



**UNIVERSIDAD DE SANCTI SPÍRITUS “JOSÉ
MARTÍ PÉREZ”
FACULTAD DE CIENCIAS TÉCNICAS Y
EMPRESARIALES.**

Aplicación Web para la Gestión de los Registros Primarios de Información de la Universidad de Sancti Spiritus “José Martí Pérez”.

Trabajo de Diploma en opción al título de Ingeniería Informática

Autor: Joan Betancourt Balbín

Tutores: Dr. Roberto Carlos Rodríguez

Ing. Livan Ramirez Brito

Curso: 2018-2019

Pensamiento

“Donde haya un árbol que plantar, plántalo tú. Donde haya un error que enmendar, enmiéndalo tú. Donde haya un esfuerzo que todos esquivan, hazlo tú. Se tú el que aparta la piedra del camino”

Gabriela Mistral

Agradecimientos

A mis padres, quien le debo todo lo que soy en la vida y que sin ellos por su esfuerzo no estaría aquí,

A mi familia en general por brindarme apoyo en todo momento.

A mis tutores, a mi tutor y profesores del Departamento de Ingeniería Informática que confiaron en mí, y a otros que no dejan de ser especiales como mis amigos y compañeros de curso.

Al Dr. C. Manuel que también me brindó sus conocimientos para lograr este trabajo,

Al ingeniero Liván Ramírez Brito que a pesar de ser mi tutor, es también mi amigo y sin él no hubiese podido hacer este trabajo,

A Dios por mantenerme vivo y permitirme realizar mi sueño y a todos los que de una forma u otra han intervenido en este recorrido por la universidad, a todos les doy GRACIAS.

Joan Betancourt Balbin

Resumen

En la Universidad de Sancti Spíritus José Martí Pérez se gestionan los datos mediante sistemas informáticos como el Assets, el Sigenu y el Directorio Activo, lo que provoca que la información está dispersa, siendo esto una dificultad a la hora de crear reportes y hacer consultas a los datos que se requieren, lo cual imposibilita el fácil manejo de esta información. Esto trae consigo la descentralización, la dispersión de los datos ya que para consultarlos se requiere de buscar los datos en varias fuentes además resulta compleja la búsqueda de la información lo cual toma un tiempo considerable debido a que hay que buscar la información en varias fuentes. Por estas razones se desarrolló una aplicación web para contribuir a la gestión de la información, utilizando tecnologías web. Para el desarrollo de este sistema se utilizó la metodología Extreme Programming (XP) y el Lenguaje Unificado de Modelado (UML). En la implementación se utilizó el framework de desarrollo Django 3.6.2, los lenguajes de programación Python 2.7, HTML 5 y Pycharm 5 como IDE de desarrollo, el gestor de base de datos seleccionado fue MySQL.

Abstract

At the University Jose Marti Perez from Sancti Spíritus, the information about the Informational Primary Recording is being managed by informatics systems like Assets, Sigenu and Active Directories which is a difficulty when creating reports and consulting the data required, which makes impossible to easily manage this information. This brings along the decentralization, redundancy and it is also complex the recovery and validation of the required information and it takes a considerable time because the information is some ways. For these reasons a web application was developed to contribute to the management of information, using web technologies. For the development of this system was used Extreme Programming (XP) methodology and Unified Modeling Language (UML). In the implementation was used the framework Django 3.6.2, the programming languages were Python 2.7, HTML 5 and as development IDE Pycharm 4.5, the database manager selected was MySQL.

Índice

Introducción	1
CAPÍTULO I: Fundamentación teórico-metodológica que sustenta el desarrollo de una aplicación informática para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.	7
Introducción	7
1.1 Antecedentes de la gestión de la información de los Registros Primarios de Información.	7
1.2 Gestión de la Información	8
1.3 Universidad José Martí Pérez	9
1.4 Estructura de los registros primarios	11
1.5 Aplicaciones Informáticas	11
1.5.1 Aplicaciones Desktop	12
1.5.2 Aplicaciones Web	12
1.6 Metodología de desarrollo de software. XP	14
1.7 Arquitectura Cliente Servidor.	15
1.8 Modelo Vista Controlador (MVC) Y Modelo Vista Plantilla (MVT).	15
1.8.1 Modelo Vista Controlador (MVC)	16
1.8.2 Modelo Vista Plantilla(MVT)	16
1.9 Framework Web DJANGO 1.6	17
1.10 HTML5	18
1.11 CSS3	19
1.12 Lenguaje de programación Python 3.6.2	19
1.13 Entorno De Desarrollo Web (Wamp Server 2.5).	20
1.14 Gestor De Base De Datos MySQL.	21
1.15 JavaScript	23
1.15.1 Manejo de las bases de datos en DJANGO	23
1.16 Herramientas a utilizar.	24
1.16.1 Navicat Premium	24
1.16.2 PyCharm 5.0	24
Conclusiones Parciales.	25

CAPÍTULO 2: DESCRIPCIÓN DE LA APLICACIÓN PROPUESTA	28
2.1 Reglas Del Negocio	28
2.2 Procesos Del Negocio	28
2.3 Lista De Reserva Del Sistema	29
2.4 Historias De Usuario	31
2.5 Diagrama Entidad Relación De La Base De Datos	35
2.6 Tareas De Ingeniería	36
Conclusiones Parciales	42
CAPÍTULO 3: Construcción de la aplicación	43
Introducción	43
3.1 Tratamiento De Excepciones Y Seguridad. Principios De Interfaz De Usuario. Ayuda Del Sistema.	43
3.1.1 Tratamiento de excepciones	43
3.1.2 Seguridad	44
3.1.3 Interfaz de Usuario	44
3.1.4 Concepción de la ayuda	48
3.2 Fase de pruebas según XP: Pruebas de aceptación	48
Conclusiones Parciales	54
Conclusiones Generales	55
Recomendaciones	56
Bibliografía	57
Anexos	62
Anexo 1 Historias de Usuario	62
Anexo 2 Tareas de Ingeniería	70
Anexo 3 Pruebas de Aceptación	83

Índice de Tablas

Tabla 1: Historias de Usuario	32
Tabla 2: Historia de Usuario Nro. 1	33
Tabla 3: Historia de Usuario Nro. 2	33
Tabla 4: Historia de Usuario Nro. 3	34
Tabla 5: Historia de Usuario Nro. 4	34
Tabla 6: Tareas de Ingeniería	36
Tabla 7: Tarea de Ingeniería A -1	39
Tabla 8: Tarea de Ingeniería A -2	39
Tabla 9: Tarea de Ingeniería B-1	40
Tabla 10: Tarea de Ingeniería B-2	40
Tabla 11: Tarea de Ingeniería C-1	40
Tabla 12: Tarea de Ingeniería C-2	41
Tabla 13: Tarea de Ingeniería D-1	41
Tabla 14: Tarea de Ingeniería D-2	41
Tabla 15: Prueba de Aceptación P2	48
Tabla 16: Prueba de Aceptación P3	49
Tabla 17: Prueba de Aceptación P6	52
Tabla 18: Prueba de Aceptación P7	53
Tabla 19: Prueba de Aceptación P7	53

Índice de Figuras

Figura 1: Diagrama Entidad-Relación de la Base de Datos	35
Figura 2: Vista General de la Gestión de Roles por cargo	39
Figura 3: Portada de la autenticación al sistema.	46
Figura 4: Portada de la página principal.	47
Figura 5: Portada de gestión de trabajadores docentes.	47

Introducción

En los últimos años el avance de la tecnología ha sido parte del día a día tal que se ha convertido en un recurso vital para las organizaciones y empresas que se ven obligadas a mantenerse en la vanguardia, en diversas áreas como la industria, robótica, seguridad y por supuesto el uso de sistemas de cómputo con la finalidad de automatizar procesos manuales y rutinas pesadas, agilizando así el procesamiento de la información para facilitar la toma de decisiones y poder competir en el presente mercado laboral.

Esto trae consigo que hoy en día el auge que han alcanzado las nuevas Tecnologías de la Informática y las Comunicaciones (TIC) a nivel mundial sea visible. Es por eso que diferentes instituciones se han inclinado hacia la realización de sistemas automatizados que hagan más eficiente y económico el trabajo. Debido a estas circunstancias, se ha recurrido al proceso de informatización. Con el surgimiento de estas tecnologías, en conjunto a las exigencias de cambios constantes para lograr resultados relevantes, se ha llevado a cabo la expansión de conocimientos en todas las ramas de la sociedad, tanto en los países desarrollados como en los subdesarrollados.

La incorporación de las tecnologías de la información y de la comunicación (TIC) al mundo educativo ha sido quizás más lenta y más tardía de lo deseable, pero indudablemente el proceso es imparable y generalizado tanto cualitativa como cuantitativamente. La preocupación reside en cómo orientarlo para que la integración de las TIC resulte efectiva, se garantice un mejor aprovechamiento de su potencial formativo y de la mano de dichos recursos pueda, incluso, vehicularse la innovación y la mejora educativas (González Sanmamed, 2009).

La universidad ha de estar preparada para hacer frente a los profundos cambios que caracterizan a la sociedad de la información (Sangrà y González Sanmamed, 2007). Estas transformaciones en buena medida van a venir provocadas por los efectos de

las TIC sobre las funciones desempeñadas por la universidad, a saber: formación, investigación y, en general, prestación de servicios a la sociedad (Barro, 2004: 16).

El informe de la UNESCO de 1998, en su artículo 12, dedicado a comentar el potencial y los desafíos de la tecnología, señala que las TIC modificarán la forma de elaboración, adquisición y transmisión de los conocimientos, además de que brindarán la posibilidad de renovar el contenido de los cursos y los métodos pedagógicos, y de ampliar el acceso a la educación superior. En este mismo artículo se alude a la figura del profesor, indicando que las TIC alteran su papel en relación con el proceso de aprendizaje.

Por ahora, **los avances más significativos de las TIC en las universidades conllevan:**

- La simplificación de la gestión, reduciendo tiempos de espera y mejorando la efectividad: realizar una matrícula, solicitar un certificado o consultar una duda con el personal administrativo son tareas completamente digitalizadas.
- Material didáctico en mayor cantidad y calidad. El material docente es de fácil acceso y consulta para todos los estudiantes a través de un dispositivo digital al igual que las bibliotecas y los centros de documentación también.
- Entorno virtual para la comunicación fluida y efectiva: desde tutoriales virtuales a foros por asignaturas para debatir y compartir la información impartida en clase.
- La creación y uso de entornos colaborativos globales que tengan una doble vertiente: acceder a conocimientos y contribuir con experiencias y proyectos propios.
- La gestión más efectiva del big data que promueve la innovación, la investigación y la detección de problemas y soluciones con mayor margen de acción.

El manejo y la normalización de este tipo de sistemas mientras se realiza una carrera universitaria supone que el estudiante ya sepa manejar este tipo de tecnologías con

un fin profesional y las aproveche para mejorar su rendimiento y acceder a mejores puestos de trabajo.

El uso de las TIC en la gestión de las universidades cubanas se remonta la década de los 90, con la introducción, por decisión del MES, del sistema Assets para las áreas de Recursos Humanos, Economía, Contabilidad, Finanzas y Planificación. Este sistema es un ERP propietario de factura extranjera que aún se encuentra en uso en todo el sistema MES y por el que anualmente deben erogarse pagos en moneda libremente convertible.

Otro hito importante fue el esfuerzo realizado, desde el inicio de la década del 2000, en la planificación e implementación del sistema SIGENU (Sistema de Gestión de la Nueva Universidad) que se perfilaba, en su diseño original, como el resultado de la acción mancomunada de varios centros de Educación Superior (IES), en el desarrollo de módulos que automatizarían la gestión de los procesos fundamentales y de apoyo de las universidades: matrícula, proceso docente, postgrado, relaciones internacionales, investigación, etc.

No obstante, los esfuerzos realizados por los desarrolladores y la dirección del MES, hasta la actualidad, sólo se ha extendido a todos las IES el módulo de matrícula y promoción. Esfuerzos adicionales interesantes se han realizado por un grupo de universidades. La UCI, por ejemplo, cuenta con una plataforma integrada de sistemas de gestión denominada AKADEMOS, desarrollada a imagen y semejanza de esta universidad que, con sus procesos de producción y exportación y sus condiciones tecnológicas y de recursos humanos, resulta una singularidad difícil de replicar en los demás IES. La CUJAE, la UCLV, y muchos otros centros poseen diferentes sistemas que soportan la gestión y los flujos de información de algunos procesos. Sin embargo, la gestión de las universidades está lejos, desde el punto de vista de su automatización, de lo que debe ser una universidad moderna, a la altura de las demandas actuales y las posibilidades que brindan las TIC.

La transversalidad del proceso de informatización contribuye al perfeccionamiento de la Educación Superior, es por ello que en nuestro organismo la aplicación de las TIC de forma pertinente y novedosa es una de sus principales estrategias.

El “Plan nacional de informatización” para implementar esta política en el período 2017-2021, incluye un grupo de proyectos y sistemas básicos, dentro de los cuales la UNISS contribuye al sistema de gestión integral para la docencia mediante la informatización de procesos de gestión universitarios; así como al desarrollo de plataformas públicas de aprendizaje e interacción en línea.

En la institución se cuenta con algunos de sus procesos parcialmente informatizados, aunque se requiere de una mayor sistematicidad en su informatización. El sub-proceso de informatización, dirigido desde la Dirección de Informatización (DInfo), contiene las bases metodológicas para la informatización de los procesos universitarios. Este sub-proceso se integra a los sub-proceso de información y comunicación a través del **proceso de información, comunicación e informatización**. El mismo se basa en la participación de todos los usuarios de los servicios informáticos de la institución para lograr la informatización de los procesos universitarios.

