

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE
CONTROL DE ACCESO BASADO EN CRIPTOMEMORIAS PARA
EL LDICC**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

CHAMORRO FUELTALA MANUEL PATRICIO
patriciochmorrof@gmail.com

DIRECTOR:
ING. MONTENEGRO ARMAS CARLOS ESTALESMIT WILLAM
carlos.montengro@epn.edu.ec

Quito, 31 de julio de 2014

DECLARACIÓN

Yo, Manuel Patricio Chamorro Fweltala, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Manuel Patricio Chamorro Fweltala

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Manuel Patricio Chamorro Fualta, bajo mi supervisión.

Ing. Carlos Montenegro

DIRECTOR DE PROYECTO

AGRADECIMIENTOS

Agradezco a Dios por haberme dado padres incondicionales, hermanos solidarios, y haberme rodeado de amigos que me han apoyado en los buenos y malos momentos.

A mis profesores que a más de impartir sus conocimientos, me enseñaron a ser un profesional con valores como la humildad y respeto. De manera especial agradezco al Ing. José Luis Carrera por compartir sus conocimientos y la guía brindada para la realización de este proyecto.

DEDICATORIA

Recuerdo que cuando era un niño de unos cuatro años, mi profesión anhelada era ser “paracaidista”, en otras palabras quería ser un militar, pero luego para cuando terminé la escuela mis sueños cambiaron totalmente y mi gusto se inclinó por la tecnología. Y todo esto fue gracias a que mis padres tomaron la decisión de comprar una computadora para tener en la casa.

Mi hermano mayor seguía Ingeniería Informática y cada vez que llegaba a la casa me quedaba acompañándolo en las noches a “programar”, aunque sinceramente no sabía lo que hacía, pero así fue como iba teniendo curiosidad por aprender esta profesión.

Y finalmente al ser un chico que ha crecido en el campo, soy consciente del trabajo duro realizado por mis padres y por todo esto les dedico el presente trabajo a Manuel Chamorro mi padre y Cristina Fweltala mi madre, por el apoyo que me han sabido brindar desde niño para cumplir una meta más en mi vida profesional.

CONTENIDO

CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA.....	1
1.1. DESCRIPCIÓN DEL SISTEMA DE CONTROL DE ACCESO	2
1.1.1 SISTEMAS DE CONTROL DE ACCESO.....	2
1.1.1.1. Control de acceso	2
1.1.2 TECNOLOGÍA DE IDENTIFICACIÓN POR RADIO FRECUENCIA RFID.....	3
1.1.2.1. Historia.....	3
1.1.2.2. Arquitectura.....	4
1.1.2.3. Clasificación de etiquetas RFID.....	6
1.1.2.4. Ventajas de la tecnología RFID.....	8
1.2. JUSTIFICACIÓN DE LA METODOLOGÍA DE DESARROLLO	9
1.2.1 METODOLOGÍA DE DESARROLLO XP.....	11
1.3. SELECCIÓN DE LAS HERRAMIENTAS A UTILIZAR.....	21
1.3.1 COMPONENTES DE HARDWARE.....	22
1.3.1.1. Tarjeta RFID	22
1.3.1.2. Lector RFID	22
1.3.1.3. Cerradura Eléctrica.....	22
1.3.1.4. Microcontrolador.....	23
1.3.2 COMPONENTE DE SOFTWARE.....	23
1.3.2.1. Plataforma de Java SE.....	23
1.3.2.2. MVC.....	24
1.3.2.3. NetBeans 7.0	25
1.3.2.4. Gestor de Base de Datos MySQL.....	25
CAPÍTULO 2: PLANIFICACIÓN Y DISEÑO.....	26
2.1. HISTORIAS DE USUARIO	26
2.1.1 HISTORIAS DE USUARIO INICIALES.....	26
2.1.1.1. Gestión de usuarios	26

2.1.1.2.	Personalización de Tarjetas de Acceso.....	27
2.1.1.3.	Registro de accesos al LDICC.....	28
2.1.1.4.	Indicadores de estado de salas y acceso al LDICC.....	29
2.1.1.5.	Mostrar mensaje al usuario del LDICC.....	30
2.1.1.6.	Generación de reportes.....	30
2.1.2	HISTORIAS DE USUARIO FINALES	31
2.1.2.1.	Módulo de gestión de usuarios.....	31
2.1.2.2.	Módulo de reservación de laboratorios	36
2.1.2.3.	Módulo de gestión de entradas.....	39
2.1.2.4.	Módulo de gestión de control de acceso.....	40
2.1.2.5.	Módulo de reportes.....	44
2.2.	PLAN DE ENTREGAS	46
2.2.1	ESTIMACIÓN DE HISTORIAS DE USUARIO	46
2.2.2	ELABORACIÓN DEL PLAN DE ENTREGA.....	47
2.2.2.1.	Iteraciones.....	49
2.2.2.2.	Plan de Entregas.....	81
2.3.	METÁFORA DEL SISTEMA	81
2.3.1	MÓDULO DE GESTIÓN DE USUARIOS.....	82
2.3.2	MÓDULO DE RESERVACIÓN DE LABORATORIOS	83
2.3.3	MÓDULO DE GESTIÓN DE ENTRADAS	83
2.3.4	MÓDULO DE CONTROL DE ACCESO	84
2.3.5	MÓDULO DE REPORTES	84
2.4.	TARJETAS CRC	85
2.4.1	TARJETAS CRC DEL SISTEMA.....	85
2.4.2	ESTRUCTURA DEL SISTEMA DE CONTROL DE ACCESO	86
2.4.3	ESPECIFICACIÓN DE LA ARQUITECTURA DEL SISTEMA.....	88
CAPÍTULO 3: DESARROLLO Y PRUEBAS		97
3.1.	PROGRAMACIÓN.....	97
3.1.1	ESTÁNDARES DE IMPLEMENTACIÓN	97
3.1.2	CODIFICACIÓN DE HISTORIAS DE USUARIO.....	101

3.2. INTEGRACIÓN	110
3.3. IMPLANTACIÓN del Caso de Estudio	111
3.3.1 REQUISITOS DE INSATALACIÓN	111
3.4. PRUEBAS.....	112
3.4.1 PRUEBAS UNITARIAS.....	112
3.4.2 PRUEBAS DE ACEPTACIÓN	116
3.4.2.1. Pruebas de Aceptación del Sistema	116
3.5. ANÁLISIS DE RESULTADOS	140
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.....	143
4.1. CONCLUSIONES	143
4.2. RECOMENDACIONES.....	144
Bibliografía.....	146
GLOSARIO	149
ANEXOS	152

ÍNDICE DE FIGURAS

Figura 1.1: Estructura de un sistema RFID básico	5
Figura 1.2: El proceso completo es un conjunto de iteraciones	15
Figura 1.3: Fases de la metodología XP	16
Figura 1.4: Formato de Historia de Usuario.....	17
Figura 1.5: Tarjeta CRC	20
Figura 1.6: Arquitectura Java SE.....	24
Figura 2.13: Mapa del sitio del tipo usuario Administrador.....	87
Figura 2.14: Arquitectura del Sistema de Control de Acceso	88
Figura 2.15: Diseño de la placa del Circuito de Control de Acceso.....	90
Figura 2.16: Modelo Entidad Relación	91
Figura 2.17: Diagrama de Componentes	92
Figura 2.18: Diagrama de Paquetes.....	95
Figura 3.1: Organización de las capas del sistema en carpetas.	97
Figura 3.2: Tareas de la Historia de Usuario “Gestión de profesores”.	101
Figura 3.3: Persona.java	102
Figura 3.4: Clase DAO – PersonaJpaController.java	103
Figura 3.5: Interfaz Gráfica - Gestión de Usuarios	107
Figura 3.6: Refactorización del Software.....	109
Figura 3.7: Esquema de Control de Acceso.....	110
Figura 3.8: Resultado de prueba unitaria – PersonaControladorTest.java.....	115

ÍNDICE DE TABLAS

Tabla 1.1: Historia de la tecnología RFID.....	4
Tabla 1.2: Bandas de Frecuencias utilizadas en RFID.....	7
Tabla 1.3: Ranking de “agilidad”.....	10
Tabla 1.4: Comparación de metodologías ágiles	11
Tabla 2.1: Historia de usuario: Gestión de usuarios.....	26
Tabla 2.2: Historia de Usuario: Personalización de Tarjetas.....	27
Tabla 2.3: Historia de Usuario: Registro de accesos al LDICC.	28
Tabla 2.4: Indicadores de estado de salas y acceso al LDICC.	29
Tabla 2.5: Historia de Usuario: Mostrar mensaje al usuario del LDICC.	30
Tabla 2.6: Historia de Usuario: Generación de reportes.	30
Tabla 2.7: Historia de Usuario Final: Gestión de profesores.	32
Tabla 2.8: Historia de Usuario Final: Gestión de estudiantes.....	32
Tabla 2.9: Historia de Usuario Final: Gestión de otros usuarios.....	33
Tabla 2.10: Historia de Usuario Final: Gestión de usuarios del sistema.	34
Tabla 2.11: Historia de Usuario: Configuración de Tarjeta RFID.....	34
Tabla 2.12: Historia de Usuario: Registro de Tarjetas.....	35
Tabla 2.13: Historia de Usuario: Asignación de permisos a Tarjeta RFID.....	36
Tabla 2.14: Gestión de materias.	37
Tabla 2.15: Historia de Usuario: Registrar reservación de un laboratorio.	37
Tabla 2.16: Historia de Usuario: Modificar reservación de un laboratorio.	38
Tabla 2.17: Historia de Usuario: Eliminar reservación de un laboratorio.....	39
Tabla 2.18: Historia de Usuario: Gestionar punto de acceso.	39
Tabla 2.19: Historia de Usuario: Asignar punto de acceso.....	40
Tabla 2.20: Historia de Usuario: Gestionar permiso de acceso.....	41
Tabla 2.21: Historia de Usuario: Registrar acceso.	41
Tabla 2.22: Historia de Usuario: Elaboración de circuito con leds.....	42
Tabla 2.23: Historia de Usuario: Imprimir mensaje en pantalla LCD.	43
Tabla 2.24: Historia de Usuario: Generación de reportes Totales.....	44
Tabla 2.25: Historia de Usuario: Generación de reportes Detallados.....	44

Tabla 2.26: Resumen de Historias de usuario en base a la prioridad.	45
Tabla 2.27: Tiempo Calendario	47
Tabla 2.28: Esfuerzo de Desarrollo en base a una persona	47
Tabla 2.29: Historias de Usuario con número de iteración	48
Tabla 2.30: <i>Planning Poker</i> – Primera Iteración	49
Tabla 2.31: Plan de Entregas – Primera Iteración.....	50
Tabla 2.32: Criterios de aceptación – Primera Iteración.....	51
Tabla 2.33: Resultado prueba de aceptación - Primera Iteración.....	51
Tabla 2.34: <i>Planning Poker</i> – Segunda Iteración	54
Tabla 2.35: Plan de Entregas – Segunda Iteración	55
Tabla 2.36: Criterios de aceptación – Segunda Iteración.....	55
Tabla 2.37: Resultado prueba de aceptación - Segunda Iteración.....	56
Tabla 2.38: <i>Planning Poker</i> – Tercera Iteración.	60
Tabla 2.39: Plan de Entregas – Tercera Iteración	61
Tabla 2.40: Criterios de aceptación – Tercera Iteración.....	61
Tabla 2.41: Resultado prueba de aceptación - Tercera Iteración.....	62
Tabla 2.42: <i>Planning Poker</i> – Cuarta Iteración.....	63
Tabla 2.43: Plan de Entregas – Cuarta Iteración.....	65
Tabla 2.44: Criterios de aceptación – Tercera Iteración.....	65
Tabla 2.45: Resultado prueba de aceptación - Cuarta Iteración	66
Tabla 2.46: <i>Planning Poker</i> – Quinta Iteración.....	69
Tabla 2.47: Plan de Entregas – Quinta Iteración.....	70
Tabla 2.48: Criterios de aceptación – Quinta Iteración.....	70
Tabla 2.49: Resultado prueba de aceptación - Quinta Iteración.....	70
Tabla 2.50: <i>Planning Poker</i> – Sexta Iteración.	72
Tabla 2.51: Plan de Entregas – Sexta Iteración	73
Tabla 2.52: Criterios de aceptación – Quinta Iteración.....	74
Tabla 2.53: Resultado prueba de aceptación - Sexta Iteración.....	74
Tabla 2.54: <i>Planning Poker</i> – Séptima Iteración.	77
Tabla 2.55: Plan de Entregas – Séptima Iteración	79
Tabla 2.56: Criterios de aceptación – Séptima Iteración.....	79
Tabla 2.57: Resultado prueba de aceptación - Séptima Iteración.....	80
Tabla 2.58: Plan de Entregas	81

Tabla 2.59: Tarjetas CRC del Sistema	85
Tabla 2.60: Tarjeta CRC. Persona	86
Tabla 2.61: Tarjeta CRC. Controlador Persona.....	86
Tabla 2.62: Componentes de la capa de modelo.....	92
Tabla 2.63: Componentes de la capa de controlador.....	93
Tabla 2.64: Componentes de la capa de vista.	94
Tabla 2.65: Paquetes de la capa de modelo.....	95
Tabla 2.66: Paquetes de la capa de controlador.....	95
Tabla 2.67: Paquetes de la capa de vista.	96
Tabla 3.1: Nomenclatura utilizada en la implementación de la capa de vista.....	98
Tabla 3.2: Estándares de nomenclatura.....	98
Tabla 3.3: Nomenclatura para la capa de modelo.....	99
Tabla 3.4: Nomenclatura para diseño de la base de datos.	100
Tabla 3.5: Pruebas Unitarias.....	113
Tabla 3.6: Formato de Prueba de Aceptación	116
Tabla 3.7: Prueba de aceptación - Registro de Profesores.....	117
Tabla 3.8: Prueba de aceptación - Modificar Profesor	117
Tabla 3.9: Prueba de aceptación - Eliminar Profesor.....	118
Tabla 3.10: Prueba de aceptación - Registro de Estudiante	118
Tabla 3.11: Prueba de aceptación - Modificar Estudiante	119
Tabla 3.12: Prueba de aceptación - Eliminar Estudiante.....	120
Tabla 3.13: Prueba de aceptación - Registro de Otros	120
Tabla 3.14: Prueba de aceptación - Modificar Otros	121
Tabla 3.15: Prueba de aceptación - Eliminar Otro.....	122
Tabla 3.16: Prueba de aceptación - Registro de Usuario del Sistema.....	123
Tabla 3.17: Prueba de aceptación - Modificar Usuarios del Sistema	123
Tabla 3.18: Prueba de aceptación - Eliminar Usuario del Sistema.....	124
Tabla 3.19: Prueba de aceptación - Configuración de Tarjeta RFID	125
Tabla 3.20: Prueba de aceptación - Registro de Tarjetas	125
Tabla 3.21: Prueba de aceptación - Asignación de permisos a Tarjeta RFID	126
Tabla 3.22: Prueba de aceptación - Registrar Materia.....	127
Tabla 3.23: Prueba de aceptación - Modificar Materia.....	128
Tabla 3.24: Prueba de aceptación – Eliminar materia	128

