: P- 12	Nombre Historia de Usuario: Gestionar Ediciones de Maestrías.
----------------------	--

Nombre: Comprobación de los datos en la gestión de ediciones de maestrías.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los datos de las ediciones de maestrías.

Condiciones de Ejecución: Se debe contar con un ordenador con conexión a la aplicación web. El usuario debe haberse autenticado previamente y tener permisos para realizar esa función.

Entradas/Pasos de Ejecución:

1. Solicitar en el menú la pestaña ediciones de posgrados.
2. Se elige la opción maestrías.
3. Se elige una de las opciones:
 - a. Insertar (Se llenarán todos los datos que indica la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron

insertados correctamente)

b. Modificar (Se selecciona y se cambian los datos deseados. Se verifica que los datos fueron modificados correctamente)

c. Eliminar (Se selecciona los datos a eliminar. Se verifica que fue eliminado correctamente)

Resultado Esperado: Gestionar ediciones de maestrías efectuada correctamente.

Evaluación de la Prueba: Satisfactoria.

Caso de Prueba de Aceptación

Código: P- 18	Nombre Historia de Usuario: Mostrar Base Nominalizada de Maestrías.
----------------------	--

Nombre: Probar que el sistema muestre reporte con los datos de la base nominalizada de maestrías.

Descripción: Se comprobará que la aplicación sea capaz de mostrar los datos de la base nominalizada de maestrías.

Condiciones de Ejecución: Para esta acción es necesario que se encuentre los datos necesarios correspondientes a la base nominalizada de maestrías.

Entradas/Pasos de Ejecución:

1. Se ejecutará el sistema informático
2. Se selecciona la opción bases nominalizadas
3. Se selecciona la opción BN. de maestrías
4. Se ejecutará la acción escogida por el usuario

Resultado Esperado: Mostrar los datos de la la base nominalizada de maestrías correctamente.

Evaluación de la Prueba: Satisfactoria.