El **proceso de ciencia, tecnología e innovación** cuenta con una serie de herramientas informáticas que lo apoyan, aunque en su mayoría son elaboradas por el equipo dedicado a su gestión en la UNISS o adquiridas mediante terceros que gestionan este proceso en otras universidades del país. Actualmente no se encuentra ningún módulo de un sistema informático capaz de interoperar con otros sistemas en el ámbito universitario para la gestión de la estructura de los registros primarios de la UNISS.

Asimismo, todos estos procesos que aquí se manejan se hacen de forma manual hasta la actualidad, a través de planillas y documentos de tipo Office por lo que se hace difícil la búsqueda y control de la información dado el gran volumen de datos a manejar. Esto trae consigo la descentralización, el deterioro, la pérdida de documentos en la institución, entre otras consecuencias.

Tomando como punto de partida lo planteado anteriormente, se desprende como **Situación Problemática**: Actualmente los datos referentes a los trabajadores, las áreas, los locales se gestionan mediante los sistemas Assets, SIGENU y el Directorio Activo (*Active Directory*), no existe actualmente un sistema informático para contribuir

a la gestión de los registros primarios de información de la Universidad “José Martí Pérez” de Sancti Spíritus, de modo que esta información pueda ser consultada desde una base de datos única. Consecuentemente, la información referida se encuentra dispersa, lo que repercute en su organización, uso y manejo, limitando a la entidad en sus capacidades de gestionarla.

De la situación Problemática planteada anteriormente, se deriva el siguiente **Problema de Investigación:** ¿Cómo contribuir a la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus?

En correspondencia con lo anterior se plantea entonces como **Objetivo General** de la investigación: Desarrollar una aplicación web, que gestione los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.

Preguntas Científicas:

1. ¿Cuáles son los fundamentos teóricos-metodológicos que sustentan el desarrollo de una aplicación informática para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus?
2. ¿Cómo diseñar una aplicación web para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus?
3. ¿Cómo implementar una aplicación web para gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus?

Tareas de Investigación:

1. Determinar los fundamentos teóricos-metodológicos que sustentan el desarrollo de una aplicación informática para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.
2. Diseñar una aplicación web para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.
3. Implementar una aplicación web para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.

El presente trabajo está estructurado de la siguiente forma: Introducción, tres capítulos, que incluyen todo lo relacionado con el trabajo investigativo, así como el

diseño e implementación del sistema, Conclusiones de cada capítulo y las Generales, recomendaciones, Bibliografía y Anexos.

En el Capítulo I: Fundamentación teórico-metodológica que sustenta el desarrollo de una aplicación informática para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.

En este capítulo se abordan los principales conceptos asociados al dominio del problema. Se describe el objeto de estudio y la situación actual del mismo. También incluye un estudio sobre las principales metodologías y lenguajes que se pueden utilizar para la solución del problema, así como las tecnologías y la justificación de las herramientas seleccionadas para el análisis, diseño e implementación de la aplicación.

En el Capítulo II: Descripción de la aplicación propuesta.

Se describe el Modelo del negocio, identificando los procesos involucrados en él y las reglas que lo rigen. Se realiza la descripción de la Lista de reserva del producto. Además, se realizan las Historias de usuario con su debida estimación de tiempos, se muestra el Diagrama Entidad Relación de la Base de Datos y la descripción de las Tareas de ingeniería.

En el Capítulo III: Construcción de la aplicación

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: Fundamentación teórico-metodológica que sustenta el desarrollo de una aplicación informática para la gestión de los registros primarios de información de la Universidad José Martí Pérez de Sancti Spíritus.

Introducción

El presente capítulo contiene la fundamentación teórico-metodológica sobre el tema a desarrollar. Describe los conceptos asociados al problema, brinda un análisis de los sistemas automatizados existentes vinculados al campo de acción. Además, se realiza un estudio sobre las tecnologías, lenguajes y herramientas existentes determinando cuáles van a ser las utilizadas en el desarrollo del sistema.

1.1 Antecedentes de la gestión de la información de los Registros Primarios de Información.

Los antecedentes referentes a la gestión de los registros primarios de información en la universidad José Martí Pérez de Sancti Spíritus se encuentran los sistemas informáticos Assets, Sigenu y el Directorio Activo que entre sus funcionalidades se encuentra la de gestionar los datos referentes a los trabajadores, a las áreas, los locales y los programas académicos, estos sistemas cumplen con las funcionalidades del sistema que se quiere realizar pero al estar la información dispersa se pretende realizar un sistema informático que permita unir toda esta información para tener mayor accesibilidad a estos datos y agilizar el manejo de ellos.

1.1.1 Algo de historia: Universidad José Martí Pérez.

La Universidad de Sancti Spíritus fue un centro para la clase pudiente con carácter religioso, nombrado por aquel entonces el Colegio el Apostolado del Sagrado Corazón de Jesús, lugar donde se organizaron las brigadas de alfabetización, primaria, FOC(Facultad Obrera Campesina) y también secundaria. Fue fundada en el año 1976 de la Filial Universitaria de Sancti Spíritus, como campus de la Universidad Central Marta Abreu de Las Villas. Marcó un momento importante en el desarrollo de los estudios universitarios en la joven provincia de Sancti Spíritus. Las carreras de

Economía y de Agropecuaria constituyeron una fuente importante para la matrícula del territorio en cursos para trabajadores.

En 1989 se iniciaron los estudios en Cursos Regulares Diurnos (CRD) con la carrera de Licenciatura en Contabilidad y Finanzas. EL desarrollo alcanzado y las proyecciones de trabajo favorecieron la transformación de la Filial en Sede Universitaria, asumiendo carreras de perfil agropecuario, económico y la culminación de estudios de espírituanos que cursaban estudios en (CRD) en otros Centros de Educación Superior del país.

El 6 de junio de 2003 se aprobó por el Comité Ejecutivo del Consejo de Ministro la transformación de Filial a Centro Universitario de Sancti Spíritus "José Martí Pérez" que agrupa a cuatro facultades que asumen a su vez carreras humanísticas, contables, agropecuarias e ingenieriles, hasta un total de doce. En estos momentos existe una fuerte presencia en sus acciones de pregrado, postgrado, ciencia e innovación tecnológica y extensionismo universitario, dando respuesta a las necesidades de una provincia que se caracteriza por su dinámica de desarrollo y por la diversidad geográfica y socio-cultural, con una presencia particularmente importante en el macizo montañoso Guamuhaya a través de la Facultad de Montaña del Escambray.

1.2 Gestión de la Información

En el *Harrod's Librarian's Glossary*, gestión de la información se define como un "término impreciso" que sirve para designar un conjunto de actividades orientadas a la generación, coordinación, almacenamiento o conservación, búsqueda y recuperación de la información tanto interna como externa contenida en cualquier soporte (Prytherch, 2000:372)

Para (Vidal Ledo & Pérez, 2012), la Gestión de la Información debe garantizar que la información esté disponible para cada persona de la organización en el momento requerido, para lo cual desarrolla las siguientes tareas:

- a) Implantar sistemas para conservar, organizar y recuperar cualquier tipo de información interna, de carácter técnico, informes de inteligencia competitiva o cualquier otro tipo de información para lo cual utiliza el formato y los niveles adecuados de acceso según el usuario.
 - b) Garantizar el acceso a la información externa bien en formato electrónico o no, incluyendo el acceso a la Web, a lo que añado: en cualquier otro soporte.
 - c) Mantener un sistema de expertos sobre información actualizada en cuanto a las limitaciones, legislación y condiciones del uso y explotación de la información por lo que refiere a propiedad intelectual y legislación sobre la protección de datos.
 - d) Desarrollar sistemas modernos y flexibles de diseminación selectiva de la información.
 - e) Crear y mantener sistemas de comunicación para que la información fluya con rapidez y eficacia entre los miembros de la organización, por ejemplo, mediante la creación de una Intranet.
 - f) Evaluar de forma continua el sistema de información para mantener los niveles de calidad esperados, y para eliminar aquellos recursos de información subutilizados.
- (Marcial, 2006)

1.3 Universidad José Martí Pérez

El 22 de septiembre de 1983 se funda en Sancti Spíritus, la primera institución de Educación Superior, con personalidad institucional propia, por el acuerdo No 1447 del Consejo de Ministros y que consta en la Resolución Ministerial 424/83, y que en este 2018 arriba 35 años al servicio de la sociedad («Nuestra historia», 2019).

Su estructura incluye dos Vicerrectorados, tres Direcciones Generales, cinco facultades y siete Centros Universitarios Municipales (CUM), dos residencias estudiantiles y una residencia de posgrado.

Cuenta la universidad espirituana con un claustro de excelencia integrado por más de 100 Doctores en Ciencias y de 500 Másteres en Ciencias, cerca de un centenar de Profesores Titulares y más de 200 Profesores Auxiliares. En ella se cursan 38 carreras universitarias.

La formación de profesionales para el desarrollo de la sociedad espirituana se concreta en un total que supera los treinta y seis mil egresados, de diferentes carreras, y más de tres mil másteres y especialistas como parte de la formación posgraduada.

En materia de cooperación internacional puede hablarse de la existencia de 40 convenios que representan las intenciones de trabajo en redes académicas conjuntas, la publicación en revistas científicas y el intercambio institucional en áreas como la formación profesional, las Ciencias Pedagógicas, el uso eficiente de la energía y las fuentes renovables, la producción sustentable de alimentos y el desarrollo humano sostenible, fundamentalmente con universidades de Venezuela, México, Alemania, Argentina, Brasil, Italia, Bélgica y España.

En esta arista del trabajo, 2017 abrió las puertas también al “Lehman College”, institución universitaria norteamericana (New York) con la cual se firmó un Memorando de Entendimiento (MDE) entre cuyos aspectos claves se encuentran el intercambio de estudiantes y el desarrollo de dos proyectos de investigación conjunta: uno centrado en profundizar en los estudios sobre la presencia latina en los EE.UU., (que tiene como proyecciones la creación de una plataforma interactiva), y otro que versa sobre raza y género, y el 2018 comenzó con la firma de un acuerdo de cooperación con la Facultad de Educación de la Universidad de Besançon/ Francia.

Como resultado de los procesos de Evaluación y Acreditación, la universidad espirituana ostenta la categoría de Calificada y cuenta con nueve programas acreditados

Cinco en la formación de posgrado:

- Maestría en Ciencias de la Educación (Cecess). Certificada
- Maestría en Dirección. Certificada
- Maestría en Ingeniería Industrial. Certificada
- Maestría en Ciencias Pedagógicas. Certificada
- Doctorado en Ciencias Pedagógicas. Certificado

Cuatro en la formación de pregrado:

- Carrera Pedagogía Psicología. Certificada
- Carrera Ingeniería Informática. Certificada
- Carrera Agronomía. Certificada
- Carrera Logopedia. De Excelencia

1.4 Estructura de los registros primarios

Proceso para la definición de la estructura de registros primarios. Se define una estructura organizada en Áreas, Institutos, Departamentos y Grupos de trabajo, que tendrán responsabilidades en la enseñanza de grado y posgrado, la investigación y la extensión.

La estructura de los registros primarios constituye el armazón legal, institucional y humano en el que se apoya el proceso de formación académica.

1.5 Aplicaciones Informáticas

En informática, una aplicación es un tipo de programa informático diseñado como herramienta para permitir un usuario realizar uno o diversos tipos de trabajos. Esto lo diferencia principalmente de otros tipos de programas, como los sistemas operativos (que hacen funcionar la computadora), los utilitarios (que realizan tareas de mantenimiento o de uso general), y los lenguajes de programación (para crear programas informáticos). (Cornejo, 2006)

En la actualidad las aplicaciones informáticas se pueden clasificar como aplicación desktop (o de escritorio), aplicaciones para móviles (apps) o aplicaciones web. (Cornejo, 2006)

1.5.1 Aplicaciones Desktop

Una aplicación de escritorio es aquella que se encuentra instalada en el ordenador o sistema de almacenamiento (USB) y se puede ejecutar sin internet en el sistema operativo, al contrario que las aplicaciones en la nube que se encuentran en otro ordenador (servidor) al que se accede a través de la red o internet. (Gassée, 2013)

Es un programa encargado de realizar la funcionalidad del software implementado que se instalará en cada puesto de trabajo y se conectará a través de Internet con la base de datos. La principal ventaja de este sistema será la rapidez de uso ya que se podrá incorporar todos los controles de escritorio y todos los eventos asociados a ellos. (Gassée, 2013)

Como principal desventaja se tiene la gestión de actualizaciones que nos obligará a actualizar todos los programas instalados en cada puesto de la empresa cuando se implementen evoluciones o se corrijan fallos. Esto nos obligará a diseñar un sistema automático de gestión de actualizaciones ya que un usuario con un software obsoleto puede dañar la base de datos. (Gassée, 2013)

Otra desventaja importante es la escasa portabilidad ya que, si se implementa para un entorno Windows, solo en equipos de ese tipo funcionará y no se puede usar en una Tablet o un teléfono.

1.5.2 Aplicaciones Web

Conceptualmente, una aplicación web ha sido tratada por varios especialistas. A continuación, se presentará algunos de estos criterios.

“Las aplicaciones web generan dinámicamente una serie de páginas en un formato estándar, soportado por navegadores Web comunes como HTML o XHTML. Se utilizan lenguajes interpretados del lado del cliente, tales como Java Script, para añadir elementos dinámicos a la interfaz de usuario. Generalmente cada página web individuales enviada al cliente como un documento estático, pero la secuencia de páginas provee de una experiencia interactiva.” (Peñafiel, 2016)

“Una aplicación web (web-base aplicación) es un tipo especial de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP)

están estandarizados y no han de ser creados por el programador de aplicaciones.” (Luján Mora, 2002)

“Las aplicaciones web son populares debido al gran uso del navegador web como cliente que necesita pocos requerimientos para su funcionamiento. La facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

La aplicación web no sustituye a los sistemas informáticos que ya tienen las empresas, es la interfaz que los transforma en servicio, permite acceder a los datos de la empresa para ofrecer información actualizada.” (Granados, 2016)

Basados en los conceptos anteriores, se podrá concluir que una aplicación web es en su conjunto una capa de datos, un servidor web y una interfaz gráfica. La primera es donde se almacenará todos los datos e informaciones de la misma. La segunda es la que se encarga de traducir el código interno de la aplicación al diseño propio de la interfaz entendible por el usuario; y la tercera la que muestra la información de un modo entendible al usuario. En otras palabras, debido a su contenido dinámico, una aplicación web permite la interacción del usuario con el sistema.