Tabla 3.25: Prueba de aceptación - Registrar reservación de un laboratorio.....	129
Tabla 3.26: Prueba de aceptación - Modificar reservación de un laboratorio.....	130
Tabla 3.27: Prueba de aceptación - Eliminar reservación de un laboratorio	130
Tabla 3.28: Prueba de aceptación - Registrar punto de acceso.....	131
Tabla 3.29: Prueba de aceptación - Modificar punto de acceso.....	132
Tabla 3.30: Prueba de aceptación - Eliminar punto de acceso	132
Tabla 3.31: Prueba de aceptación - Asignar punto de acceso	133
Tabla 3.32: Prueba de aceptación - Registrar permiso de acceso	134
Tabla 3.33: Prueba de aceptación - Modificar permiso de acceso	134
Tabla 3.34: Prueba de aceptación - Eliminar permiso de acceso.....	135
Tabla 3.35: Prueba de aceptación - Registrar acceso.....	136
Tabla 3.36: Prueba de aceptación - Elaboración de circuito con leds	137
Tabla 3.37: Prueba de aceptación - Imprimir mensaje en pantalla LCD	137
Tabla 3.38: Prueba de aceptación - Generación de reportes Totales	138
Tabla 3.39: Prueba de aceptación - Generación de reportes Detallados	138
Tabla 3.40: Resultados de pruebas de aceptación	139

RESUMEN

El presente proyecto de titulación ***Desarrollo de un sistema de control de acceso basado en criptomemorias. Aplicación a un caso de estudio en el Laboratorio del Departamento de Informática y Ciencias de la Computación (LDICC)***, abarca los siguientes temas en cada uno de sus capítulos.

En el Capítulo uno, se describe el actual proceso de control de acceso que se lleva a cabo en el caso de estudio, el LDICC y se describe la problemática asociada a este proceso debido a que no cuenta con un sistema que permita realizar el control de acceso.

Además, contiene la descripción de la solución propuesta, los elementos que se empleará para llevar a cabo el proceso de control de acceso, se justifica la metodología seleccionada y se describe las herramientas empleadas para desarrollar el sistema.

En el Capítulo dos, se inicia el proceso de desarrollo con la planificación y diseño del sistema siguiendo los lineamientos establecidos por la metodología XP. Los artefactos generados son: Historias de Usuario, Plan de Entregas, Metáfora del Sistema, Tarjetas CRC, Diagrama de Base de Datos, Diagrama de Componentes y Diagrama de Paquetes.

En el Capítulo tres, se realiza la fase de desarrollo del sistema siguiendo los estándares definidos en éste capítulo. Luego se realiza la integración e implantación del caso de estudio. Finalmente se realiza un análisis de resultados en base a las pruebas realizadas.

En el Capítulo cuatro, se presenta las conclusiones y recomendaciones, que se ha llegado tras culminar el presente proyecto de titulación.

CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA

El Laboratorio del Departamento de Informática y Ciencias de la Computación (LDICC) se encuentra ubicado en el tercer piso del edificio de Sistemas de la Escuela Politécnica Nacional (EPN). El LDICC está formado por la sala de recepción, cinco salas de computación (LAN, DESARROLLO, WIFI, BDD y ARQUITECTURA) para uso de estudiantes, profesores o personas externas a la EPN. Además el LDICC posee la sala de servidores donde sólo pueden acceder los auxiliares de laboratorio, ayudantes y el jefe del LDICC.

Cuatro años atrás el acceso al LDICC se lo hacía mediante un control físico por una persona en turno en la sala de recepción. Esta persona poseía un listado de todos los estudiantes de la Facultad de Sistemas (FIS) matriculados en el semestre actual, en la que verificaba si el estudiante pertenecía a la FIS, en caso de pertenecer a la FIS el estudiante tomaba el turno para un laboratorio con tiempo máximo de dos horas. Si el laboratorio se encontraba vacío el estudiante podía tomar más horas para uso de un computador en un laboratorio. El problema con éste control, radicó en que todo el registro se lo hacía mediante papel, era lento y no había la posibilidad de almacenar los datos para tomar decisiones en un futuro.

Actualmente en el LDICC no existe un sistema que permita controlar el acceso de los usuarios del laboratorio, esto ha generado problemas en el adecuado control de registro de los usuarios que acceden a las instalaciones del LDICC y sobre todo han existido pérdidas de implementos del laboratorio. Por ésta razón es necesario implementar un sistema que permita controlar el acceso al LDICC, registrando los datos del estudiante y hora de ingreso.

Para poder satisfacer la necesidad del LDICC se desarrollará el Sistema de Control de Acceso, para lo cual en este capítulo abarca la descripción del Sistema Informático, los principios y procesos establecidos por la metodología a utilizar, los conceptos fundamentales, las prácticas, ciclos, fases, roles y pilares de XP como metodología ágil de desarrollo; así como los fundamentos principales de las

herramientas utilizadas en éste sistema.

1.1. DESCRIPCIÓN DEL SISTEMA DE CONTROL DE ACCESO

1.1.1 SISTEMAS DE CONTROL DE ACCESO

Los sistemas de control de acceso (SCA) son una medida de seguridad física, cuyo objetivo es proteger recursos específicos de entes no autorizados. Estos sistemas son usados generalmente para el control de puertas tanto internas como externas en un edificio o instalación, mediante el uso de dispositivos de entrada, tales como lectores de proximidad, magnéticos, códigos de barras o biométricos; identifican y autentican al usuario, permitiendo proceder o denegar el acceso a un recinto [1].

1.1.1.1. Control de acceso

Control de acceso es evitar el uso no autorizado de un recurso por una agente en particular. Existen varios mecanismos de control que pueden ser usados en el manejo de recursos físicos (áreas restringidas a la cual solo personal autorizado tiene acceso) o recursos lógicos (una cuenta bancaria, con un limitado número de personas autorizadas a realizar un retiro o una transacción) o recursos digitales (pueden ser documentos privados, a la cual solo ciertos usuarios tienen permiso para acceder) [1].

De manera específica un control de acceso físico permite que sólo el personal no autorizado, se restrinja la entrada a un bien inmueble, edificio, cuarto o un área específica, únicamente a personal autorizado, además de esta función, sirve para tener un control histórico de entradas de personas a todas las áreas (buscar sospechosos en caso de algún incidente).

El control de acceso físico se engloba en tres hechos fundamentales: quién, cómo y cuándo. Un sistema de control determina quien está permitido a entrar o salir, dónde está permitido entrar o salir y cuando está permitido entrar o salir.

Históricamente esto era parcialmente conseguido a través de cerraduras y llaves, cuando una puerta está cerrada sólo la persona con la llave correcta puede abrirla. Sin embargo estos sistemas no han podido satisfacer las necesidades de

hoy en día, ya que no existe forma de registrar información acerca de quien utilizó la llave, la fecha de uso, tiempo que estuvo utilizando. Debido a esto la tecnología ha evolucionado constantemente para cubrir las demandas.

En la actualidad, tecnológicamente han cambiado ciertas cosas, pero en el fondo persisten las razones y motivos para mantener mecanismos de control de acceso sobre áreas e información que se desea proteger. Los mecanismos de validación han sufrido modificaciones: hoy en día existen controles biométricos, magnéticos, visuales, de radio frecuencia, entre otros [2].

1.1.2 TECNOLOGÍA DE IDENTIFICACIÓN POR RADIO FRECUENCIA RFID

La tecnología de Identificación por Radiofrecuencia es una tecnología que permite capturar e identificar automáticamente información de etiquetas o tags electrónicos. La captura de información se inicia cuando un lector RFID envía una señal a la etiqueta que se encuentra en la proximidad definida, para que la etiqueta proceda con el envío de datos almacenados en ella.

1.1.2.1. Historia

La tecnología RFID no es nueva, tiene sus orígenes en la segunda Guerra Mundial, cuando León Theremin inventó una herramienta para la Unión Soviética ante la necesidad de distinguir cuales eran los aviones amigos o enemigos, desarrolló un sistema de identificación llamado IFF (Identification, Friend or Foe), el cual utilizaba un equipo que los aviones aliados llevaban a bordo (nombre en código "loro", Parrot) y que emitía señales codificadas (graznidos o chillidos, squawk) [3].

En el artículo titulado en inglés "Shrouds of Time The history of RFID" [3], se recopila la evolución de la tecnología RFID hasta el año 2000, ver Tabla 1.1.

Tabla 1.1: Historia de la tecnología RFID.

Década	Evento
1920 - 1930	Pilotos del ejército alemán balanceaban sus aviones al volver a la base para cambiar la señal de radio reflejada en el radar y poder identificarse
1930 - 1940	Británicos inventan el sistema de identificación IFF
1940 - 1950	Uso y perfeccionamiento del radar, importante desarrollo de la II Guerra Mundial. RFID inventado en 1948. Harry Stockman publica artículo titulado "Comunicación por medio de la energía reflejada"
1950 - 1960	Exploraciones iniciales de la tecnología RFID, experimentos de laboratorio.
1960 - 1970	Desarrollo de la teoría de RFID. Inicio de pruebas en aplicaciones de campo.
1970 - 1980	Rápido desarrollo de la tecnología RFID. Se da al inicio de la implementación de la tecnología RFID.
1980 - 1990	Aparecen aplicaciones comerciales RFID.
1990 - 2000	Se norma la tecnología RFID. Se populariza la tecnología RFID. RFID se convierte en parte de la vida cotidiana.
2000 - 2010	Se mejora la capacidad de almacenamiento. Implantación de RFID a nivel mundial [4]. Surgen problemas de seguridad [5].

Fuente: Línea de tiempo de la tecnología RFID [3].

1.1.2.2. Arquitectura

La estructura básica de la tecnología RFID consta de un lector, una tarjeta RFID (tag) y una interfaz inalámbrica, como se muestra en la Figura 1.1.

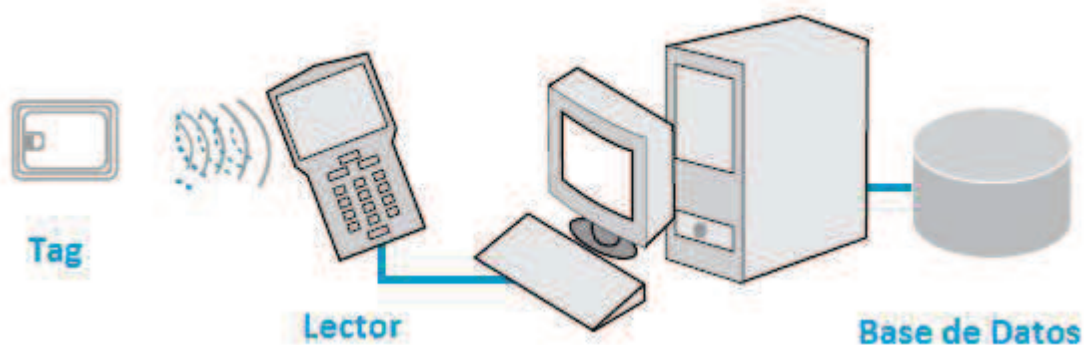


Figura 1.1: Estructura de un sistema RFID básico

Fuente: Elaborado por el Autor

1.1.2.2.1. Tag o etiqueta RFID

En electrónica se denomina “transpondedor”, por su modo de operación que tiene la capacidad de recibir y transmitir señales, pero sólo transmitirá a modo de respuesta ante una posible petición de un dispositivo “transceptor” o lector RFID [6].

Un tag está compuesto por:

- **Una bobina:** Funciona como antena, emite y/o recibe las ondas RF. La energía para activar el chip la recoge del campo RF (en HF del campo electromagnético y en UHF del campo eléctrico). Este proceso es llamado acoplamiento (coupling).
- **Un chip:** que ejecuta los comandos específicos.
- **Una memoria:** para el almacenamiento de información de cada etiqueta
- **Sustrato:** Material que mantiene el chip y la antena juntos y protegidos. Tanto el chip como la antena quedan adjuntados a este material. Además, es el sustrato el que determina la forma física del TAG

Dependiendo de la aplicación final del sistema de identificación, una etiqueta RFID puede tomar formas como: tarjeta, botón, pulsera, o incluso puede ser

inyectado bajo la piel [7].

1.1.2.2.2. Dispositivo lector

En terminología electrónica se denomina “transceptor” y es un dispositivo radio receptor/transmisor, que permite establecer una comunicación con el tag o tarjeta RFID para la obtención de la información necesaria [6].

Los receptores RFID se encuentran generalmente encendidos, transmitiendo continuamente ondas de radio, esperando a que alguna etiqueta ingrese en su área de cobertura. Pero en dispositivos que necesitan conservar energía, esta característica es indeseable; para esto es posible configurar al receptor RFID para que envíe señales de pulsos de radio solo en respuesta a un evento externo [7].

Se puede encontrar lectores con la antena RF integrada en su propio hardware y lectores con conectores de antena RF externos. Dependiendo de la aplicación final se tendrá una configuración u otra.

1.1.2.2.3. Base de datos

Es una plataforma software adicional que permite almacenar sistemáticamente la información procesada por el subsistema hardware (tag y lector). El subsistema software se encarga de gestionar la información generada por el lector RFID.

