



UNIVERSIDAD DE SANCTI SPÍRITUS
José Martí Pérez

UNIVERSIDAD DE SANCTI SPÍRITUS “JOSÉ MARTÍ PÉREZ”

FACULTAD DE CIENCIAS TÉCNICAS

CARRERA INGENIERÍA INFORMÁTICA

**APLICACIÓN WEB PARA AUTOMATIZAR LA ADMINISTRACIÓN DE LA
BIBLIOTECA Y EL CONTROL DE LA RESIDENCIA ESTUDIANTIL EN LA
UNIVERSIDAD DE CIENCIAS MÉDICAS EN SANCTI SPÍRITUS**

Trabajo de diploma en opción al título de Ingeniero Informático

Autor: Humberto González Padilla

Tutor: Ing. Vladimir Gómez Fumero

Consultante: Ing. Juan Antonio González Herrera

Sancti Spíritus

Junio 2017

“Año 59 de la Revolución”

PENSAMIENTO

Piensa diferente, porque las personas que están tan locas como para creer que pueden cambiar al mundo, son los que lo logran.

Steve Jobs

AGRADECIMIENTOS

A mis padres, a quienes amo y debo todo lo que soy.

A mi tía Lisbeidy, mi otra madre

A mi hermana Suleny, que la amo.

A mis abuelos, los adoro.

Especialmente a mi hermano Serafín, quién ha estado todo el tiempo conmigo ayudándome a cumplir este sueño.

A mi familia y amigos que de una forma u otra siempre estuvieron brindándome su apoyo incondicionalmente.

A mi tutor y consultante quienes siempre me apoyaron y aconsejaron.

A todos mis maestros y profesores, sin ellos no hubiese llegado hasta aquí.

A mis profesoras y amigas, Lissette y Melba.

A todos los que me comprendieron y apoyaron, de corazón;

GRACIAS

RESUMEN

Las Tecnologías de la Informática y las Comunicaciones (TIC) han hecho posible el desarrollo acelerado de los países por medio de la automatización y digitalización de los procesos, esto hace que sean aplicables a cada uno de los sectores. En Cuba las TICs tienen un trato especial y están en constante fusión con cada uno de los sectores del país a medida que se van desarrollando. La salud en Cuba es uno de los sectores más privilegiados ya que cuenta con varios recursos para su desarrollo, la informática médica realiza grandes proyectos para hacer más fluido el sistema de trabajo del Ministerio de Salud Pública (MINSAP), uno de los cuales es el Sistema Integral de Información (SII) que se encarga de gestionar la información del MINSAP vía web; como sistema integral, este abarca todas las instituciones de salud del país, el caso que ocupa este trabajo tiene lugar en la Universidad de Ciencias Médicas en Sancti Spíritus (UCM). Este proyecto se encargará de realizar los módulos correspondientes a la biblioteca y la residencia estudiantil de la UCM para automatizar los procesos de administración de los mismos, los cuales se integran al SII y complementan el sistema. El proyecto es desarrollado con el *framework* Django, en el lenguaje de programación *Python* y siguiendo la metodología de *Extreme Programming (XP)*. Con el desarrollo de dicho proyecto se espera conseguir una mayor concentración de la información para que pueda ser usada por los usuarios con disímiles propósitos de gestión.

ABSTRACT

Information and Communication Technologies (ICT) have made possible to accelerate the development of countries through the automation and digitization of processes, making them applicable to each sector. In Cuba ICTs have a special treatment and are in constant fusion with each one of the sectors of the country as they are developed. Health in Cuba is one of the most privileged sectors, it has several resources for its development, medical informatics makes big projects to make more fluid the work system of the Ministry of Public Health (MINSAP), one of which is the Integral Information System (SII) that is responsible for managing MINSAP information through the web; as an integral system, this covers all the health institutions of the country, the case that occupies this work takes place in the University of Medical Sciences in Sancti Spíritus (UCM). This project will be responsible for making the modules corresponding to the library and the student residence of the UCM to automate the administration processes of the same, which are integrated into the SII and complement the system. The project is developed with the Django framework, in the programming language Python and following the methodology of Extreme Programming (XP). With the development of this project is expected to achieve a greater concentration of information so that it can be used by users with different management purposes.

CONTENIDO

Pensamiento	I
Agradecimientos	II
Resumen.....	III
<i>Abstract</i>	IV
Introducción.....	1
CAPÍTULO 1: Fundamentos teóricos-metodológicos que sustentan la elaboración de una aplicación web para automatizar la administración de la biblioteca y el control de la residencia estudiantil.....	6
1.1 Antecedentes	6
1.1.1 Historia	8
1.1.2 Sistema Integral de Información (SII)	9
1.1.3 Administración de la biblioteca.....	9
1.1.4 Administración de la residencia estudiantil	10
1.2 Tecnologías Informáticas utilizadas en el proceso de diseño e implementación de la aplicación web	10
1.2.1 Aplicaciones informáticas	10
1.2.2 Aplicaciones Desktop.....	11
1.2.3 Aplicaciones Web	11
1.2.4 Metodología para el desarrollo de software <i>eXtreme Programing (XP)</i>	12
1.2.5 Modelo Vista Controlador (MVC) y Modelo Vista Plantilla (MVT)	13
1.3 Tecnologías de desarrollo del servidor	15
1.3.1 Framework	15
1.3.2 Lenguaje de programación Python 2.7.....	19
1.4 Servidor de Aplicaciones (Apache 2 con mod_python).....	20
1.5 El Sistema Gestor de Base de Datos (MySQL).....	20
1.6 Manejo de bases de datos en django.....	21
1.7 Herramientas para el desarrollo del software	22
1.7.1 Interfaces de desarrollo de aplicaciones (JetBrain PyCharm 5.0)	22
1.7.2 Herramientas para el modelado (Navicat Premium)	23
Conclusiones parciales.....	25

CAPÍTULO 2: Descripción de la aplicación web propuesta para automatizar la administración de la biblioteca y el control de la residencia estudiantil en la universidad de ciencias médicas en Sancti Spíritus	26
2.1 Reglas del negocio	26
2.2 Descripción del Proceso del Negocio	26
2.3 Lista de reserva del sistema	27
2.4 Según XP	29
2.5 Historias de Usuarios	29
2.6 Diagrama Entidad Relación de la Base de Datos	31
2.7 Tareas de Ingeniería	33
Conclusiones parciales	43
CAPÍTULO 3: Construcción de la aplicación propuesta para automatizar la administración de la biblioteca y el control de la residencia estudiantil en la universidad de ciencias médicas en Sancti Spíritus	45
3.1 Tratamiento de excepciones y seguridad. Principios de interfaz de usuario. Ayuda del sistema.	45
3.1.1 Tratamiento de excepciones	45
3.1.2 Seguridad	47
3.1.3 Interfaz de usuario	47
3.1.4 Concepción de la ayuda	48
3.2 Fase de pruebas XP, Pruebas de Aceptación.	49
Conclusiones parciales	56
Conclusiones	57
Recomendaciones	58
Bibliografía	59
Anexos	62
Anexo 1: Historias de usuario	62
Anexo 2: Tareas de ingeniería	69
Anexo 3: Pruebas de Aceptación	88

INTRODUCCIÓN

Las Tecnologías de la Información y Comunicación (TIC) no son más que los métodos técnicos para el almacenamiento, recuperación, proceso y comunicación de la información que se han convertido, a una gran velocidad, en parte importante de nuestras vidas. Este concepto que también se llama sociedad de la información se debe principalmente a un invento que apareció en 1969: Internet. Las TIC se desarrollan, entonces, a partir de estos avances científicos producidos en los ámbitos de la informática y las telecomunicaciones(Ortí, 2015).

Prácticamente no hay un solo ámbito de la vida humana que no se haya visto impactada por este desarrollo: la salud, las finanzas, los mercados laborales, las comunicaciones, el gobierno, la productividad industrial, etc. El conocimiento se multiplica más rápido que nunca antes y se distribuye de manera prácticamente instantánea. El mundo se ha vuelto un lugar más pequeño e interconectado. Para bien y para mal, las buenas y las malas noticias llegan antes: los hallazgos de la ciencia, nuevos remedios y soluciones, descubrimientos e innovaciones, pero también las crisis económicas, las infecciones, nuevas armas y formas de control(Ortí, 2015).

Las características que más resaltan en las TIC según (Cabero, 1998) son:

- **Inmaterialidad:** En líneas generales se puede decir que las TIC realizan la creación (aunque en algunos casos sin referentes reales, como pueden ser las simulaciones), el proceso y la comunicación de la información. Esta información es básicamente inmaterial y puede ser llevada de forma transparente e instantánea a lugares lejanos.
- **Interactividad:** Esta es posiblemente la característica más importante de las TIC para su aplicación en el campo educativo. Mediante las TIC se consigue un intercambio de información entre el usuario y el ordenador. Esta característica permite adaptar los recursos utilizados a las necesidades y características de los sujetos, en función de la interacción concreta del sujeto con el ordenador.
- **Interconexión:** La interconexión hace referencia a la creación de nuevas posibilidades tecnológicas a partir de la conexión entre dos tecnologías. Por ejemplo, la telemática es la interconexión entre la informática y las tecnologías de comunicación, propiciando con ello, nuevos recursos como el correo electrónico, los IRC, etc.
- **Mayor Influencia sobre los procesos que sobre los productos:** Es posible que el uso de diferentes aplicaciones de la TIC presente una influencia sobre los procesos mentales que realizan los usuarios para la adquisición de conocimientos, más que sobre los propios conocimientos adquiridos.

- **Penetración en todos los sectores (culturales, económicos, educativos, industriales...):** El impacto de las TIC no se refleja únicamente en un individuo, grupo, sector o país, sino que, se extiende al conjunto de las sociedades del planeta. Los propios conceptos de "la sociedad de la información" y "la globalización", tratan de referirse a este proceso. Así, los efectos se extenderán a todos los habitantes, grupos e instituciones conllevando importantes cambios, cuya complejidad está en el debate social hoy en día.
- **Innovación:** Las TIC están produciendo una innovación y cambio constante en todos los ámbitos sociales. Sin embargo, es de reseñar que estos cambios no siempre indican un rechazo a las tecnologías o medios anteriores, sino que en algunos casos se produce una especie de simbiosis con otros medios. Por ejemplo, el uso de la correspondencia personal se había reducido ampliamente con la aparición del teléfono, pero el uso y potencialidades del correo electrónico ha llevado a un resurgimiento de la correspondencia personal (Beck, 1998).
- **Tendencia hacia automatización:** La propia complejidad empuja a la aparición de diferentes posibilidades y herramientas que permiten un manejo automático de la información en diversas actividades personales, profesionales y sociales. La necesidad de disponer de información estructurada hace que se desarrollen gestores personales o corporativos con distintos fines y de acuerdo con unos determinados principios (Severin, 2013).

Las TIC son un elemento imprescindible para el desarrollo de cualquier país pues todas las ramas de la economía, salud, educación, entre otras, están sujetas a ellas. En Cuba se dan pasos para el ordenamiento de un trabajo continuo, destinado a impulsar el uso y desarrollo de las TIC en cada uno de los centros laborales del país, atendiendo no solo a las grandes ventajas que posee el uso de estas, sino también con el fin de incorporarse en el mercado a nivel mundial. Una de las organizaciones interesadas en la utilización de las TIC, existente hoy en el país, es la Dirección Provincial de Salud en Sancti Spíritus.

En nuestro país la informática médica se encuentra en un lugar privilegiado y se realizan proyectos de informatización de alto nivel. La dirección nacional de salud pública brinda facilidades a sus trabajadores para la conectividad a internet y para los servicios de correo electrónico internacional y cuenta con líneas de desarrollo de software en pos de la informatización de la institución. Como parte del proceso de informatización, el departamento informático de Salud Pública Provincial de Sancti Spíritus, perteneciente al Ministerio de Salud

Pública (MINSAP) desarrolla varias soluciones informáticas con el fin de automatizar los procesos que se desarrollan en las instituciones de salud de la provincia.

El Sistema Integral de Información (SII), es un proyecto vía web encargado de automatizar la gestión de información del MINSAP en la provincia. El mismo cuenta con varias opciones para optimizar el trabajo de los que laboran en el área de salud (Economía, Estructura, Recursos Humanos, Equipamiento, Inspección, entre otros). Se encuentra en constante actualización y mejora de sus módulos y aspira a poder ser utilizado con carácter nacional. Para el logro de los objetivos de dicho proyecto este debe incluir a todas las instituciones, pues cada una tributa al correcto funcionamiento del ministerio, en el caso que ocupa el presente trabajo la institución en cuestión es la Universidad de Ciencias Médicas (UCM) de Sancti Spíritus.

El SII persigue la homogeneidad en el procesamiento de datos y gestión de la información, y como un único MINSAP existirá un único sistema informático que llevará a cabo el control de todos los recursos de forma dinámica y global, lo cual traerá consigo una mayor concentración de la información para el control interno del ministerio.

Para que el SII se considere un sistema integral este debe abarcar la mayor cantidad de áreas por institución posibles, en el presente caso se hace referencia a la biblioteca y la residencia estudiantil de la UCM de Sancti Spíritus como dos sectores fundamentales en el control interno de la institución.

Primeramente, hay que destacar que la biblioteca de la UCM no cuenta con un software interno que facilite el control de los servicios de préstamos y la gestión de la documentación impresa en la biblioteca de dicha universidad. Estos procesos se llevan a cabo mediante plantillas y boletas que testifican el préstamo de un libro u otro documento perteneciente al local, lo cual es engorroso y lento de realizar por parte del personal que lo lleva a cabo, además está expuesto a la pérdida de la boleta o registro de control.

Por otra parte, se puede mencionar el proceso de control de la residencia estudiantil universitaria en el instituto, el cual se realizan en soporte físico ya sea un libro de control o libreta. De esta forma se lleva el control de diferentes elementos necesarios para la residencia estudiantil, dígame los edificios, la capacidad total de cada uno, la cantidad de habitaciones de los mismos y el responsable de cada piso el cual lleva ese registro, para el acceso a este registro de control es necesario acudir al personal que posea dicha información.

Las dos situaciones anteriormente descritas tienen varias dificultades, sin embargo, el problema de la disponibilidad de la información es común en ambas, esto trae como consecuencia la obstaculización del proceso de control de la organización.

Basados en la anterior **Situación Problemática** descrito se plantea el siguiente **Problema de Investigación**:

- ¿Cómo contribuir a los procesos de administración en la biblioteca y el control de la residencia estudiantil en la Universidad de Ciencias Médicas en Sancti Spíritus?

Para darle respuesta a esta situación problemática se proyecta el siguiente **objetivo general**:

- Desarrollar una aplicación web que automatice los procesos de administración en la biblioteca y el control de la residencia estudiantil en la Universidad de Ciencias Médicas en Sancti Spíritus.

De los objetivos anteriores se generan las siguientes **Preguntas de Investigación**:

1. ¿Cuáles son los fundamentos teóricos-metodológicos que sustentan la elaboración de un software que facilite la administración en la biblioteca y el control de la residencia estudiantil en la Universidad de Ciencias Médicas en Sancti Spíritus?
2. ¿Cómo diseñar una aplicación web que facilite la administración en la biblioteca y el control de la residencia estudiantil en dicha universidad?
3. ¿Cómo implementar una aplicación web que facilite la administración de la biblioteca y el control de la residencia estudiantil en la entidad mencionada anteriormente?

Teniendo como **Tareas de Investigación** las siguientes:

1. Determinar los fundamentos teóricos metodológicos que sustentan la elaboración de un software que facilite la administración en la biblioteca y el control de la residencia estudiantil.
2. Diseñar una aplicación web que facilite la administración en la biblioteca y el control de la residencia estudiantil en la Universidad de Ciencias Médicas en Sancti Spíritus.
3. Implementar una aplicación web que facilite la administración en la biblioteca y el control de la residencia estudiantil en la Universidad de Ciencias Médicas en Sancti Spíritus.

Para dar solución a la problemática existente la investigación se estructuró de la siguiente manera: introducción, tres capítulos, conclusiones, recomendaciones, bibliografías y anexos.

A continuación, se explica brevemente el contenido de los capítulos:

En el **Capítulo 1: FUNDAMENTOS TEÓRICOS-METODOLÓGICOS QUE SUSTENTAN LA ELABORACIÓN DE UNA APLICACIÓN WEB PARA AUTOMATIZAR LA ADMINISTRACIÓN DE LA BIBLIOTECA Y EL CONTROL DE LA RESIDENCIA ESTUDIANTIL**. Se abordan los principales conceptos asociados al dominio del problema. Se describe el objeto de estudio y se incluye un análisis 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 2: DESCRIPCIÓN DE LA APLICACIÓN WEB PROPUESTA PARA AUTOMATIZAR LA ADMINISTRACIÓN DE LA BIBLIOTECA Y EL CONTROL DE LA RESIDENCIA ESTUDIANTIL EN LA UNIVERSIDAD DE CIENCIAS MÉDICAS EN SANCTI SPÍRITUS**. Se describe el modelo del negocio, identificando los procesos involucrados en él y las reglas que lo rigen. Se realiza la descripción del modelo de casos de uso, identificando y describiendo los actores, trabajadores y casos de uso del negocio mediante el diagrama de casos de uso y el diagrama de actividades. Además de la obtención del modelo del sistema, partiendo de los requerimientos funcionales y no funcionales, utilizando para ello el Lenguaje Unificado de Modelado (UML). Se definen los diferentes tipos de usuario y el rol que desempeña cada uno, describiendo la interacción de los mismos con el sistema a partir de los diagramas de casos de uso y su descripción textual.