Con el desarrollo actual de las plataformas web y la evolución cada vez mayor hacia un mundo sobre la internet, una aplicación web es la opción más adecuada y la seleccionada para resolver el problema de investigación.

También se debe tener en cuenta que, mientras una aplicación desktop corre solamente sobre un sistema operativo determinado y una aplicación móvil sobre un sistema para móvil preestablecido; las aplicaciones web son totalmente multiplataforma, solo se necesita un navegador web (de los cuales existen múltiples versiones para todos los sistemas operativos existentes ya sean de computadoras, tabletas o móviles) en el dispositivo mediante el cual se vaya a acceder a la misma. Adicionalmente se debe tener en cuenta que los módulos desarrollados en esta tesis tienen como finalidad integrarse al SII, del cual ya se ha hablado previamente y se ha descrito como una aplicación web.

El desarrollo de una aplicación web resulta una tarea compleja tanto desde el punto de vista de la tecnología y de la ingeniería. Como consecuencia se hace necesario el uso de metodologías que guíen el proceso de desarrollo de software, durante el

desarrollo de todo el ciclo de vida del proyecto, para así desarrollar un producto de calidad y que cumpla con las necesidades exigidas por el cliente.

1.6 Metodología de desarrollo de software. XP

La metodología de desarrollo de software es utilizada por lo general para el desarrollo de sistemas corporativos para el cumplimiento de los objetivos y se define como un conjunto de procedimientos, herramientas y técnicas que se usan para el desarrollo de software. Estas técnicas están pensadas para que el desarrollo del proyecto esté más estructurado, planeado y controlado y como consecuencia de esto la calidad del mismo es mayor.

Este trabajo está encaminado por la necesidad de encontrar una metodología que se adapte al proceso de gestión de los registros primarios de información en la Universidad José Martí Pérez de Sancti Spíritus, lo que llevó a una revisión de la literatura sobre metodologías ágiles para decidir la más adecuada a este tipo de proyectos, decidiéndose por Extreme Programming (XP).

“XP fue introducida como metodología ágil de desarrollo de software sobre finales de los 1990s. Uno de los conocidos “caso de éxito” fue publicado a fines de 1998, cuando Kent Beck introdujo la nueva metodología en el proyecto de desarrollo denominado C3 (Chrysler Comprehensive Compensation) para la firma Chrysler.” (Joskowicz, 2008)

Esta es una metodología ágil que fue desarrollada, buscando guiar equipos de desarrollo de software pequeño o mediano, entre dos y diez desarrolladores, en ambientes de requerimientos imprecisos o cambiantes. XP tiene como base cinco valores: Simplicidad, Comunicación, Retroalimentación, Respeto y Coraje

Para la realización de este trabajo se selecciona esta metodología como herramienta de desarrollo ya que permite tomar los requerimientos del usuario y realizar entregas pequeñas del proyecto, iniciando con una pequeña parte útil del proyecto que proporcione funcionalidad al sistema y que poco a poco va incrementando su funcionalidad en función a los entregables; teniendo en la entrega final el sistema completo con las funcionalidades requeridas.

1.7 Arquitectura Cliente Servidor.

Una aplicación Web ofrece la posibilidad al usuario de actuar sobre la lógica del negocio en el servidor a través de lenguajes que son interpretados por el navegador y basándose en un modelo de arquitectura Cliente / Servidor, a continuación, se expone la definición y características de esta arquitectura.

Para entender bien esta arquitectura se analiza las palabras que la componen:

Según (R. A. L. Acosta & Ureche, 2015) en esta arquitectura cada ordenador o proceso en la red es cliente o servidor:

- ✓ Servidor: Normalmente los servidores son ordenadores potentes dedicados a gestionar unidades de disco, impresoras, tráfico de red, datos o incluso aplicaciones.
- ✓ Clientes: los clientes son máquinas menos potentes y usan los recursos que ofrecen los servidores

La señalada arquitectura implica la existencia de una relación entre procesos que solicitan servicios (clientes) y procesos que responden a estos servicios (servidores). Dicha arquitectura permite a su vez crear aplicaciones distribuidas para facilitar la separación de las funciones según su servicio, permitiendo situar cada función en la plataforma más adecuada para su ejecución.

Teniendo en cuenta lo expuesto anteriormente desde el punto de vista funcional, se puede definir la computación Cliente/Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma, es decir, un cliente hace una petición de un servicio y recibe la respuesta a dicha petición; un servidor recibe y procesa la petición, y devuelve la respuesta.

1.8 Modelo Vista Controlador (MVC) Y Modelo Vista Plantilla (MVT).

“Los procesos de desarrollo de software constituyen una tarea compleja, llevada a cabo en muchos casos por un equipo de desarrollo. Por otro lado, un sistema informático consta, normalmente, de tres componentes separables de acuerdo a su funcionalidad: la vista del sistema, la lógica del dominio o negocio y los datos. Debido

a esta descomposición lógica de los sistemas surge el patrón de arquitectura n-capas (3-capas en este caso).” (Amargot Valdivia, 2015)

Entre los modelos de arquitectura n-capas existentes, uno de los más usados por su robustez es el modelo-vista-controlador.

1.8.1 Modelo Vista Controlador (MVC)

“En pocas palabras, el patrón de diseño MVC organiza el código en base a su función.

De hecho, este patrón separa el código en tres capas:

- La capa del modelo define la lógica de negocio (la base de datos pertenece a esta capa).
- La vista es lo que utilizan los usuarios para interactuar con la aplicación (los gestores de plantillas pertenecen a esta capa).
- El controlador es un bloque de código que realiza llamadas al modelo para obtener los datos y se los pasa a la vista para que los muestre al usuario.” (Potencier, 2016)

En Django este modelo arquitectónico se convierte en un nuevo modelo que se describirá a continuación: el modelo-vista-plantilla.

1.8.2 Modelo Vista Plantilla (MVT)

En próximos acápite se hará referencia a nuestro framework de desarrollo: DJANGO 1.6. El mismo utiliza el MVC tan al pie de la letra que es considerado un Framework MVC. En el caso de este framework la C (de controlador) es manejada internamente por el framework y la parte más interesante se produce en los modelos, las vistas y las plantillas (templates en inglés). Por esto Django es conocido como un Framework Modelo Vista Plantilla (o MVT por sus siglas en inglés).

“En el patrón de diseño MVT:

- La capa del modelo es la capa de acceso a la base de datos. Contiene toda la información sobre los datos: cómo acceder a estos, cómo validarlos, cuál es el comportamiento que tiene, y las relaciones entre los datos.

- La capa de plantilla es la capa de presentación. Contiene las decisiones relacionadas a la presentación: como algunas cosas son mostradas sobre una página web u otro tipo de documento.
- La capa de vista es la capa de la lógica de negocios. Contiene la lógica que accede al modelo y la delega a la plantilla apropiada: puedes pensar en esto como un puente entre los modelos y las plantillas.” (García, 2015)

Si se quisiera establecer una comparación entre ambos modelos para hacer más comprensible su compatibilidad, se podría establecer que en las vistas y las plantillas en el MVT pueden ser el controlador y las vistas respectivamente en el MVC.

La implementación del patrón MVC (y MVT más específicamente) junto al paradigma orientado a objetos fueron algunos de las características tomadas en cuenta en la selección de las tecnologías de desarrollo de la presente investigación.

1.9 Framework Web DJANGO 1.6

Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo–vista–controlador. Fue desarrollado en origen para gestionar varias páginas orientadas a noticias de la World Company de Lawrence, Kansas, y fue liberada al público bajo una licencia BSD en julio de 2005; el framework fue nombrado en alusión al guitarrista de jazz gitano Django Reinhardt.

Algunas características de Django son:

- Un mapeador objeto-relacional.
- Aplicaciones "enchufables" que pueden instalarse en cualquier página gestionada con Django.
- Una API de base de datos robusta.
- Un sistema incorporado de "vistas genéricas" que ahorra tener que escribir la lógica de ciertas tareas comunes.
- Un sistema extensible de plantillas basado en etiquetas, con herencia de plantillas.

- Un despachador de URLs basado en expresiones regulares.
- Un sistema "middleware" para desarrollar características adicionales; por ejemplo, la distribución principal de Django incluye componentes middleware que proporcionan cacheo, compresión de la salida, normalización de URLs, protección CSRF y soporte de sesiones.
- Soporte de internacionalización, incluyendo traducciones incorporadas de la interfaz de administración.
- Documentación incorporada accesible a través de la aplicación administrativa (incluyendo documentación generada automáticamente de los modelos y las bibliotecas de plantillas añadidas por las aplicaciones).(García, 2015)

En resumen, es un sólido framework, con una arquitectura que lo convierte en un framework multiplataforma, una sólida API de bases de datos, un sistema extensible de plantillas basado en etiquetas, con herencia de plantillas, un despachador de URLs basado en expresiones regulares, soporte de internacionalización, incluyendo traducciones incorporadas de la interfaz de administración, consola de Administración de Proyectos, mapeador objeto-relacional ORM propio; características que llevan a que se seleccione este framework en su versión 1.6 como framework para el desarrollo de la aplicación web de esta investigación y próximamente se hablará sobre Python que es su lenguaje de programación.

1.10 HTML5

HTML5, hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que, en sus diferentes versiones, define una estructura básica y un código para la definición de contenido de una página web, como texto e imágenes.

HTML dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas y tablas. Podríamos decir que sirve para crear páginas web, darles estructura y contenido. De esta manera, se define el documento de una forma abstracta y es responsabilidad del cliente (normalmente el navegador web) el mostrar el documento de una forma gráfica, escogiendo apropiadamente las fuentes, el uso de negrita, diferentes tamaños. (Gauchat, 2012)

1.11 CSS3

Es un lenguaje que describe la presentación de los documentos estructurados en hojas de estilo para diferentes métodos de interpretación, es decir, describe cómo se va a mostrar un documento en pantalla.

Tecnología empleada en la creación de páginas web, permite un mayor control sobre el lenguaje HTML. Permite crear hojas de estilo que definen como cada elemento, como por ejemplo los encabezados o los enlaces, se tiene que mostrar. El término “en cascada” indica que diferentes hojas de estilo se pueden aplicar sobre la misma página. Este mismo autor agrega que posee ventajas como:

- ✓ Control centralizado de la presentación de un sitio web completo con lo que se a Los navegadores web permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad.
- ✓ Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario.
- ✓ El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño. (Gauchat, 2012)

1.12 Lenguaje de programación Python 3.6.2

“Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 cuyo nombre está inspirado en el grupo de cómicos ingleses “Monty Python”. Es un lenguaje similar a Perl, pero con una sintaxis muy limpia y que favorece un código legible. Se trata de un lenguaje interpretado o de script, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos.” (González Duque, 2012)

El 5 de diciembre de 2015, fue lanzado Python 3.6.2 el cual incluye:

- Manipulación numérica mejorada tanto para los números de punto flotante como para la clase Decimal.

- Adiciones útiles a la biblioteca estándar, tales como un muy mejorado módulo unittest, el módulo agruparse para analizar las opciones de línea de comandos, OrderedDict contador de clases en el módulo de colecciones.
- Período de mantenimiento mucho más largo en comparación con las versiones anteriores 2.x. Python 2.7 está actualmente espera que se mantenga el apoyo del equipo de desarrollo del núcleo (recibiendo actualizaciones de seguridad y otras correcciones de errores) hasta al menos 2020.
- Se silencian las advertencias que sólo interesan a los desarrolladores de forma predeterminada.
- Permite construir aplicaciones en Windows sin errores cuando no están disponibles las librerías externas.

Una aplicación web trabaja sobre una arquitectura cliente-servidor. Ya se ha explicado las especificidades de la parte del cliente, lo cual nos lleva a aventurarnos a partir del próximo acápite a explicar las tecnologías del lado del servidor.

1.13 Entorno De Desarrollo Web (Wamp Server 2.5).

Se debe tener en cuenta para el desarrollo de una aplicación web el entorno de desarrollo a utilizar. Debe estar equipado con las características necesarias ya que este juega un papel importante. En este epígrafe se describirá brevemente el entorno Wamp Server 2.5.(Martínez, 2012)

Wamp Server, provee a los desarrolladores con los cuatro elementos necesarios para un servidor web: un Sistema Operativo(Windows),un manejador de base de datos(MySQL), un software para servidor web(Apache) y un software de programación scriptWeb (PHP, Python o PERL), debiendo su nombre a dichas herramientas.(Martínez, 2012)

El uso de un WAMP permite servir páginas HTML a internet, además de poder gestionar datos en ellas, al mismo tiempo un WAMP, proporciona lenguajes de programación para desarrollar aplicaciones web. Es un entorno de desarrollo Web. Le permite crear aplicaciones web con Apache, PHP y la base de datos SQL. También viene con SQLiteManager y PHPMyAdmin para que administre más fácilmente sus bases de datos.

Wamp Server tiene funcionalidades que lo hacen muy completo y fácil de usar.