Entre la base de datos y el dispositivo lector es necesario un interfaz middleware que ejecute un tratamiento previo sobre los datos en bruto que genera el lector [8].

1.1.2.3. Clasificación de etiquetas RFID

Las etiquetas RFID tienen características o capacidades diferentes, para su clasificación se ha tomado sus características principales [8]:

- La tipología
- Frecuencias de trabajo
- Tipo de memoria

1.1.2.3.1. Por la tipología

Una característica de las etiquetas RFID es el modo de alimentación y es uno de los principales factores que determina el coste y vida de la etiqueta [7].

Estas pueden ser:

- **Etiquetas pasivas:** obtienen la energía de la transmisión del lector.
- **Etiquetas activas:** utilizan una batería propia.
- **Etiquetas semipasivas:** utilizan una batería para activar los circuitos del chip, pero la energía para generar la comunicación es la que recoge de las ondas radio del lector (como en las pasivas).

1.1.2.3.2. Frecuencia de trabajo

Las etiquetas RFID de acuerdo a la banda de frecuencia que operan, determina su rango y sector de aplicación, como se muestra en la Tabla 1.2.

Tabla 1.2: Bandas de Frecuencias utilizadas en RFID.

Banda de frecuencias	Descripción	Rango	Uso
125 kHz - 134 KHz	LF(Baja Frecuencia)	Hasta 45 cm.	En animales y llaves inalámbricas.
13,553MHz - 13,567 MHz	HF (Alta Frecuencia)	De 1 a 3 m.	Trazabilidad de productos, movimientos de equipajes de avión o acceso a edificios.
400 MHz - 1000 MHz	UHF (Ultra Alta Frecuencia)	De 3 a 10 m.	Aplicaciones de trazabilidad con tags activos
2.45 GHz 5.4 GHz	Microondas	Más de 10 m.	Para seguimiento y trazabilidad de personas u objetos

Fuente: Tomado de [7].

1.1.2.3.3. Tipo de memoria

Las etiquetas por el tipo de memoria se pueden ser [7]:

- **Solo Lectura (RO):** Las etiquetas no pueden ser personalizadas, sus datos son grabados durante su fabricación y ya no pueden reescribirse.
- **Una escritura, Muchas Lecturas (WORM):** Una etiqueta WORM, permite al usuario configurar o escribir su valor una sola vez; después de modificar el inicial, es imposible cambiarlo.
- **Lectura y Escritura (RW):** Estas etiquetas, pueden ser reprogramadas varias veces, típicamente este número varía entre 10,000 y 100,000 veces, incluso mayores.

1.1.2.4. Ventajas de la tecnología RFID

Las principales ventajas son:

- **Seguridad:** Los datos almacenados en una tarjeta inteligente están protegidos por sofisticados mecanismos de seguridad. Por ello resulta difícil y costoso modificar fraudulentamente los datos o copiar la tarjeta. Los chips están diseñados para realizar cálculos avanzados de criptografía, por ejemplo, para una autenticación básica de una tarjeta es necesario que esta realice al menos 1.600.000 multiplicaciones 16×16 dígitos [9].
- **Capacidad:** Las tarjetas inteligentes pueden almacenar más datos que cualquier otro dispositivo de su tipo. Las tarjetas de banda magnética más utilizadas pueden almacenar hasta 135 bytes de datos, en cambio las tarjetas inteligentes más comunes son capaces de almacenar 8000 bytes de datos. Esta capacidad ha permitido el desarrollo de tecnologías para multiplicación, lo que implica que varias aplicaciones distintas puedan residir al mismo tiempo en una sola tarjeta: Tarjeta de crédito, de débito puntos de fidelidad e identificación de seguridad. Actualmente pueden existir tarjetas inteligentes de hasta 2 Megabytes de capacidad [9].

- *Flexibilidad*: La tecnología de tarjetas Inteligentes es compatible con los principales tipos de sistemas operativos de computadoras y dispositivos móviles. Estas tarjetas poseen un entorno de programación que permite crear, almacenar o suprimir aplicaciones en ellas lo que significa que es posible hacer tarjetas inteligentes “a la medida” seleccionando para la tarjeta las aplicaciones que se adapten a las circunstancias y necesidades de cada persona [9].
- *Adaptabilidad*: Las tarjetas inteligentes son elementos que dadas sus características se pueden insertar de forma natural en toda la cadena de generación y gestión de información en cualquier sector económico, educativo, financiero, comercial, identificación, telefonía, salud, etc.

1.2. JUSTIFICACIÓN DE LA METODOLOGÍA DE DESARROLLO

El uso de una metodología de desarrollo en un proyecto de software se logra que éste sea de calidad y sobre todo a tener éxito en el desarrollo del proyecto. Pero hoy en día existen varias metodologías para el desarrollo de proyectos de software, por tanto surge la necesidad de hacer una comparación entre varias metodologías y elegir la más conveniente.

En los estudios realizados por James Highsmith en su libro “*Agile Software Development Ecosystems*” compara distintas metodologías ágiles en base a tres parámetros: vista del sistema como algo cambiante, según la colaboración entre los miembros del equipo y características más específicas de la propia metodología como la simplicidad, adaptabilidad, excelencia técnica y prácticas de colaboración, como se muestra en la Tabla 1.3.

Tabla 1.3: Ranking de “agilidad”

	ASD	Crystal	DSDM	FDD	LD	Scrum	XP
Sistema como algo cambiante	5	4	3	3	4	5	5
Colaboración	5	5	4	4	4	5	5
Características Metodología							
Resultados	5	5	4	4	4	5	5
Simplicidad	4	4	3	5	3	5	5
Adaptabilidad	5	5	3	3	4	4	3
Excelencia técnica	3	3	4	4	4	3	4
Prácticas de colaboración	5	5	4	3	3	4	5
Media CM	4,4	4,4	3,6	3,8	4	4,2	4
Media Total	4,8	4,5	3,6	3,6	4	4,7	5

(Los valores más altos representan una mayor agilidad)

Fuente: Agile Software Development Ecosystems [10].

También se ha considerado el estudio realizado por el Instituto de Ingeniería de Software de la Universidad “The Carnegie Mellon University”, en el cual de acuerdo a sus experiencias en desarrollo se asigna valores numéricos a diferentes características de las metodologías, determinando así la más “ágil”. La Tabla 1.4 muestra los resultados obtenidos en el análisis elaborado por “The Carnegie Mellon University”.

Tabla 1.4: Comparación de metodologías ágiles

Características de la metodología	DSD	FDD	SCRUM	XP
Resultados	4	4	5	5
Simplicidad	3	5	5	5
Adaptabilidad	3	3	4	3
Excelencia Técnica	4	4	3	4
Prácticas de colaboración	4	3	4	5
Media Total	3.6	3.8	4.2	4.4

(Los valores más altos representan una mayor agilidad)

Fuente: Carnegie Mellon University

Tomando en cuenta los resultados del análisis de la Tabla 1.3 y Tabla 1.4, se concluye que la metodología que tiene mayor ranking en cuanto a características de agilidad, es la metodología XP y debido a las características que tiene éste proyecto, como la interacción continua con el cliente, manejo de prototipos, el uso de una tecnología poco conocida y el tiempo para su entrega se determinó que la metodología XP es la mejor.

1.2.1 METODOLOGÍA DE DESARROLLO XP

Una metodología se define como un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar nuevo software de calidad [11].

Metodología ágil

El término "ágil" aplicado al desarrollo de software, nace en una reunión celebrada en febrero de 2001 en Utah, EEUU por un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y

dirigidos por la documentación que se genera en cada una de las actividades desarrolladas [12].

Metodología de desarrollo Extreme Programming XP

Ésta metodología se basa principalmente en la simplicidad y agilidad. Promueve trabajo en equipo, la capacitación de los desarrolladores y se centra en fomentar las relaciones interpersonales como parte fundamental para el éxito del proyecto [13].

La metodología XP exige que el cliente sea involucrado con el equipo de desarrollo, el grado de colaboración del cliente es otro factor que determina el éxito del proyecto.

- Las cuatro variables

La metodología XP define cuatro variables para proyectos software: *coste*, *tiempo*, *calidad* y *ámbito* [13]. El coste, tiempo y calidad puede ser definido por fuerzas externas (jefes de proyecto y clientes), mientras la variable ámbito debe ser definida por el equipo de desarrollo en función de las otras tres variables.

- El coste del cambio

Generalmente se sabe que el coste del cambio para un proyecto software va incrementando exponencialmente a medida que se va avanzando. La metodología XP propone actuar de manera distinta frente a los cambios que puede sufrir un proyecto software, empleando buenas prácticas de programación y una adecuada tecnología [13].

Realizar un diseño lo más sencillo, de tal forma que se obtenga lo más indispensable y mediante los test continuos en cada iteración del proyecto permite que los cambios no sean un problema sino algo útil en el proceso de desarrollo. Con esto se evita tomar decisiones grandes al inicio del proyecto ya que se van tomando acciones de cambio en la marcha del proyecto.

- Los cuatro valores

En el ciclo de vida de un proyecto software los cambios están inmersos, provocando un cambio en los requisitos, las reglas de negocio, el personal, la tecnología, en sí todo va a cambiar [13].

Frente a ésta situación mencionada, la filosofía de XP menciona los cuatro valores siguientes [13]:

- Comunicación: XP exige que el equipo de desarrollo mantenga una comunicación constante durante todo el ciclo de vida del proyecto software.
- Sencillez: Los programadores deben realizar software funcional lo más simple que se pueda. XP enseña hacer una cosa sencilla hoy y pagar un poco más para mañana, si es necesario, que hacer una cosa complicada hoy y no utilizarla después.
- Retroalimentación o Feedback: Existen varios niveles de retroalimentación.
 - Retroalimentación del sistema: Después de implementar cambios se necesita hacer pruebas unitarias e integrales de manera periódica y así conocer el estado del sistema software [14].
 - Pruebas de aceptación: El cliente participa en el momento de realizar las pruebas, para que exista un intercambio de opiniones con el equipo de desarrollo que ayude a determinar el fallo y así realizar los cambios [14].
 - Nuevos requerimientos por parte del cliente: Tras la captura de nuevos requerimientos el equipo de desarrollo debe hacer una estimación del tiempo que va a tomar realizar estos cambios [14].
- Coraje: Se necesita de coraje ante los problemas o cambios que surgen para enfrentarlos. En ocasiones es necesario desechar lo que ya se tiene y

empezar haciendo algo mejor y más sencillo que cumpla con los requerimientos del usuario [13].

- Las cuatro actividades básicas

XP propone cuatro actividades que deben ser cumplidas tomando en cuenta los valores mencionados anteriormente [13], como se ilustra en la Figura 1.2.

- *Codificar*: Mediante el código se expresa las ideas del programador para brindar una solución a un problema. Con el código fuente se puede aprender y realizar mejoras al sistema [13].
- *Pruebas al sistema*: Mediante estas pruebas el desarrollador determina si el sistema hace lo que tiene que hacer. Las pruebas se las realiza mientras se va programando aunque esto provoca una disminución en la productividad del desarrollador. Pero se logra disminuir errores, obteniendo mayor precisión y rapidez ante un error y un sistema tal y como desea el cliente [13].
- *Escuchar*: El cliente es el que mejor sabe sobre los problemas que se tienen que resolver. Los desarrolladores deben saber escuchar ya que el éxito del proyecto dependerá mucho de la interpretación correcta que hagan sobre los requisitos mencionados por el cliente [13].
- *Diseñar*: El diseño del sistema a implementar debe ser sencillo o si es complejo se debe dividir en varios. Con un diseño sencillo se puede comprender la lógica del sistema [13].

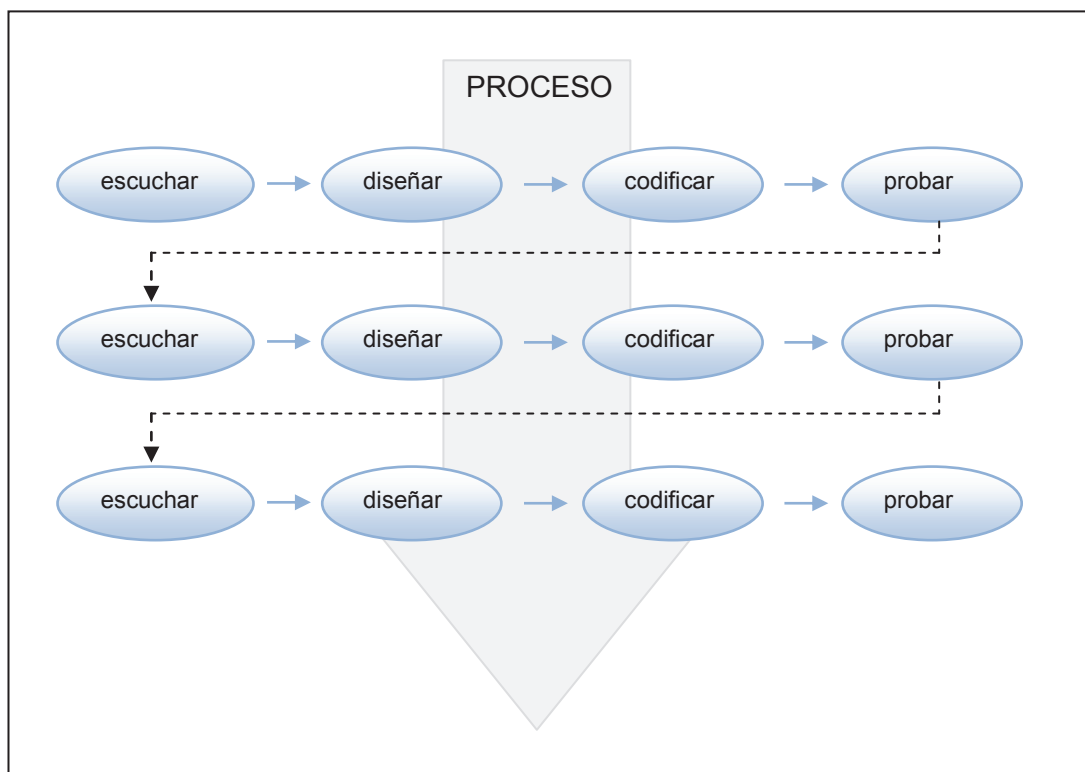


Figura 1.2: El proceso completo es un conjunto de iteraciones

Fuente: Metodos Ágiles [15].