En el **Capítulo 3: CONSTRUCCIÓN DE LA APLICACIÓN PROPUESTA PARA AUTOMATIZAR LA ADMINISTRACIÓN DE LA BIBLIOTECA Y EL CONTROL DE LA RESIDENCIA ESTUDIANTIL EN LA UNIVERSIDAD DE CIENCIAS MÉDICAS EN SANCTI SPÍRITUS**. 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 1: FUNDAMENTOS TEÓRICOS-METODOLÓGICOS QUE SUSTENTAN LA ELABORACIÓN DE UNA APLICACIÓN WEB PARA AUTOMATIZAR LA ADMINISTRACIÓN DE LA BIBLIOTECA Y EL CONTROL DE LA RESIDENCIA ESTUDIANTIL

Introducción

El presente capítulo contiene la fundamentación teórica 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 y análisis de las tecnologías, lenguajes y herramientas disponibles, determinando cuáles van a ser las utilizadas en el desarrollo del sistema.

1.1 Antecedentes

“Los sistemas de automatización de bibliotecas se consolidaron a finales de la década de los años 1970. En los albores de los años 1980 se establecieron las bases del concepto de sistema integrado. Estos sistemas para la automatización de bibliotecas surgieron como una evolución de los sistemas monofuncionales que se emplearon hasta finales de los años 1970, los cuales tenían por objetivo resolver el problema de la gestión mecánica de funciones que suponían un mayor costo de recursos humanos a las grandes bibliotecas. A partir de la década de los años 1980, se comenzó a considerar el momento de los sistemas integrados, completos, centrados y únicos” (Ó. A. Navarrete & Yáñez, 2013).

“Para lograr la integración de las tareas bibliotecarias en un solo sistema compartiendo una misma base de datos, se desarrollaron varios intentos. Uno de los primeros esfuerzos sistemáticos en la automatización de bibliotecas comenzó en la Universidad de Chicago en 1965, cuando el Dr. Herman H. Fussler, director de la Biblioteca de la Universidad de Chicago, presentó una propuesta a la Fundación Nacional de Ciencias para desarrollar un enfoque integrado, basado en bases de datos bibliográficas en computadora” (D. Ó. A. Navarrete & Aguilar, 2014).

“Los sistemas integrales de automatización actuales están diseñados para facilitar la gestión de las bases de datos a través de sus módulos, que abarcan las diferentes actividades que se llevaban a cabo dentro de ellas” (Mostaccio & Engler, 2010).

“Algunos de los sistemas propietarios tienen costos elevados y a veces dificulta su adquisición. Por esta y otras razones la tendencia actual se enmarca en una clara evolución hacia los sistemas de gestión basados en software libre (el *software* libre no es más que el conjunto de *softwares* que por elección de su autor puede ser utilizado libremente con cualquier fin y redistribuido con o sin cambios o mejora sin efectuar ningún tipo de pago). Se debe especialmente a las atractivas opciones a las que tiene acceso el usuario sobre el programa, amparadas por este tipo de licencia: libertad de usar el programa en distintos ordenadores, libertad de estudiar el funcionamiento del programa y adaptarlo a sus necesidades, libertad de distribuir copias y libertad de mejorar el programa” (Gelguero, 2011).

Existen diversas opciones de *softwares* libre que están centrados en la integración de gestión de bibliotecas, de las más destacadas se encuentran las siguientes:

- **Koha** destaca por ser el primer Sistema Integrado de Gestión bibliotecaria (SIGB) de código fuente abierto, liberado bajo la licencia GPL (Licencia Pública General). Fue creado en 1999 por Katipo Communications para la Horowhenua Library Trust en Nueva Zelanda. Reúne las características propias de un SIGB, permitiendo al bibliotecario manejar la mayoría de los procedimientos administrativos de una biblioteca, y además proveer a los visitantes con un catálogo público para la consulta de ejemplares y circulación (Gelguero, 2011).
- **PMB** es un SIGB basado en un servidor HTTP, una base de datos y PHP. Está realizado conforme con las normas y estándares establecidos de biblioteconomía, respetando la norma UNIMARC. Es portátil y funciona sobre todas las plataformas: Windows, MacOs, Linux y más generalmente todo UNIX. Incorpora módulos para catalogación, circulación, administración, publicaciones periódicas, adquisiciones, edición, tesoro, control de autoridades, estadísticas de uso, difusión selectiva de información y OPAC. Está adaptado para las redes de centros de documentación o de bibliotecas (Chitasko, 2013) .
- **OpenBiblio** es una aplicación libre bajo licencia de GNU para la gestión integral de bibliotecas a través de web escrito en PHP, multiplataforma, utilizado tanto en entorno Linux como en Microsoft Windows. Creado por Dave Stevens en 2002. Su objetivo principal es ofrecer un sistema para la gestión de bibliotecas que sea fácil de usar, que esté bien documentado y que sea fácil de instalar. Openbiblio fue diseñado cubriendo las funciones típicas que se piden a la mayor parte de las bibliotecas escolares y públicas.

Está indicado para la automatización de pequeñas y medianas bibliotecas (Arzamendi, 2003).

1.1.1 Historia

“A inicios de la década de los años 90 se inician proyectos de automatización por el Centro Nacional de Información de Ciencias Médicas (CNICM). Desde las primeras etapas se combinó el desarrollo de redes locales y una red amplia que permitiera la vinculación del conjunto de instituciones del sistema entre sí y con otras redes nacionales e internacionales” (Infomed, 2016).

“El proyecto de desarrollar una red para mejorar los servicios de información en el país se da en el contexto de una revolución en esta esfera a nivel internacional, un potencial humano con calificación acumulada durante más de 20 años, la existencia de determinada infraestructura y organización, así como la decisión política de no abandonar los esfuerzos por desarrollar los servicios de información y los esfuerzos para el desarrollo. En este contexto, el 18 de diciembre de 1992 se comenzó a desarrollar en el CNICM una red de computadoras llamada Red Electrónica de Información en Salud (Infomed). Su objetivo entonces era facilitar el intercambio de información entre los profesionales, los académicos, los investigadores y los funcionarios del Sistema Nacional de Salud cubano” (Infomed, 2016).

El surgimiento de esta red vino, por una parte, a dar respuesta a la difícil situación que sufría el país por la aguda crisis económica que lo afectaba desde finales de 1989, la cual conllevó muchas dificultades para adquirir y diseminar la información científico médica y, por otra parte, a posibilitar la asimilación gradual de las modernas tecnologías que se estaban imponiendo respecto a la generación y el uso de nuevos productos y servicios de información. Por ello, desde el mismo momento en que Infomed comenzó a funcionar se orientaron los esfuerzos a ampliar y fortalecer su infraestructura, con una estrategia de progresiva incorporación de servicios soportados en esas tecnologías, que ofrecen alternativas relativamente económicas para diseminar información a muchos grupos de personas con intereses comunes.

El proyecto de desarrollo de la Red Infomed, se ha caracterizado por el uso de las tecnologías de la información y la comunicación con una visión social y a partir del desarrollo de las capacidades locales. Fue la primera red nacional cubana que utilizó el

sistema operativo GNU/Linux en todos sus servidores, facilitando el acceso a contenidos nacionales y servicios adecuados a su sistema de salud (Infomed, 2016).

La voluntad política y el apoyo del Gobierno cubano y la colaboración recibida, tanto a escala nacional como internacional, han constituido factores clave para la creación y desarrollo de Infomed.

En una nueva etapa de desarrollo de la red, Infomed 2.0 es el nombre que sintetiza la propuesta de asumir una filosofía de trabajo, que se sustente cada vez más en el trabajo colectivo de sus miembros.

“Como parte del proceso de informatización, el departamento informático de Salud Pública Provincial de Sancti Spíritus, perteneciente al Ministerio de Salud Pública (MINSAP), desarrolla varias soluciones informáticas con el fin de automatizar el funcionamiento de las instituciones de salud en la provincia. Entre los proyectos de desarrollo podemos mencionar el Sistema Integral de Información (SII), para la automatización de los procesos informáticos en la provincia” (Infomed, 2016).

1.1.2 Sistema Integral de Información (SII)

El SII es un proyecto web que se encuentra en desarrollo por el Departamento de Informática de Salud Provincial en Sancti Spíritus. El sistema responde a las necesidades de concentración de información y automatización de los procesos del MINSAP. El mismo cuenta con una serie de módulos que se integran a él de forma independiente, interconectados a una misma base de datos centralizada. El SII pone a disposición de los usuarios todo tipo de información con respecto al MINSAP en dependencia de los permisos de cada cual, esto permite una gran disponibilidad de la información a la hora de llevar a cabo procesos de gestión y control de cualquier entidad dentro del ministerio.

1.1.3 Administración de la biblioteca

El primero de los módulos a desarrollar es el de los procesos administrativos de la biblioteca. La cual no cuenta con un sistema informático para llevar a cabo el control de los libros, revistas y tesis que se encuentran físicamente en dicha área.

La aplicación permitirá que los procesos engorrosos de búsqueda de libros, revistas y tesis, sea más eficiente ya que la base de datos almacena la localización exacta de estos medios. Además, la gestión de los préstamos se encuentra disponible en una misma interfaz que brinda información de las devoluciones, los atrasos, los vigentes y la adición de uno nuevo, manteniendo la plantilla oficial de los préstamos.

1.1.4 Administración de la residencia estudiantil

El segundo de los módulos a desarrollar es el de la residencia estudiantil. Este proceso comienza con la gestión de cada edificio con que cuentan las instalaciones de la UCM, partiendo desde su estado constructivo hasta la cantidad de pisos y habitaciones con que cuenta. Posteriormente se prosigue con las capacidades disponibles por habitación para la ubicación del becado, sin embargo, la selección de la ubicación del estudiante se realizará según la distribución que haya hecho el Jefe de Beca con la información que obtuvo del estudiante en la secretaría (desde el módulo de secretaría)

1.2 Tecnologías Informáticas utilizadas en el proceso de diseño e implementación de la aplicación web

1.2.1 Aplicaciones informáticas

“Una aplicación informática es un tipo de software que permite al usuario realizar uno o más tipos de trabajo. Son, aquellos programas que permiten la interacción entre usuario y computadora (comunicación), dando opción al usuario a elegir opciones y ejecutar acciones que el programa le ofrece existen innumerable cantidad de tipos de aplicaciones. Los procesadores de texto y las hojas de cálculo son ejemplos de aplicaciones informáticas, mientras que los sistemas operativos o los programas de utilidades (que cumplen tareas de mantenimiento) no forman parte de estos programas. Las aplicaciones pueden haber sido desarrolladas a medida (para satisfacer las necesidades específicas de un usuario) o formar parte de un paquete integrado (como el caso de *Microsoft Office*)” (Jiménez, 2015).

1.2.2 Aplicaciones Desktop

Una aplicación de escritorio es aquella que se encuentra instalado en el ordenador o sistema de almacenamiento (USB) y podemos ejecutarlo sin internet en nuestro sistema operativo, al contrario que las aplicaciones en la nube que se encuentran en otro ordenador (servidor) al que accedemos a través de la red o internet a su software. Será un programa encargado de realizar la funcionalidad del software implementado que instalaremos 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 podremos incorporar todos los controles de escritorio y todos los eventos asociados a ellos. Como principal desventaja tendremos la gestión de actualizaciones que obligará a actualizar todos los programas instalados en cada puesto de la empresa cuando implementemos evoluciones o corriamos fallos. Esto obligará a diseñar un sistema automático de gestión de actualizaciones ya que un usuario con un software obsoleto puede dañar la base de datos. Otra desventaja importante es la escasa portabilidad ya que, si lo implementamos para un entorno Windows, solo en equipos de ese tipo funcionará y no podremos usarla en una *tablet* o un teléfono.

1.2.3 Aplicaciones Web

Una **aplicación web** tiene varias acepciones según los especialistas del tema y a continuación se presentan diversos criterios:

“Una aplicación web (*web-based application*) 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.” (Mora, 2002)

“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 individual es enviada al cliente como un documento estático, pero la secuencia de páginas provee de una experiencia interactiva.” (Peñafiel, 2016)

De forma general son aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador, o sea, una aplicación

de software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador. Debido a su contenido dinámico 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 el internet, una aplicación web es la opción más adecuada y la seleccionada para resolver nuestro problema de investigación. También debemos 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 debemos tener en cuenta que los módulos desarrollados en esta tesis tienen como finalidad integrarse al SII, del cual hemos hablado previamente y hemos 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.

1.2.4 Metodología para el desarrollo de software *eXtreme Programming (XP)*

“Las metodologías de desarrollo de software son marcos o modelos de trabajos que se utilizan para construir, planificar y controlar el proceso de desarrollo de sistemas. Hoy en día existen infinitudes de metodologías para desarrollar software. Entre ellas encontramos las Metodologías Tradicionales, las Metodologías Iterativas/Evolutivas, las Metodologías basadas en Tecnología Web, y las Metodologías Ágiles” (Dayana & C., 2014).

“Existen numerosas propuestas de metodología para desarrollar software. Tradicionalmente estas metodologías se centran en el control del proceso, estableciendo rigurosamente las actividades, herramientas y notaciones al respecto, dado estas reglas estas metodologías se caracterizan por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas” (Dayana & C., 2014).

“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)

“XP es una metodología ágil para pequeños y medianos equipos, desarrollando software cuando los requerimientos son ambiguos o rápidamente cambiantes. A diferencia de los procesos tradicionales para desarrollar software, XP asume el cambio como algo natural, y que, indefectiblemente, en alguna etapa de un proyecto sucede. En XP se realiza el software que el cliente solicita y necesita, en el momento que lo precisa, alentando a los programadores a responder a los requerimientos cambiantes que plantea el cliente en cualquier momento. Esto es posible porque está diseñado para adaptarse en forma inmediata a los cambios, con bajos costos asociados, en cualquier etapa del ciclo de vida” (Calabria & Pìriz, 2003).

Al haber seleccionado esta metodología de trabajo se prosigue a determinar la arquitectura, lenguajes, tecnologías y herramientas a utilizar para el desarrollo del sistema integrado.

Después del análisis comparativo de algunas metodologías de desarrollo de software, ágiles y tradicionales, determinamos utilizar *Extreme Programming* (XP) por las características que mostramos a continuación:

- Se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.
- Se aplica de manera dinámica durante el ciclo de vida del software.
- Es capaz de adaptarse a los cambios de requisitos.
- Los individuos e interacciones son más importantes que los procesos y herramientas.
- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.

1.2.5 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.

1.2.5.1 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:

- 1- La capa del modelo define la lógica de negocio (la base de datos pertenece a esta capa).
- 2- La vista es lo que utilizan los usuarios para interactuar con la aplicación (los gestores de plantillas pertenecen a esta capa).
- 3- 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.”

En el *Framework* Django este modelo de arquitectura se convierte en un nuevo modelo que describiremos a continuación: el modelo-vista-plantilla.

1.2.5.2 Modelo Vista Plantilla (MVT)

El patrón llamado MVT en django hace referencia a modelo vista *template*, y es sencillo de entender, el modelo sigue siendo el modelo, en este caso, la vista no es una vista, sino que más bien es un controlador que se llama vista, y el *template* son las vistas del MVC, es decir, los formularios van en *template*, los formularios hacen peticiones a las vistas’, y las vistas obtienen datos de los modelos, es así de sencillo.

En el patrón de diseño MVT,

- *M* significa "*Model*" (Modelo), la capa de acceso a la base de datos. Esta capa 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 (Matías, 2016).
- *V* significa "*View*" (Vista), la capa de la lógica de negocios. Esta capa 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 (Matías, 2016).

- *T* significa "*Template*" (Plantilla), la capa de presentación. Esta capa contiene las decisiones relacionadas a la presentación: como algunas cosas son mostradas sobre una página web o otro tipo de documento (Matías, 2016).

1.3 Tecnologías de desarrollo del servidor

1.3.1 Framework

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

“En general, con el término *framework*, se hace referencia a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un *framework* se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta” (Gutiérrez, 2015).

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones

“Un framework Web, por tanto, podemos definirlo como un conjunto de componentes (por ejemplo clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web” (Gutiérrez, 2015).

Algunas de las ventajas de utilizar un *framework* son:

1. Estructura predefinida de la aplicación

“El programador no necesita plantearse la estructura global de la aplicación, ya que esta es proporcionada por el propio *framework*. Esto tiene la ventaja de que pasado un tiempo, si tenemos que tocar algo en la aplicación, sabremos donde encontrar el archivo en cuestión de forma rápida” (acensTechnologies, 2012).

2. Código altamente testado

“Todo el código que forma parte del *framework* está altamente probado, lo que garantiza el buen funcionamiento del mismo. Se podría desarrollar esas mismas funcionalidades, pero

nunca podremos garantizar ese nivel de testeo que ofrecen los *frameworks*” (acensTechnologies, 2012).

3. Comunidad de usuarios detrás de cada *framework*.

“La gran mayoría de los *frameworks* tienen detrás a una amplia comunidad de usuarios, de los cuales muchos ayudan en su desarrollo o creando extensiones con funcionalidades extra que podremos utilizar de forma sencilla sin tener que desarrollarlas por nuestra cuenta” (acensTechnologies, 2012).