- ✓ Gestionar sus servicios de Apache y MySQL,
- ✓ Cambiar de línea / fuera de línea (dar acceso a todos o sólo local)
- ✓ Instalar y cambiar emisiones de Apache, MySQL y PHP emisiones
- ✓ Gestión de la configuración de sus servidores
- ✓ Acceder a sus registros
- ✓ Acceder a sus archivos de configuración
- ✓ Crear alias
- ✓ Wamp Server cambiar el idioma del menú
- ✓ Acceder a la página principal

Su utilidad es importante a la hora de desarrollar aplicaciones web, ya que funciona como si se trabajara en un servidor web, ya se puede ejecutar estas aplicaciones de manera local y ver cómo sería el funcionamiento antes de ser subidas a un hosting o servidor web. Además de ello se puede gestionar datos con la ayuda del motor de base de datos (MySQL) y su administrador (PHPMyAdmin). (Martínez, 2012)

Por todas estas características que lo hacen fácil de usar y a la vez una potente herramienta se eligió para la creación de la aplicación web.

1.14 Gestor De Base De Datos MySQL.

Es un sistema de administración de bases de datos, una base de datos es una colección estructurada de tablas que contienen datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, accederá y procesar datos guardados en un computador, usted necesita un administrador como MySQL.

Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

Es uno de los sistemas de gestión de bases de datos relacionales de mayor aceptación mundial, ya que permite una creación asequible y fiable de aplicaciones de bases de datos integradas basadas en web de alto rendimiento y fácilmente

ampliables. Algunas de las características más importantes según (Encinas, 2015) son:

- ✓ Su facilidad de uso e instalación.
- ✓ El bajo coste total de propiedad.
- ✓ La capacidad de ampliación y rendimiento en los sitios web con más tráfico y las aplicaciones más exigentes.
- ✓ El soporte de producción.

Grandes sitios web como Wikipedia, Google, Facebook, Twitter y YouTube utilizan MySQL. Además, posee una licencia GNU GPL (software libre) para aplicaciones no comerciales.

A demás posee soporte completo para cláusulas groupby y orderby, soporte de funciones de agrupación. Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor(CORTES. & ENCISO., 2008).

Algunas de sus ventajas que plantea (Espinoza & Sarango, 2013) son:

- Compatible con la mayoría de las plataformas debido a su estabilidad general del uso de bibliotecas de subprocesos.
- Capacidad del núcleo del S.O (Sistema Operativo) para administrar procesos o subprocesos en diferentes CPU's, permitiendo ejecutar procesos que bloqueen y liberen procesos de una región crítica.
- Rendimiento y estabilidad general del sistema de archivos.
- Escalabilidad y límites con respecto a grandes bases de datos.
- Tiene una gran comunidad para soporte

Y como desventaja:

- Carece de procedimientos almacenados, en la parte técnica.

Por todas estas particularidades se eligió este gestor de base de datos para la realización de este trabajo.

MySQL es el sistema gestor de bases de datos de código abierto por excelencia. La razón: las empresas que utilizan MySQL consiguen recortar el gasto dedicado a este tipo de software en un 90%, y por regla general su uso es gratuito, siempre que se

sigan unas reglas muy sencillas. La velocidad y la flexibilidad en diferentes sistemas operativos hacen muy popular a MySQL (J. D. Gutiérrez, 2004).

Para la realización de este trabajo fue seleccionado el gestor de bases de datos MySQL porque permite una creación asequible y fiable de aplicaciones de bases de datos integradas basadas en web gracias a sus características teniendo en cuenta las ventajas y cualidades que posee.

1.15 JavaScript

JavaScript se utiliza principalmente del lado del cliente (es decir, se ejecuta en nuestro ordenador, no en el servidor) permitiendo crear efectos atractivos y dinámicos en las páginas web. Los navegadores modernos interpretan el código JavaScript integrado en las páginas web.

La ventaja de JavaScript es que al estar alojado en el ordenador del usuario los efectos son muy rápidos y dinámicos. Al ser un lenguaje de programación permite toda la potencia de la programación como uso de variables, condicionales, bucles, etc.(Gauchat, 2012)

1.15.1 Manejo de las bases de datos en DJANGO

“Django es un framework donde cada uno de los modelos de datos utilizados tiene su correspondencia en una tabla de la Base de Datos, por ello Django está preparado para soportar las más comunes en el mercado simplemente cambiando la variable DATABASE_ENGINE en el settings.py.” (Cumba Armijos & Barreno Pilco, 2012)

Antes de la parte de configuración se debe instalar y configurar nuestro servidor MySQL. Django solo admite la versión 4 o superior de MySQL, y además se debe descargar e instalar el paquete MySQLdb, disponible en: <http://www.djangoproject.com/r/python-mysql/>.

Posteriormente solo se necesitará ir al archivo settings.py en la raíz de nuestro proyecto y cambiar los valores del diccionario DATABASES con los específicos para MySQL.

1.16 Herramientas a utilizar.

Las consideraciones básicas que se tuvieron en cuenta en la elección de una herramienta para construir la aplicación fueron, por una parte, la obligación de efectuar una equiparación entre las características del problema con las de la tarea a realizar para resolver el problema, con el usuario y con las herramientas en particular seleccionadas. Por otra, la consideración de la fase del desarrollo del sistema en que se está trabajando.

A continuación, se abordará las herramientas seleccionadas para la codificación y el modelado de la aplicación.

1.16.1 Navicat Premium

Navicat es un administrador gráfico de base de datos y un software de desarrollo producido por PremiumSoft CyberTech Ltd. para MySQL, MariaDB, Oracle, SQLite, PostgreSQL y Microsoft SQL Server. Cuenta con un Explorador como interfaz gráfica de usuario soportando múltiples conexiones para bases de datos locales y remotas. Su diseño está pensado para satisfacer las diferentes necesidades de un amplio sector del public; desde administradores y programadores de bases de datos a diferentes empresas que dan soporte y o comparten información con clientes o socios. (PremiumSoft™ CyberTech Ltd., 2017)

Se utilizó para crear el Modelo-Entidad-Relación.

1.16.2 PyCharm 5.0

Dentro de las herramientas IDEs utilizadas se encontró a JetBrains PyCharm 5.0, ambiente de desarrollo integrado para programar en Python con soporte para varios frameworks de desarrollo, entre ellos Django, web2py, Flask, entre otros. PyCharm 5, posee soporte para todas las versiones liberadas de Python. Incluye análisis de código, unidad de pruebas integrada y un depurador gráfico.

Entre sus principales características se tiene:

- Asistencia y Análisis durante la codificación, con completamiento de código, resaltado de sintaxis y errores y las reparaciones rápidas.

- Navegador de Código y del proyecto. Vistas especializadas de proyecto. Vistas estructuradas de archivos. Salto rápido entre archivos, clases, métodos, etc.
- Integración de un controlador de versiones
- Unidad de Pruebas Integrada, con cobertura línea-a-línea.

Posee una serie de combinaciones de teclas que permiten una mayor soltura al programador y facilita la programación, prácticamente sin tocar el mouse. Soporta varios lenguajes anidados como HTML 5, CCS 3, JavaScript; y algunos de los frameworks JavaScript más populares como Angula, React, entre otros.

Para la realización de este proyecto se utilizó el JetBrains PyCharm 5.0, una herramienta que proporciona un editor de código rico e inteligente para Python con resaltado de sintaxis, configuración de formateo del código extendido, comprobación de errores sobre la marcha y terminación de código inteligente, además de ofrecer numerosas opciones para depurar el código Python, lenguaje que se utiliza en el desarrollo del sistema propuesto y para gestionar las plantillas y sus estilos también se utilizó como editor de HTML 5 y CSS3.(Salamea., 2003)

Conclusiones Parciales.

En este capítulo se determinaron los fundamentos teórico-metodológicos necesarios para el desarrollo de una aplicación web para la gestión de los Registros Primarios de Información de la Universidad José Martí Pérez De Sancti Spíritus. Se decidió utilizar la metodología de desarrollo XP ya que resalta por su corto tiempo de duración, un equipo reducido y gran cantidad de información disponible y aceptación, por lo que se determinó que es la mejor opción para este trabajo.

Fue elegido Django como Framework web por ser rápido, seguro y profesional, además de ser compatible con casi cualquier sistema de base de datos y permite construir aplicaciones robustas. Utiliza el patrón de arquitectura Modelo-Vista-Plantilla, que es una modificación del patrón Modelo-Vista-Controlador; el lenguaje de programación Python; tiene una curva de aprendizaje muy baja y altas prestaciones. Se determinó utilizar MySQL como sistema gestor de base de datos, ya que proporciona el diseño de una base de datos robusta y segura. Su integración con el

entorno de desarrollo web Wamp Server a través de PHPMyAdmin permite la ejecución de aplicaciones web en una computadora con escasos recursos.

CAPÍTULO 2: DESCRIPCIÓN DE LA APLICACIÓN PROPUESTA

Introducción

Después de analizado el marco teórico referencial, escogidas las herramientas, la metodología a utilizar y la situación actual en que se encuentra la entidad, se está en condiciones de comenzar el desarrollo del siguiente capítulo. En este se desarrollan las fases de Planificación y Diseño propias de la metodología propuesta para el desarrollo del sistema, así como los artefactos generados en dichas fases.

2.1 Reglas Del Negocio

Las reglas del negocio describen y fijan las principales políticas que deberán cumplirse en pos de un adecuado funcionamiento del negocio. A continuación, se exponen las que fueron identificadas:

- Los usuarios deberán estar registrados en la aplicación web.
- Para agregar a los trabajadores se deben presentar 1 fotos, chequeo médico, síntesis biográfica, expediente laboral, aval del centro de trabajo anterior, llenar el modelo de pre-empleo docente o no docente.
- El estado de los datos que se manejan se actualizan de forma constante.
- Los administradores son los encargados de realizar todos los cambios pertinentes y consultar la información.

2.2 Procesos Del Negocio

“Cuando se habla de procesos de negocio se puede decir que son un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y forma, y que emplean los recursos de la organización para dar resultados que apoyen sus objetivos.” (Jacobson, Booch, & Rumbaugh, 2006)

Los procesos de negocio también se pueden ver como el conjunto de actividades que se realizan con el objetivo de alcanzar un resultado de valor para un cliente de un negocio.

A partir de estos conceptos se identifica el siguiente proceso de negocio:

- Proceso de Gestión de los Registros Primarios de Información.

Este Proceso ocurre en la Universidad de Sancti Spíritus José Martí Pérez.

Este proceso comienza cuando un trabajador solicita entrar a trabajar en la universidad a este se le entrega una planilla llamada modelo de pre empleo ya sea docente o no docente. Estos modelos son entregados a los administradores del departamento de Recursos Humanos que son los encargados de facilitar los datos referentes a los trabajadores, estos datos son guardados en el sistema Assets.

Los trabajadores son ubicados en las áreas pertenecientes a la universidad y a su vez también a los distintos locales que se subordinan a ellas. Los trabajadores docentes son asignados a los distintos programas académicos que son los programas de pregrado y de postgrado.

En estos momentos existen varias fuentes que gestionan los datos referentes a los trabajadores, a las áreas, a los locales entre otros datos que manejan, toda esta información se encuentra dispersa y lo que se pretende es unificar en un sistema toda la información referente a estos datos y a otros que se le incorporan para tener mayor facilidad a la hora de consultar la información que se desea. El sistema que se realizará tendrá como objetivo analizar el estado actual de estos datos e interoperará con otros sistemas por lo cual no existe un sistema informático para contribuir a la gestión de los registros primarios de información de la Uniss.

2.3 Lista De Reserva Del Sistema

Apariencia o interfaz externa: La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.

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

Rendimiento: No se requiere de una capacidad de procesamiento alta, pues la aplicación no ejecutará algoritmos complejos.

Soporte: Se requiere un servidor de bases de datos con soporte de volúmenes medianos de información. Se documentará la aplicación para garantizar su soporte. Se realizará mantenimiento a fin de aumentar las funcionalidades del mismo a través de versiones posteriores y según las nuevas necesidades de los clientes.

Portabilidad: El producto podrá ser utilizado sobre plataforma Windows, Linux u otro sistema operativo. La estandarización del protocolo de TCP/IP y HTTP permite la interacción del lado del cliente para los sistemas operativos más difundidos como los Sistemas GNU/Linux (Debian, Ubuntu, Nova, etc.), Windows o MacOS.

Seguridad: Debe garantizar la conectividad e integridad de los datos almacenados a través de la red usando el protocolo de comunicación HTTPS y el SGBD respectivamente. Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos. El sistema impondrá un estricto control de acceso que permitirá a cada usuario tener disponible solamente las opciones relacionadas con su actividad.

La información deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo con la política de seguridad del sistema.

Integridad: La información manejada está 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.

Requisitos legales: La herramienta propuesta responderá a los intereses del Departamento de Informática de Salud Pública Provincial de Sancti Spíritus.

Confiabilidad: La aplicación en caso de fallos garantiza que las pérdidas de información sean mínimas y los datos almacenados no se pierden ni se modifiquen ya que los mismos solo son modificados cuando se confirma la acción requerida.

Software: Del lado del Servidor: Se requiere una computadora que cuente con un servidor web con soporte para Python 3.6.2. Además, se requiere de un servidor de base de datos MySQL para Python 3.6.2 (versión 1.2.4 o superior), ReportLap

(versión 3.6.2 o superior) para Python 3.6.2 usado para la generación de Reportes. Todo lo anterior para una eficiencia óptima. Del lado del cliente: Se requiere un navegador que interprete las funciones básicas de JavaScript, css3 y html5, recomendado Mozilla Firefox 25 o superior.

Hardware: Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo un procesador Pentium IV, 512 Mb de memoria RAM. Estas máquinas deben estar conectadas en red con el servidor.

Como servidor se requerirá un computador con un procesador Pentium IV, 2 Gb de memoria RAM y al menos 80 Gb de disco duro.

2.4 Historias De Usuario

Las historias de usuario son utilizadas por la metodología XP como una técnica para especificar los requisitos del mismo, tanto requisitos no funcionales como funcionales. Se trata de tarjetas en las cuales el usuario describe las características que el sistema debe poseer, de forma muy breve. Las características fundamentales del método son (Mendoza C. , 2019)

Desarrollo iterativo e incremental: pequeñas mejoras, unas tras otras.