- Fases de la metodología XP

La metodología XP define el ciclo de vida para un proyecto software con las siguientes fases [15], como se muestra en la Figura 1.3.

- Planificación.
- Diseño.
- Desarrollo.
- Pruebas.

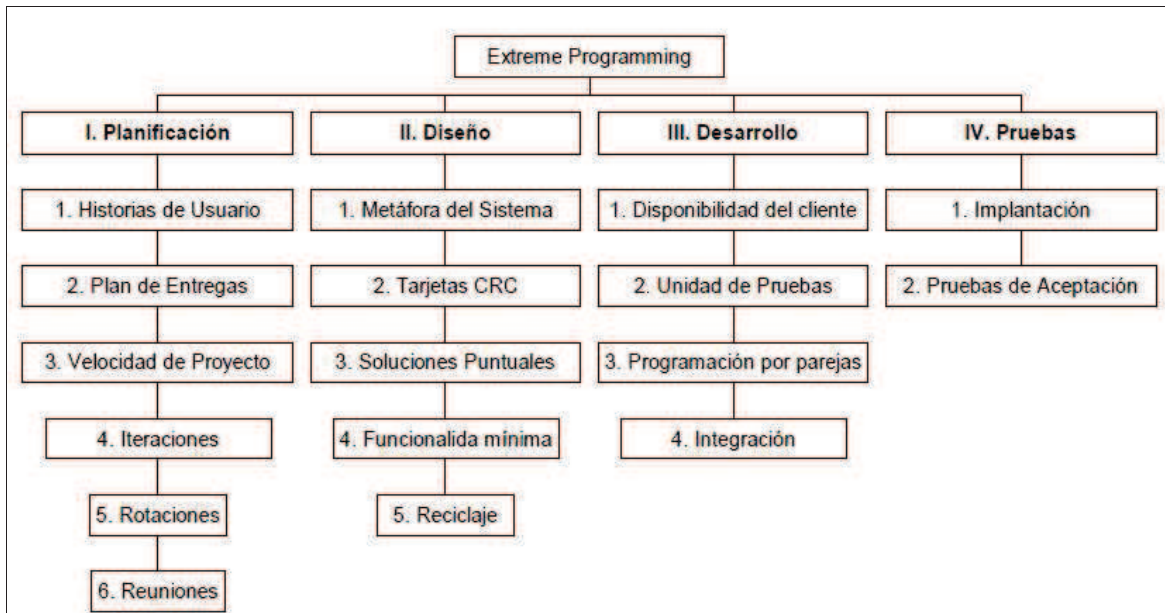


Figura 1.3: Fases de la metodología XP

Fuente: Métodos Ágiles [15].

- Planificación

La metodología XP plantea esta fase como un proceso de constante diálogo entre el cliente y el equipo de desarrollo del proyecto.

El equipo de desarrollo debe hacer la captura de requerimientos, hacer una estimación para implementar las funcionalidades especificadas por el cliente, organizar la cultura de trabajo y realizar la planificación detallada dentro de cada versión [15].

La fase de planificación comprende de cinco procesos [15]:

- Redacción de Historias de usuarios.
- Elaboración del Plan de Entregas.
- Control de la velocidad del proyecto.
- División del proyecto en iteraciones.
- Definición del Plan de Iteración.

- Rotación de personal.
- Corrección de la propia metodología XP cuando falla.
- Redacción de Historias de Usuarios

La elaboración de las Historias de Usuario tiene similar propósito que los casos de uso. Las Historias de Usuario son escritas por el cliente en un lenguaje no técnico sin hacer hincapié en los detalles. Se utilizan para especificar los requerimientos que el sistema debe cumplir [15].

Historia N° :	
Número:	Usuario:
Nombre Historia	
Estimación: # puntos	Prioridad: Baja/Media/Alta
Programador Responsable:	
Descripción:	
Criterios de Aceptación:	

Figura 1.4: Formato de Historia de Usuario.

Fuente: Metodos Ágiles [15].

En cada Historia de Usuario, el equipo de desarrolladores debe hacer una estimación del tiempo de desarrollo, el tiempo ideal es entre 1 y 3 semanas. Si el tiempo sobrepasa las tres semanas se debe dividir en dos partes, si el tiempo es menor a una semana, entonces se debe unir dos o más Historias de Usuario [15].

En la fase de pruebas, las Historias de Usuario se utilizan para verificar si el programa fue implementado correctamente en base a los Criterios de Aceptación que cada Historia de Usuario tiene [15].

- Elaboración de Plan de Entregas

Una vez redactado las Historias de Usuario se debe hacer el Plan de Entregas, en inglés "Release Plan", donde se especifica la prioridad con la que serán implementadas las Historias de Usuario y las fechas en las que se publicarán estas versiones [15].

El Plan de Entregas debe dejar claro cuatro factores: las Historias de Usuario que se deben implementar en cada versión, el tiempo que tardarán en desarrollarse y publicarse las versiones del programa, el número de personas que trabajarán en el desarrollo y cómo se evaluará la calidad del trabajo realizado [15].

- Control de la velocidad del proyecto

La velocidad del proyecto está dada por el número de Historias de Usuario que son implementadas en un tiempo determinado. Esta medida permite controlar que todas las tareas se puedan implementar en el tiempo que dispone la iteración [15].

Es conveniente reevaluar esta medida cada 3 ó 4 iteraciones y si se aprecia que no es adecuada hay que negociar con el cliente un nuevo Plan de Entregas [15].

- División del Proyecto en Iteraciones

Este proceso consiste en dividir el proyecto en iteraciones entre 1 y 3 semanas aproximadamente de duración. También las Historias de Usuario se deben dividir en tareas entre 1 y 3 días de duración que se deben asignar a los programadores [15].

Para determinar si una iteración se encuentra sobrecargada se utiliza la velocidad del proyecto y la suma de los días que costará desarrollar todas las tareas de la iteración no debería sobrepasar la velocidad del proyecto de la iteración anterior [15].

Si la iteración está sobrecargada, el cliente deberá decidir que historias de usuario retrasar a una iteración posterior. De lo contrario se debe adicionar historias de usuario para completar la iteración [15].

- Definición del Plan de Iteración

En cada iteración se debe empezar con la selección de las Historias de Usuario a implementar definidas en el Plan de Entregas y con esto generar el Plan de Iteraciones. Todas las tareas que no constan en el Plan de Iteraciones no deben ser implementadas. En el Plan de Iteraciones se debe también incluir las pruebas de aceptación fallidas para que sean corregidas [15].

- Rotación de personal

Las rotaciones del personal permiten que las personas se involucren y conozcan el funcionamiento del sistema en general. Y con la programación en parejas se reduce el riesgo de que solo una persona sea capaz de trabajar en un área concreta, ya que así se entrena a un nuevo miembro [15].

- Corrección de la propia metodología XP cuando falla

Se debe corregir el proceso cuando éste falle. Al iniciar un proyecto aplicando la metodología XP es posible cambiar aquello que no funciona en el proyecto y todos los cambios realizados deben ser comunicados al resto del equipo [15].

- Diseño

Generalmente en esta fase se crea la parte visual del proyecto como la interfaz que tendrá el usuario o cliente del proyecto [15].

XP sugiere las siguientes recomendaciones en el momento del diseño [15]:

- Simplicidad.
- Elección de una metáfora para el sistema.
- Usar tarjetas CRC en las reuniones de diseño.

- Simplicidad

El diseño debe ser simple en lo posible, un diseño simple es fácil de comprender e implementar y toma menos tiempo y esfuerzo para implementarlo. Si alguna parte de la implementación es compleja, se debe replantear (divide y vencerás) [15].

- Elección de una metáfora para el sistema

“Una metáfora para el sistema es una historia que todo el mundo puede contar acerca de cómo funciona el sistema” (Kent Beck). Las metáforas permiten entender de mejor manera lo que se va a implementar, esto es describir la funcionalidad de cada historia de usuario empleando un lenguaje técnico [15].

- Usar tarjetas CRC en las reuniones de diseño

Las tarjetas C.R.C (Class, Responsibilities and Collaboration) permiten que los desarrolladores de software obtengan hábitos de desarrollo orientada a objetos. Las tarjetas CRC representan objetos en los que se especifica la clase del objeto, las responsabilidades u objetivos que debe cumplir el objeto y las clases que colaboran con cada responsabilidad [15]. En la Figura 1.5 se ilustra el modelo de una tarjeta CRC.

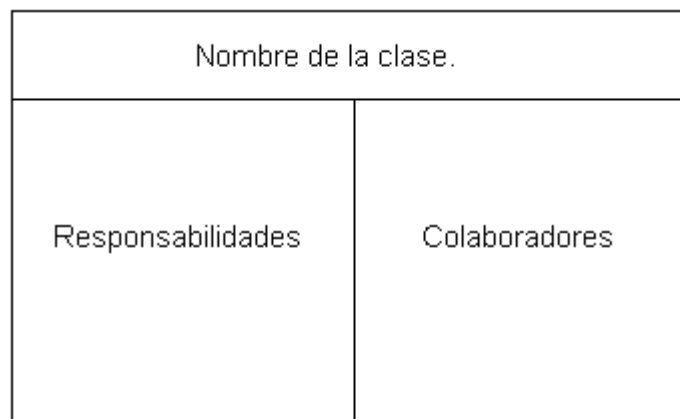


Figura 1.5: Tarjeta CRC

Fuente: Metodos Ágiles [15].

- Desarrollo

En la fase de desarrollo la metodología XP exige que se cumpla con las siguientes características [15]:

- Disponibilidad del cliente.
- Escritura del código según los estándares.
- Implementar unidad de pruebas previo el desarrollo.
- Programar todo el código por parejas.
- Una sola pareja es encargada de la integración del código.
- Integración frecuente con las versiones recientes.
- Dejar las optimizaciones para el final.
- No trabajar más de 40 horas semanales.

- Pruebas

Las pruebas o test en la metodología XP es el punto principal, ya que es el medio por el cual se comprueba el funcionamiento implementado se ejecute correctamente [15].

Para esta fase se debe cumplir con las siguientes características [15]:

- Todo el código debe ir con su unidad de pruebas.
- Hacer todas las pruebas antes de implantar.
- Se deben ejecutar pruebas de aceptación a menudo y publicar los resultados.

1.3. SELECCIÓN DE LAS HERRAMIENTAS A UTILIZAR

Las herramientas de desarrollo a utilizar son seleccionadas en base a factores de compatibilidad entre los componentes de hardware y software del sistema de control de acceso.

1.3.1 COMPONENTES DE HARDWARE

Comprenden todos los dispositivos de hardware empleados en el sistema de control de acceso, estos son:

- Tarjeta RFID.
- Lectora RFID.
- Cerradura Eléctrica.
- Microcontrolador.

1.3.1.1. Tarjeta RFID

Rol del componente: Proporciona la información de identificación del usuario almacenada en su memoria.

Componente empleado.- La tarjeta RFID empleada es de la serie MIFARE MF1S503 de 1KB; está dividida en 16 zonas y por cada zona existen 4 bloques de memoria de 16 bytes cada uno. De los 4 bloques, 3 son para datos y uno para configuración de claves y acceso a la zona. Para más información revisar el ANEXO A.

1.3.1.2. Lector RFID

Rol del componente.- Este dispositivo permite capturar la información de la tarjeta RFID y enviar la información al computador.

Componente empleado.- El lector utilizado es de la serie OMNIKEY 5321 CL que se conecta con el computador a través de una interfaz USB.

1.3.1.3. Cerradura Eléctrica

Rol del componente.- Luego de procesar la información recibida del lector se debe elegir qué operación hacer sobre la cerradura eléctrica, si abrir o no.

Componente empleado.- El componente empleado es una cerradura eléctrica.

1.3.1.4. Microcontrolador

Rol del componente.- Este componente permite abrir la chapa eléctrica y encender leds mediante comandos enviados desde la CPU, utilizando una comunicación serial.

Componente empleado.- El componente fue construido utilizando un microcontrolador de serie PIC16F877A –I/P de Microchip y ensamblado en una placa impresa con un puerto serial para la comunicación.

1.3.2 COMPONENTE DE SOFTWARE

En esta sección se describe las herramientas utilizadas para el desarrollo del sistema de control de acceso.

1.3.2.1. Plataforma de Java SE

Java SE (Java Platform, Standard Edition) es una plataforma de programación parte de la plataforma Java, permite desarrollar y ejecutar aplicaciones cliente/servidor [16]. Java SE ofrece un conjunto de componentes como se muestra en la Figura 1.6.