1.3.1.1 Django

“Django nació naturalmente de aplicaciones de la vida real escritas por un equipo de desarrolladores Web en Lawrence, Kansas. Nació en el otoño boreal de 2003, cuando los programadores Web del diario *Lawrence Journal-World*, *Adrian Holovaty* y *Simon Willison*, comenzaron a usar Python para crear sus aplicaciones. Lo liberaron en julio de 2005 y lo llamaron Django, por el guitarrista de jazz “Django Reinhardt”.”(Kaplan-Moss, 2008)

“Django es un *framework* de desarrollo de aplicaciones web libre, de código abierto que promueve las buenas prácticas de diseño y permite el desarrollo rápido. Se trabaja para optimizar su desarrollo de aplicaciones y ayuda a garantizar un producto final eficiente, extensible y sostenible”.(Kaplan-Moss, 2008)

La estabilidad de este proyecto, el rendimiento y la comunidad han crecido enormemente en los últimos diez años desde la creación del *framework*. Detallados tutoriales y buenas prácticas son fáciles de encontrar en la web y en los libros. El *framework* sigue añadiendo nuevas funcionalidades importantes como las migraciones de bases de datos con cada versión.

Django te permite enfocarte en la parte divertida al mismo tiempo que mitiga el esfuerzo de las partes repetitivas. De esta forma, provee un alto nivel de abstracción de patrones comunes en el desarrollo Web, atajos para tareas frecuentes de programación y convenciones claras sobre cómo solucionar problemas. Al mismo tiempo, intenta no entrometerse, dejándote trabajar fuera del ámbito del *framework* según sea necesario. (Kaplan-Moss, 2008)

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.

En el desarrollo de esta aplicación web se integra en conjunto con el *framework* Django otros *framework* para el desarrollo de la interfaz visual y validaciones opcionales llamado Twitter Bootstrap el cual se hace mención en el próximo acápite.

1.3.1.2 Bootstrap

“Bootstrap fue desarrollado por Mark Otto y Jacob Thornton de Twitter, como un marco de trabajo (*framework*) para fomentar la consistencia entre las herramientas internas. Antes de Bootstrap, se usaban varias librerías para el desarrollo de interfaces de usuario, las cuales llevaban a inconsistencias y a una gran carga de trabajo en su mantenimiento”.(Cochran, 2012)

“Twitter Bootstrap es un *framework* o conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript adicionales”.(Mestras, 2013)

De esta forma se muestran las características y ventajas que ofrece este *framework*:

- **Sencillo y ligero:** Puede bastar con un fichero CSS y uno JavaScript.
- **Basado en los últimos estándares de desarrollo de Web:** HTML5, CSS3 y JavaScript/JQuery.
- Plugins de jQuery para validar entrada de datos, visualización tablas, grafos, etc.
- Compatible con todos los navegadores habituales.
- **Abierto:** Publicado en 2011 con licencia Apache.
- **Responsive web design:** Diseño de páginas web para que el usuario las visualice perfectamente en un amplio rango de dispositivos (Navegador en el PC, tableta, Smartphone)
- **Bootstrap 3: Mobile first:** Se ha rediseñado para adaptarse desde el principio a tamaños de pantalla típicos de los móviles

HyperText Markup Language (HTML5)

HTML5 no es una nueva versión del antiguo lenguaje de etiquetas, ni siquiera una mejora de esta ya antigua tecnología, sino un nuevo concepto para la construcción de sitios web y aplicaciones en una era que combina dispositivos móviles, computación en la nube y trabajos en red.(Gauchat, 2013)

“HTML5 provee básicamente tres características: estructura, estilo y funcionalidad. Nunca fue declarado oficialmente, pero, incluso cuando algunas APIs (Interface de Programación de Aplicaciones) y la especificación de CSS3 por completo no son parte del mismo, HTML5 es considerado el producto de la combinación de HTML, CSS y Javascript. Estas tecnologías son altamente dependientes y actúan como una sola unidad organizada bajo la especificación de HTML5. HTML está a cargo de la estructura, CSS presenta esa estructura y su contenido en la pantalla y *Javascript* hace el resto que es extremadamente significativo” (Gauchat, 2013)

Cascading Style Sheets (CSS3)

“CSS es un lenguaje que trabaja junto con HTML para proveer estilos visuales a los elementos del documento, como tamaño, color, fondo, bordes, etc...” (Gauchat, 2013)

“La versión 3 de CSS sigue el mismo camino, pero esta vez con un mayor compromiso. La especificación de HTML5 fue desarrollada considerando CSS a cargo del diseño. Debido a esta consideración, la integración entre HTML y CSS es ahora vital para el desarrollo web y esta es la razón por la que cada vez que mencionamos HTML5 también estamos haciendo referencia a CSS3, aunque oficialmente se trate de dos tecnologías completamente separadas”.(Gauchat, 2013)

Javascript

“Javascript es un lenguaje interpretado usado para múltiples propósitos pero solo considerado como un complemento hasta ahora. Una de las innovaciones que ayudó a cambiar el modo en que vemos Javascript fue el desarrollo de nuevos motores de interpretación, creados para acelerar el procesamiento de código. La clave de los motores más exitosos fue transformar el código Javascript en código máquina para lograr velocidades de ejecución similares a aquellas encontradas en aplicaciones de escritorio. Esta mejorada capacidad permitió superar viejas

limitaciones de rendimiento y confirmar el lenguaje Javascript como la mejor opción para la web”.(Gauchat, 2013)

1.3.2 Lenguaje de programación Python 2.7

“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, 2012)

El 5 de diciembre de 2015, fue lanzado Python 2.7.11 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 argparse 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 conlleva a explicar las tecnologías del lado del servidor en próximos acápite.

1.4 Servidor de Aplicaciones (Apache 2 con mod_python)

Un aspecto inherente a las aplicaciones Web lo constituye su arquitectura distribuida conocida como cliente-servidor. En este epígrafe se describirá las diferentes tecnologías usadas específicamente como servidores para la implementación y despliegue de la aplicación Web desarrollada en el presente trabajo.

“Para el despliegue de una aplicación web es necesario un servidor web encargado de recibir la petición desde un navegador utilizando el protocolo HTTP, localizar el archivo de entrada a la aplicación Web y pasar el control de la aplicación al sistema encargado de ejecutar la petición.” (Valdivia, 2015)

“Django incluye un servidor web ligero que se puede usar mientras se está desarrollando un sitio web. Este servidor de desarrollo vigila el código a la espera de cambios y se reinicia automáticamente, ayudándote a hacer algunos cambios rápidos al proyecto sin necesidad de reiniciar nada.” (Kaplan-Moss, 2008)

Este servidor es recomendable para la etapa de desarrollo y pruebas, pero para la fase de despliegue se decide utilizar el servidor web Apache 2 con *mod_python*, un módulo para este servidor que le permite correr aplicaciones hechas en Python.

“El servidor Apache es desarrollado y mantenido por una comunidad de usuarios bajo la supervisión de la *Apache Software Foundation* dentro del proyecto HTTP Server (httpd). Este presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido. Permite múltiples lenguajes de script tales como PHP, Perl, Tcl, Python. La arquitectura del servidor Apache es muy modular el cual consta de una sección core y diversos módulos que aportan mucha de la funcionalidad que podría considerarse básica para un servidor web”. (Valdivia, 2015)

Este servidor establecerá la conexión para la base de datos pero para la misma se necesita utilizar un lenguaje de programación SQL específico.

1.5 El Sistema Gestor de Base de Datos (MySQL)

“Un sistema gestor de base de datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos. Los usuarios pueden

acceder a la información usando herramientas específicas de consulta y de generación de informes, o bien mediante aplicaciones al efecto”(Valdivia, 2015).

“Inicialmente, MySQL carecía de algunos elementos esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de esto, atrajo a los desarrolladores de páginas web con contenido dinámico, debido a su simplicidad, de tal manera que los elementos faltantes fueron complementados por la vía de las aplicaciones que la utilizan. Poco a poco estos elementos faltantes, están siendo incorporados tanto por desarrolladores internos, como por desarrolladores de software libre”(Alma, Jesús, Yunko, & Goretty, 2014).

Estas son las principales características de este SGBD:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas (*passwords*) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas

Debido a lo anteriormente expuesto y a que el SII usa este SGBD, se es coge el mismo para el desarrollo de la aplicación web en cuestión.

1.6 Manejo de 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 está preparado para soportar las más comunes en el mercado simplemente cambiando la variable `DATABASE_ENGINE` en el `settings.py` (Armijos & Pilco, 2012).”

Antes de la parte de configuración se debe instalar y configurar el 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 necesitamos ir al archivo settings.py en la raíz de nuestro proyecto y cambiar los valores del diccionario DATABASES con los específicos para MySQL.

1.7 Herramientas para el desarrollo del software

Un entorno de desarrollo integrado (IDE), es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI).

Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, PHP, Python, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto (Lidia Fuentes, 2013).

En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto (Lidia Fuentes, 2013).

Las principales características que definen los IDE son:

- Multiplataforma
- Soporte para diversos lenguajes de programación
- Integración con Sistemas de Control de Versiones
- Reconocimiento de Sintaxis
- Extensiones y Componentes para el IDE
- Integración con Framework populares
- Depurador
- Importar y Exportar proyectos
- Múltiples idiomas
- Manual de Usuarios y Ayuda

1.7.1 Interfaces de desarrollo de aplicaciones (JetBrain PyCharm 5.0)

Dentro de las herramientas IDEs utilizadas encontramos 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 tenemos:

- 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.

1.7.2 Herramientas para el modelado (Navicat Premium)

“Navicat Premium es un Administrador de Base de datos rápido, fiable y asequible. Especialmente diseñada para simplificar la gestión de bases de datos y reducir los costes de administración. Con una interfaz gráfica del usuario intuitiva, Navicat le proporciona una manera más fácil de gestionar, diseñar y manipular datos en MySQL, MariaDB, SQL Server, SQLite, Oracle y PostgreSQL”(Gil, 2012)

Características principales de Navicat Premium

- **Diseño de Objetos Inteligentes:** proporciona una Interfaz de Usuario Intuitiva y bien diseñada para crear, modificar y gestionar todos los objetos de la base de datos, como tablas, vistas, funciones / procedimientos, índices, triggers (disparadores) y secuencias. El Diseñador de Tablas ayuda a crear y modificar tablas de la base de datos, permite diseñar y visualizar una sola tabla en la base de datos a la que esté conectado, y facilita

la configuración de opciones avanzadas como las relaciones, restricciones, disparadores y más.

- **Simplifique la Edición de Datos:** Permite navegar y modificar los datos existentes en la base de datos. Al insertar, editar, eliminar datos o copiar y pegar un número de registros en nuestra hoja de balance, como editor de datos , Navicat ejecutará los comandos apropiados (por ejemplo, INSERTAR o ACTUALIZAR) en el servidor de la base de datos. Así no tendrá que escribir más SQL complicados. Además, obtendrá una amplia gama de herramientas de edición para facilitar dicho trabajo, como la búsqueda de clave externa, selector de set o enumerador, filtro de registros y un conjunto de editores asistentes como un editor de memoria con una sintaxis destacada para el valor de un texto grande en datos y un editor hexadecimal útil para datos binarios.
- **Fácil Edición de SQL:** Nos facilita crear, editar, ejecutar consultas y ver los resultados. Con la increíble función de Finalización de Código Contextual, no sólo ayuda a completar la consulta que esté escribiendo, haciendo sugerencias razonables, también auxiliará rápidamente en los errores de código. El Generador de Consultas permite crear y editar consultas visualmente sin conocimientos de SQL. Proporcionando un conjunto de métodos para construir las partes individuales de una instrucción de SELECCION. También, reúne las diferentes partes de una instrucción válida de SQL. Sólo con unos pocos clics, tendrá una consulta completada.
- **Seis bases de datos, un solo usuario:** Navicat es compatible para trabajar con la gran mayoría de los sistemas de base de datos conocidos – MySQL, MariaDB, SQL Server, SQLite, Oracle y PostgreSQL, proporcionando una conectividad nativa directa a sus bases de datos. Brinda una experiencia muy eficiente y de fácil comprensión. Mejora la experiencia del usuario a través de una interfaz de usuario altamente interactiva que une procesos, flujos de trabajo e integración.
- **Diseño de Modelado Inteligente:** Dé una vuelta a sus bases de datos en una representación gráfica utilizando la herramienta de diseño y modelado. Utiliza diagramas de entidad de relación para visualizar estructuras de bases de datos e ilustrar las

relaciones entre los datos, por lo que nos permite modelar, crear bases de datos complejas y comprenderlas de una manera mucho más sencilla. También posee funciones flexibles y fiables de modelización, como la ingeniería inversa de bases de datos existentes, y generar un SQL más accesible.

Como se puede apreciar el software Navicat Premium se consideró como la mejor opción para la gestión y modelado de la base de datos para el presente proyecto debido a la amplia gama de herramientas y facilidades que el mismo nos proporciona.

Conclusiones parciales

Mediante el estudio realizado en el presente capítulo se llega a las siguientes conclusiones parciales:

- El análisis del marco teórico permitió definir a las aplicaciones web como las más adecuadas para el desarrollo de las aplicaciones en cuestión.
- Se determina que la metodología eXtreme Programming (XP) es la metodología ágil más destacada para el desarrollo del proyecto. La cual cuenta con cuatro etapas (planificación, diseño, codificación y pruebas) que la hacen más adecuada para el desarrollo del software.
- Se fijó la utilización del framework Django 1.6 y Bootstrap debido a que es un software libre, de código abierto, con una gran cantidad de bibliografía y de comunidades de desarrollo web.
- Se selecciona el gestor de base de datos *Navicat Premium* y el editor de código *JetBrain PyCharm 5.0* como herramienta para la programación de la aplicación ya que ofrece un alto nivel de integración con Python.

CAPÍTULO 2: Descripción de la aplicación web propuesta para automatizar la administración de la biblioteca y el control de la residencia estudiantil en la universidad de ciencias médicas en Sancti Spíritus

Introducción

Después de analizado el marco teórico referencial y escogidas las herramientas y metodología a utilizar, se está en condiciones de comenzar el desarrollo del siguiente capítulo. En este capítulo 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 el Sistema Informático Integral.
- Los usuarios solo podrán acceder a la información relativa a su nivel de acceso.
- Los estudiantes que accedan el sistema solo pueden visualizar la documentación con la que cuenta la biblioteca.
- Los presentes módulos del SII van a ser administrado por el departamento de informática de la UCM.

2.2 Descripción del Proceso 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 identifican los siguientes procesos de negocio:

- Proceso de administración de la biblioteca.
- Proceso de gestión de residencia estudiantil.

Ambos procesos ocurren en la UCM de Sancti Spíritus.

El proceso de administración de la biblioteca comienza cuando se recibe una solicitud de préstamo por parte de un usuario, posteriormente el responsable del área busca el libro, revista o tesis en la sección correspondiente y una vez que lo encuentre se continua a tomar los datos del usuario para llevar a cabo el préstamo llenándose el modelo 0207 que es la plantilla oficial para justificarlo y llevar un control de los mismos, este proceso se repite en un libro que funciona como registro de control de respaldo en caso de pérdida de la plantilla.

El proceso de gestión de la residencia estudiantil está sujeto a las variaciones que se realicen en la secretaría docente con respecto a la información de cada estudiante nuevo o continuante, el Jefe de Beca se basa en esta información para realizar la distribución de los estudiantes en los edificios, esto se lleva a cabo teniendo en cuenta el año y la especialidad que cursará. Seguidamente se determinan las capacidades disponibles en las habitaciones del edificio, las cuales son conocidas por la persona responsable de cada piso, estos responsables también llevan el control de los medios básico de cada habitación.

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.

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 2.7. Además, se requiere de un servidor de base de datos MySQL para Python 2.7 (versión 1.2.4 o superior). Todo lo anterior para una eficiencia óptima. Del lado del cliente: se requiere un navegador que interprete las funciones básicas de JavaScript, CCS3 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 Según XP

El proceso de desarrollo de software se divide en cuatro tipos de actividades:

- Planificación
- Diseño
- Codificación
- Pruebas

Y entre sus principales artefactos se encuentran:

- Historias de usuarios
- Tareas de Ingeniería
- Pruebas de aceptación

Teniendo en cuenta lo mencionado se detallarán algunos de los artefactos anteriores.

2.5 Historias de Usuarios

“Una historia de usuario describe una funcionalidad que, por sí misma, aporta valor al usuario. Se compone de: una descripción escrita de la historia usada como recordatorio y para planificar, conversaciones acerca de la historia que sirven para aclarar los detalles y un criterio de aceptación (idealmente automatizado) que permita determinar cuándo la historia ha sido completada.”(Valdivia, 2015)

Una vez culminado este proceso fueron creadas un total de 21 historias de usuarios, las cuales se relacionan a continuación en la tabla 2.1, donde se establecen los puntos estimados y los puntos reales de cada historia de acuerdo a la complejidad de cada una.