- Pruebas unitarias continuas: frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión: Se aconseja escribir el código de la prueba antes de la codificación.
- Programación en parejas: se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.
- Frecuente integración del equipo de programación en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.

- Simplicidad en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario.

Las historias de usuarios deben contener el detalle mínimo para que los programadores puedan realizar la estimación de tiempo que llevará su desarrollo.

En esta investigación identificaron un total de Historias de Usuarios, las cuales se enumeran en la tabla No1, donde se especifican además los puntos estimados y los reales de cada Historia de Usuario.

Tabla 1: Historias de Usuario

Iteración	Historias de Usuario (Número y Nombre)	Ptos.	Ptos.	
		Estim.	Reales	
Gestionar				
1	1	Gestionar Roles por cargo	0.4	0.30
	2	Gestionar Trabajadores Docentes	0.7	0.6
	3	Gestionar Trabajadores no Docentes	0.7	0.6
	4	Gestionar Areas	0.4	0.4
	5	Gestionar Tipos de area	0.3	0.3
	6	Gestionar Locales	0.3	0.3
	7	Gestionar Programas Académicos	0.5	0.4
	8	Gestionar Usuarios	0.4	0.4
	9	Gestionar provincias	0.3	0.2
	10	Gestionar países	0.3	0.3
	11	Gestionar municipios	0.3	0.3
	12	Gestionar categorías académicas	0.3	0.3
	13	Gestionar categorías científicas	0.3	0.3
	14	Gestionar categorías docentes	0.3	0.3
	15	Gestionar grado de escolaridad	0.3	0.3
	16	Gestionar cargos	0.3	0.3

17	Gestionar graduaciones	0.3	0.3
18	Gestionar grados científicos	0.3	0.3
19	Gestionar función que desempeñará	0.3	0.3
20	Mostrar las trazas de todos los usuarios	0.3 0	0.15

Nota: Se considera 0.5 puntos como 1 día de trabajo

Tabla 2: Historia de Usuario Nro. 1

Historia de Usuario	
Número:1	Nombre de Historia de Usuario: Gestionar Roles por cargo
Modificación(o extensión)de Historia de Usuario(Nro.y Nombre):	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en Negocio: Alta	PuntosEstimados: 0.45
Riesgo en Desarrollo: Media	PuntosReales: 0.3
Descripción: El administrador tendrá la posibilidad de gestionar los roles por cargo además de asignarle a estos los permisos de acceso y modificación en el sistema.	
Observaciones: El administrador es el único que puede insertar, modificar o eliminar el rol de una persona	

Tabla 3: Historia de Usuario Nro. 2

Historia de Usuario	
Número:2	Nombre de Historia de Usuario: Gestionar Trabajadores Docentes
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: administrador	IteraciónAsignada: 1
Prioridad en Negocio: Alta	PuntosEstimados: 0.75
Riesgo en Desarrollo: Alta	PuntosReales: 0.6

Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar los trabajadores docentes de la universidad.
Observaciones: Los trabajadores docentes de la universidad solo deben ser gestionados y consultados por los administradores

Tabla 4: Historia de Usuario Nro. 3

Historia de Usuario	
Número: 3	Nombre de Historia de Usuario: Gestionar Trabajadores no Docentes
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 0.75
Riesgo en Desarrollo: Alta	Puntos Reales: 0.6
Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar los trabajadores docentes de la universidad.	
Observaciones: Los trabajadores docentes de la universidad solo deben ser gestionados y consultados por los administradores.	

Tabla 5: Historia de Usuario Nro. 4

Historia de Usuario	
Número: 4	Nombre de Historia de Usuario: Gestionar Áreas
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Estadística	Iteración Asignada: 1
Prioridad en Negocio: Media	Puntos Estimados: 0.4
Riesgo en Desarrollo: Media	Puntos Reales: 0.4
Descripción: Los administradores son los encargados de crear, mostrar, modificar y eliminar las áreas de la universidad	
Observaciones: Las áreas solo deben ser gestionadas y consultadas por los administradores	

El resto de las Historias de Usuario se encuentran en el [Anexo 1 Historias de Usuario](#)

2.5 Diagrama Entidad Relación De La Base De Datos

Un diagrama o modelo entidad-relación es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información, así como sus interrelaciones y propiedades.

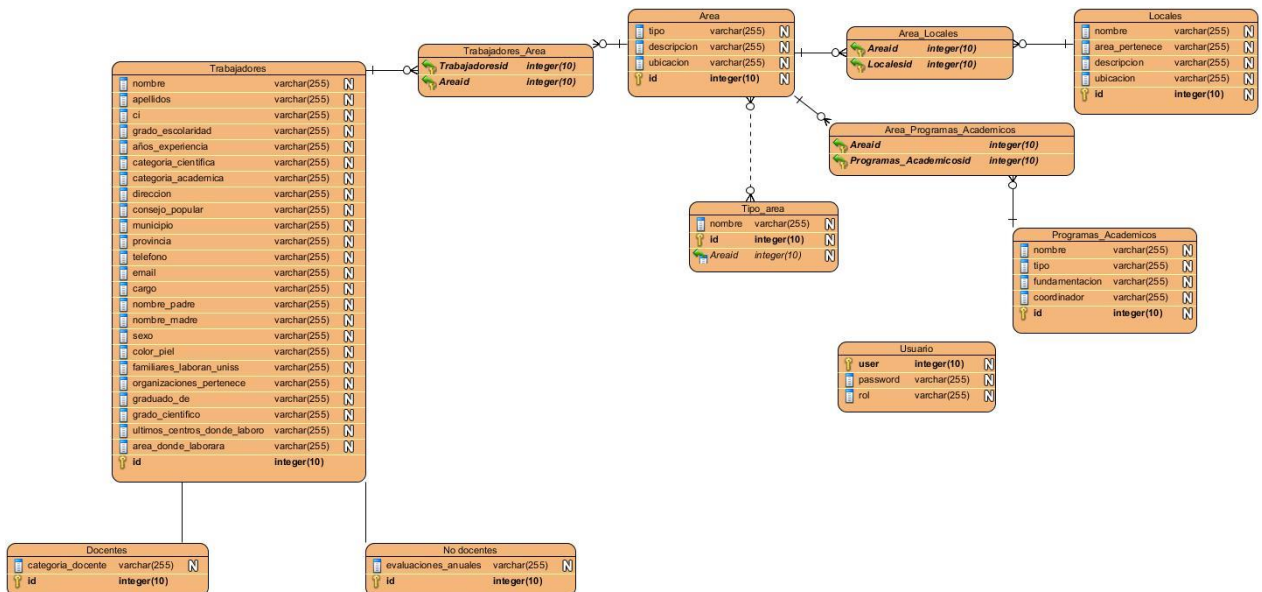


Figura 1: Diagrama Entidad-Relación de la Base de Datos

2.6 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.

Tabla 6: Tareas de Ingeniería

HU		Tareas de Ingeniería	Ptos.	Ptos.
1	A-1	Diseño de la interfaz de gestión de roles por cargo.	0.15	0.05
	A-2	Implementación de la gestión de roles por cargo.	0.3	0.25
2	B-1	Diseño de la interfaz de gestión de los trabajadores docentes.	0.3	0.25
	B-2	Implementación de la gestión de los trabajadores docentes.	0.3	0.5
3	C-1	Diseño de la interfaz de gestión de los trabajadores no docentes.	0.3	0.25
	C-2	Implementación de la gestión de los trabajadores no docentes.	0.23	0.5
4	D-1	Diseño de la interfaz de la gestión de las áreas	0.1	0.2
		Implementación de la gestión de las áreas	0.2	0.2
5	E-1	Diseño de la interfaz de gestión de tipos de áreas	0.1	0.1
		Implementación de la gestión de tipos de áreas	0.2	0.2
6	F-1	Diseño de la interfaz de gestión de los locales	0.1	0.1
	F-2	Implementación de la gestión de los locales	0.2	0.2
7	G-1	Diseño de la interfaz de gestionar los programas académicos	0.2	0.2
		Implementación de gestionar los programas académicos	0.3	0.2

8	H-1	Diseño de la interfaz de gestión de usuario.	0.2	0.2
	H-2	Implementación de la gestión de usuario.	0.2	0.2
9	I-1	Diseño de la interfaz de gestión de Provincias.	0.1	0.1
	I-2	Implementación de la gestión de Provincias.	0.2	0.1
1	J-1	Diseño de la interfaz Gestionar Países	0.1	0.1
0	J-2	Implementación de Gestionar Países	0.2	0.2
1 1	K-1	Diseño de la interfaz de gestión de municipios	0.1	0.1
	K-2	Implementación de la gestión de municipios	0.2	0.2
1 2	L-1	Diseño de la interfaz de gestión categorías académicas	0.1	0.1
		Implementación de gestión categorías académicas	0.2	0.2
1 3	M-	Diseño de la interfaz de Gestionar categorías científicas	0.1	0.1
	M-	Implementación de Gestionar categorías científicas	0.2	0.2
1 4	N-1	Diseño de la interfaz Gestionar categorías docentes	0.1	0.1
	N-2	Implementación de Gestionar categorías docentes	0.2	0.2
1 5	Ñ-1	Diseño de la interfaz Gestionar grado de escolaridad	0.1	0.1
	Ñ-2	Implementación de gestionar grado de escolaridad	0.2	0.2
1 6	O-1	Diseño de la interfaz Gestionar cargos	0.1	0.1
	O-2	Implementación de Gestionar cargos	0.2	0.2

1 7	P-1	Diseño de la interfaz Gestionar graduaciones	0.1	0.1
	P-2	Implementación de Gestionar graduaciones	0.2	0.2
1 8	Q-1	Diseño de la interfaz Gestionar grados científicos	0.1	0.1
	Q-2	Implementación de Gestionar grados científicos	0.2	0.2
1 9	R-1	Diseño de la interfaz Gestionar función que desempeñará	0.1	0.1
	R-2	Implementación de Gestionar función que desempeñará	0.2	0.2
2 0	S-1	Diseño de la interfaz de Mostrar Trazas	0.1	0.1
	S-2	Implementación de Mostrar Trazas	0.2	0.2

Tabla 7: Tarea de Ingeniería A -1

Tarea de Ingeniería	
Nro.deTarea: A-1	Nro.HistoriadeUsuario: 1
Nombre de Tarea: Diseño de la interfaz de Gestionar Roles por cargo	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.15
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita al administrador gestionar los roles por cargo del sistema	



Figura 2: Vista General de la Gestión de Roles por cargo

Tabla 8: Tarea de Ingeniería A -2

Tarea de Ingeniería	
Nro.deTarea: A-2	Nro.HistoriadeUsuario: 1
Nombre de Tarea: Implementación de la gestión de roles por cargo.	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.30
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará la gestión de Roles necesaria para limitar el acceso a la información a los usuarios específicos	

Tabla 9: Tarea de Ingeniería B-1

Tarea de Ingeniería	
Nro.de Tarea: B-1	Nro. Historia de Usuario: 2
Nombre de Tarea: Diseño de la interfaz de gestión de los trabajadores docentes	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.30
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los trabajadores docentes	

Tabla 10: Tarea de Ingeniería B-2

Tarea de Ingeniería	
Nro. de Tarea: B-2	Nro. Historia de Usuario: 2
Nombre de Tarea: Implementación de gestión de los trabajadores docentes	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.30
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementaría una página para la gestión de los trabajadores docentes	

Tabla 11: Tarea de Ingeniería C-1

Tarea de Ingeniería	
Nro. de Tarea: C-1	Nro. Historia de Usuario: 3
Nombre de Tarea: Diseño de la interfaz de gestión de los trabajadores no docentes	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.30
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los trabajadores	

Tabla 12: Tarea de Ingeniería C-2

Tarea de Ingeniería	
Nro. de Tarea: C-2	Nro. Historia de Usuario: 3
Nombre de Tarea: Implementación de gestión de los trabajadores no docentes	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.30
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementaría una página para la gestión de los trabajadores no docentes	

Tabla 13: Tarea de Ingeniería D-1

Tarea de Ingeniería	
Nro. de Tarea: D-1	Nro. Historia de Usuario: 4
Nombre de Tarea: Diseño de la interfaz de gestión de las áreas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las áreas.	

Tabla 14: Tarea de Ingeniería D-2

Tarea de Ingeniería	
Nro. de Tarea: D-2	Nro. Historia de Usuario: 4
Nombre de Tarea: Implementación de la gestión de las áreas	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementaría una página para la gestión de las áreas.	

El resto de las descripciones de cada tarea de Ingeniería se encuentran en el [Anexo 2 Tareas de Ingeniería](#).

Conclusiones Parciales

A través del estudio de los procesos que ocurren en la Universidad José Martí Pérez de Sancti Spíritus se definieron el modelo y las reglas del negocio, lo que conllevó a un mejor entendimiento de las funcionalidades del software a implementar. Estas funcionalidades se especifican en las Historias de Usuario, que dan paso a las Tareas de Ingeniería y al Diagrama Entidad Relación de la Base de Datos, artefactos definidos por XP y que quedaron especificados a través del trabajo realizado en el Capítulo. También se definió la Lista de reserva del producto.

CAPÍTULO 3: Construcción de la aplicación

Introducción

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 pruebas.

3.1 Tratamiento De Excepciones Y Seguridad. Principios De Interfaz De Usuario. Ayuda Del Sistema.

Cuando se va a construir una aplicación para un cliente determinado hay que tener 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 y muy importante el tema de la seguridad. 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 Django, además se procuró realizar la validación de la información gestionada de modo tal que la aplicación no permitiera a los usuarios de la misma entrar información en un formato que no fuera válido evitando así la pérdida 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 forma entendible el 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.