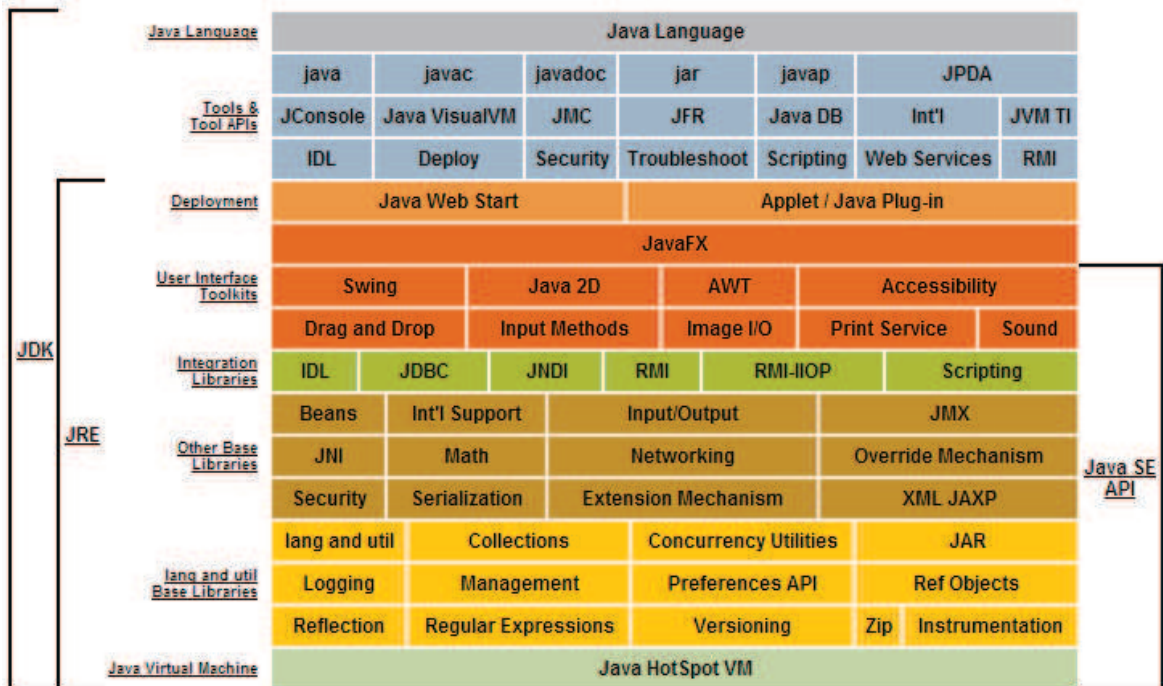


Figura 1.6: Arquitectura Java SE

Fuente: Tomado de <http://docs.oracle.com> [16]

1.3.2.2. MVC

El desarrollo del proyecto se va a realizar empleando el patrón de diseño Modelo-Vista-Controlador. Este patrón de diseño separa claramente tres elementos claves en el desarrollo de aplicaciones: la parte del diseño (Vista), la parte del negocio (Modelo) y la parte del control (Controlador). A continuación se detalla cada una de ellas [17].

- **Modelo:** Implementa la lógica del negocio. Son un conjunto de clases que representan la información del mundo real sin tomar en cuenta ni la forma en la que esa información se va a mostrar, ni los mecanismos que permiten que esos datos estén dentro del Modelo [17].
- **Vista:** Hace referencia comúnmente a las interfaces de usuario que se utilizan para representar la información almacenada en el Modelo [17].
- **Controlador:** Es un objeto que se encarga de dirigir el flujo de eventos de la aplicación. A partir de estos eventos el controlador se encarga de actualizar el modelo o de actualizar la vista [17].

1.3.2.3. NetBeans 7.0

Es un entorno de desarrollo integrado (IDE) de código abierto que permite escribir, compilar, depurar y ejecutar programas que emplean un lenguaje de programación Java, pero puede ser utilizado para otros lenguajes de desarrollo [18].

1.3.2.4. Gestor de Base de Datos MySQL

Se utiliza el motor de base de datos MySQL 5 para el desarrollo del presente proyecto, por ser un software Open Source, brindar estabilidad, confiabilidad [19] y sobre todo por la disponibilidad del servidor en el laboratorio del LDICC.

-

CAPÍTULO 2: PLANIFICACIÓN Y DISEÑO

En la fase de planificación se realizará los artefactos correspondientes a:

- Historias de Usuario
- Plan de Entregas

Y en la parte de diseño se realizará los artefactos correspondientes a:

- Metáfora del Sistema
- Tarjetas CRC
- Diagrama de Base de Datos
- Diagrama de Componentes
- Diagrama de Paquetes

2.1. HISTORIAS DE USUARIO

Mediante el empleo de las Historias de Usuario se realiza la captura de requerimientos del usuario que posteriormente deberán ser implementadas o desarrolladas. Cabe recalcar que existen dos tipos de Historias de Usuario: las iniciales, son redactadas por el usuario empleando un lenguaje natural y las finales, son redactadas en un lenguaje técnico.

2.1.1 HISTORIAS DE USUARIO INICIALES

2.1.1.1. Gestión de usuarios

Tabla 2.1: Historia de usuario: Gestión de usuarios

HISTORIA DE USUARIO	
Número: 01	Usuario: Administrador
Nombre Historia: Gestión de usuarios	

Estimación: N/A	Prioridad: Baja/Media/Alta
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar o modificar la información de un estudiante, profesor, personal de limpieza o invitados para tener una base de datos del personal.	
Criterios de Aceptación: Para registrar una persona para que pueda acceder al LDICC se ingresa la cédula, nombres, apellidos, permiso de acceso, fecha de registro, fecha de modificación y fecha de caducidad. Adicionalmente si se trata de estudiantes, se toma en cuenta el número único, carrera, semestre, materias inscritas y horario de materias. Si es personal docente, se adiciona los registros de materias que dicta. Para el personal administrativo/limpieza/invitados no se registra ningún dato adicional.	

Fuente: Elaborador por el Autor.

2.1.1.2. Personalización de Tarjetas de Acceso

Tabla 2.2: Historia de Usuario: Personalización de Tarjetas

HISTORIA DE USUARIO	
Número: 02	Usuario: Administrador
Nombre Historia: Personalización de Tarjetas.	
Estimación: N/A	Prioridad: Baja/Media/Alta
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito personalizar la tarjeta RFID para cada persona que quiera acceder al LDICC, con la finalidad de identificar el acceso.	
Criterios de Aceptación:	

- Se puede asignar una tarjeta RFID a una persona que consta en la BDD del sistema a elaborar.
- Se registra el número de cédula, el número único, nombres, apellidos, fecha de emisión y caducidad de la tarjeta y tipo de acceso en la tarjeta RFID.
- Los datos se deben capturar mediante un formulario

Fuente: Elaborador por el Autor.

2.1.1.3. Registro de accesos al LDICC

Tabla 2.3: Historia de Usuario: Registro de accesos al LDICC.

HISTORIA DE USUARIO	
Número: 03	Usuario: Administrador
Nombre Historia: Registro de accesos al LDICC.	
Estimación: N/A	Prioridad: Baja/Media/Alta
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito controlar el acceso físico al LDICC para determinar si ingresaron estudiantes, personal administrativo, limpieza o invitados.</p>	
<p>Criterios de Aceptación:</p> <p>Cualquier persona para poder acceder a un laboratorio o a las oficinas del LDICC debe presentar una tarjeta RFID al lector, seguido de esto el sistema se encarga de recoger los datos de la tarjeta RFID a través del lector para posteriormente sean analizados y el sistema se encargue de permitir o denegar el acceso.</p> <p>Cuando el sistema permite el acceso:</p> <ul style="list-style-type: none"> • Para un estudiante que ingresa a recibir clases se indica la sala asignada encendiendo un led azul, y se registra la fecha y hora de ingreso. En cambio sí es un estudiante que ingresa a realizar tareas, se indica un laboratorio que se encuentra disponible y se registra la fecha y 	

hora de acceso.

- Al personal administrativo, limpieza o invitados se registra la fecha y hora de ingreso.
- Se debe mantener un historial de los accesos al LDICC.

Fuente: Elaborador por el Autor.

2.1.1.4. Indicadores de estado de salas y acceso al LDICC

Tabla 2.4: Indicadores de estado de salas y acceso al LDICC.

HISTORIA DE USUARIO	
Número: 04	Usuario: Sistema
Nombre Historia: Indicadores de estado de salas y acceso al LDICC.	
Estimación: N/A	Prioridad: Baja/Media/Alta
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito que el sistema alerte por medio de leds el estado de uso de las salas del LDICC y el resultado del acceso físico, para informar a los usuarios del LDICC.</p>	
<p>Criterios de Aceptación:</p> <p>Para el caso de estudio se debe fabricar un dispositivo en donde se debe colocar leds con el objetivo de informar al usuario el estado de las salas y además informar al usuario si se le negó o no el acceso.</p> <p>Los leds muestran los siguientes estados:</p> <ul style="list-style-type: none"> • Estados para control de acceso: <ul style="list-style-type: none"> ○ Led Rojo: La puerta no se abre, deniega el acceso y se enciende el led durante dos segundos ○ Led Verde: La puerta se abre para permitir el acceso. Permite el acceso y se enciende el led durante dos segundos. • Estados para una sala de computación: <ul style="list-style-type: none"> ○ Led Rojo: Sala en clases, sala ocupada. Permanece encendido y se apaga solamente cuando cambia a un estado de sala libre. 	

- Led Verde: Sala libre. Permanece encendido mientras la sala no tiene reservado hora de clases. Se apaga sólo cuando pasa a sala ocupada.
- Led Azul: Sala asignada, se enciende durante dos segundos y se informa al usuario a que aula debe dirigirse.
- La tarjeta de acceso debe ser personalizada previamente.

Fuente: Elaborador por el Autor.

2.1.1.5. Mostrar mensaje al usuario del LDICC

Tabla 2.5: Historia de Usuario: Mostrar mensaje al usuario del LDICC.

HISTORIA DE USUARIO	
Número: 05	Usuario: Sistema
Nombre Historia: Mostrar mensaje al usuario del LDICC.	
Estimación: N/A	Prioridad: Baja/Media/Alta
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito que el sistema imprima un mensaje en una pantalla LCD, para informar a la persona que quiere ingresar el resultado del proceso de control acceso.	
Criterios de Aceptación: Cuando una persona presente una tarjeta al lector, en la pantalla LCD del dispositivo a fabricar se debe mostrar un mensaje de bienvenida si el acceso es correcto.	

Fuente: Elaborador por el Autor.

2.1.1.6. Generación de reportes

Tabla 2.6: Historia de Usuario: Generación de reportes.

HISTORIA DE USUARIO	
Número: 06	Usuario: Administrador
Nombre Historia: Generación de reportes.	
Estimación: N/A	Prioridad: Baja/Media/Alta

Programador Responsable: Patricio Chamorro
<p>Descripción:</p> <p>Yo como administrador, necesito generar reportes de acceso al laboratorio del LDICC en un intervalo de fechas especificadas, para saber quién hizo uso de las instalaciones del LDICC.</p>
<p>Criterios de Aceptación:</p> <p>Los reportes deben detallar la persona, la hora, fecha de ingreso y resumirse utilizando tablas y gráficos.</p>

Fuente: Elaborador por el Autor.

2.1.2 HISTORIAS DE USUARIO FINALES

Tomando en cuenta las primeras Historias de Usuario se acuerda dividir las historias en los siguientes módulos:

- Módulo de gestión de usuarios.
- Módulo de gestión de reservación de laboratorios.
- Módulo de gestión de entradas.
- Módulo de gestión de control de acceso.
- Módulo de gestión de reportes.

2.1.2.1. Módulo de gestión de usuarios

Este módulo permite a un administrador registrar, modificar y eliminar usuarios tipo estudiante, profesor, personal administrativo, limpieza y otros. Adicionalmente se registra y actualiza la información de una persona en la tarjeta RFID asignada, así como también gestionar cada tarjeta emitida y cambiar o eliminar sus permisos de acceso asignados por entrada o punto de acceso.

2.1.2.1.1. *Registro de usuarios*

Tabla 2.7: Historia de Usuario Final: Gestión de profesores.

HISTORIA DE USUARIO	
Número: 07	Usuario: Administrador
Nombre Historia: Gestión de profesores.	
Estimación: N/A	Prioridad: 1
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar, modificar o eliminar la información de un profesor, para tener una base de datos de los profesores.	
Criterios de Aceptación: Mediante una interfaz gráfica, se registra la siguiente información para un profesor: <ul style="list-style-type: none"> • Número de cédula, nombres, apellidos y materias que dicta. • La asignación de materias a un profesor es opcional. La información de un profesor que se modifica son: <ul style="list-style-type: none"> • Número de cédula, nombres, apellidos y materias que dicta. Se puede eliminar un profesor, solamente si no tiene registros de accesos.	

Fuente: Elaborador por el Autor.

Tabla 2.8: Historia de Usuario Final: Gestión de estudiantes.

HISTORIA DE USUARIO	
Número: 08	Usuario: Administrador
Nombre Historia: Gestión de estudiantes.	
Estimación: N/A	Prioridad: 2
Programador Responsable: Patricio Chamorro	
Descripción:	

Yo como administrador, necesito registrar, modificar o eliminar la información de un estudiante, para tener una base de datos de estudiantes.

Criterios de Aceptación:

Mediante una interfaz gráfica, la información de un Estudiante a registrar es:

- Número de cédula, número único, nombres, apellidos y lista de materias que recibe en el periodo actual.
- Los campos obligatorios son el número único, nombres, apellidos y la lista de materias que recibe.

La información de un estudiante a modificar es:

- Número de cédula, número único, nombres, apellidos y lista de materias que recibe en el periodo actual.

Se puede eliminar un estudiante, solamente si no tiene registros de accesos.

Fuente: Elaborador por el Autor.

Tabla 2.9: Historia de Usuario Final: Gestión de otros usuarios

HISTORIA DE USUARIO	
Número: 09	Usuario: Administrador
Nombre Historia: Gestión de otros usuarios.	
Estimación: N/A	Prioridad: 2
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito registrar, modificar o eliminar la información de usuarios como personal administrativo, de limpieza u otros, para tener una base de datos de dicho personal.</p>	
<p>Criterios de Aceptación:</p> <p>Mediante una interfaz gráfica, la información para el Personal Administrativo, Personal de Limpieza u Otros, se registra:</p> <ul style="list-style-type: none"> • Número de cédula, nombres y apellidos. • Los campos obligatorios son los nombres, apellidos y la cédula. 	

La información a modificar es:

- Número de cédula, nombres y apellidos.

Se puede eliminar a un Personal Administrativo, Personal de Limpieza u Otros, solamente si no tiene registros de accesos.

Fuente: Elaborador por el Autor.

Tabla 2.10: Historia de Usuario Final: Gestión de usuarios del sistema.

HISTORIA DE USUARIO	
Número: 10	Usuario: Administrador
Nombre Historia: Gestión usuarios del sistema.	
Estimación: N/A	Prioridad: 8
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar, modificar o eliminar la información de los usuarios del sistema, para restringir el acceso a la administración del sistema.	
Criterios de Aceptación: Mediante una interfaz gráfica, la información de un Usuario del Sistema a registrar es: <ul style="list-style-type: none"> • Nombre de Usuario y contraseña. • Los campos obligatorios son el nombre de usuario y contraseña. La información de un Usuario del Sistema a modificar es: <ul style="list-style-type: none"> • Nombre de Usuario y contraseña. • El nombre de usuario no se puede duplicar. Se puede eliminar un usuario del sistema, solamente si no se trata del usuario administrador.	