Iteración	Historias de usuario (número y nombre)		Ptos. Estim.	Ptos. Reales
Seguridad				
1	1	Gestionar Roles y permisos	0.5	0.4
	2	Gestionar usuario	0.5	0.5
	3	Autenticar usuario	0.5	0.5
	4	Libre acceso para estudiantes	0.3	0.3
	5	Cambiar contraseña de usuario	0.5	0.4
	6	Base de Datos	1.5	1.5
Gestionar Datos				
2	7	Gestionar Edificios	0.5	0.3
	8	Gestionar Habitación	0.5	0.3

	9	Gestionar Capacidad	0.5	0.4
	10	Gestionar Becado	0.5	0.3
	11	Gestionar Libros	0.8	0.8
	12	Gestionar Revistas	0.8	0.6
	13	Gestionar Artículos	0.3	0.3
	14	Gestionar Tesis	0.3	0.1
	15	Control de Prestamos Libros	1.5	1.5
	16	Control de Prestamos Revistas	1.5	1
	17	Control de Prestamos Tesis	1.5	1
		Reportes		
3	18	Mostrar Modelo de Control 0207	0.4	0.1
	19	Reporte de préstamos de libros sin devolver	0.1	0.1
	20	Reporte de préstamos de revistas sin devolver	0.2	0.1
	21	Reporte de préstamos de tesis sin devolver	0.2	0.1
	22	Graficas de vía de préstamos de libros	0.1	0.1
	23	Graficas de vía de préstamos de tesis	0.1	0.1
	24	Graficas de vía de préstamos de revistas	0.1	0.1
	25	Reporte de capacidad becas	0.1	0.2
	26	Reporte de mapa de residencia	0.5	0.3
	27	Reporte de por ciento de ocupación	0.1	0.1
	28	Reporte grafico de estado constructivo	0.1	0.1
	29	Listar Trazas de usuarios del sistema	1	0.8

Tabla 2.1: Historias de Usuario

Historia de Usuario	
Número: 1	Nombre Historia de Usuario: Gestionar Roles y Permisos
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Alto	Puntos Reales: 0.5
Descripción: El administrador tendrá la posibilidad de gestionar los roles además de asignarle a estos los permisos de acceso y modificación en el sistema.	
Observaciones:	

Tabla 2.2: Historia de Usuario No.1

Historia de Usuario	
Número: 2	Nombre Historia de Usuario: Gestionar Usuario
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Medio	Puntos Reales: 0.4

Descripción: Permite guardar los diferentes datos sobre los usuarios que trabajan con la aplicación, además se le da una cuenta de usuario y una contraseña para que pueda entrar al software.

Observaciones:

Tabla 2.3: Historia de Usuario No.2

Historia de Usuario	
Número: 3	Nombre Historia de Usuario: Autenticar Usuario
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Todos	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Medio	Puntos Reales: 0.4
Descripción: Permite la entrada a la aplicación para gestionar los datos.	
Observaciones:	

Tabla 2.4: Historia de Usuario No.3

El resto de las historias de usuario se encuentran en el [Anexo 1](#)

2.6 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.

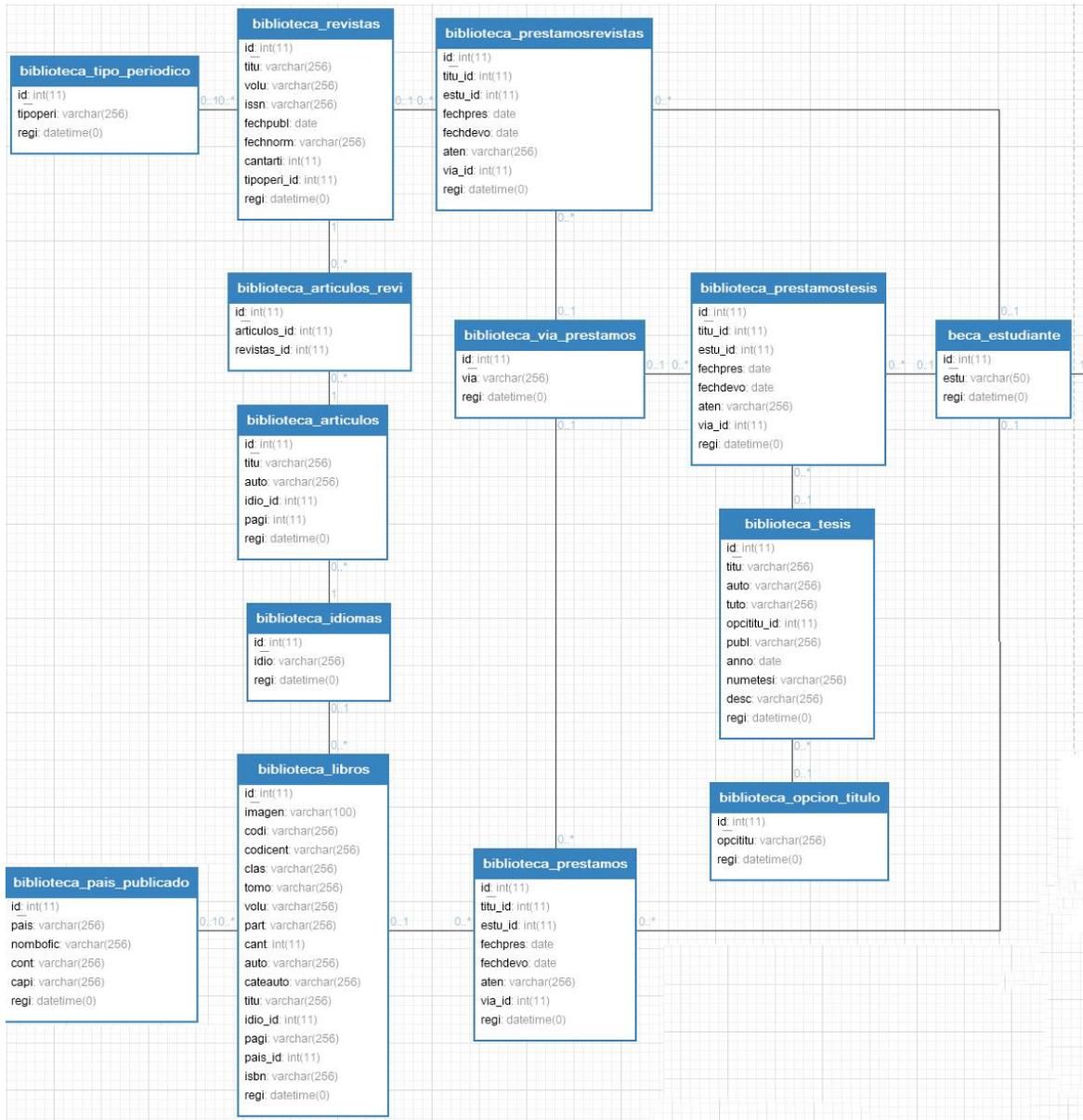


Figura 2.1: Diagrama Entidad Relación para el Módulo Biblioteca.

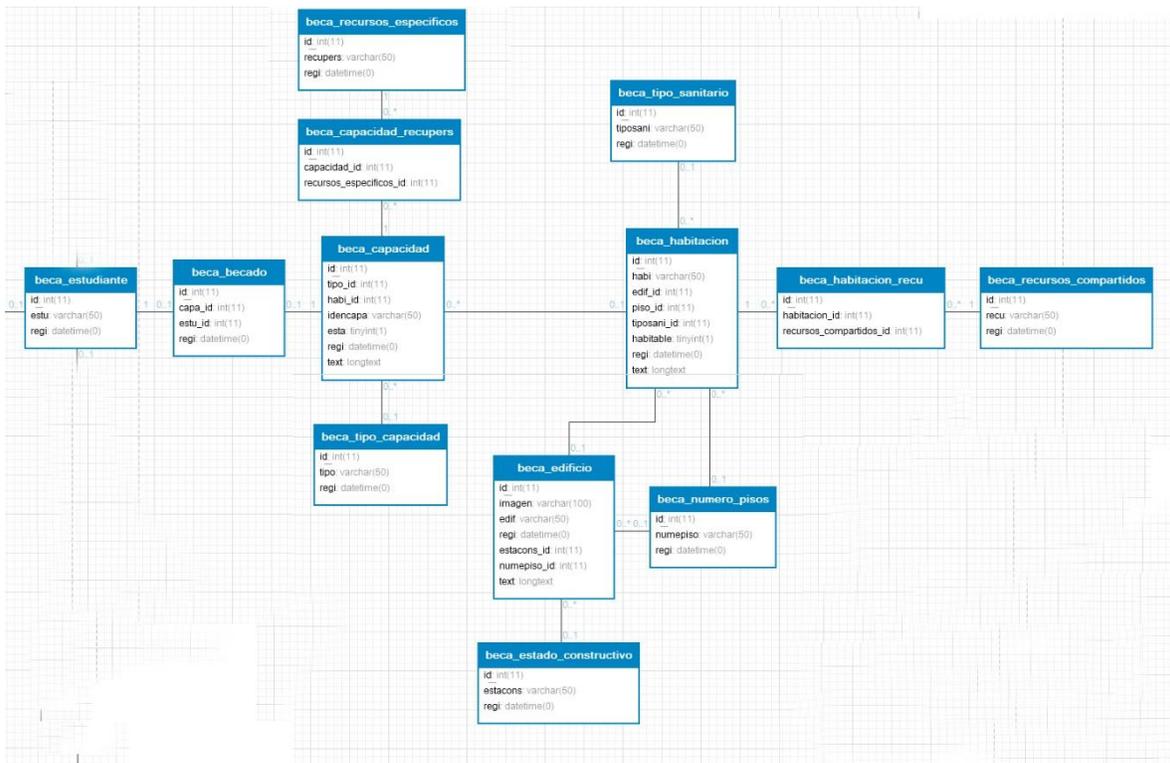


Figura 2.2: Diagrama Entidad Relación para el Módulo Residencia Estudiantil

2.7 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 (ver tabla 2.5).

HU	Tareas de ingeniería		Ptos. Estim.	Ptos. Reales.
1	A-1	Diseño de la interfaz de gestión de roles y permisos	0.5	0.4
	A-2	Implementación de operaciones CRUD para la gestión de usuarios y permisos	0.5	0.5
2	B-1	Diseño de la interfaz de gestión de usuarios	0.3	0.2

	B-2	Implementación de operaciones CRUD para la gestión de usuarios	0.5	0.4
3	C-1	Diseño de la interfaz de autenticación de usuario	0.3	0.1
	C-2	Implementación de la autenticación de usuario	0.4	0.3
4	D-1	Diseño de la interfaz Libre acceso a estudiantes	0.2	0.1
	D-2	Implementación de operaciones para Libre acceso a estudiantes	0.2	0.1
5	E-1	Diseño de la interfaz de Cambiar contraseña de usuario	0.1	0.2
	E-2	Implementación de operaciones CRUD para Cambiar contraseña de usuario	0.2	0.3
6	F-1	Diseño y Confección del Diagrama de Entidad Relación	0.1	0.1
7	G-1	Diseño de la interfaz de Gestionar Edificios	0.3	0.1
	G-2	Implementación de operaciones CRUD Gestionar Edificios	0.3	0.1
8	H-1	Diseño de la interfaz de Gestionar Habitaciones	0.2	0.1
	H-2	Implementación de operaciones CRUD para Gestionar Habitaciones	0.2	0.1
9	I1	Diseño de la interfaz de Gestionar Capacidad	0.1	0.3
	I-2	Implementación de operaciones CRUD para Gestionar Capacidad	0.1	0.2
10	J-1	Diseño de la interfaz de Gestionar Becados	0.1	0.1
	J-2	Implementación de operaciones CRUD para Gestionar Becados	0.2	0.3
11	K-1	Diseño de la interfaz de Gestionar Libros	0.2	0.3
	K-2	Implementación de operaciones CRUD para Gestionar Becados	0.1	0.2
12	L-1	Diseño de la interfaz de Gestionar Revistas	0.1	0.2
	L-2	Implementación de operaciones CRUD para Gestionar Revistas	0.5	0.6

13	M-1	Diseño de la interfaz de Gestionar Artículos	0.1	0.1
	M-2	Implementación de operaciones CRUD para Gestionar Artículos	0.6	0.6
14	N-1	Diseño de la interfaz de Gestionar Tesis	0.1	0.1
	N-2	Implementación de operaciones CRUD para Gestionar Tesis	0.5	0.5
15	O-1	Diseño de la interfaz para el Control de Préstamos de Libros	0.1	0.1
	O-2	Implementación de operaciones CRUD para el Control de préstamos de Libros	0.5	0.5
16	P-1	Diseño de la interfaz para el Control de Préstamos de Revistas	0.5	0.2
	P-2	Implementación de operaciones CRUD para el Control de préstamos de Revistas	0.3	0.6
17	Q-1	Diseño de la interfaz para el Control de Préstamos de Tesis	0.3	0.3
	Q-2	Implementación de operaciones CRUD para el Control de préstamos de Tesis	0.3	0.5
18	R-1	Diseño de la interfaz para Mostrar el modelo de control 0207	0.1	0.1
	R-2	Implementación del mostrar el Modelo de Control 0207	0.5	0.5
19	S-1	Diseño de la interfaz para listar el Reporte de Préstamos de Libros sin devolver	0.1	0.1
	S-2	Implementación de la interfaz para listar Reporte de Préstamos de Libros sin devolver	0.4	0.4
20	T-1	Diseño de la interfaz para listar el Reporte de Préstamos de Tesis sin devolver	0.1	0.1
	T-2	Implementación de la interfaz para listar Reporte de Préstamos de Tesis sin devolver	0.4	0.4
21	U-1	Diseño de la interfaz para listar el Reporte de Préstamos de Revistas sin devolver	0.1	0.1

	U-2	Implementación de la interfaz para listar Reporte de Préstamos de Revistas sin devolver	0.4	0.4
22	V-1	Diseño de la interfaz de gráficas de vías de préstamos de Libros	0.1	0.1
	V-2	Implementación de mostrar las gráficas de vías de préstamos de Libros	1	0.8
23	W-1	Diseño de la interfaz de gráficas de vías de préstamos de Tesis	0.6	0.6
	W-2	Implementación de mostrar las gráficas de vías de préstamos de Tesis	0.6	0.8
24	X-1	Diseño de la interfaz de gráficas de vías de préstamos de Revistas	0.2	0.2
	X-2	Implementación de mostrar las gráficas de vías de préstamos de Revistas	0.7	0.7
25	Y-1	Diseño de la interfaz para listar la capacidad de becas en cada edificio	0.5	0.5
	Y-2	Implementación de listar la capacidad de becas en cada edificio	0.3	0.3
26	Z-1	Diseño de la interfaz para mostrar el mapa de residencia y capacidades	0.5	0.2
	Z-2	Implementación de mostrar el mapa de residencia y capacidades.	0.5	0.5
27	AA-1	Diseño de la interfaz para mostrar el porciento de ocupación.	0.1	0.1
	AA-2	Implementación de mostrar el porciento de ocupación	0.1	0.1
28	BB-1	Diseño de la interfaz de reporte gráfico de estado constructivo	0.1	0.1
	BB-2	Implementación de reporte gráfico de estado constructivo	0.1	0.1
29	CC-1	Diseño de listar trazas de usuarios del sistema	0.8	0.5
	CC-2	Implementación de listar trazas de usuarios del sistema	1	1

Tabla 2.5: Tareas de Ingeniería

Tarea de ingeniería	
Número Tarea: A-1	Número Historia de Usuario: 1
Nombre Tarea: Diseño de la interfaz de gestión de roles y permisos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita al administrador gestionar los roles del sistema	

Tabla 2.6: Tarea de Ingeniería A-1

En las figuras 2.3 y 2.4 se muestran las interfaces para gestionar roles y permisos respectivamente.

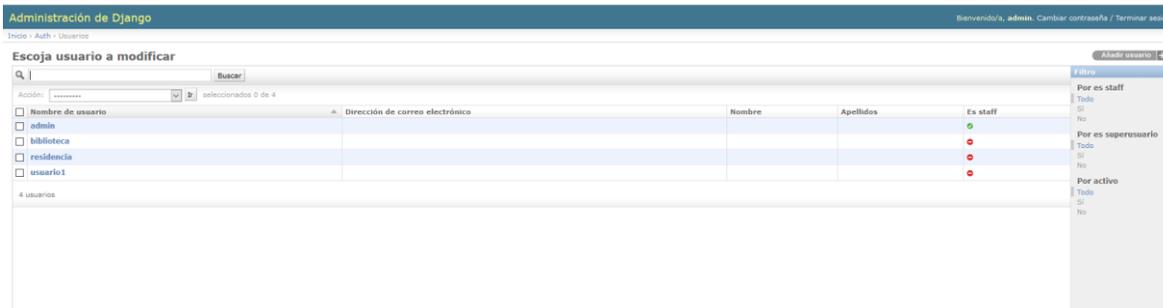


Figura 2.3: Gestionar roles

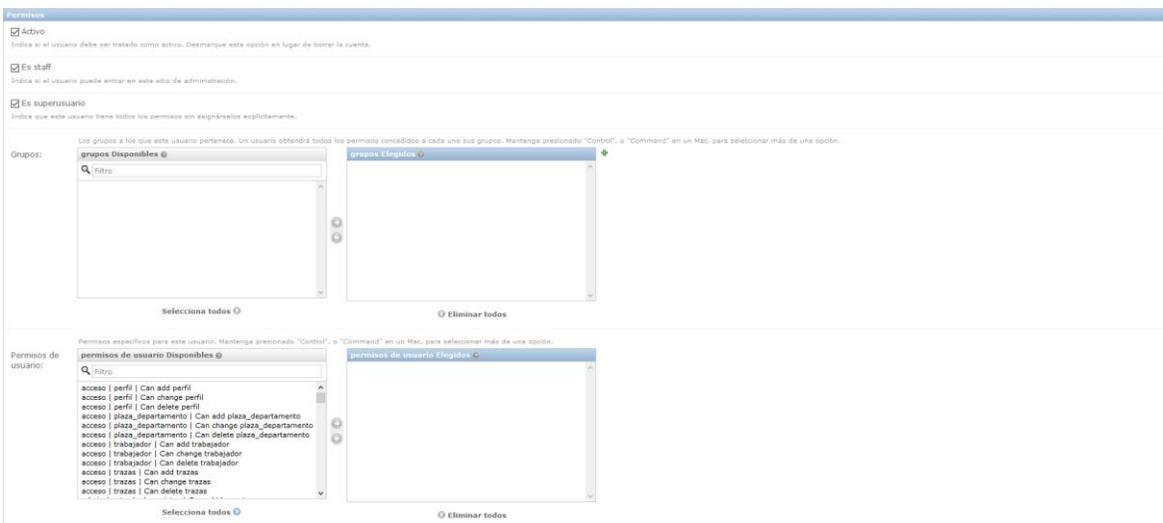


Figura 2.4: Gestionar permisos específicos.