3.1.2 Seguridad

La seguridad del sistema se gestionó a través de la autenticación de usuarios, mediante la cual el usuario debe registrar su nombre y contraseña de manera correcta para poder tener acceso a las opciones que brinda 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.

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 para su rápida comprensión, además se utiliza un lenguaje conocido por el usuario, sin emplear términos informáticos.

Para el diseño de las interfaces se siguieron las tres reglas para el diseño de interfaces gráficas según (Pressman, 2012)

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

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

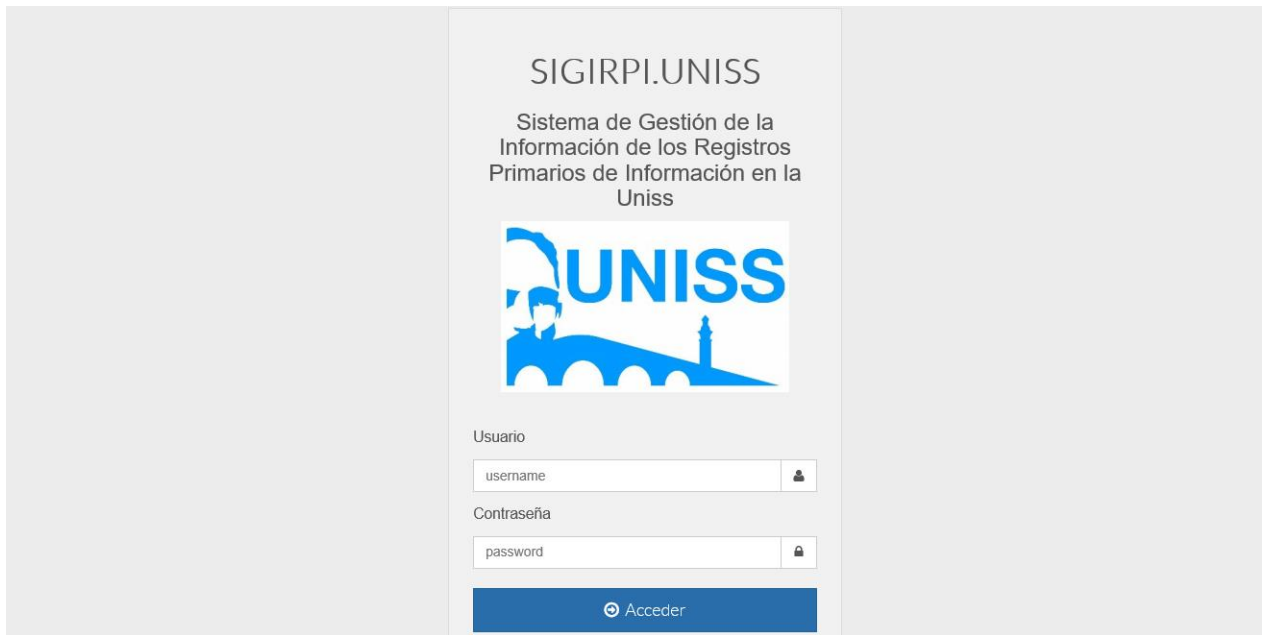


Figura 3: Portada de la autenticación al sistema.



Figura 4: Portada de la página principal.

The screenshot shows a web application interface. At the top left is the UNISS logo. At the top right is a user profile icon for 'joann B B'. The left sidebar contains a navigation menu with the following items: Portada, Gestión de Usuarios, Administración, Locales, Áreas, Programas Academicos, Trabajadores docentes, Trabajadores no docentes, and Gestión de nombradores. The main content area is titled 'Reportes: Trabajadores Docentes'. It features a 'Reportes' tab and a green button labeled '+ Agregar trabajadores docentes'. Below this is a search bar and a dropdown menu set to '10 registros por página'. A table displays one record with the following data:

Nombre	Apellidos	Cargo	CI	Categoría docente	Ubicación	Acciones
ww	aa	Administrador	93042314062	eee	Rectorado	 

Below the table, it says 'Mostrando del 1 al 1 de 1 registros'. At the bottom right of the table area are navigation buttons: 'Anterior', '1', and 'Siguiente'.

Figura 5: Portada de gestión de trabajadores docentes.

3.1.4 Concepción de la ayuda

La aplicación cuenta con un Manual de Usuario, de interés para todos los usuarios del sistema, explica el funcionamiento permitido por cada rol, así como la vía para acceder a cada una de las funcionalidades de la aplicación, haciéndole más fácil el trabajo del usuario y se encuentra disponible en todo momento.

3.2 Fase de pruebas según XP: Pruebas de aceptación

Las pruebas de aceptación o funcionales prueban todas las capas de la aplicación: el sistema de enrutamiento, el modelo, las acciones y las plantillas. En realidad, son muy similares a lo que se hace manualmente cada vez que se añade o modifica una acción y se prueban dichos cambios en el navegador para comprobar que todo funciona bien al pulsar sobre los enlaces y botones y que todos los elementos se muestran correctamente en la página. En otras palabras, lo que se hace es probar un escenario correspondiente a la historia de usuario que se acaba de implementar en la aplicación.

Tabla 15: Prueba de Aceptación P2

Caso de Prueba de aceptación	
Código:P2	Historia de Usuario: 2
Nombre: Comprobación de los datos de la gestión de los trabajadores docentes .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los trabajadores docentes.	
Condiciones de ejecución: Debe existir al menos un usuario con el rol de administrador para realizar esta acción.	
Entradas/Pasos de ejecución: <ol style="list-style-type: none">1. Se ejecutará el sistema informático.2. El Administrador entrará con su usuario y contraseña.	

<ol style="list-style-type: none"> 3. Se selecciona la “Administración ” 4. Se selecciona la opción “Trabajadores Docentes” 5. Se elige una de las siguientes opciones: <ul style="list-style-type: none"> • Añadir Nuevo Trabajador Docente (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) • Editar Trabajador Docente (Se selecciona el Trabajador Docente a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente) • Eliminar Trabajador Docente (Se selecciona el Trabajador Docente a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)
Resultado esperado: Evaluación del Trabajador Docente Gestionada satisfactoriamente
Evaluación de la prueba: Prueba satisfactoria

Tabla 16: Prueba de Aceptación P3

Caso de Prueba de aceptación	
Código:P3	Historia de Usuario: 3
Nombre: Comprobación de los datos de la gestión de los trabajadores no docentes.	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los trabajadores no docentes.	
Condiciones de ejecución: Debe existir al menos un usuario con el rol de administrador para realizar esta acción.	
Entradas/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Se ejecutará el sistema informático. 2. El Administrador entrará con su usuario y contraseña. 	

3. Se selecciona la “**Administración** ”
4. Se selecciona la opción “**Trabajadores no Docentes**”
5. Se elige una de las siguientes opciones:
 - **Añadir Nuevo Trabajador no Docente** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - **Editar Trabajador no Docente** (Se selecciona el **Trabajador no Docente** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
 - **Eliminar Trabajador no Docente** (Se selecciona el **Trabajador no Docente** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Evaluación del Trabajador no Docente Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:P3

Historia de Usuario: 4

Nombre: Comprobación de los datos de la gestión de **Áreas**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las **Áreas**.

Condiciones de ejecución: Debe existir al menos un usuario con el rol de estadística para realizar esta acción.

Entradas/Pasos de ejecución:

1. Se ejecutará el sistema informático.
2. La Estadística entrará con su usuario y contraseña.
3. Se selecciona la “**Administración**”
4. Se selecciona la opción “**Áreas**”
5. Se elige una de las siguientes opciones:

- **Añadir Nueva Áreas** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
- **Editar Áreas** (Se selecciona el **Área** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
- **Eliminar Área** (Se selecciona el **Área** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Área Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:P5

Historia de Usuario: 5

Nombre: Comprobación de los datos de **Gestionar Tipo de Área.**

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los tipos de áreas.

Condiciones de ejecución: Debe existir al menos un usuario con el rol de estadística para realizar esta acción.

Entradas/Pasos de ejecución:

1. Se ejecutará el sistema informático.
2. La Estadística entrará con su usuario y contraseña.
3. Se selecciona la “**Administración**”
4. Se selecciona la opción “**Tipo de Área**”
5. Se elige una de las siguientes opciones:
 - **Añadir Nuevo Tipo de Área** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - **Editar Tipo de Área** (Se selecciona el **Tipo de Área** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)

- **Eliminar Tipo de Área** (Se selecciona el **Tipo de Área** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Tipo de Área Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Tabla 17: Prueba de Aceptación P6

Caso de Prueba de aceptación	
Código:P6	Historia de Usuario: 6
Nombre: Comprobación de los datos de la gestión de locales .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los locales.	
Condiciones de ejecución: Debe existir al menos un usuario con el rol de estadística para realizar esta acción.	
Entradas/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Se ejecutará el sistema informático. 2. La Estadística entrará con su usuario y contraseña. 3. Se selecciona “Administración” 4. Se selecciona “Locales” 5. Se elige una de las siguientes opciones: <ul style="list-style-type: none"> • Añadir Nuevo Local (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) • Editar Local (Se selecciona el Local a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente) • Eliminar Local (Se selecciona el Local a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos) 	
Resultado esperado: Local Gestionado satisfactoriamente	

Evaluación de la prueba: Prueba satisfactoria

Tabla 18: Prueba de Aceptación P7

Caso de Prueba de aceptación	
Código:P7	Historia de Usuario: 7
Nombre: Comprobación de los datos de la gestión de Programas Académicos .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los programas académicos .	
Condiciones de ejecución: Debe existir al menos un usuario con el rol de estadística para realizar esta acción.	
Entradas/Pasos de ejecución: <ol style="list-style-type: none">1 Se ejecutará el sistema informático.2 La Estadística entrará con su usuario y contraseña.3 Se selecciona la “Administración”4 Se selecciona la opción “Programas Académicos”6. Se elige una de las siguientes opciones:<ol style="list-style-type: none">a. Añadir Nuevo Programa Académico (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)b. Editar Programa Académico (Se selecciona el Programa Académico a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)c. Eliminar Programa Académico (Se selecciona el Programa Académico a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)	
Resultado esperado: Programa académico Gestionado satisfactoriamente	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 19: Prueba de Aceptación P7

El resto de las pruebas de aceptación se encuentran en el [Anexo 3: Pruebas de Aceptación](#)

Conclusiones Parciales

Durante el desarrollo de este capítulo se arribaron a las siguientes conclusiones parciales:

1. Se implementó una aplicación web adecuada a los estándares establecidos y a los requerimientos del cliente con un correcto tratamiento de excepciones, la adecuada seguridad tan necesaria para la protección de datos y una Interfaz de Usuario dinámica, agradable y accesible al usuario estándar.
2. Se llevó a cabo la fase de pruebas según la metodología XP, más específicamente las pruebas de aceptación, obteniendo en cada una de ellas resultados satisfactorios.

Conclusiones Generales

- ❖ El estudio de los fundamentos teóricos y metodológicos para la elaboración del sistema permitió determinar que la metodología XP es la adecuada para el análisis, diseño, implementación y documentación del sistema. Se seleccionaron el lenguaje Python, el Framework de desarrollo Django y el Sistema Gestor de Base de Datos MySQL.
- ❖ Se diseñó una aplicación web para la Gestión de la Información de la Secretaría Docente de la Universidad de Ciencias Médicas en Sancti Spíritus, sustentado en las bases del software libre, la programación orientada a objetos y la arquitectura modelo-vista-plantilla.
- ❖ Se implementó una aplicación web utilizando el lenguaje de programación Python 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 empleando la arquitectura de red cliente-servidor de manera que se pueda acceder a la información desde cualquier máquina.

Recomendaciones

Añadir una nueva funcionalidad que permita generar informes en formato PDF con relación al documento oficial de matrícula del estudiante MS-1.

Bibliografía

04SoftwareSociometrico.doc. (s. f.).

Alianza SIDALC. (2019). Recuperado 21 de mayo de 2019, a partir de <http://www.sidalc.net/cgi-bin/wxis.exe/?IsisScript=AGRIUAN.xis&method=post&formato=2&cantidad=1&expresion=mfn=028874>

Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. *Ciencias Holguín, XX(2)*. Recuperado a partir de <http://www.redalyc.org/resumen.oa?id=181531232001>

Cobo, Á. (2005). *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web*. Ediciones Díaz de Santos.

Cornejo, C., & Arturo, D. (2006). Diseño de una red de comunicaciones de voz y datos para el almacén central del Ministerio de Salud. *Universidad Nacional de Ingeniería*. Recuperado a partir de <http://cybertesis.uni.edu.pe/handle/uni/10382>

DSpace ESPOCH.: Estudio Comparativo de las Tecnologías Python y Perl para Desarrollar Aplicaciones Web Implementado al Programa de Alfabetización del Consejo Provincial de Chimborazo. (2019). Recuperado 21 de mayo de 2019, a partir de <http://dspace.esPOCH.edu.ec/handle/123456789/463>

Fernández Marcial, V. (2006). Gestión del conocimiento versus gestión de la información. *Investigación bibliotecológica, 20(41)*, 44-62.

Fernández Marcial, V. (2008). La gestión de la información y las habilidades informacionales: binomio esencial en la formación universitaria. *Reencuentro, (51)*. Recuperado a partir de <http://www.redalyc.org/resumen.oa?id=34005103>

Gauchat, J. D. (2012). *El gran libro de HTML5, CSS3 y Javascript*. Marcombo.

González, Y. D., & Romero, Y. F. (2012). Patrón Modelo-Vista-Controlador. *Revista Telemática, 11(1)*, 47-57.