Fuente: Elaborador por el Autor.

2.1.2.1.2. Personalización de Tarjetas de Acceso

Tabla 2.11: Historia de Usuario: Configuración de Tarjeta RFID.

HISTORIA DE USUARIO	
	Usuario: Sistema

Número: 11	
Nombre Historia: Configuración de Tarjeta RFID.	
Estimación: N/A	Prioridad: 5
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito que el sistema configure las credenciales de acceso que vienen por defecto con una nueva configuración, para evitar el uso no autorizado de las zonas de la memoria de la tarjeta RFID.</p>	
<p>Criterios de satisfacción:</p> <p>Mediante una interfaz gráfica se debe realizar la configuración de una tarjeta RFID, esto consiste en cambiar las claves de acceso por defecto (FFFFFFFFFFFF) hexadecimal a una nueva contraseña en los sectores que se va a registrar la información</p>	

Fuente: Elaborador por el Autor.

Tabla 2.12: Historia de Usuario: Registro de Tarjetas.

HISTORIA DE USUARIO	
Número: 12	Usuario: Administrador
Nombre Historia: Registro de Tarjetas.	
Estimación: N/A	Prioridad: 5
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito registrar la información de una persona que consta en la base de datos en la tarjeta RFID, para identificar el acceso.</p>	
<p>Criterios de Aceptación:</p> <p>Mediante una interfaz gráfica se debe realizar el registro de información de una tarjeta. El registro consta de dos partes: Un registro en la tarjeta RFID y un registro en la base de datos.</p> <ul style="list-style-type: none"> • La información en la Tarjeta RFID a registrar son: Código de la Persona, Código del Tipo de Persona, nombres, apellidos, cédula, número único, 	

una clave generada aleatoriamente, y el código de la tarjeta.

- La información en la base de datos son: Código de la Persona, Código del Tipo de Persona, Código de la tarjeta, fecha de emisión, fecha de vencimiento y clave de acceso.

Fuente: Elaborador por el Autor.

Tabla 2.13: Historia de Usuario: Asignación de permisos a Tarjeta RFID.

HISTORIA DE USUARIO	
Número: 13	Usuario: Administrador.
Nombre Historia: Asignación de permisos a Tarjeta RFID.	
Estimación: N/A	Prioridad: 5
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar o eliminar permisos a una tarjeta RFID, para controlar los niveles de acceso que tiene una tarjeta.	
Criterios de Aceptación: Mediante una interfaz gráfica, se registra en la base de datos los permisos asignados a la tarjeta RFID. Una tarjeta RFID puede tener varios permisos de acceso.	

Fuente: Elaborador por el Autor.

2.1.2.2. Módulo de reservación de laboratorios

La captura de requisitos, análisis y diseño de éste módulo ya se encuentran realizadas en la tesis desarrollada por Fabricio Arellano con el tema “*Desarrollo e implementación del Sistema de Reservación de Laboratorios para el LDICC*”¹, motivo por el cual se toma en cuenta parte de este trabajo como referencia.

¹ “[Desarrollo e implantación del sistema de reservación de laboratorios...](#)”

2.1.2.2.1. *Creación de Materias*

Tabla 2.14: Gestión de materias.

HISTORIA DE USUARIO	
Número: 14	Usuario: Administrador
Nombre Historia: Gestión de materias.	
Estimación: N/A	Prioridad: 1
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar, modificar o eliminar la información de una materia, para tener un listado de las materias que hacen uso del LDICC.	
Criterios de Aceptación: Mediante una interfaz gráfica, la Información a registrar de una materia es: <ul style="list-style-type: none"> • Código de la materia, el grupo asignado, el nombre de la materia, horario de la materia y profesor. La Información a modificar de una materia es: <ul style="list-style-type: none"> • Código de la materia, el grupo asignado, el nombre de la materia, horario de la materia y profesor. Una materia se puede eliminar, solamente si no tiene reservaciones realizadas.	

Fuente: Elaborador por el Autor.

2.1.2.2.2. *Reservación de un laboratorio.*

Tabla 2.15: Historia de Usuario: Registrar reservación de un laboratorio.

HISTORIA DE USUARIO	
Número: 15	Usuario: Administrador
Nombre Historia: Registrar reservación de un laboratorio.	
Estimación: N/A	Prioridad: 7

Programador Responsable: Patricio Chamorro
<p>Descripción:</p> <p>Yo como administrador, necesito registrar la información de la materia con el horario que se desea reservar, para informar a los usuarios el estado de las salas del LDICC.</p>
<p>Criterios de Aceptación:</p> <p>Mediante una interfaz gráfica, la información de una reserva a registrar es:</p> <ul style="list-style-type: none"> • La materia, periodo semestral, sistema operativo junto a los requerimientos técnicos y el horario a reservar. • El horario a reservar se restringe a los horarios de la materia.

Fuente: Elaborador por el Autor.

Tabla 2.16: Historia de Usuario: Modificar reservación de un laboratorio.

HISTORIA DE USUARIO	
Número: 16	Usuario: Administrador
Nombre Historia: Modificar reservación de un laboratorio.	
Estimación: N/A	Prioridad: 8
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito modificar la reservación de una sala, para actualizar la información del estado de uso de las salas del LDICC.</p>	
<p>Criterios de Aceptación:</p> <p>Mediante una interfaz gráfica, la Información de una reserva a modificar es:</p> <ul style="list-style-type: none"> • Cambiar de requerimientos técnicos y el horario a reservar. 	

Fuente: Elaborador por el Autor.

Tabla 2.17: Historia de Usuario: Eliminar reservación de un laboratorio.

HISTORIA DE USUARIO	
Número: 17	Usuario: Administrador
Nombre Historia: Eliminar reservación de un laboratorio.	
Estimación: N/A	Prioridad: 8
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito eliminar una reservación de una sala para actualizar la información del estado de uso de las salas del LDICC.	
Criterios de Aceptación: Solo las reservaciones registradas se pueden eliminar.	

Fuente: Elaborador por el Autor.

2.1.2.3. Módulo de gestión de entradas

El módulo de gestión de entradas permite Registrar o Modificar la información de los puntos de acceso en los que se debe realizar un control, tomando en cuenta sus permisos de acceso asignados.

Tabla 2.18: Historia de Usuario: Gestionar punto de acceso.

HISTORIA DE USUARIO	
Número: 18	Usuario: Administrador
Nombre Historia: Gestionar punto de acceso.	
Estimación: N/A	Prioridad: 9
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar o modificar la información de un punto	

de acceso, para administrar los puntos de acceso.

Criterios de Aceptación:

Mediante una interfaz gráfica, la información de un punto de acceso a registrar es:

- Nombre del punto de acceso, descripción, número de serie del dispositivo RFID y la lista de permisos asignados al punto de acceso.

La Información de un Punto de Acceso a modificar es:

- Nombre del punto de acceso, descripción, número de serie del dispositivo RFID y la lista de permisos asignados al punto de acceso.

Fuente: Elaborador por el Autor.

Tabla 2.19: Historia de Usuario: Asignar punto de acceso.

HISTORIA DE USUARIO	
Número: 19	Usuario: Administrador
Nombre Historia: Asignar punto de acceso.	
Estimación: N/A	Prioridad: 9
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como administrador, necesito especificar el punto de acceso para que el sistema cuando inicie el proceso de control, lo realice desde el punto especificado.</p>	
<p>Criterios de Aceptación:</p> <p>Mediante una interfaz gráfica, el punto de acceso asignado se guarda en un archivo de propiedades.</p>	

Fuente: Elaborador por el Autor.

2.1.2.4. Módulo de gestión de control de acceso

Este módulo se encarga de interactuar con el sistema hardware compuesto de un lector RFID que permite capturar los datos de una tarjeta, un micro controlador

que permite encender o apagar leds para informar el estado de las salas y un conmutador electrónico que permite la apertura de una chapa eléctrica

2.1.2.4.1. Permisos de Acceso

Tabla 2.20: Historia de Usuario: Gestionar permiso de acceso.

HISTORIA DE USUARIO	
Número: 20	Usuario: Administrador
Nombre Historia: Gestionar permiso de acceso.	
Estimación: N/A	Prioridad: 2
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito registrar, modificar o eliminar la información de un permiso de acceso, para tener varios perfiles de acceso.	
Criterios de Aceptación: Mediante una interfaz gráfica, la información de un permiso de acceso a registrar es: <ul style="list-style-type: none"> • Nombre del permiso y la lista de horarios permitidos para el acceso. La información de un permiso de acceso a modificar es: <ul style="list-style-type: none"> • Nombre del permiso y la lista de horarios permitidos para el acceso. Al eliminar un permiso de acceso se quita el permiso en las entradas y en las tarjetas que contengan dicho permiso.	

Fuente: Elaborador por el Autor.

2.1.2.4.2. Control de accesos al LDICC

Tabla 2.21: Historia de Usuario: Registrar acceso.

HISTORIA DE USUARIO	
Número: 21	Usuario: Administrador
Nombre Historia: Registrar acceso.	
Estimación: N/A	Prioridad: 9
Programador Responsable: Patricio Chamorro	

<p>Descripción:</p> <p>Yo como administrador, necesito que el sistema valide y registre el acceso de los usuarios del LDICC, para identificar las personas que ingresan.</p>
<p>Criterios de Aceptación:</p> <p>El sistema se encarga de ejecutar una tarea en un intervalo de tiempo de un segundo para detectar cuando una tarjeta RFID se presenta al lector. En el momento que se presenta la tarjeta, el sistema realiza la lectura de la tarjeta RFID a través del lector, verifica y valida los datos y según este resultado se permite o deniega el acceso.</p> <p>Cuando el sistema permite el acceso se registra los siguientes datos:</p> <ul style="list-style-type: none"> • A un estudiante o profesor se registra la sala asignada, la fecha y hora de ingreso. • Al personal administrativo, limpieza o para otro tipo de persona se registra la fecha y hora de ingreso. • Se debe mantener un historial de los accesos al LDICC.

Fuente: Elaborador por el Autor.

2.1.2.4.3. *Indicadores de estado de salas y acceso al LDICC*

Tabla 2.22: Historia de Usuario: Elaboración de circuito con leds.

HISTORIA DE USUARIO	
Número: 22	Usuario: Desarrollador
Nombre Historia: Elaboración de circuito con leds.	
Estimación: N/A	Prioridad: 6
Programador Responsable: Patricio Chamorro	
<p>Descripción:</p> <p>Yo como desarrollador, necesito un panel para que el sistema informe el estado de uso de las salas.</p>	
<p>Criterios de Aceptación:</p> <p>Para el caso de estudio, se debe fabricar un circuito electrónico que contenga un</p>	

micro controlador que permita encender o apagar un led determinado.

Los leds muestran los siguientes estados:

- Se necesita dos leds para informar el tipo de acceso al LDICC:
 - Led Rojo: La puerta no se abre, deniega el acceso y se prende el led durante dos segundos
 - Led Verde: La puerta se abre. Permite el acceso y se enciende el led durante dos segundos.
- Se necesita tres leds por cada sala para informar sus estados:
 - Led Rojo: Sala en clases, sala ocupada. Permanece prendido y se apaga solamente cuando cambia a un estado de sala libre.
 - Led Verde: Sala libre. Permanece prendido mientras la sala no tiene reservado hora de clases. Se apaga sólo cuando pasa a un estado de sala ocupada.
 - Led Azul: Sala asignada, se enciende durante dos segundos y se informa al usuario a que aula debe dirigirse.

Fuente: Elaborador por el Autor.

2.1.2.4.4. *Mostrar mensaje al usuario LDICC*

Tabla 2.23: Historia de Usuario: Imprimir mensaje en pantalla LCD.

HISTORIA DE USUARIO	
Número: 23	Usuario: Desarrollador
Nombre Historia: Imprimir mensaje en pantalla LCD.	
Estimación: N/A	Prioridad: 6
Programador Responsable: Patricio Chamorro	
Descripción: Yo como desarrollador, necesito una pantalla LCD para que el sistema muestre un mensaje al usuario del LDICC.	
Criterios de Aceptación: En el circuito electrónico, se debe colocar una pantalla LCD para que el micro controlador se encargue de mostrar en pantalla un mensaje recibido desde el computador. El mensaje a mostrar es un mensaje de bienvenida.	

- Los mensajes deben ser borrados del LCD después de dos segundos.

Fuente: Elaborador por el Autor.

2.1.2.5. Módulo de reportes

Este módulo permite al usuario acceder a reportes resumiendo los eventos de control de acceso en cada uno de los puntos de acceso. Para ello se accede a los registros almacenados en la base de datos y se filtra de acuerdo al intervalo de fecha definido por el usuario.

2.1.2.5.1. Generación de reportes

Tabla 2.24: Historia de Usuario: Generación de reportes Totales.

HISTORIA DE USUARIO	
Número: 24	Usuario: Administrador.
Nombre Historia: Generación de reportes Totales.	
Estimación: N/A	Prioridad: 10
Programador Responsable: Patricio Chamorro	
Descripción: Yo como administrador, necesito generar un reporte resumiendo el total de accesos concedidos/denegados en un punto de acceso especificado por el usuario para saber la estadística de uso de las instalaciones del LDICC.	
Criterios de Aceptación: Mediante una interfaz gráfica, los reportes totales se generan en un intervalo de fechas especificadas por el usuario: fecha inicio, fecha fin.	

Fuente: Elaborador por el Autor.

Tabla 2.25: Historia de Usuario: Generación de reportes Detallados.

HISTORIA DE USUARIO	
Número: 25	Usuario: Administrador
Nombre Historia: Generación de reportes Detallados.	
Estimación: N/A	Prioridad: 10
Programador Responsable: Patricio Chamorro	
Descripción:	

Yo como administrador, necesito generar un reporte de accesos detallando el nombre de la persona, la fecha y hora, y finalmente si fue concedido o no el acceso para identificar las personas que han accedido al LDICC.

Criterios de Aceptación:

Mediante una interfaz gráfica, el reporte se debe generar en un intervalo de fechas especificadas por el usuario: fecha inicio, fecha fin.

Fuente: Elaborador por el Autor.

A continuación se presenta un resumen de las Historias de Usuario con la prioridad, ver Tabla 2.26.