Tarea de ingeniería	
Número Tarea: A-2	Número Historia de Usuario: 1
Nombre Tarea: Implementación de operaciones CRUD para Gestión de roles y permisos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se implementará la gestión de Roles necesaria para limitar el acceso a la información a los usuarios específicos.	

Tabla 2.7: Tarea de Ingeniería A-2

Tarea de ingeniería	
Número Tarea: B-1	Número Historia de Usuario: 2
Nombre Tarea: Diseño de la interfaz de gestión de usuarios	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Crear una interfaz amigable que permita al administrador del sistema gestionar los usuarios.	

Tabla 2.8: Tarea de Ingeniería B-1

En las figuras 2.5, 2.6 y 2.7 se muestran las interfaces para gestionar, modificar y eliminar un usuario respectivamente.

Trabajador	Carnet de Identidad	Correo	Direccion	Plaza que Ocupa	Acciones
Biblioteca	92031566894	biblioteca@uniss.edu.cu		Biblioteca	[+], [U], [X]
Humberto González Padilla	92021538343	humberto.gonzalez@mauta.cu	Hector Castellanos #09	Administrador	[+], [U], [X]
Residencia	84031688245	residencia@uniss.edu.cu		Residencia	[+], [U], [X]

Figura 2.5: Gestionar Usuario

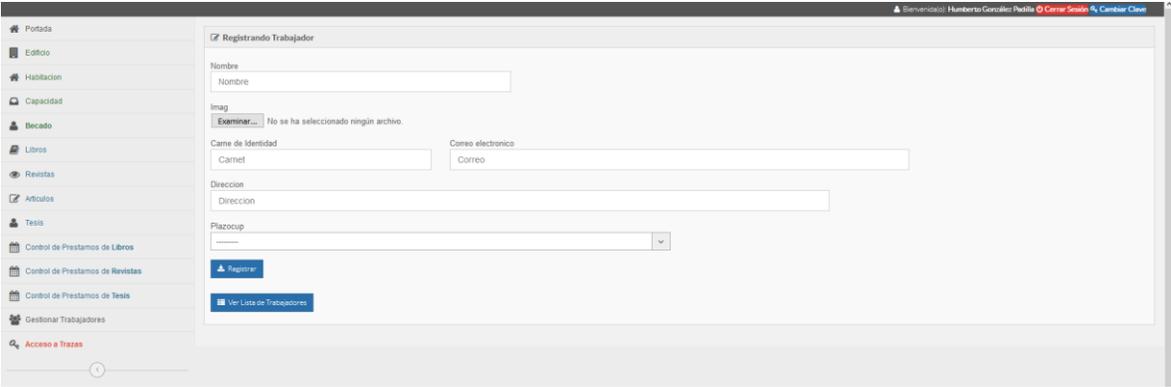


Figura 2.6: Modificar Usuario

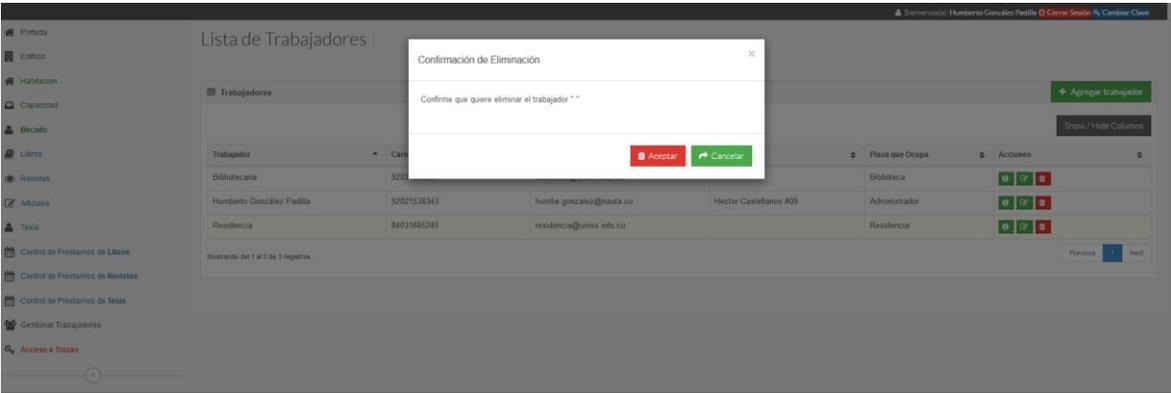


Figura 2.7: Confirmación de eliminar usuario.

Tarea de ingeniería	
Número Tarea: B-2	Número Historia de Usuario: 2
Nombre Tarea: Implementación de la gestión de usuarios	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una página en el sistema para que el administrador del sistema gestione los usuarios.	

Tabla 2.8: Tarea de Ingeniería B-2

Tarea de ingeniería	
Número Tarea: C-1	Número Historia de Usuario: 3

Nombre Tarea: Diseño de la interfaz para Libre acceso a estudiantes	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz amigable y de fácil entendimiento para los usuarios en el proceso de autenticación del sistema.	

Tabla 2.9: Tarea de Ingeniería C-1

En las figuras 2.8 y 2.9 se muestran las interfaces de usuario para la autenticación y la interfaz de libre acceso a estudiantes respectivamente.

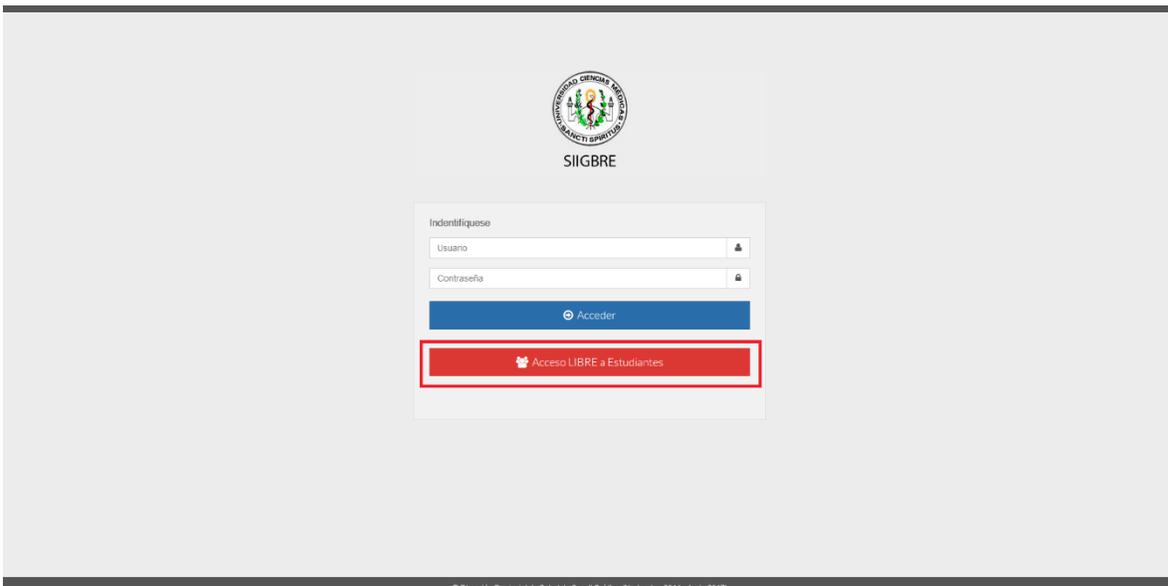


Figura 2.8: Acceso a Estudiantes.

Caratula	Ubicación	Título	Autor	ISBN
	Estante A, S1	CORPORACIÓN UNIVERSITARIA DE IBAGUÉ CENTRO DE INVESTIGACIONES, CONSULTORÍA Y EXTENSIÓN A LA COMUNIDAD	Ricardo Grau Abalo	679-8-89-006754-6
	Estante 1, Sección B	El Libro de Django	Alexander Pérez Ullota	545-1-55-645456-3
	Estante 1, Sección A	Introducción a la Programación Orientada a Objetos	Colectivo de Autores	564-0-91-815151-6
	Estante 1, Sección A	Cómo Programar en C/C++	Colectivo de Autores	654-5-12-318156-1
	Estante 1, Sección B	Matemática numérica	Manuel Alvarez Blanco	156-2-31-615615-6
	Estante 1, Sección B	Programación Java 2.J2SE 1.4	John Zukowski	156-1-16-151551-5
	Estante 2, Sección A	Economía Política de la Construcción del Socialismo: Fundamentos Generales	Colectivo de Autores	154-6-48-456468-4
	Estante 2, Sección A	Programación Java 2.J2SE 1.4	John Zukowski	556-1-51-56151-8

Figura 2.8: Visualizar libros.

Tarea de ingeniería	
Número Tarea: C-2	Número Historia de Usuario: 3
Nombre Tarea: Implementación de la interfaz para Libre acceso a estudiantes	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz amigable y de fácil entendimiento para los usuarios en el proceso de autenticación del sistema.	

Tabla 2.10: Tarea de Ingeniería C-2

Tarea de ingeniería	
Número Tarea: P-1	Número Historia de Usuario: 18
Nombre Tarea: Diseño de la interfaz para el Control de Préstamos de Libros	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una página para gestionar el Control de Préstamos de libros a los estudiantes dentro y fuera de la biblioteca	

Tabla 2.11: Tarea de Ingeniería P-1

En las figuras 2.9, 2.10 y 2.11 se muestran las interfaces visuales para listar, efectuar y eliminar préstamos respectivamente.

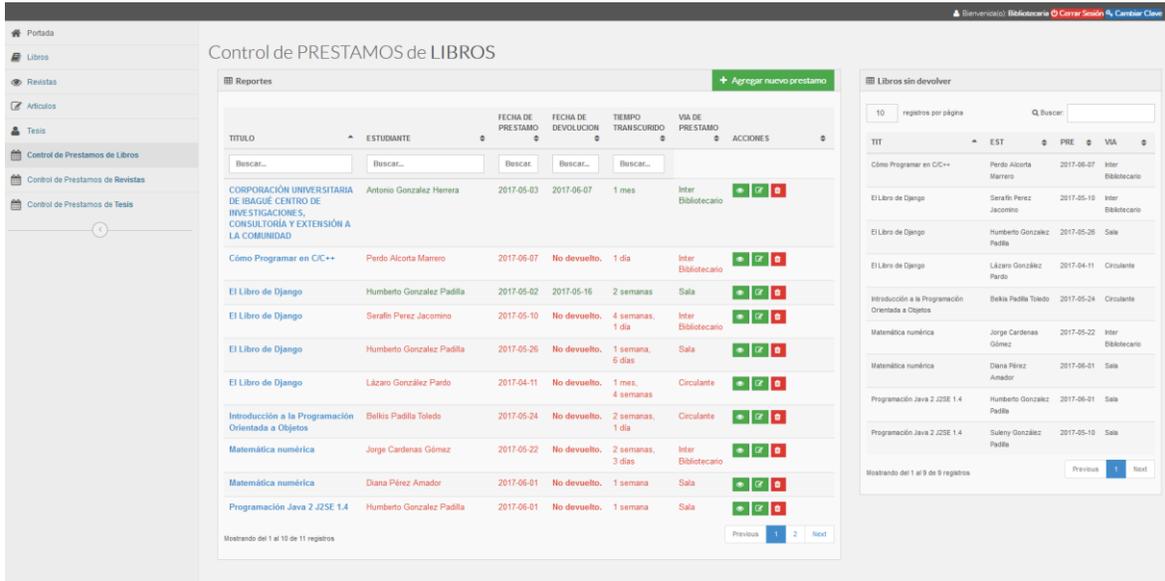


Figura 2.9: Control de préstamo de libros

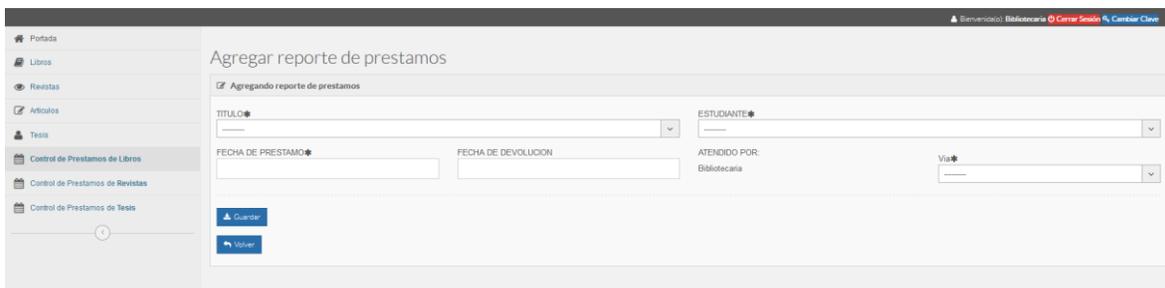


Figura 2.10: Agregar nuevo préstamo.

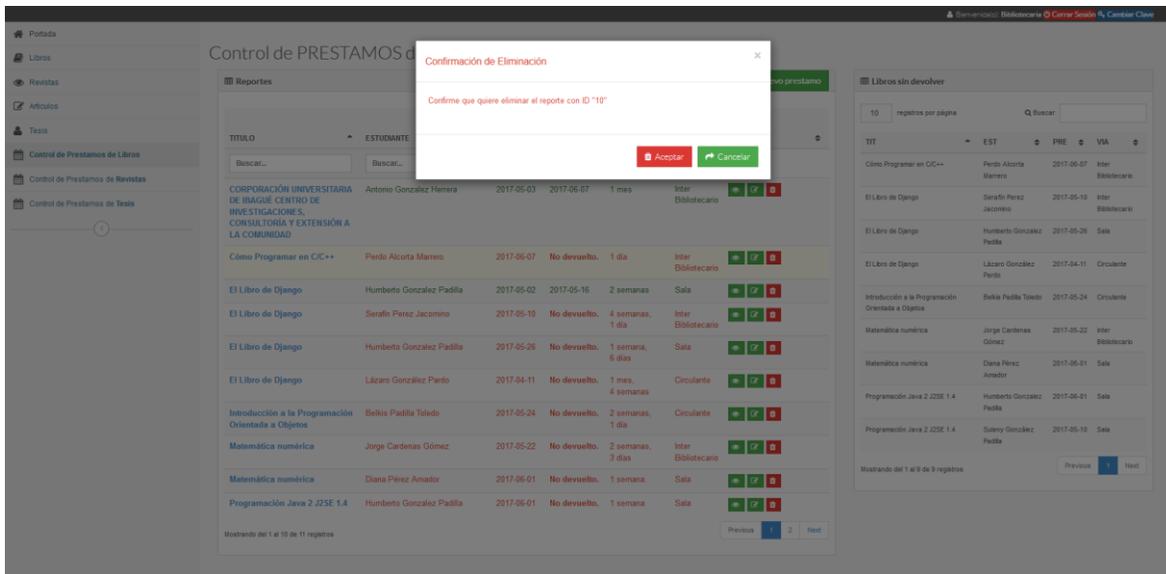


Figura 2.11: Eliminar préstamo de libros.

Tarea de ingeniería	
Número Tarea: P-1	Número Historia de Usuario: 18
Nombre Tarea: Implementación de operaciones CRUD para el Control de préstamos de libros	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable:Humberto González Padilla	
Descripción: Se implementará una página para gestionar el Control de Préstamos de libros a los estudiantes dentro y fuera de la biblioteca	

Tabla 2.12: Tarea de Ingeniería P-2

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

Conclusiones parciales

Queda claro cómo funciona el negocio a través de los artefactos proporcionados por la metodología XP y las reglas del negocio, entre otros. Además, la descripción general del sistema identifica los requerimientos funcionales, la lista de reserva del sistema y los procesos del negocio. La construcción de todos estos artefactos propició que:

- Se esclareciera cómo es el flujo de eventos que se realiza en cada uno de los procesos del negocio
- Se establecieron las relaciones de cada uno de los actores del sistema con las diferentes funcionalidades a implementar

CAPÍTULO 3: Construcción de la aplicación propuesta para automatizar la administración de la biblioteca y el control de la residencia estudiantil en la universidad de ciencias médicas en Sancti Spíritus.

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:

- No culpar al usuario del problema ocurrido.
- Escribir los mensajes de error en forma entendible el usuario.

- Explicarle en el mensaje cuál ha sido el error y cómo recuperarse.
- Utilizar siempre el mismo color y la misma forma para lograr una consecuencia entre los errores y lograr una detección casi intuitiva del error.