Letelier, P., & Penadés, M. C. (2012). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Recuperado a partir de <http://10.22.1.21:8080/jspui/handle/123456789/477>

Liesa Orús, M., Lloret-Gazo, J., & Vázquez-Toledo, S. (2016). Identificación de las fortalezas y debilidades de la competencia digital en el uso de aplicaciones de internet del alumno de primer curso del Grado de Magisterio. En *Rev. complut. educ.* https://doi.org/10.5209/rev_RCED.2016.v27.n2.48409

Luján-Mora, S. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web*. Editorial Club Universitario. Recuperado a partir de <http://rua.ua.es/dspace/handle/10045/16995>

Luján-Mora, S. (2013). De la clase magistral tradicional al MOOC: doce años de evolución de una asignatura sobre programación de aplicaciones web. Recuperado a partir de <http://rua.ua.es/dspace/handle/10045/41439>

Martín, R. C., Gasset, D. I., & Gálvez, I. E. (2013). Inclusión de jóvenes con discapacidad intelectual en la Universidad. *Revista de investigación en educación*, 11(1), 41-57.

Martínez, D. A. A. (2012). *PHP básico y práctico*. Instituto Tecnológico Metropolitano. Memoria.pdf. (s. f.).

METODOLOGIAS DE DESARROLLO DE SOFTWARE by CAROLINE MENDOZA PLAZA - issuu. (2019). Recuperado 20 de marzo de 2019, a partir de https://issuu.com/carolinemendozaplaza/docs/metodologias_de_desarrollo_de_softw

Nuestra historia. (2019). Recuperado 21 de mayo de 2019, a partir de <http://intranet.uniss.edu.cu/nuestra-historia/>

Pineda, S., & Cecibel, L. (2017). Desarrollo de software para la gestión de proyectos de estudio de mercado. Recuperado a partir de <http://repositorio.utmachala.edu.ec/handle/48000/10964>

Vidal Ledo, M. J., & Pérez, A. B. A. (2012). Gestión de la información y el conocimiento. *Revista Cubana de Educación Médica Superior*, 26(3), 474-484.

Wikstrom Pujante, E. (2014). Diseño y desarrollo en django de una aplicación web para la edición, creación y gestión de cuestionarios tipo test on-line. Recuperado a partir de <https://e-archivo.uc3m.es/handle/10016/26673>

XP - Jose Joskowicz.pdf. (s. f.).

Analyst: There's a great future in iPhone apps. (2015). Recuperado el 1 de Abril de 2016, de Venture Beat.: <http://venturebeat.com/2008/06/11/analyst-theres-a-great-future-in-iphone-apps/>

Beas, J. M. (2012). Recuperado el 16 de Marzo de 2016, de <http://jmbeas.es/guias/historias-de-usuario/#>

Bourdon, R. (13 de 6 de 2017). *Wamp Server*. Obtenido de <http://www.wampserver.com/en/>

Calderón, A., Dámaris, S., Rebaza, V., & Carlos, J. (2007). *Metodologías Ágiles*.

Cornejo, A. (2006). *Tecnologías de la Información y la Comunicación*. Obtenido de <http://nticsanjo.blogspot.com/p/que-es-el-software-en-computacion-el.html>

Cumba Armijos, P. D., & Barreno Pilco, B. A. (2012). *“Análisis De Python Con Django Frente A Ruby On Rails Para Desarrollo Ágil De Aplicaciones Web. Caso Práctico: Dech.”*. Robamba, Ecuador.

Díaz, O. F. (2013). Recuperado el 6 de Febrero de 2016, de Slideshare.net: <http://es.slideshare.net/ofebles/visual-paradign>

Gallego Vázquez, J. A. (2003). *Desarrollo Web con PHP y MySQL*. España: Grupo Anaya.

García, A. A. (2007). *Desarrollo de herramientas web de gestión docente*. .

García, S. (2015). *La guía definitiva de Django: Desarrolla aplicaciones web de forma rápida y sencilla*. Django Software Corporation.

Gassée, J.-L. (2013). *The Silly Web vs. Native Apps Debate*.

Gauchat, J. D. (Enero de 2012). *El gran libro de HTML5, CSS3 y Javascript*.

González Duque, R. (2012). *Python para todos*. España: Creative Commons Reconocimiento 2.5.

Granados, D. (2016). *Desarrollo de Aplicaciones Web II*. Recuperado el 5 de Febrero de 2016, de Slideshare.net: <http://es.slideshare.net/difagram/desarrollo-de-aplicaciones-web-ii-sesin-01-introduccion>

Gutierrez, J. J. (2006). *¿Que es un Framwork Web?* Recuperado el 5 de Febrero de 2016, de Departamento de Lenguajes y Sistemas informáticos. Universidad de Sevilla.: http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf

Jacobson, I., Booch, G., & Rumbaugh, J. (2006). *El Proceso Unificado de Desarrollo de software. Volumen I*. La Habana: Félix Varela.

Joskowicz, J. (2008). *Reglas y Prácticas en eXtreme Programming*.

Luján Mora, S. (2002). *Programación de APLICACIONES Web: Historia, Principios Básicos y Clientes Web*. España: Editorial Club Universitario.

Marcial, V. F. (2006). *Gestión del conocimiento versus gestión de la información*. Obtenido de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2006000200003

Martínez Méndez, J. (2012). *El salto de la gestión de Información a la Gestión del Conocimiento*. Recuperado el 12 de Febrero de 2016, de <http://www.um.es/gtiweb/fjmm>

Michael Kennedy, C. A. (2017). *Pycharm, Python IDE for professional developers*. Obtenido de <https://www.jetbrains.com/pycharm/?fromMenu>

Muñoz Castillo, V. D., García Barrios, H., & Rubiera Hernández, O. (2015). *Ciencias de la Información* (2, ed., Vols. Vol. 46,).

Peñafiel, M. (2016). *Aplicaciones Web*. Recuperado el 5 de Febrero de 2016, de Scribd.com: <http://es.scribd.com/doc/136052164/APLICACIONES-WEB-pdf#scribd>

Potencier, F. (2016). *El tutorial Jobeet*. Recuperado el 5 de Febrero de 2016, de LibrosWeb.es:

http://librosweb.es/libro/jobeeet_1_4/capitulo_4/la_arquitectura_mvc.html

PremiumSoft™ CyberTech Ltd. (2017). Obtenido de <https://www2.navicat.com/es/products/navicat-premium>

Pressman, R. S. (2012). *Ingeniería de software. Un enfoque práctico*. Recuperado el 12 de Abril de 2016, de Slieshare.net: <http://es.slideshare.net/jdbg16/ingenieria-de-software-un-enfoque-prctico-pressman-5thed>

Anexos

Anexo 1. Historias de Usuario

Historia de Usuario	
Número:	Nombre de Historia de Usuario: Gestionar tipos de Áreas
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Estadística	Iteración Asignada: 1
Prioridad en Negocio: Media	PuntosEstimados: 0.4
Riesgo en Desarrollo: Media	PuntosReales: 0.4
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar los tipos de áreas que existen en la universidad.	
Observaciones: Los tipos de áreas de la universidad solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:	Nombre de Historia de Usuario: Gestionar Locales
Modificación(o extensión) de Historia de Usuario(Nro.yNombre):	
Usuario: Estadística	Iteración Asignada: 1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar los locales que existen en la universidad.	
Observaciones: Los locales de la universidad solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:	Nombre de Historia de Usuario: Gestionar Programas Académicos
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Estadística	Iteración Asignada: 1
Prioridad en Negocio: Media	PuntosEstimados: 0.5
Riesgo en Desarrollo: Media	PuntosReales: 0.4
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar los programas académicos que existen en la universidad.	
Observaciones: Los programas académicos de la universidad solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número: 8	Nombre de Historia de Usuario: Gestionar Usuarios
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en Negocio: Media	Puntos Estimados: 0.4
Riesgo en Desarrollo: Media	Puntos Reales: 0.4
Descripción: El Administrador debe ser capaz de crear y eliminar los Usuarios.	
Observaciones: El administrador puede cambiar la Contraseña al Usuario también. Los usuarios de la universidad solo deben ser gestionados y consultados por los administradores	

Historia de Usuario	
Número:9	Nombre de Historia de Usuario: Gestionar Provincia
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.2
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar las provincias que existen.	
Observaciones: las provincias solo deben ser gestionados y consultados por los administradores	

Historia de Usuario	
Número:10	Nombre de Historia de Usuario: Gestionar Países
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar los países que existen.	
Observaciones: los países solo deben ser gestionados y consultados por los administradores	

Historia de Usuario	
Número:11	Nombre de Historia de Usuario: Gestionar Municipio
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3

Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar los municipios que existen.

Observaciones: Los municipios solo deben ser gestionados y consultados por los administradores.

Historia de Usuario	
Número:12	Nombre de Historia de Usuario: Gestionar categorías académicas
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar las categorías académicas que existen.	
Observaciones: Las categorías académicas solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:13	Nombre de Historia de Usuario: Gestionar categorías científicas
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar las categorías científicas que existen.	
Observaciones: Las categorías científicas solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:14	Nombre de Historia de Usuario: Gestionar categorías docentes
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3

Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar las categorías docentes que existen en la universidad.

Observaciones: Las categorías docentes solo deben ser gestionados y consultados por los administradores.

Historia de Usuario

Número:15 **Nombre de Historia de Usuario:** Gestionar grado de escolaridad

Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):

Usuario: Administrador

Iteración Asignada:1

Prioridad en Negocio: Media

Puntos Estimados: 0.3

Riesgo en Desarrollo: Media

Puntos Reales: 0.3

Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar el grado de escolaridad que existen.

Observaciones: Las categorías académicas solo deben ser gestionados y consultados por los administradores.

Historia de Usuario

Número:16 **Nombre de Historia de Usuario:** Gestionar cargos

Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):

Usuario: Administrador

Iteración Asignada:1

Prioridad en Negocio: Media

Puntos Estimados: 0.3

Riesgo en Desarrollo: Media

Puntos Reales: 0.3

Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar los cargos que existen.

Observaciones: los cargos solo deben ser gestionados y consultados por los administradores.

Historia de Usuario	
Número:17	Nombre de Historia de Usuario: Gestionar graduaciones
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar las graduaciones que existen.	
Observaciones: Las graduaciones solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:18	Nombre de Historia de Usuario: Gestionar grados científicos
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver , crear, modificar y eliminar los grados científicos que existen.	
Observaciones: Los grados científicos solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:19	Nombre de Historia de Usuario: Gestionar función que desempeñará
Modificación(o extensión)de Historia de Usuario(Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los administradores deben ser capaces de ver, crear, modificar y eliminar las funciones a desempeñar que existen.	
Observaciones: Las funciones a desempeñar solo deben ser gestionados y consultados por los administradores.	

Historia de Usuario	
Número:20	Nombre de Historia de Usuario: Mostrar trazas
Usuario: Administrador	Iteración Asignada:1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.15
Descripción: Los administradores deben ser capaces de ver las trazas de las acciones del sistema.	
Observaciones: Las trazas solo deben ser consultadas por los administradores.	

Anexo 2. Tareas de Ingeniería

Tarea de Ingeniería	
Nro. De Tarea: E-1	Nro. Historia de Usuario: 5
Nombre de Tarea: Diseño de la interfaz de la gestión de los tipos de áreas	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los tipos de área.	

Tarea de Ingeniería	
Nro. De Tarea: E-2	Nro. Historia de Usuario: 5
Nombre de Tarea: Implementación de la gestión de los tipos de área.	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los tipos de área.	

Tarea de Ingeniería	
Nro. De Tarea: F-1	Nro. Historia de Usuario: 6
Nombre de Tarea: Diseño de la interfaz de la gestión de los locales.	

Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los locales de la universidad	

Tarea de Ingeniería	
Nro. De Tarea: F-2	Nro. Historia de Usuario: 6
Nombre de Tarea: Implementación de la gestión de los locales.	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los locales	

Tarea de Ingeniería	
Nro. De Tarea: G-1	Nro. Historia de Usuario: 7
Nombre de Tarea: Diseño de la interfaz de gestión los programas académicos	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los programas académicos.	

Tarea de Ingeniería	
Nro. De Tarea: G-2	Nro. Historia de Usuario: 7
Nombre de Tarea: Implementación de la gestión de los programas académicos.	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.30
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los programas académicos.	

Tarea de Ingeniería	
Nro. De Tarea: H-1	Nro. Historia de Usuario: 8
Nombre de Tarea: Diseño de la interfaz Gestionar Usuarios	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita al administrador gestionar los Usuarios.	

Tarea de Ingeniería	
Nro. De Tarea: H-2	Nro. Historia de Usuario: 8

Nombre de Tarea: Implementación de Gestionar Usuarios.	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de Usuarios.	

Tarea de Ingeniería	
Nro. De Tarea: R-1	Nro. Historia de Usuario: 20
Nombre de Tarea: Diseño de la interfaz de Mostrar las trazas de todos los usuarios.	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita al administrador mostrar las trazas de todos los usuarios.	

Tarea de Ingeniería	
Nro. De Tarea: R-2	Nro. Historia de Usuario: 20
Nombre de Tarea: Implementación de Mostrar las trazas o Logs de todos los	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para Mostrar las trazas de todos los usuarios.	

Tarea de Ingeniería	
Nro. De Tarea: I-1	Nro. Historia de Usuario: 9
Nombre de Tarea: Diseño de la interfaz de Gestionar provincia.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las provincias	

Tarea de Ingeniería	
Nro. De Tarea: I-2	Nro.HistoriadeUsuario:9
Nombre de Tarea: Implementación de Gestionar provincia	
Tipo de Tarea: Desarrollo	Puntos Estimados:0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de las provincias.	

Tarea de Ingeniería	
Nro.de Tarea: J-1	Nro. Historia de Usuario: 10
Nombre de Tarea: Diseño de la interfaz Gestionar países	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	

Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los países .