Tabla 2.26: Resumen de Historias de usuario en base a la prioridad.

Módulo	Nro.	Historia de Usuario	Prioridad
Gestión de Usuarios	07	Gestión de profesores	1
	08	Gestión de estudiantes	2
	09	Gestión de otros usuarios	2
	10	Gestión usuarios del sistema	8
	11	Configuración de Tarjeta RFID.	5
	12	Registro de Tarjetas.	5
	13	Asignación de permisos a Tarjeta RFID.	5
Módulo de reservación de laboratorios.	14	Gestión de materias	1
	15	Registrar reservación de un laboratorio.	7
	16	Modificar reservación de un laboratorio.	8
	17	Eliminar reservación de un laboratorio.	8
Módulo de	18	Gestión de puntos de	9

Módulo	Nro.	Historia de Usuario	Prioridad
gestión de entradas		acceso	
	19	Asignar punto de acceso	9
Módulo de control de acceso.	20	Gestión de permisos de acceso	2
	21	Registrar acceso.	9
	22	Elaboración de circuito con leds.	6
	23	Imprimir mensaje en pantalla LCD.	6
Módulo de reportes	24	Generación de reportes Totales.	10
	25	Generación de reportes Detallados.	10

Fuente: Elaborador por el Autor.

2.2. PLAN DE ENTREGAS

En esta sección se elabora el Plan de Entrega, en donde se especifica una estimación del tiempo por cada Historia de Usuario y además se especifica el número de iteración en la que se desarrolla.

2.2.1 ESTIMACIÓN DE HISTORIAS DE USUARIO

Por cada iteración se mantiene reuniones para realizar la estimación de las Historias de Usuario y para ello se emplea la técnica *Planning Poker*, en donde el cliente se encarga de leer una historia de usuario, luego se abre un debate para clarificar la historia de usuario si es necesario, cada programador escribe su propia estimación en una tarjeta sin dar a conocerla. Lugo cuando todos los programadores tengan las estimaciones, se ponen sobre la mesa y si todos están de acuerdo, se registra la estimación y se pasa a la siguiente historia de usuario, de lo contrario se puede dividir la historia y volver a estimarla [20].

El cliente lee la cada historia de usuario en este caso el Ing. Patricio Proaño, a los integrantes del equipo, Patricio Chamorro (A) e Ing. José Luis Carrera (B),

quienes realizan preguntas para entender el alcance y cada miembro efectúa la votación del esfuerzo de manera simultánea.

2.2.2 ELABORACIÓN DEL PLAN DE ENTREGA

Antes de proceder con la elaboración del Plan de Entregas se define el tiempo calendario para el desarrollo del proyecto. De ésta manera se establece el mes de 4 semanas, una semana de 5 días y un día de 8 horas.

Tiempo Calendario

Para la estimación del Tiempo Calendario, se presenta la siguiente plantilla como modelo a utilizar en horas, días y semanas, la misma que permitirá calcular el esfuerzo utilizado para el desarrollo del presente proyecto.

Tabla 2.27: Tiempo Calendario

Horas Calendario	Días Calendario	Semanas Calendario
8 horas (horas diarias que se va a dedicar al desarrollo del proyecto)	5 días (días laborables que se va a dedicar al desarrollo del proyecto)	4 semanas (semanas al mes que se va a dedicar al desarrollo del proyecto).

Fuente: Elaborador por el Autor.

Esfuerzo de Desarrollo

Se considera: Personas en el equipo: 1 persona; por lo tanto:

Esfuerzo por semana de desarrollo: $1 \text{ personas} = 1 \text{ semanas}$

Esfuerzo por días de desarrollo: $1 \text{ personas} = 5 \text{ días}$

Esfuerzo por horas calendario de desarrollo (día calendario): $1 \text{ personas} = 8 \text{ horas}$

En consecuencia a este análisis, se tiene:

Tabla 2.28: Esfuerzo de Desarrollo en base a una persona

Personas en el Equipo	Horas de Esfuerzo de Desarrollo	Días de Esfuerzo de Desarrollo	Semanas de Esfuerzo de Desarrollo

1 persona	8 horas	5 días	1 semana
-----------	---------	--------	----------

Fuente: Elaborador por el Autor.

Elaboración del plan de Entrega

Para elaborar el Plan de Entregas se realiza una clasificación de las historias de usuario asignando una iteración, tal y como se muestra en la Tabla 2.29. De esta manera resulta sencillo realizar el Plan de Entregas.

Tabla 2.29: Historias de Usuario con número de iteración

Módulo	Nro.	Historia de Usuario	Iteración Asignada					
			1	2	3	4	5	6
Gestión de Usuarios	07	Gestión de profesores	x					
	08	Gestión de estudiantes		x				
	09	Gestión de otros usuarios		x				
	10	Gestión usuarios del sistema				x		
	11	Configuración de Tarjeta RFID.			x			
	12	Registro de Tarjetas.			x			
	13	Asignación de permisos a Tarjeta RFID.				x		
Módulo de reservación de laboratorios.	14	Gestión de materias	x					
	15	Registrar reservación de un laboratorio.				x		
	16	Modificar reservación de un laboratorio.				x		
	17	Eliminar reservación de un laboratorio.				x		
Módulo de gestión de entradas	18	Gestión de puntos de acceso						x
	19	Asignar punto de acceso						x
Módulo de control de acceso.	20	Gestión de permisos de acceso		x				
	21	Registrar acceso.						x
	22	Elaboración de circuito con leds.					x	

Módulo	Nro.	Historia de Usuario	Iteración Asignada					
			1	2	3	4	5	6
	23	Imprimir mensaje en pantalla LCD.					x	
Módulo de reportes	24	Generación de reportes Totales.						x
	25	Generación de reportes Detallados.						x

Fuente: Elaborador por el Autor.

2.2.2.1. Iteraciones

Por cada iteración se elige las Historias de Usuario que se van a implementar, luego se descompone en tareas para poder estimar el tiempo de desarrollo y en base a esto se presenta el Plan de Entregas.

2.2.2.1.1. Primera Iteración

El objetivo en la primera iteración es diseñar el modelo de base de datos para el sistema en base a los requerimientos recolectados a través de las historias de usuario. Y finalmente se realiza la implementación de la primera iteración para el desarrollo del sistema.

Las Historias de Usuario para esta iteración son las siguientes:

- Gestión de profesores
- Gestión de materias

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.30: *Planning Poker* – Primera Iteración

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Gestión de Profesores.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			

Diseño de la Interfaz Gestión de Profesores	1	1	1
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1
Gestión de Materias			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Gestión de Profesores	1	1	1
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1

Fuente: Elaborador por el Autor.

Plan de Entregas

Con la estimación realizada para las Historias de Usuario asignadas a esta iteración se elabora el plan de entregas:

Tabla 2.31: Plan de Entregas – Primera Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Fecha Entrega
07	Gestión de profesores	8	03/07/2013
14	Gestión de materias	8	12/07/2013
Total		16	

Fuente: Elaborador por el Autor.

Criterios de aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.32: Criterios de aceptación – Primera Iteración.


Nro.	Historia de Usuario	Acción	Criterios de Aceptación
07	Gestión de profesores	Registrar	Ver Tabla 3.7
		Modificar	Ver Tabla 3.8
		Eliminar	Ver Tabla 3.9
14	Gestión de materias	Registrar	Ver Tabla 3.22
		Modificar	Ver Tabla 3.23
		Eliminar	Ver Tabla 3.24

Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.33: Resultado prueba de aceptación - Primera Iteración

Gestión de profesores - Registro	Resultado
Resultado esperado 1: El sistema presenta interfaz de gestión de profesores. El sistema registra los datos del profesor ingresados por el usuario	

en la base de datos.	
El sistema muestra el nuevo registro en la tabla de lista de profesores.	✓
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si ya existe el profesor.	✓
Gestión de profesores - Modificar	Resultado
Resultado esperado 1:	
El sistema muestra el listado de profesores registrados en la BDD.	
El sistema llena el formulario con los datos del registro seleccionado.	✓
El sistema modifica los datos del profesor de acuerdo con los datos ingresados por el administrador.	
El sistema actualiza los datos en la tabla de lista de profesores.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si no existe registro seleccionado.	
Gestión de profesores - Eliminar	
Resultado esperado 1:	
El sistema muestra el listado de profesores registrados en la BDD.	
El sistema llena el formulario con los datos del registro seleccionado.	✓
El sistema elimina los datos del profesor.	
El sistema elimina los datos en la tabla de lista de profesores.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no existe registro seleccionado.	✓
El sistema muestra mensaje de error si el profesor tiene registros de acceso.	Pendiente
Gestión de materias - Registro	Resultado

Resultado esperado 1:	
El sistema muestra interfaz de gestión de materias.	✓
El sistema registra los datos de la materia ingresados por el usuario.	
El sistema muestra el nuevo registro en la tabla de lista de materias.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si ya existe materia.	
Gestión de materias – Modificar.	Resultado
Resultado esperado 1:	
El sistema muestra el listado de materias registradas en la BDD.	
El sistema llena el formulario con los datos del registro seleccionado.	✓
El sistema modifica los datos de la materia.	
El sistema actualiza los datos en la tabla de lista de materias.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si se duplica la materia (código materia y grupo).	
Gestión de materias - Eliminar	Resultado
Resultado esperado 1:	
El sistema muestra el listado de materias registradas en la BDD.	
El sistema llena el formulario con los datos del registro seleccionado.	✓
El sistema elimina los datos de la materia.	
El sistema elimina los datos en la tabla de lista de materias.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no existe registro seleccionado.	✓
El sistema muestra mensaje de error si la materia tiene reservado un laboratorio.	Pendiente

Fuente: Elaborador por el Autor.

En la Tabla 2.31 se determina que dos casos de prueba quedan pendientes

debido a que en ésta iteración no es posible realizar la prueba ya que depende de funcionalidades que aún no han sido implementadas.

2.2.2.1.2. Segunda Iteración

El objetivo en la segunda iteración es implementar el registro, actualización y eliminación de estudiantes, personal administrativo y de otro tipo de personas. También se realiza la implementación para registrar, actualizar y eliminar permisos de acceso.

Las Historias de Usuario para esta iteración son las siguientes:

- Gestión de estudiantes
- Gestión de otros usuarios
- Gestión de permisos de acceso

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.34: *Planning Poker* – Segunda Iteración

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Gestión de Estudiante.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Gestión de Profesores	2	1	2
Validación de datos de la Interfaz	2	1	2
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1
Gestión de usuarios			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Gestión de Profesores	2	1	2
Validación de datos de la Interfaz	2	1	2
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1

Pruebas de aceptación	1	1	1
Gestión de permisos de acceso.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Gestión de Profesores	2	1	1
Validación de datos de la Interfaz	2	2	2
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1

Fuente: Elaborador por el Autor.

Plan de Entregas

Con la estimación realizada para las Historias de Usuario asignadas a esta iteración se elabora el plan de entregas:

Tabla 2.35: Plan de Entregas – Segunda Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Fecha Entrega
08	Gestión de estudiantes	10	25/07/2013
09	Gestión de otros usuarios	10	08/08/2013
20	Gestión de permisos de acceso	10	18/082013
Total		30 días	

Fuente: Elaborador por el Autor.

Criterios de Aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.36: Criterios de aceptación – Segunda Iteración.

Nro.	Historia de Usuario	Acción	Criterios de Aceptación
------	---------------------	--------	-------------------------

Nro.	Historia de Usuario	Acción	Criterios de Aceptación
08	Gestión de estudiantes	Registrar	Ver Tabla 3.10
		Modificar	Ver Tabla 3.11
		Eliminar	Ver Tabla 3.12
09	Gestión de otros usuarios	Registrar	Ver Tabla 3.13
		Modificar	Ver Tabla 3.14
		Eliminar	Ver Tabla 3.15
20	Gestión de permisos de acceso	Registrar	Ver Tabla 3.32
		Modificar	Ver Tabla 3.33
		Eliminar	Ver Tabla 3.34

Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.37: Resultado prueba de aceptación - Segunda Iteración

Gestión de estudiantes - Registro	Resultado
<p>Resultado esperado 1:</p> <p>El sistema muestra interfaz de gestión de estudiantes.</p> <p>El sistema registra los datos del estudiante, ingresados por el usuario.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de estudiantes.</p>	✓
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p>	✓

El sistema muestra mensaje de error si ya existe el estudiante (número único).	
Gestión de estudiantes - Modificar	Resultado
Resultado esperado 1:	
El sistema muestra el listado de estudiantes registrados en la BDD.	
El sistema llena el formulario con los datos del registro seleccionado.	✓
El sistema modifica los datos del estudiante.	
El sistema actualiza los datos en la tabla de lista de estudiantes.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si no existe registro seleccionado.	
Gestión de estudiantes - Eliminar	
Resultado esperado 1:	
El sistema muestra el listado de estudiantes registrados en la BDD.	
El sistema llena el formulario con los datos del registro seleccionado.	✓
El sistema elimina los datos del estudiante.	
El sistema elimina los datos en la tabla de lista de estudiantes.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no existe registro seleccionado.	✓
El sistema muestra mensaje de error si el estudiante tiene registros de acceso.	Pendiente
Gestión de otros usuarios - Registro	
Resultado esperado 1:	
El sistema muestra interfaz de gestión de otros usuarios.	
El sistema actualiza la tabla de lista de personas de acuerdo al tipo seleccionado.	✓
El sistema registra los datos del tipo de persona seleccionado.	
El sistema muestra el nuevo registro en la tabla de lista de personas	

de acuerdo al tipo de persona seleccionado.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si ya existe la persona (cédula).	
Gestión de otros usuarios - Modificar	Resultado
Resultado esperado 1:	
El sistema actualiza la tabla de lista de personas de acuerdo al tipo seleccionado.	✓
El sistema llena el formulario con los datos del registro seleccionado.	
El sistema modifica los datos de la persona.	
El sistema actualiza los datos en la tabla de lista de personal.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si no existe registro seleccionado.	
Gestión de otros usuarios - Eliminar	Resultado
Resultado esperado 1:	
El sistema actualiza la tabla de lista de personas de acuerdo al tipo seleccionado.	✓
El sistema llena el formulario con los datos del registro seleccionado.	
El sistema elimina los datos de la persona.	
El sistema elimina los datos en la tabla de lista de personal.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no existe registro seleccionado.	✓
El sistema muestra mensaje de error si el usuario tiene registros de acceso.	Pendiente
Gestión de permisos de acceso - Registro	Resultado
Resultado esperado 1:	
El sistema muestra interfaz de gestión de permisos de acceso.	✓

El sistema registra los datos del Permiso. El sistema muestra el nuevo registro en la tabla de lista de permisos.	
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si ya existe el permiso de acceso.	✓
Gestión de permisos de acceso - Modificar	Resultado
Resultado esperado 1: El sistema muestra el listado de permisos de acceso registrados en la BDD. El sistema llena el formulario con los datos del registro seleccionado. El sistema modifica los datos del permiso. El sistema actualiza los datos en la tabla de lista de permisos.	✓
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si no existe registro seleccionado.	✓
Gestión de permisos de acceso - Eliminar	Resultado
Resultado esperado 1: El sistema muestra listado de permisos de acceso registrados en la BDD El sistema llena el formulario con los datos del registro seleccionado. El sistema elimina los datos del permiso de acceso. El sistema elimina los datos en la tabla de lista de permisos de acceso.	✓
Resultado esperado 2: El sistema muestra mensaje de error si no existe registro seleccionado.	✓

Fuente: Elaborador por el Autor.