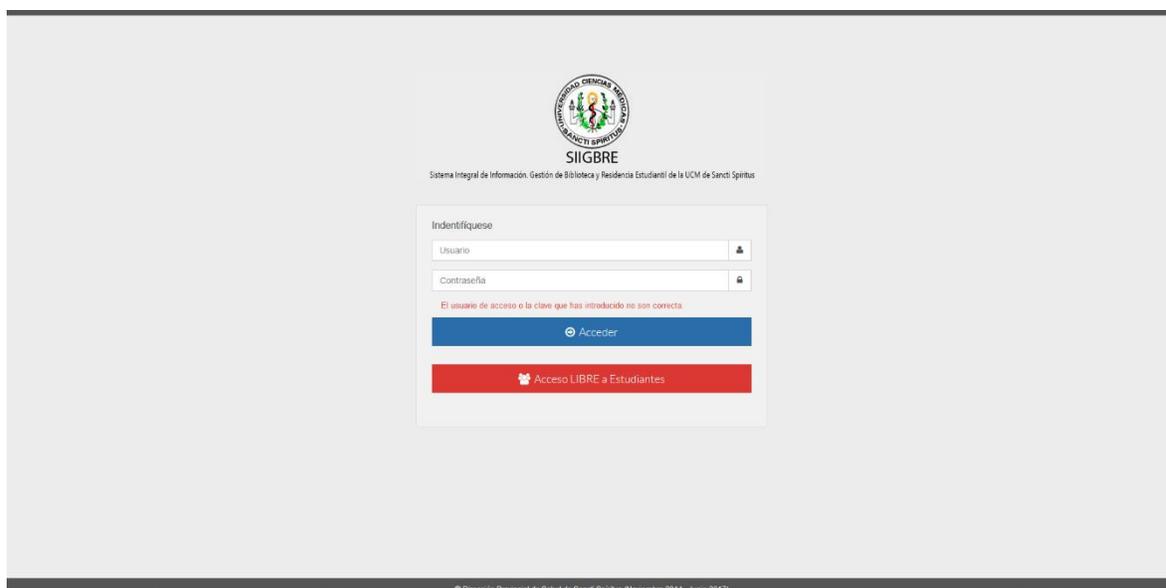


Figura 3.1: Pantalla de acceso al sistema. Error de Usuario o Contraseña.



Figura 3.2: Pantalla de acceso al sistema de administración. Error de Usuario o Contraseña.

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

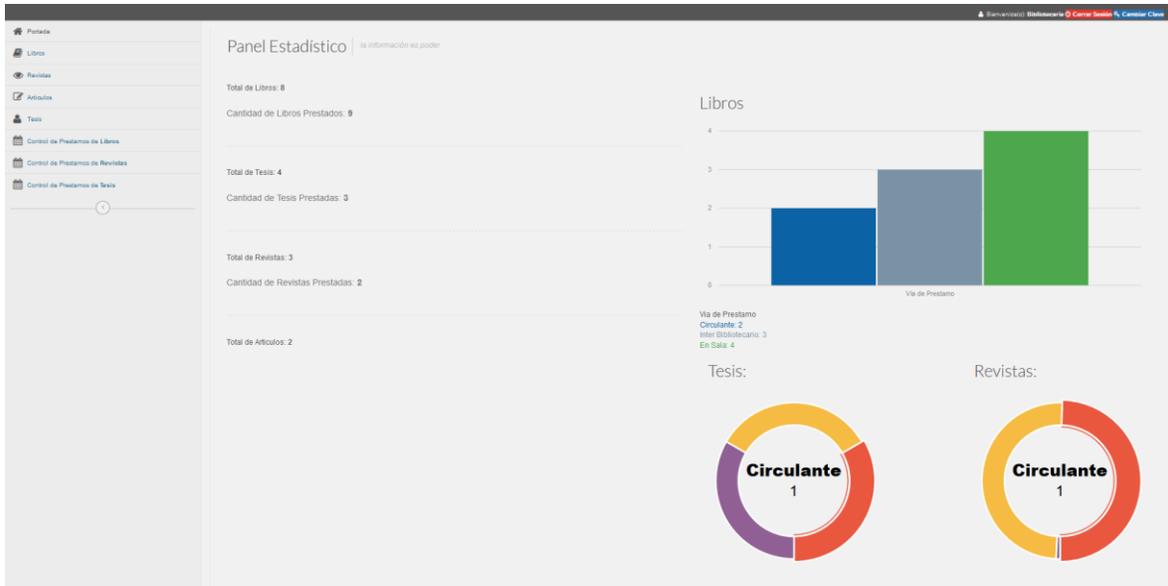


Figura 3.3: Portada del Módulo de Biblioteca



Figura 3.4: Portada del Módulo de Residencia Estudiantil.

3.1.4 Concepción de la ayuda

La aplicación cuenta con una ayuda, de interés para todos los usuarios del sistema, explica al usuario cómo es la manipulación y funcionamiento del software, así como la vía para acceder a cada una de las funcionalidades de la aplicación. Está formada por un menú interno que contiene

todas las funcionalidades del sistema. El usuario podrá navegar por cada uno de ellos sin restricciones, haciéndole más fácil el trabajo en el sistema y se encuentra disponible en todo momento.

3.2 Fase de pruebas 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.

Pruebas de Aceptación	
Código :P1	Número de Historia de usuario:1
Nombre: Comprobación de los datos de gestión de roles y permisos.	
Condiciones de Ejecución: Debe existir al menos un usuario administrador para realizar esa acción.	
Entrada/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. Selecciona la opción "Gestionar Trabajadores" 4. Se realizan los siguientes pasos: <ol style="list-style-type: none"> a. Agregar trabajador (Se llenan los campos de la interfaz, se selecciona la plaza que este ocupa y se verifica que los datos fueron insertados correctamente). b. Editar trabajador (Se selecciona el trabajador a modificar y se realizan los cambio deseados. Se verifican los cambios realizados). c. Eliminar trabajador (Se selecciona el trabajador a eliminar, se muestra un mensaje de confirmación de la operación. Se verifica la eliminación). 	
Resultados esperados: Rol y permisos gestionados correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 3.1: Prueba de aceptación P1

Pruebas de Aceptación	
Código :P2	Número de Historia de usuario:2
Nombre: Comprobación de datos de gestión de usuarios del sistema.	
Condiciones de Ejecución: Debe existir al menos un usuario administrador para realizar esa acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El Administrador inicia sección en el sistema con su usuario y contraseña. 2. Selecciona la opción “Usuarios”. 3. Se elige una de las acciones diferentes: <ul style="list-style-type: none"> - Añadir Usuario (Se llenaran los campos de la interfaz. Se insertan los datos en la base de datos y se verifica que los datos del usuario fueron insertados correctamente). - Modificar Usuario (Se modifican los datos introducidos de los usuarios registrados anteriormente en el sistema. Se comprueban los campos y se verifica en la base de datos satisfactoriamente). - Eliminar Usuario (Se elimina el usuario seleccionado y se verifica que la eliminación resultó satisfactoria). 	
Resultados esperados: Gestión de usuarios	
Evaluación de la prueba: Prueba Satisfactoria	

Tabla 3.2: Prueba de aceptación P2

Pruebas de Aceptación	
Código :P3	Número de Historia de usuario:3
Nombre: Comprobación de los datos de Libre Acceso a Estudiantes	
Condiciones de Ejecución: Cualquier usuario puede acceder a esta sección sin necesidad de autenticación en el sistema.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se ejecuta el sistema informático. 2. Se accede en la sección “Libre Acceso a Estudiantes”. 3. Se visualiza la información de la biblioteca. 4. Se realizan las diferentes opciones: 	

<p>a. Visualizar Libros (El sistema muestra los libros que con que cuenta la entidad).</p> <p>b. Visualizar Revistas (El sistema muestra las revistas que con que cuenta la entidad).</p> <p>c. Visualizar Tesis (El sistema muestra las tesis que con que cuenta la entidad.)</p> <p>d. Visualizar Artículos (El sistema muestra los artículos que con que cuenta las revistas de la unidad).</p> <p>e. Realizar búsquedas (Se realizan búsquedas por toda la base de datos de libros, artículos, revistas y tesis insertadas en el sistema para mayor precisión).</p>
Resultados esperados: Libre acceso a estudiantes gestionado correctamente
Evaluación de la prueba: Prueba satisfactoria

Tabla 3.3: Prueba de aceptación P3

Pruebas de Aceptación	
Código :P4	Número de Historia de usuario:4
Nombre: Comprobación de los datos de cambiar contraseña.	
Condiciones de Ejecución: El administrador o usuario del sistema puede realizar esta operación.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se accede al sistema mediante usuario y contraseña. 2. Selecciona la opción cambiar clave (Se llenan los campos requeridos en por la opción y se verifican los cambios en la base de datos de una forma satisfactoria). 	
Resultados esperados: Cambio de contraseña gestionada correctamente	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.4: Prueba de aceptación P1

Pruebas de Aceptación	
Código :P5	Número de Historia de usuario:5
Nombre: Comprobación de los datos en la BD.	
Condiciones de Ejecución: Solo el administrador puede realizar esta opción	
Entrada/Pasos de Ejecución:	

1. Se ejecuta el servidor XAMPP.
2. Se inicia sección con usuario y contraseña en el phpMyAdmin.
3. Se comprueba la conexión de la base de datos con la aplicación web.
4. Se inicia el servidor de Python.
Resultados esperados: Base de datos gestionada correctamente
Evaluación de la prueba: Prueba satisfactoria

Tabla 3.5: Prueba de aceptación P5

Pruebas de Aceptación	
Código :P6	Número de Historia de usuario:6
Nombre: Comprobación de los datos de Gestionar Edificios	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la residencia estudiantil tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
1. Se inicia el sistema.	
2. El trabajador inicia sección con usuario y contraseña.	
3. Selecciona la opción Edificios.	
4. Se realizan las diferentes opciones:	
a. Agregar nuevo edificio (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos).	
b. Eliminar edificio (Se elimina el edificio y se verifica al eliminación en la base de datos de forma satisfactoria).	
c. Modificar edificio (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).	
Resultados esperados: Edificios gestionados correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 3.6: Prueba de aceptación P6

Pruebas de Aceptación	
Código :P7	Número de Historia de usuario:7
Nombre: Comprobación de los datos de Gestionar Habitaciones	

<p>Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la residencia estudiantil tiene permisos para realizar esta acción.</p>
<p>Entrada/Pasos de Ejecución:</p> <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Habitaciones. 4. Se realizan las diferentes opciones: <ol style="list-style-type: none"> a. Agregar nueva habitación (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar habitación (Se elimina la habitación y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar habitación (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).
<p>Resultados esperados: Habitaciones gestionadas correctamente.</p>
<p>Evaluación de la prueba: Prueba satisfactoria</p>

Tabla 3.7: Prueba de aceptación P7

Pruebas de Aceptación	
Código :P8	Número de Historia de usuario:8
Nombre: Comprobación de los datos de Gestionar Capacidad	
<p>Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la residencia estudiantil tiene permisos para realizar esta acción.</p>	
<p>Entrada/Pasos de Ejecución:</p> <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Capacidad. 4. Se realizan las diferentes opciones: <ol style="list-style-type: none"> a. Agregar nueva capacidad (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar capacidad (Se elimina la capacidad y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar capacidad (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria). 	

Resultados esperados: Capacidades gestionadas correctamente.
Evaluación de la prueba: Prueba satisfactoria

Tabla 3.8: Prueba de aceptación P8

Pruebas de Aceptación	
Código :P9	Número de Historia de usuario:9
Nombre: Comprobación de los datos de Gestionar Becados	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la residencia estudiantil tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Becados. 4. Se realizan las diferentes opciones: <ol style="list-style-type: none"> a. Agregar nuevo registro (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar registro (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar registro (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria). 	
Resultados esperados: Becados gestionados correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 3.9: Prueba de aceptación P9

Pruebas de Aceptación	
Código :P10	Número de Historia de usuario:10
Nombre: Comprobación de los datos de Gestionar Libros	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 	

<p>3. Selecciona la opción Libros.</p> <p>4. Se realizan las diferentes opciones:</p> <p>a. Agregar nuevo registro (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos).</p> <p>b. Eliminar registro (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria).</p> <p>c. Modificar registro (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).</p> <p>d. Visualizar libro (Se muestra la pantalla informativa de los detalles del documento insertado).</p>
Resultados esperados: Libros gestionados correctamente.
Evaluación de la prueba: Prueba satisfactoria

Tabla 3.10: Prueba de aceptación P10

Pruebas de Aceptación	
Código :P11	Número de Historia de usuario:11
Nombre: Comprobación de los datos de Gestionar Revistas	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<p>1. Se inicia el sistema.</p> <p>2. El trabajador inicia sección con usuario y contraseña.</p> <p>3. Selecciona la opción Revistas.</p> <p>4. Se realizan las diferentes opciones:</p> <p>a. Agregar nuevo registro (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos).</p> <p>b. Eliminar registro (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria).</p> <p>c. Modificar registro (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).</p>	
Resultados esperados: Revistas gestionadas correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 3.11: Prueba de aceptación P11

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

Conclusiones parciales

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

- 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.
- 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

Con la realización del presente proyecto se arriban a las siguientes conclusiones:

1. Los fundamentos teóricos y metodológicos permitieron determinar que la metodología XP es la más adecuada para el desarrollo del sistema, así como el lenguaje *Python*, el *framework* Django y el sistema Gestor de Base de Datos MySQL son las tecnologías seleccionadas por ser de código libre y tener la capacidad de integrarse para funcionar en cualquier plataforma.
2. Se diseñó una aplicación web que contribuya al proceso de gestión bibliotecaria y de residencia estudiantil de la UCM, respaldado por softwares libres, programación orientada a objetos y la arquitectura Modelo Vista Plantilla, lo cual hace más sostenible la aplicación.
3. Se implementó una aplicación web que responde a las áreas de la biblioteca y residencia estudiantil de la UCM en Sancti Spíritus la cual se integra al SII y mantiene la concentración de la información para que pueda ser usada por varios usuarios con diversos fines, así como la automatización de los procesos para agilizar las tareas. La aplicación brinda al usuario la posibilidad de interactuar dinámicamente con los datos empleando la arquitectura cliente-servidor de manera que se pueda acceder a la información desde cualquier máquina con acceso a la red.

RECOMENDACIONES

Como recomendaciones se estimula a:

- Integrar el Módulo de Biblioteca la posibilidad de procesar el estado físico de los documentos impresos que allí se gestionan.
- Agregar una funcionabilidad de impresión del modelo 0207 por parte de los préstamos.
- Agregar al Módulo de Residencia Estudiantil la posibilidad de gestionar el proceso para la realización de la guardia en los edificios.

BIBLIOGRAFÍA

- acensTechnologies. (2012). Framework para el desarrollo ágil de aplicaciones.
- Alma, E. T., Jesús, M. A., Yunko, N. O., & Goretty, N. T. (2014). MySQL.
- Armijos, P. D. C., & Pilco, B. A. B. (Eds.). (2012). *Análisis De Python Con Django Frente A Ruby On Rails Para Desarrollo Ágil De Aplicaciones Web. Caso Práctico: Dech.*”. Ecuador.
- A. CALDERÓN, A., DÁMARIS, S., REBAZA, V. & CARLOS, J. 2007. Metodologías Ágiles.
- ALCALA, M. L. 2007. *MEDIDA DE LA USABILIDAD EN APLICACIONES DE ESCRITORIO. UN MÉTODO PRÁCTICO.*, UNIVERSIDAD NACIONAL DE EDUCACION A DISTANCIA.
- Arzamendi, A. (2003). *SISTEMA VASCO DE BIBLIOTECAS*
- Beck, U. (1998). *¿Qué es la globalización? Falacias del globalismo, respuestas a la globalización*
- Cabero, J. (1998). *Impacto de las nuevas tecnologías de la información y la comunicación en las organizaciones educativas.*
- Calabria, L., & Píriz, P. (2003). *Metodología XP*. Universidad ORT Uruguay.
- Cochran, D. (2012). *Twitter Bootstrap Web Development* r. Edición (Ed.) Retrieved from <http://www.packtpub.com/twitter-bootstrap-web-development/book>
- Chitasko, F. E. (2013). *Sistemas gestores de bases de dato*
- Dayana, B., & C., R. J. (2014). *Metodología actual, metodología XP*
- Duran, A. (2007). Qué es una aplicación web.
- Gauchat, J. D. (2013). *El gran libro de HTML5, CSS3 y Javascript* MARCOMBO (Ed.)
- Gelguero, M. P. (2011). *La evolución al software libre en la gestión de una biblioteca escolar.*
- Gil, F. J. C. (2012). Qué es Navicat? , from danysoft.com
- González, D. (2012). *Python para todos* E. C. C. Reconocimiento (Ed.)
- Gutiérrez, J. J. (2015). ¿Qué es un framework web?