Tarea de Ingeniería	
Nro. De Tarea: J-2	Nro. Historia de Usuario: 10
Nombre de Tarea: Implementación de Gestionar países	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los países	

Tarea de Ingeniería	
Nro. De Tarea: K-1	Nro. Historia de Usuario: 11
Nombre de Tarea: Diseño de la interfaz Gestionar municipios	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los municipios .	

Tarea de Ingeniería	
Nro. De Tarea: J-2	Nro. Historia de Usuario: 11
Nombre de Tarea: Implementación de Gestionar municipios	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los municipios	

Tarea de Ingeniería	
Nro.de Tarea: K-1	Nro. Historia de Usuario: 12
Nombre de Tarea: Diseño de la interfaz Gestionar categorías académicas	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las categorías académicas.	

Tarea de Ingeniería	
Nro. De Tarea: K-2	Nro. Historia de Usuario: 12
Nombre de Tarea: Implementación de Gestionar categorías académicas	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	

Descripción: Se implementará una página para la gestión de las categorías académicas

Tarea de Ingeniería	
Nro.de Tarea: M-1	Nro. Historia de Usuario: 13
Nombre de Tarea: Diseño de la interfaz Gestionar categorías científicas	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las categorías científicas.	

Tarea de Ingeniería	
Nro. De Tarea: M-2	Nro. Historia de Usuario: 13
Nombre de Tarea: Implementación de Gestionar categorías científicas	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de las categorías científicas	

Tarea de Ingeniería	
Nro. De Tarea: N-1	Nro. Historia de Usuario: 14
Nombre de Tarea: Diseño de la interfaz Gestionar categorías docentes	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las categorías docentes.	

Tarea de Ingeniería	
Nro. De Tarea: N-2	Nro. Historia de Usuario: 14
Nombre de Tarea: Implementación de Gestionar categorías docentes	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de las categorías docentes	

Tarea de Ingeniería	
Nro. De Tarea: Ñ-1	Nro. Historia de Usuario: 15
Nombre de Tarea: Diseño de la interfaz Gestionar grado de escolaridad	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10

Programador Responsable: Joan Betancourt Balbin
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los grados de escolaridad

Tarea de Ingeniería	
Nro. De Tarea: Ñ-2	Nro. Historia de Usuario: 15
Nombre de Tarea: Implementación de Gestionar grado de escolaridad	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los grados de escolaridad	

Tarea de Ingeniería	
Nro.de Tarea: O-1	Nro. Historia de Usuario: 16
Nombre de Tarea: Diseño de la interfaz Gestionar cargos	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los cargos	

Tarea de Ingeniería	
Nro.de Tarea: O-2	Nro. Historia de Usuario: 16
Nombre de Tarea: Implementación de Gestionar cargos	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los cargos	

Tarea de Ingeniería	
Nro. De Tarea: P-1	Nro. Historia de Usuario: 17
Nombre de Tarea: Diseño de la interfaz Gestionar graduaciones	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las graduaciones	

Tarea de Ingeniería	
Nro.de Tarea: P-2	Nro. Historia de Usuario: 17
Nombre de Tarea: Implementación de Gestionar graduaciones	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20

Programador Responsable: Joan Betancourt Balbin
Descripción: Se implementará una página para la gestión de las graduaciones

Tarea de Ingeniería	
Nro.de Tarea: Q-1	Nro. Historia de Usuario: 18
Nombre de Tarea: Diseño de la interfaz Gestionar grados científicos	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar los grados científicos	

Tarea de Ingeniería	
Nro.de Tarea: Q-2	Nro. Historia de Usuario: 18
Nombre de Tarea: Implementación de Gestionar grados científicos	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de los grados científicos	

Tarea de Ingeniería	
Nro. De Tarea: R-1	Nro. Historia de Usuario: 19
Nombre de Tarea: Diseño de la interfaz Gestionar función a desempeñar	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.10
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a los administradores gestionar las funciones a desempeñar	

Tarea de Ingeniería	
Nro. De Tarea: R-2	Nro. Historia de Usuario: 19
Nombre de Tarea: Implementación de Gestionar funciones a desempeñar	
Tipo de Tarea: Desarrollo	PuntosEstimados: 0.20
Programador Responsable: Joan Betancourt Balbin	
Descripción: Se implementará una página para la gestión de las funciones a desempeñar	

Anexo 3. Pruebas de Aceptación

Caso de Prueba de aceptación	
Código: A-1	Historia de Usuario: 1
Nombre: Comprobación de los datos de la gestión de Roles por cargo .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los roles por cargo .	
Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.	
Entradas/Pasos de ejecución: <ol style="list-style-type: none">5 Se ejecutará el sistema informático en la pantalla de administración.6 El administrador entrará con su usuario y contraseña.7 Se selecciona la opción “Nomencladores”8 Se elige una de las siguientes opciones:<ol style="list-style-type: none">a. Añadir Nuevo Cargo (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)b. Editar Cargos (Se selecciona el Cargo a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)c. Eliminar Cargos (Se selecciona el Cargo a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)	
Resultado esperado: Rol por cargo Gestionado satisfactoriamente	
Evaluación de la prueba: Prueba satisfactoria	

Caso de Prueba de aceptación

Código:H-1**Historia de Usuario:** 8**Nombre:** Comprobación de los datos de la gestión de **usuarios**.**Descripción:** Se comprobará que la aplicación sea capaz de gestionar correctamente los usuarios.**Condiciones de ejecución:** Debe existir al menos un usuario con el rol de administrador para realizar esta acción.**Entradas/Pasos de ejecución:**

- 1 Se ejecutará el sistema informático.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la “**Gestión de usuarios**”
- 4 Se selecciona la opción “**Usuarios**”
- 5 Se elige una de las siguientes opciones:
 - a. **Añadir Nuevo Usuario** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - b. **Eliminar Usuario** (Se selecciona el **Usuario** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: **Usuario** Gestionado satisfactoriamente**Evaluación de la prueba:** Prueba satisfactoria

Caso de Prueba de aceptación

Código:S-1**Historia de Usuario:** 20**Nombre:** Comprobación de los datos de **Mostrar las trazas de todos los usuarios**.**Descripción:** Se comprobará que la aplicación sea capaz de Mostrar las trazas

de todos los usuarios.

Condiciones de ejecución: Debe existir al menos un usuario con el rol de administrador para realizar esta acción.

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático.
- 2 La Estadística entrará con su usuario y contraseña.
- 3 Se selecciona la “**Gestión de Usuarios**”
- 4 Se selecciona la opción “**Trazas o Logs**”
- 5 Se selecciona el usuario.
- 6 Se selecciona “**Filtrar**”
- 7 Se visualiza las trazas del usuario.

Resultado esperado: Trazas listadas correctamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:I-1

Historia de Usuario: 9

Nombre: Comprobación de los datos de la gestión de **Provincia**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las **provincias**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático en la pantalla de administración.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la opción “**Nomencladores**”
- 4 Se elige una de las siguientes opciones:
 - a. **Añadir Nuevo Provincia** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos

fueron insertados correctamente)

b. Editar Provincia (Se selecciona la **Provincia** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)

c. Eliminar Provincia (Se selecciona la **Provincia** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Provincia Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:J-1

Historia de Usuario: 10

Nombre: Comprobación de los datos de la gestión de **Países**

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los **Países**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático en la pantalla de administración.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la opción "**Nomencladores**"
- 4 Se elige una de las siguientes opciones:
 - a. Añadir Nuevo País** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - b. Editar Países** (Se selecciona el **País** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
 - c. Eliminar País** (Se selecciona el **País** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: País Gestionado satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:K-1

Historia de Usuario: 11

Nombre: Comprobación de los datos de la gestión de **Municipio**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los **municipios**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático en la pantalla de administración.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la opción “**Nomencladores**”
- 4 Se elige una de las siguientes opciones:
 - a. **Añadir Nuevo Municipio** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - b. **Editar Municipio** (Se selecciona el **Municipio** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
 - c. **Eliminar Municipio** (Se selecciona el **Municipio** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Municipio Gestionado satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:L-1

Historia de Usuario: 12

Nombre: Comprobación de los datos de la gestión de Categorías Académicas .
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las Categorías Académicas .
Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.
Entradas/Pasos de ejecución: <ol style="list-style-type: none"> 1 Se ejecutará el sistema informático en la pantalla de administración. 2 El administrador entrará con su usuario y contraseña. 3 Se selecciona la opción “Nomencladores” 4 Se elige una de las siguientes opciones: <ol style="list-style-type: none"> a. Añadir Nueva categoría Académica (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) b. Editar Categoría Académica (Se selecciona la Categoría Académica a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente) c. Eliminar Categoría Académica (Se selecciona la Categoría Académica a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)
Resultado esperado: Categoría Académica Gestionada satisfactoriamente
Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación	
Código: M-1	Historia de Usuario: 13
Nombre: Comprobación de los datos de la gestión de Categoría Científica .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las Categorías Científicas .	
Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.	

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático en la pantalla de administración.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la opción “**Nomencladores**”
- 4 Se elige una de las siguientes opciones:
 - a. **Añadir Nueva Categoría Científica** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - b. **Editar Categoría Científica** (Se selecciona la **Categoría Científica** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
 - c. **Eliminar Categoría Científica** (Se selecciona la **Categoría Científica** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Categoría Científica Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:N-1

Historia de Usuario: 14

Nombre: Comprobación de los datos de la gestión de **Categorías Docentes**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las **Categorías Docentes**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático en la pantalla de administración.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la opción “**Nomencladores**”
- 4 Se elige una de las siguientes opciones:

- a. **Añadir Nueva Categoría Docente** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
- b. **Editar Categoría Docente** (Se selecciona la **Categoría Docente** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
- c. **Eliminar Categoría Docente** (Se selecciona la **Categoría Docente** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Categoría Docente Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:Ñ-1

Historia de Usuario: 15

Nombre: Comprobación de los datos de la gestión de **Grado de Escolaridad**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los **Grados de Escolaridad**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

- 1 Se ejecutará el sistema informático en la pantalla de administración.
- 2 El administrador entrará con su usuario y contraseña.
- 3 Se selecciona la opción **“Nomencladores”**
- 4 Se elige una de las siguientes opciones:
 - a. **Añadir Nuevo Grado de Escolaridad** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 - b. **Editar Grado de Escolaridad** (Se selecciona el **Grado de Escolaridad** a modificar y se realizan los cambios deseados. Se

inserta en la base de datos y se verifica que los datos han sido insertados correctamente)

- c. Eliminar Grado de Escolaridad** (Se selecciona el **Grado de Escolaridad** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Grado de Escolaridad Gestionado satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:O-1

Historia de Usuario: 16

Nombre: Comprobación de los datos de la gestión de **cargos**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los **Cargos**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

1. Se ejecutará el sistema informático en la pantalla de administración.
2. El administrador entrará con su usuario y contraseña.
3. Se selecciona la opción "**Nomencladores**"
4. Se elige una de las siguientes opciones:
 1. **Añadir Nuevo Cargo** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 2. **Editar Cargo** (Se selecciona el **Cargo** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
 3. **Eliminar Cargo** (Se selecciona el **Cargo** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Cargo Gestionado satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación	
Código: P-1	Historia de Usuario: 17
Nombre: Comprobación de los datos de la gestión de Graduaciones .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las Graduaciones .	
Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.	
Entradas/Pasos de ejecución:	
<ol style="list-style-type: none"> 5. Se ejecutará el sistema informático en la pantalla de administración. 6. El administrador entrará con su usuario y contraseña. 7. Se selecciona la opción “Nomencladores” 8. Se elige una de las siguientes opciones: <ol style="list-style-type: none"> 1. Añadir Nueva Graduaciones (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente) 2. Editar Graduaciones (Se selecciona la Graduación a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente) 3. Eliminar Graduaciones (Se selecciona la Graduación a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos) 	
Resultado esperado: Graduaciones Gestionada satisfactoriamente	
Evaluación de la prueba: Prueba satisfactoria	

Caso de Prueba de aceptación	
Código: Q-1	Historia de Usuario: 18
Nombre: Comprobación de los datos de la gestión de Grados Científicos .	
Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente los Grados Científicos .	
Condiciones de ejecución: Debe existir al menos un administrador para realizar	

esta acción.

Entradas/Pasos de ejecución:

9. Se ejecutará el sistema informático en la pantalla de administración.
10. El administrador entrará con su usuario y contraseña.
11. Se selecciona la opción “**Nomencladores**”
12. Se elige una de las siguientes opciones:
 1. **Añadir Nuevo Grado Científico** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
 2. **Editar Grado Científico** (Se selecciona el **Grado Científico** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
 3. **Eliminar Grado Científico** (Se selecciona el **Grado Científico** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Grado Científico Gestionado satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria

Caso de Prueba de aceptación

Código:R-1

Historia de Usuario: 19

Nombre: Comprobación de los datos de la gestión de **Función a Desempeñar**.

Descripción: Se comprobará que la aplicación sea capaz de gestionar correctamente las **Funciones a Desempeñar**.

Condiciones de ejecución: Debe existir al menos un administrador para realizar esta acción.

Entradas/Pasos de ejecución:

13. Se ejecutará el sistema informático en la pantalla de administración.
14. El administrador entrará con su usuario y contraseña.
15. Se selecciona la opción “**Nomencladores**”

16. Se elige una de las siguientes opciones:

- 1. Añadir Nueva Función a Desempeñar** (Se llenarán los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos fueron insertados correctamente)
- 2. Editar Función a Desempeñar** (Se selecciona la **Función a Desempeñar** a modificar y se realizan los cambios deseados. Se inserta en la base de datos y se verifica que los datos han sido insertados correctamente)
- 3. Eliminar Función a Desempeñar** (Se selecciona la **Función a Desempeñar** a eliminar y se elimina. Se verifica que fue eliminado correctamente de la base de datos)

Resultado esperado: Función a Desempeñar Gestionada satisfactoriamente

Evaluación de la prueba: Prueba satisfactoria