- Jacobson, I., et al. (2006). "El Proceso Unificado de Desarrollo de software".
- Infomed. (2016). Historia. from http://www.sld.cu/acerca-de?quicktabs_pagina_acercade=1#quicktabspagina_acercade
- Jiménez, E. B. (2015). APLICACIONES INFORMÁTICAS.
- Joskowicz, J. (2008). Reglas y Prácticas en eXtreme Programming.
- Kaplan-Moss, A. H. y. J. (2008). *El Libro de Django* J. Dunck (Ed.)
- Lidia Fuentes, J. e. M. T. y. A. V. (2013). *Desarrollo de Software Basado en Componentes*
Retrieved from <https://fergarcia.com/2013/01/25/entorno-de-desarrollo-integrado-ide/>
- Matías, J. G. (2016). El patrón de diseño MTV from http://librosweb.es/libro/django_1_0/capitulo_5/el_patron_de_diseno_mtv.html
- Mateu, C. (2004). *Desarrollo de aplicaciones web*
- Mestras, J. P. (2013). Aplicaciones Web/Sistemas Web: Bootstrap 3.0. In F. d. I. Universidad Complutense Madrid (Ed.).
- Mora, S. L. (2002). *Programación de Aplicaciones Web: Historia, Principios Básicos y Clientes Web. Club Universitario*
- Mostaccio, M. R., & Engler, E. (2010). Del catálogo en línea al Sistema Integrado de Bibliotecas de la Facultad de Filosofía y Letras.
- Navarrete, D. Ó. A., & Aguilar, L. E. M. d. O. (2014). *Sistemas Integrales de Automatización de Bibliotecas: una descripción suscita* Vol. 1.
- Navarrete, Ó. A., & Yáñez, K. B. (2013). Sistemas integrales para la automatización de bibliotecas basados en software libre.
- Ortí, C. B. (2015). *Las tecnologías de la información y las comunicación (TIC)*
- Palacios, B. M. d. S. (2012). Sistemas Integrados de Gestión Bibliotecaria para jóvenes "hiperconectados".
- Peñafiel, M. (2016). *Aplicaciones Web*. Retrieved from <http://es.scribd.com/doc/136052164/APLICACIONES-WEB-pdf#scribd>

- Pressman, S. R. (2012). Ingeniería de software. Un enfoque práctico., from <http://es.slideshare.net/jdbg16/ingenieria-desoftware-un-enfoque-prctico-pressman-5thed>
- Prieto, J. Á. P. (2012). *Sistemas Integrados de Gestión Bibliotecaria libres y de código abierto*. UNIVERSIDAD DE LEÓN.
- Ramírez, A. O. (2010). Python como primer lenguaje de programación.
- Severin, E. (2013). *Enfoques estratégicos sobre las TICs en educación en América Latina y el Caribe*
- Valdivia, D. A. (2015). *Aplicación web que contribuya a la gestión de la información de los cadetes del Ministerio del Interior en Sancti Spíritus*. . Sancti Spíritus.
- Vázquez, I. M. (2015). *Definición de un Framework para aplicaciones Web con navegación sensible a concerns*

ANEXOS

Anexo 1: Historias de usuario

Historia de Usuario	
Número: 4	Nombre Historia de Usuario: Libre acceso para estudiantes
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Todos	Iteración Asignada: 1
Prioridad en Negocio: Media	Puntos Estimados: 0.3
Riesgo en Desarrollo: Media	Puntos Reales: 0.3
Descripción: Los usuarios, en su mayoría estudiantes, tendrán la posibilidad de realizar consultas en la base de datos.	
Observaciones:	

Tabla 1: Historia de Usuario No.4

Historia de Usuario	
Número: 5	Nombre Historia de Usuario: Cambiar contraseña de usuario
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Todos	Iteración Asignada: 1
Prioridad en Negocio: Media	Puntos Estimados: 0.5
Riesgo en Desarrollo: Media	Puntos Reales: 0.4
Descripción: El usuario autenticado podrá cambiar su contraseña actual por otra.	
Observaciones:	

Tabla 2: Historia de Usuario No.5

Historia de Usuario	
Número: 6	Nombre Historia de Usuario: Base de Datos
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 1.5
Riesgo en Desarrollo: Alta	Puntos Reales: 1.5
Descripción: La base de datos permitirá almacenar cada uno de los registros con una gran cantidad de información cada uno.	
Observaciones:	

Tabla 3: Historia de Usuario No.6

Historia de Usuario	
Número: 7	Nombre Historia de Usuario: Gestionar Edificios

Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alto	Puntos Estimados: 1.5
Riesgo en Desarrollo: Medio	Puntos Reales: 1.5
Descripción: Permite gestionar la cantidad de edificios disponibles para la residencia estudiantil.	
Observaciones:	

Tabla 4: Historia de Usuario No.7

Historia de Usuario	
Número: 8	Nombre Historia de Usuario: Gestionar Habitación
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alto	Puntos Estimados: 0.5
Riesgo en Desarrollo: Medio	Puntos Reales: 0.3
Descripción: Permite gestionar las habitaciones disponibles en cada edificio para la residencia estudiantil.	
Observaciones:	

Tabla 5: Historia de Usuario No.8

Historia de Usuario	
Número: 9	Nombre Historia de Usuario: Gestionar Capacidad
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alto	Puntos Estimados: 0.5
Riesgo en Desarrollo: Medio	Puntos Reales: 0.4
Descripción: Permite gestionar las capacidades, ocupadas o no, de cada habitación.	
Observaciones:	

Tabla 6: Historia de Usuario No. 9

Historia de Usuario	
Número: 10	Nombre Historia de Usuario: Gestionar Becado
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alto	Puntos Estimados: 0.5
Riesgo en Desarrollo: Medio	Puntos Reales: 0.3
Descripción: Permite gestionar la cantidad de becados con que cuentan las instalaciones.	
Observaciones:	

Observaciones:

Tabla 7: Historia de Usuario No.10

Historia de Usuario	
Número: 11	Nombre Historia de Usuario: Gestionar Libros
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alto	Puntos Estimados: 0.8
Riesgo en Desarrollo: Alto	Puntos Reales: 0.8
Descripción: El especialista tendrá la posibilidad de agregar, modificar, eliminar y visualizar los libros existentes en la biblioteca.	
.	
Observaciones:	

Tabla 8: Historia de Usuario No. 11

Historia de Usuario	
Número: 12	Nombre Historia de Usuario: Gestionar Revista
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alto	Puntos Estimados: 0.8
Riesgo en Desarrollo: Medio	Puntos Reales: 0.6
Descripción: Permite gestionar las revistas de la biblioteca.	
Observaciones:	

Tabla 9: Historia de Usuario No. 12

Historia de Usuario	
Número: 13	Nombre Historia de Usuario: Gestionar Artículo
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Bajo	Puntos Estimados: 0.3
Riesgo en Desarrollo: Medio	Puntos Reales: 0.3
Descripción: Permite gestionar los artículos de la biblioteca.	
Observaciones:	

Tabla 10: Historia de Usuario No.13

Historia de Usuario	
Número: 14	Nombre Historia de Usuario: Gestionar Tesis
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Bajo	Puntos Estimados: 0.3
Riesgo en Desarrollo: Medio	Puntos Reales: 0.1

Descripción: Permite gestionar las tesis de los años precederos que se encuentran en la biblioteca.

Observaciones:

Tabla 11: Historia de Usuario No. 14

Historia de Usuario	
Número: 15	Nombre Historia de Usuario: Control de Préstamo Libros
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1.5
Riesgo en Desarrollo: Alta	Puntos Reales: 1.5
Descripción: Permite controlar los préstamos de libros que se realicen la biblioteca.	
Observaciones:	

Tabla 12: Historia de Usuario No. 15

Historia de Usuario	
Número: 16	Nombre Historia de Usuario: Control de Préstamo Revistas
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1.5
Riesgo en Desarrollo: Alta	Puntos Reales: 1.0
Descripción: Permite controlar los préstamos de revistas que se realicen la biblioteca.	
Observaciones:	

Tabla 13: Historia de Usuario No. 16

Historia de Usuario	
Número: 17	Nombre Historia de Usuario: Control de Préstamo Tesis
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1.5
Riesgo en Desarrollo: Alta	Puntos Reales: 1.0
Descripción: Permite controlar los préstamos de tesis que se realicen la biblioteca.	
Observaciones:	

Tabla 14: Historia de Usuario No. 17

Historia de Usuario	
Número: 18	Nombre Historia de Usuario: Mostrar Modelo de Control 0207
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Media	Puntos Estimados: 0.4
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Muestra el modelo de control 0207 con los respectivos datos del préstamo.	
Observaciones:	

Tabla 15: Historia de Usuario No. 18

Historia de Usuario	
Número: 19	Nombre Historia de Usuario: Reporte de préstamos de libros sin devolver
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Media	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Realiza un reporte sobre los libros prestados que no se han devuelto.	
Observaciones:	

Tabla 16: Historia de Usuario No. 19

Historia de Usuario	
Número: 20	Nombre Historia de Usuario: Reporte de préstamos de revistas sin devolver
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Media	Puntos Estimados: 0.2
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Realiza un reporte sobre las revistas prestadas que no se han devuelto.	
Observaciones:	

Tabla 17: Historia de Usuario No. 20

Historia de Usuario	
Número: 21	Nombre Historia de Usuario: Reporte de préstamos de tesis sin devolver
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Media	Puntos Estimados: 0.2
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1

Descripción: Realiza un reporte sobre las tesis prestadas que no se han devuelto.

Observaciones:

Tabla 18: Historia de Usuario No 21.

Historia de Usuario	
Número: 22	Nombre Historia de Usuario: Gráficas de vía de préstamos de libros
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Baja	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Muestra una gráfica que informa la situación de los préstamos de libros realizados.	
Observaciones:	

Tabla 19: Historia de Usuario No. 22

Historia de Usuario	
Número: 23	Nombre Historia de Usuario: Gráficas de vía de préstamos de tesis
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Baja	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Muestra una gráfica que informa la situación de los préstamos de tesis realizados.	
Observaciones:	

Tabla 20: Historia de Usuario No. 23

Historia de Usuario	
Número: 24	Nombre Historia de Usuario: Gráficas de vía de préstamos de revistas
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Baja	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Muestra una gráfica que informa la situación de los préstamos de revistas realizados.	
Observaciones:	

Tabla 21: Historia de Usuario No. 24

Historia de Usuario	
Número: 25	Nombre Historia de Usuario: Reporte de capacidad becas
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Medio	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.2
Descripción: Muestra un reporte sobre las capacidades en la beca.	
Observaciones:	

Tabla 22: Historia de Usuario No. 25

Historia de Usuario	
Número: 26	Nombre Historia de Usuario: Reporte de mapa de residencia
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Medio	Puntos Estimados: 0.5
Riesgo en Desarrollo: Baja	Puntos Reales: 0.3
Descripción: Muestra un reporte sobre las capacidades en la beca en forma de un mapa gráfico.	
Observaciones:	

Tabla 23: Historia de Usuario No. 26

Historia de Usuario	
Número: 27	Nombre Historia de Usuario: Reporte de por ciento de ocupación
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Medio	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1
Descripción: Muestra un reporte sobre la ocupación de la beca en por ciento.	
Observaciones:	

Tabla 24: Historia de Usuario No. 27

Historia de Usuario	
Número: 28	Nombre Historia de Usuario: Reporte gráfico del estado constructivo
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador y especialista	Iteración Asignada: 3
Prioridad en Negocio: Medio	Puntos Estimados: 0.1
Riesgo en Desarrollo: Baja	Puntos Reales: 0.1

Descripción: Muestra un reporte gráfico sobre el estado constructivo de los edificios
Observaciones:

Tabla 25: Historia de Usuario No. 28

Historia de Usuario	
Número: 29	Nombre Historia de Usuario: Listar Trazas de usuarios del sistema
Modificación (o extensión) de Historia de Usuario (Nro. y Nombre):	
Usuario: Administrador	Iteración Asignada: 3
Prioridad en Negocio: Medio	Puntos Estimados: 1.0
Riesgo en Desarrollo: Baja	Puntos Reales: 0.8
Descripción: Genera una lista de las trazas de cada usuario que use el sistema.	
Observaciones:	

Tabla 26: Historia de Usuario No. 29

Anexo 2: Tareas de ingeniería

Tarea de ingeniería	
Número Tarea: A-1	Número Historia de Usuario: 1
Nombre Tarea: Diseño de la interfaz de gestión de roles y permisos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita al administrador gestionar los roles del sistema	

Tabla 1: Tarea de Ingeniería A-1

Tarea de ingeniería	
Número Tarea: A-2	Número Historia de Usuario: 1
Nombre Tarea: Implementación de operaciones CRUD para la gestión de usuarios y permisos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5

Programador responsable: Humberto González Padilla
Descripción: Se implementarán las operaciones de creación, lectura, actualización y eliminación de usuarios y permisos

Tabla 2: Tarea de Ingeniería A-2

Tarea de ingeniería	
Número Tarea: B-1	Número Historia de Usuario: 2
Nombre Tarea: Diseño de la interfaz de gestión de usuarios	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita al administrador gestionar los usuarios del sistema.	

Tabla 3: Tarea de Ingeniería B-1

Tarea de ingeniería	
Número Tarea: B-2	Número Historia de Usuario: 2
Nombre Tarea: Implementación de operaciones CRUD para la gestión de usuarios y permisos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se implementarán las operaciones de creación, lectura, actualización y eliminación de usuarios y permisos	

Tabla 4: Tarea de Ingeniería B-2

Tarea de ingeniería	
Número Tarea: C-1	Número Historia de Usuario: 3
Nombre Tarea: Diseño de la interfaz de autenticación de usuario.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	

Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita a cada usuario autenticarse.

Tabla 5: Tarea de Ingeniería C-1

Tarea de ingeniería	
Número Tarea: C-2	Número Historia de Usuario: 3
Nombre Tarea: Implementación de la autenticación de usuario.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Humberto González Padilla	
Descripción: Se implementará la interfaz para la autenticación de los usuarios.	

Tabla 6: Tarea de Ingeniería C-2

Tarea de ingeniería	
Número Tarea: D-1	Número Historia de Usuario: 4
Nombre Tarea: Diseño de la interfaz Libre acceso a estudiantes	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que brinde la información adecuada a cualquier usuario que se autentique (en su mayoría estudiantes).	

Tabla 7: Tarea de Ingeniería D-1

Tarea de ingeniería	
Número Tarea: D-2	Número Historia de Usuario: 4
Nombre Tarea: Implementación de operaciones para Libre acceso a estudiantes	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	

Descripción: Se implementará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que brinde la información adecuada a cualquier usuario que se autentique (en su mayoría estudiantes).

Tabla 8: Tarea de Ingeniería D-2

Tarea de ingeniería	
Número Tarea: E-1	Número Historia de Usuario: 5
Nombre Tarea: Diseño de la interfaz de Cambiar contraseña de usuario	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que posibilite realizar el cambio de contraseña de los usuarios.	

Tabla 9: Tarea de Ingeniería E-1

Tarea de ingeniería	
Número Tarea: E-2	Número Historia de Usuario: 5
Nombre Tarea: Implementación de operaciones CRUD para Cambiar contraseña de usuario	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD con la contraseña de cada usuario autenticado.	

Tabla 10: Tarea de Ingeniería E-2

Tarea de ingeniería	
Número Tarea: F-1	Número Historia de Usuario: 6
Nombre Tarea: Diseño y Confección del Diagrama de Entidad Relación	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	

Descripción: Se diseñará el diagrama de Entidad Relación

Tabla 11: Tarea de Ingeniería F-1

Tarea de ingeniería	
Número Tarea: G-1	Número Historia de Usuario: 7
Nombre Tarea: Diseño de la interfaz de Gestionar Edificios	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los edificios.	

Tabla 12: Tarea de Ingeniería F-2

Tarea de ingeniería	
Número Tarea: G-2	Número Historia de Usuario: 7
Nombre Tarea: Implementación de operaciones CRUD Gestionar Edificios	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar los edificios.	

Tabla 13: Tarea de Ingeniería G-2

Tarea de ingeniería	
Número Tarea: H-1	Número Historia de Usuario: 8
Nombre Tarea: Diseño de la interfaz de Gestionar Habitaciones	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar las habitaciones.	

Tabla 14: Tarea de Ingeniería H-1

Tarea de ingeniería	
Número Tarea: H-2	Número Historia de Usuario: 8
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Habitaciones	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar las habitaciones.	

Tabla 15: Tarea de Ingeniería H-2

Tarea de ingeniería	
Número Tarea: I-1	Número Historia de Usuario: 9
Nombre Tarea: Diseño de la interfaz de Gestionar Capacidad	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar las capacidades en las habitaciones.	

Tabla 16: Tarea de Ingeniería I-1

Tarea de ingeniería	
Número Tarea: I-2	Número Historia de Usuario: 9
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Capacidad	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar las capacidades en las habitaciones.	

Tabla 17: Tarea de Ingeniería I-2

Tarea de ingeniería	
Número Tarea: J-1	Número Historia de Usuario: 10
Nombre Tarea: Diseño de la interfaz de Gestionar Becados	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los becados.	

Tabla 18: Tarea de Ingeniería J-1

Tarea de ingeniería	
Número Tarea: J-2	Número Historia de Usuario: 10
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Becados.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a los becados.	

Tabla 19: Tarea de Ingeniería J-2

Tarea de ingeniería	
Número Tarea: K-1	Número Historia de Usuario: 11
Nombre Tarea: Diseño de la interfaz de Gestionar Libros	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los libros de la biblioteca.	

Tabla 20: Tarea de Ingeniería K-1

Tarea de ingeniería	
Número Tarea: K-2	Número Historia de Usuario: 11

Nombre Tarea: Implementación de operaciones CRUD para Gestionar Libros	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.8
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a los libros de la biblioteca.	

Tabla 21: Tarea de Ingeniería K-2

Tarea de ingeniería	
Número Tarea: L-1	Número Historia de Usuario: 12
Nombre Tarea: Diseño de la interfaz de Gestionar Revistas	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar las Revistas de la biblioteca.	

Tabla 22: Tarea de Ingeniería L-1

Tarea de ingeniería	
Número Tarea: L-2	Número Historia de Usuario: 12
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Revistas	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a las Revistas de la biblioteca.	

Tabla 23: Tarea de Ingeniería L-2

Tarea de ingeniería	
Número Tarea: M-1	Número Historia de Usuario: 13
Nombre Tarea: Diseño de la interfaz de Gestionar Artículos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2

Programador responsable: Humberto González Padilla
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los Artículos de la biblioteca.

Tabla 24: Tarea de Ingeniería M-1

Tarea de ingeniería	
Número Tarea: M-2	Número Historia de Usuario: 13
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Artículos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.6
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a los Artículos de la biblioteca.	

Tabla 25: Tarea de Ingeniería M-2

Tarea de ingeniería	
Número Tarea: N-1	Número Historia de Usuario: 14
Nombre Tarea: Diseño de la interfaz de Gestionar Tesis	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar las Tesis de la biblioteca.	

Tabla 26: Tarea de Ingeniería N-1

Tarea de ingeniería	
Número Tarea: N-2	Número Historia de Usuario: 14
Nombre Tarea: Implementación de operaciones CRUD para Gestionar Tesis	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	

Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a las Tesis de la biblioteca.

Tabla 27: Tarea de Ingeniería N-2

Tarea de ingeniería	
Número Tarea: O-1	Número Historia de Usuario: 15
Nombre Tarea: Diseño de la interfaz para el Control de Préstamos de Libros	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los préstamos de libros en la biblioteca.	

Tabla 28: Tarea de Ingeniería O-1

Tarea de ingeniería	
Número Tarea: O-2	Número Historia de Usuario: 15
Nombre Tarea: Implementación de operaciones CRUD para el Control de préstamos de Libros	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a los préstamos de libros en la biblioteca.	

Tabla 29: Tarea de Ingeniería O-2

Tarea de ingeniería	
Número Tarea: P-1	Número Historia de Usuario: 16
Nombre Tarea: Diseño de la interfaz para el Control de Préstamos de Revistas	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	

Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los préstamos de revistas en la biblioteca.

Tabla 30: Tarea de Ingeniería P-1

Tarea de ingeniería	
Número Tarea: P-2	Número Historia de Usuario: 16
Nombre Tarea: Implementación de operaciones CRUD para el Control de préstamos de Revistas	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a los préstamos de revistas en la biblioteca.	

Tabla 31: Tarea de Ingeniería P-2

Tarea de ingeniería	
Número Tarea: Q-1	Número Historia de Usuario: 17
Nombre Tarea: Diseño de la interfaz para el Control de Préstamos de Tesis	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita gestionar los préstamos de Tesis en la biblioteca.	

Tabla 32: Tarea de Ingeniería Q-1

Tarea de ingeniería	
Número Tarea: Q-2	Número Historia de Usuario: 17
Nombre Tarea: Implementación de operaciones CRUD para el Control de préstamos de Tesis	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3

Programador responsable: Humberto González Padilla
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para gestionar a los préstamos de Tesis en la biblioteca.

Tabla 33: Tarea de Ingeniería Q-2

Tarea de ingeniería	
Número Tarea: R-1	Número Historia de Usuario: 18
Nombre Tarea: Diseño de la interfaz para Mostrar el modelo de control 0207	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita mostrar el modelo 0207.	

Tabla 34: Tarea de Ingeniería R-1

Tarea de ingeniería	
Número Tarea: R-2	Número Historia de Usuario: 18
Nombre Tarea: Implementación del mostrar el Modelo de Control 0207	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita realizar operaciones CRUD para mostrar el modelo 0207.	

Tabla 35: Tarea de Ingeniería R-1

Tarea de ingeniería	
Número Tarea: S-1	Número Historia de Usuario: 19
Nombre Tarea: Diseño de la interfaz para listar el Reporte de Préstamos de Libros sin devolver.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	

Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita listar un reporte de los libros prestados sin devolver.

Tabla 36: Tarea de Ingeniería S-1

Tarea de ingeniería	
Número Tarea: S-2	Número Historia de Usuario: 19
Nombre Tarea: Implementación de la interfaz para listar Reporte de Préstamos de Libros sin devolver	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita listar un reporte de los libros prestados sin devolver.	

Tabla 37: Tarea de Ingeniería S-2

Tarea de ingeniería	
Número Tarea: T-1	Número Historia de Usuario: 20
Nombre Tarea: Diseño de la interfaz para listar el Reporte de Préstamos de Tesis sin devolver	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita listar un reporte de las tesis prestadas sin devolver.	

Tabla 38: Tarea de Ingeniería T-1

Tarea de ingeniería	
Número Tarea: T-2	Número Historia de Usuario: 20
Nombre Tarea: Implementación de la interfaz para listar Reporte de Préstamos de Tesis sin devolver	

Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita listar un reporte de las Tesis prestadas sin devolver.	

Tabla 39: Tarea de Ingeniería T-2

Tarea de ingeniería	
Número Tarea: U-1	Número Historia de Usuario: 21
Nombre Tarea: Diseño de la interfaz para listar el Reporte de Préstamos de Revistas sin devolver.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita listar un reporte de las revistas prestadas sin devolver.	

Tabla 40: Tarea de Ingeniería U-1

Tarea de ingeniería	
Número Tarea: U-2	Número Historia de Usuario: 21
Nombre Tarea: Implementación de la interfaz para listar Reporte de Préstamos de Revistas sin devolver.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.4
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita listar un reporte de las revistas prestadas sin devolver.	

Tabla 41: Tarea de Ingeniería U-2

Tarea de ingeniería	
Número Tarea: V-1	Número Historia de Usuario: 22

Nombre Tarea: Diseño de la interfaz de gráficas de vías de préstamos de Libros.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita graficar las vías de préstamos de los libros.	

Tabla 42: Tarea de Ingeniería V-1

Tarea de ingeniería	
Número Tarea: V-2	Número Historia de Usuario: 22
Nombre Tarea: Implementación de mostrar las gráficas de vías de préstamos de Libros.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1.0
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita graficar las vías de préstamos de los libros.	

Tabla 43: Tarea de Ingeniería V-2

Tarea de ingeniería	
Número Tarea: W-1	Número Historia de Usuario: 23
Nombre Tarea: Diseño de la interfaz de gráficas de vías de préstamos de Tesis.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.6
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita graficar las vías de préstamos de las tesis.	

Tabla 44: Tarea de Ingeniería W-1

Tarea de ingeniería

Número Tarea: W-2	Número Historia de Usuario: 23
Nombre Tarea: Implementación de mostrar las gráficas de vías de préstamos de Tesis.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.6
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita graficar las vías de préstamos de las tesis.	

Tabla 45: Tarea de Ingeniería W-2

Tarea de ingeniería	
Número Tarea: X-1	Número Historia de Usuario: 24
Nombre Tarea: Diseño de la interfaz de gráficas de vías de préstamos de Revistas.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.2
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita graficar las vías de préstamos de las revistas.	

Tabla 46: Tarea de Ingeniería X-1

Tarea de ingeniería	
Número Tarea: X-2	Número Historia de Usuario: 24
Nombre Tarea: Implementación de mostrar las gráficas de vías de préstamos de Revistas.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.7
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita graficar las vías de préstamos de las revistas.	

Tabla 47: Tarea de Ingeniería X-2

Tarea de ingeniería	
Número Tarea: Y-1	Número Historia de Usuario: 25
Nombre Tarea: Diseño de la interfaz para listar la capacidad de becas en cada edificio	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita listar las capacidades de becas en cada edificio.	

Tabla 48: Tarea de Ingeniería Y-1

Tarea de ingeniería	
Número Tarea: Y-2	Número Historia de Usuario: 25
Nombre Tarea: Implementación de listar la capacidad de becas en cada edificio	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.3
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita listar las capacidades de becas en cada edificio.	

Tabla 49: Tarea de Ingeniería Y-2

Tarea de ingeniería	
Número Tarea: Z-1	Número Historia de Usuario: 26
Nombre Tarea: Diseño de la interfaz para mostrar el mapa de residencia y capacidades	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita mostrar un mapa de la residencia estudiantil y las capacidades.	

Tabla 50: Tarea de Ingeniería Z-1

Tarea de ingeniería	
Número Tarea: Z-2	Número Historia de Usuario: 26
Nombre Tarea: Implementación de mostrar el mapa de residencia y capacidades.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.5
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita mostrar un mapa de la residencia estudiantil y las capacidades.	

Tabla 51: Tarea de Ingeniería Z-2

Tarea de ingeniería	
Número Tarea: AA-1	Número Historia de Usuario: 27
Nombre Tarea: Diseño de la interfaz para mostrar el porcentaje de ocupación.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita mostrar el por ciento de ocupación.	

Tabla 52: Tarea de Ingeniería AA-1

Tarea de ingeniería	
Número Tarea: AA-2	Número Historia de Usuario: 27
Nombre Tarea: Implementación de mostrar el porcentaje de ocupación	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita mostrar el por ciento de ocupación.	

Tabla 53: Tarea de Ingeniería AA-2

Tarea de ingeniería	
Número Tarea: BB-1	Número Historia de Usuario: 28
Nombre Tarea: Diseño de la interfaz de reporte gráfico de estado constructivo	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita mostrar el estado constructivo.	

Tabla 54: Tarea de Ingeniería BB-1

Tarea de ingeniería	
Número Tarea: BB-2	Número Historia de Usuario: 28
Nombre Tarea: Implementación de reporte gráfico de estado constructivo	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita mostrar el estado constructivo.	

Tabla 55: Tarea de Ingeniería BB-2

Tarea de ingeniería	
Número Tarea: CC-1	Número Historia de Usuario: 29
Nombre Tarea: Diseño de listar trazas de usuarios del sistema	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se diseñará una interfaz adecuada de acuerdo a los estándares y plantilla seleccionada que permita listar las trazas de usuarios del sistema.	

Tabla 56: Tarea de Ingeniería CC-1

Tarea de ingeniería	
Número Tarea: CC-2	Número Historia de Usuario: 29
Nombre Tarea: Implementación de listar trazas de usuarios del sistema	

Tipo de Tarea: Desarrollo.	Puntos Estimados: 0.1
Programador responsable: Humberto González Padilla	
Descripción: Se implementará una interfaz que permita listar las trazas de usuarios del sistema.	

Tabla 57: Tarea de Ingeniería CC-2

Anexo 3: Pruebas de Aceptación

Pruebas de Aceptación	
Código :P12	Número de Historia de usuario:12
Nombre: Comprobación de los datos de Gestionar Revistas	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Revistas. 4. Se realizan las diferentes opciones: <ol style="list-style-type: none"> a. Agregar nuevo registro (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar registro (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar registro (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria). 	
Resultados esperados: Revistas gestionadas correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P13	Número de Historia de usuario:13
Nombre: Comprobación de los datos de Gestionar Artículos	

Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.
Entrada/Pasos de Ejecución: 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Artículos. 4. Se realizan las diferentes opciones: a. Agregar nuevo registro (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar registro (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar registro (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).
Resultados esperados: Artículos gestionados correctamente.
Evaluación de la prueba: Prueba satisfactoria

Pruebas de Aceptación	
Código :P14	Número de Historia de usuario:14
Nombre: Comprobación de los datos de Gestionar Tesis	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución: 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Tesis. 4. Se realizan las diferentes opciones: a. Agregar nuevo registro (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar registro (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar registro (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).	

Resultados esperados: Tesis gestionadas correctamente.
Evaluación de la prueba: Prueba satisfactoria

Pruebas de Aceptación	
Código :P15	Número de Historia de usuario:15
Nombre: Comprobación de los datos de Control de préstamos de libros	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Control de Préstamos de Libros. 4. Se realizan las diferentes opciones: <ol style="list-style-type: none"> a. Agregar nuevo préstamo (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos). b. Eliminar préstamo (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria). c. Modificar préstamo (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria). 	
Resultados esperados: Control de préstamo de libros gestionados correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P16	Número de Historia de usuario:16
Nombre: Comprobación de los datos de Control de préstamos de revistas	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 	

<p>2. El trabajador inicia sección con usuario y contraseña.</p> <p>3. Selecciona la opción Control de Préstamos de Revistas.</p> <p>4. Se realizan las diferentes opciones:</p> <p>a. Agregar nuevo préstamo (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos).</p> <p>b. Eliminar préstamo (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria).</p> <p>c. Modificar préstamo (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).</p>
Resultados esperados: Control de préstamo de revistas gestionados correctamente.
Evaluación de la prueba: Prueba satisfactoria

Pruebas de Aceptación	
Código :P17	Número de Historia de usuario:17
Nombre: Comprobación de los datos de Control de préstamos de tesis	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<p>1. Se inicia el sistema.</p> <p>2. El trabajador inicia sección con usuario y contraseña.</p> <p>3. Selecciona la opción Control de Préstamos de Tesis.</p> <p>4. Se realizan las diferentes opciones:</p> <p>a. Agregar nuevo préstamo (Se llenan los campos requeridos por el sistema y se verifica que los datos se introdujeron correctamente en la base de datos).</p> <p>b. Eliminar préstamo (Se elimina el registro y se verifica al eliminación en la base de datos de forma satisfactoria).</p> <p>c. Modificar préstamo (Se modifican los campos requeridos en el sistema. Se verifica los cambios en la base de datos de forma satisfactoria).</p>	
Resultados esperados: Control de préstamo de tesis gestionados correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P18	Número de Historia de usuario:18
Nombre: Comprobación de los datos de mostrar modelo 0207	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona las opciones de Control de Préstamos de Libros, Tesis y Revistas. 4. Seleccionar la opción visualizar. 5. Mostrar modelo 0207. 	
Resultados esperados: mostrando modelo 0207 correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P19	Número de Historia de usuario:19
Nombre: Comprobación de los datos de Reporte de Préstamo de libros sin devolver.	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Control de Préstamos de Libros. 4. Listando de libros sin devolver en la parte derecha de la página. 	
Resultados esperados: Reporte de préstamo de libros sin devolver listado correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P20	Número de Historia de usuario:20
Nombre: Comprobación de los datos de Reporte de Préstamo de revistas sin devolver.	

Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.
Entrada/Pasos de Ejecución: 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Control de Préstamos de Revistas. 4. Listando de libros sin devolver en la parte derecha de la página.
Resultados esperados: Reporte de préstamo de revistas sin devolver listado correctamente.
Evaluación de la prueba: Prueba satisfactoria

Pruebas de Aceptación	
Código :P21	Número de Historia de usuario:21
Nombre: Comprobación de los datos de Reporte de Préstamo de Tesis sin devolver.	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución: 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Control de Préstamos de Tesis. 4. Listando de libros sin devolver en la parte derecha de la página.	
Resultados esperados: Reporte de préstamo de tesis sin devolver listado correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P22	Número de Historia de usuario:22

Nombre: Comprobación de los datos de grafico de barras mostrando la vía de préstamo de libros.
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Portada. 4. Listando el gráfico de barras de vía de préstamos de libros. (Circulante, en sala, inter bibliotecario).
Resultados esperados: Gráfico de vía de préstamo de libros listado correctamente.
Evaluación de la prueba: Prueba satisfactoria

Pruebas de Aceptación	
Código :P23	Número de Historia de usuario:23
Nombre: Comprobación de los datos de grafico de pastel mostrando la vía de préstamo de tesis.	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Portada. 4. Listando el gráfico de barras de vía de préstamos de tesis. (Circulante, en sala, inter bibliotecario). 	
Resultados esperados: Gráfico de vía de préstamo de tesis listado correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P24	Número de Historia de usuario:24
Nombre: Comprobación de los datos de grafico de barras mostrando la vía de préstamo de libros.	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Portada. 4. Listando el gráfico de barras de vía de préstamos de revistas. (Circulante, en sala, inter bibliotecario). 	
Resultados esperados: Gráfico de vía de préstamo de revistas listado correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P25	Número de Historia de usuario:25
Nombre: Comprobación de los datos de reporte de capacidades en becas.	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Portada. 4. Listando el total de capacidades en becas. Listando cuantas ocupadas. 	
Resultados esperados: Reporte de capacidades en becas listado correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P26	Número de Historia de usuario:26
Nombre: Comprobación de los datos de reporte mapeado de los edificios.	

Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Portada. 4. Listando los edificios con sus respectivas habitaciones ocupadas o libres.
Resultados esperados: Reporte de mapeado de los edificios listado correctamente.
Evaluación de la prueba: Prueba satisfactoria

Pruebas de Aceptación	
Código :P27	Número de Historia de usuario:27
Nombre: Comprobación de los datos de reporte gráfico de barra de estado constructivo de los edificios.	
Condiciones de Ejecución: Solo el administrador o el usuario que gestiona la biblioteca tiene permisos para realizar esta acción.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se inicia el sistema. 2. El trabajador inicia sección con usuario y contraseña. 3. Selecciona la opción Portada. 4. Reporte gráfico del estado constructivo de los edificios de la residencia estudiantil. (Bueno, Regular, Malo). 	
Resultados esperados: Reporte gráfico de barra de estado constructivo de los edificios listado correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Pruebas de Aceptación	
Código :P29	Número de Historia de usuario:29
Nombre: Comprobación de los datos de listar Trazas de usuarios del sistema.	
Condiciones de Ejecución: Solo el administrador tiene permisos para realizar esta acción.	

Entrada/Pasos de Ejecución:

1. Se inicia el sistema.
2. El administrador inicia sección con usuario y contraseña.
3. Selecciona la opción Acceso a Trazas.
4. Se realizan las siguientes opciones:
 - a. Listar trazas (El sistema lista todas las acciones realizadas por los usuarios del sistema, registrando la fecha y la hora. Se registra en la base de datos satisfactoriamente).
 - b. Buscar trazas. (Se selecciona la opción buscar y se realiza el método de búsqueda específica de trazas en el sistema de forma satisfactoria).

Resultados esperados: Acceso a trazas del sistema gestionado correctamente.

Evaluación de la prueba: Prueba satisfactoria