En la Tabla 2.37 se determina que dos casos de prueba quedan pendientes debido a que en ésta iteración no es posible realizar la prueba ya que depende de funcionalidades que aún no han sido implementadas.

2.2.2.1.3. Tercera Iteración

El objetivo en la tercera iteración es configurar la tarjeta RFID a utilizar, cambiar la contraseña por defecto en los sectores a utilizar, hacer pruebas de escritura y lectura en los bloques de datos y finalmente hacer el registro de una tarjeta.

Las Historias de Usuario para esta iteración son las siguientes:

- Configuración de Tarjeta RFID.
- Registro de Tarjetas.

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.38: *Planning Poker* – Tercera Iteración.

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Configuración de Tarjeta RFID.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Configuración de Tarjeta RFID	3	2	3
Diseño de prototipo para lectura de tarjetas RFID	3	2	3
Integración del Prototipo al Sistema	1	1	1
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1
Registro de Tarjetas.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Registro de Tarjetas	1	1	1
Validación de datos de la Interfaz	2	1	2
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1

Pruebas de aceptación	1	1	1
-----------------------	---	---	---

Fuente: Elaborador por el Autor.

Plan de Entregas

Con la estimación realizada para las Historias de Usuario asignadas a esta iteración se elabora el plan de entregas:

Tabla 2.39: Plan de Entregas – Tercera Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Fecha Entrega
11	Configuración de Tarjeta RFID.	14	09/09/2013
12	Registro de Tarjetas.	9	20/09/2013
Total		24	

Fuente: Elaborador por el Autor.

Criterios de Aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.40: Criterios de aceptación – Tercera Iteración.

Nro.	Historia de Usuario	Acción	Criterios de Aceptación
11	Configuración de Tarjeta RFID.	Registrar	Ver Tabla 3.19
12	Registro de Tarjetas.	Registrar	Ver Tabla 3.20

Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.41: Resultado prueba de aceptación - Tercera Iteración

Configuración de Tarjeta RFID	Resultado
<p>Resultado esperado 1:</p> <p>El sistema se conecta con el lector RFID.</p> <p>El usuario presenta la tarjeta RFID al lector.</p> <p>El sistema configura las zonas de la memoria a utilizar de la criptomemoria de la tarjeta RFID.</p>	✓
<p>Resultado esperado 2:</p> <p>El sistema se conecta con el lector RFID.</p> <p>El usuario presenta la tarjeta RFID al lector.</p> <p>El sistema no puede configurar la tarjeta RFID.</p> <p>El sistema muestra mensaje de error.</p>	✓
<p>Resultado esperado 3:</p> <p>El sistema no se conecta con el lector RFID.</p> <p>El sistema muestra mensaje de error.</p>	✓
Registro de Tarjetas	Resultado
<p>Resultado esperado 1:</p> <p>El sistema se conecta con el lector RFID.</p> <p>El usuario presenta la tarjeta RFID al lector.</p> <p>El sistema detecta tarjeta RFID configurada.</p> <p>El sistema lista las personas registradas en la BDD de acuerdo al tipo de persona.</p> <p>El usuario selecciona una persona de la lista.</p> <p>El sistema registra los datos de la tarjeta en la BDD y en la criptomemoria de la tarjeta RFID.</p>	✓
<p>Resultado esperado 2:</p> <p>El sistema se conecta con el lector RFID.</p> <p>El usuario presenta la tarjeta RFID al lector.</p> <p>El sistema detecta tarjeta RFID sin configurar.</p>	✓

El sistema muestra mensaje de error.

Resultado esperado 3:

El sistema no se conecta con el lector RFID.



El sistema muestra mensaje de error.

Fuente: Elaborador por el Autor.

2.2.2.1.4. *Cuarta Iteración*

El objetivo en la cuarta iteración es relacionar los permisos de acceso con los registros de tarjetas emitidas, se implementa la reservación de laboratorios y el registro, actualización y eliminación de registros de usuarios del sistema.

Las Historias de Usuario para esta iteración son las siguientes:

- Asignación de permisos a Tarjeta RFID.
- Registrar reservación de un laboratorio.
- Modificar reservación de un laboratorio.
- Eliminar reservación de un laboratorio.
- Gestión usuarios del sistema.

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.42: *Planning Poker* – Cuarta Iteración.

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Asignación de permisos a Tarjeta RFID.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Asignación de permiso de Accesos	2	1	2
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1
Registrar reservación de un laboratorio.			
Diseño CRC	1	1	1

Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Reservación de Laboratorio	2	1	2
Validación de datos de la Interfaz	2	1	2
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Modificar reservación de un laboratorio.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Validación de datos de la Interfaz	2	1	2
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación			
Eliminar reservación de un laboratorio.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD			
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Gestión de usuarios del Sistema.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Gestión de usuarios del sistema	2	1	2
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación			

Fuente: Elaborador por el Autor.

Plan de Entregas

Con la estimación realizada para las Historias de Usuario asignadas a esta iteración se elabora el plan de entregas:

Tabla 2.43: Plan de Entregas – Cuarta Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Fecha Entrega
13	Asignación de permisos a Tarjeta RFID.	8	30/09/2013
15	Registrar reservación de un laboratorio.	7	08/10/2013
16	Modificar reservación de un laboratorio.	6	16/10/2013
17	Eliminar reservación de un laboratorio.	3	24/10/2013
10	Gestión usuarios del sistema	7	04/11/2013
Total		31 días	

Fuente: Elaborador por el Autor.

Criterios de Aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.44: Criterios de aceptación – Tercera Iteración.

Nro.	Historia de Usuario	Acción	Criterios de Aceptación
10	Gestión usuarios del sistema	Registrar	Ver Tabla 3.16
		Modificar	Ver Tabla 3.17
		Eliminar	Ver Tabla 3.18
13	Asignación de permisos a Tarjeta RFID.	--	Ver Tabla 3.21
15	Registrar reservación de un laboratorio.	--	Ver

			Tabla 3.25
16	Modificar reservación de un laboratorio.	--	Ver Tabla 3.26
17	Eliminar reservación de un laboratorio.	--	Ver Tabla 3.27

Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.45: Resultado prueba de aceptación - Cuarta Iteración

Gestión usuarios del sistema - Registro	Resultado
<p>Resultado esperado 1:</p> <p>El sistema muestra interfaz de gestión de usuarios.</p> <p>El sistema registra los datos del usuario.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de usuarios del sistema.</p>	✓
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe usuario.</p>	✓
Gestión usuarios del sistema - Modificar	Resultado
<p>Resultado esperado 1:</p> <p>El sistema muestra el listado de usuarios del sistema registrados en la BDD.</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos del usuario.</p> <p>El sistema actualiza los datos en la tabla de lista de usuarios del sistema.</p>	✓
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos</p>	✓

requeridos. El sistema muestra mensaje de error si no existe registro seleccionado.	
Gestión usuarios del sistema - Eliminar	Resultado
Resultado esperado 1: El sistema muestra listado de usuarios del sistema registrados en la BDD. El sistema llena el formulario con los datos del registro seleccionado. El sistema elimina los datos del usuario del sistema. El sistema elimina los datos en la tabla de lista de usuarios del sistema.	✓
Resultado esperado 2: El sistema muestra mensaje de error si no existe registro seleccionado. El sistema muestra mensaje de error si el registro a eliminar es el usuario administrador.	✓
Asignación de permisos a Tarjeta RFID.	Resultado
Resultado esperado 1: El sistema muestra interfaz de configuración de la tarjeta RFID. El sistema muestra la lista de permisos de acceso registrados en la BDD. El usuario selecciona los permisos de acceso. El sistema registra los datos de los permisos para la tarjeta.	✓
Resultado esperado 2: El sistema despliega mensaje de error si los datos requeridos no se han ingresado o son inválidos.	✓
Registrar reservación de un laboratorio.	Resultado
Resultado esperado 1: El sistema despliega lista de periodos, laboratorios y materias. El usuario selecciona un periodo, laboratorio y materia. El sistema muestra interfaz de reservación de laboratorios	✓

El sistema registra los datos de la reserva. El sistema muestra el nuevo registro en la tabla de horarios de reservas.	
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si ya existe cruce de horarios en la reserva.	✓
Modificar reservación de un laboratorio.	Resultado
Resultado esperado 1: El sistema despliega lista de periodos y laboratorios. El usuario selecciona un periodo, un laboratorio y materia. El sistema muestra interfaz de gestión de reservas. El sistema despliega formulario con los datos del registro seleccionado. El sistema modifica los datos de la reserva. El sistema actualiza los datos en la tabla de horarios.	✓
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si hay cruce de horarios.	✓
Eliminar reservación de un laboratorio.	Resultado
Resultado esperado 1: El sistema muestra el listado de materias reservadas en un periodo por laboratorio. El sistema llena el formulario con los datos del registro seleccionado. El sistema elimina los datos de la reserva. El sistema elimina los datos en la tabla de horarios.	✓
Resultado esperado 2: El sistema muestra mensaje de error si no existe registro seleccionado.	✓

Fuente: Elaborador por el Autor.

2.2.2.1.5. Quinta Iteración

El objetivo en la quinta iteración es realizar el diseño e implementación del circuito con leds que indican los estados de las salas (libre, ocupada y sala asignada), el permiso de acceso (permitido o denegado) y el mensaje de bienvenida a mostrar al usuario por medio de una pantalla LCD.

Las Historias de Usuario para esta iteración son las siguientes:

- Elaboración de circuito con leds
- Imprimir mensaje en pantalla LCD.

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.46: *Planning Poker* – Quinta Iteración.

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Elaboración de circuito con leds.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Diseño del circuito	5	3	5
Prototipo para pruebas de funcionalidad	3	1	3
Prototipo final	3	3	3
Integración	1	1	1
Pruebas de aceptación	1	1	1
Imprimir mensaje en pantalla LCD.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño del circuito	3	2	3
Prototipo final	2	2	2
Integración	1	1	1
Pruebas de aceptación			

Plan de Entregas

Con la estimación realizada para las Historias de Usuario asignadas a esta iteración se elabora el plan de entregas:

Tabla 2.47: Plan de Entregas – Quinta Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Entrega
22	Elaboración de circuito con leds	14	25/11/2013
23	Imprimir mensaje en pantalla LCD.	8	08/12/2013
Total		22 días	

Fuente: Elaborador por el Autor.

Criterios de Aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.48: Criterios de aceptación – Quinta Iteración.


Nro.	Historia de Usuario	Acción	Criterios de Aceptación
11	Elaboración de circuito con leds	Registrar	Ver Tabla 3.36
12	Imprimir mensaje en pantalla LCD.	Registrar	Ver Tabla 3.37



Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.49: Resultado prueba de aceptación - Quinta Iteración

Elaboración de circuito con leds	Resultado
<p>Resultado esperado 1:</p> <p>El circuito recibe comando de permiso de acceso concedido, abre la cerradura eléctrica y enciende led verde.</p> <p>El circuito recibe comando de permiso de acceso denegado,</p>	

enciende led rojo. El circuito recibe comando para actualizar leds de estado de uso de laboratorios, cambia estado de los leds.	
Resultado esperado 1: El circuito recibe comando de permiso de acceso concedido, abre chapa eléctrica y enciende led verde. El circuito recibe comando de permiso de acceso denegado, enciende led rojo.  El circuito recibe comando para actualizar leds de estado de uso de laboratorios, cambia estado de los leds.	
Imprimir mensaje en pantalla LCD.	Resultado
Resultado esperado 1: El circuito imprime mensaje en pantalla LCD. 	

Fuente: Elaborador por el Autor.

2.2.2.1.6. *Sexta Iteración*

El objetivo en la sexta iteración es implementar el módulo de gestión de entradas e integrar todo el sistema para realizar el control de accesos utilizando el circuito construido. Y finalmente el sistema se finaliza con la implementación del módulo de reportes.

Las Historias de Usuario para esta iteración son las siguientes:

- Gestión de puntos de acceso.
- Asignar punto de acceso.
- Registrar acceso.
- Generación de reportes Totales.
- Generación de reportes Detallados.

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.50: *Planning Poker* – Sexta Iteración.

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Gestión de puntos de acceso.			
Diseño CRC	1	1	1
Pruebas Unitarias	1	1	1
Programación por pares			
Diseño de la Interfaz Gestión de Puntos de Acceso	2	1	2
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación			
Asignar punto de acceso			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Asignar Punto de Acceso	1	1	1
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD			
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación			
Registrar acceso.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Registro de acceso	1	1	1
Validación de datos de la Interfaz	1	1	1
Almacenar datos en la BDD	1	1	1
Ejecución de pruebas Unitarias	1	1	1
Integración	1	1	1
Pruebas de aceptación	1	1	1
Generación de reportes Totales.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Reportes Totales	1	1	1
Validación de datos de la Interfaz			
Almacenar datos en la BDD			
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Generación de reportes Detallados.			
Diseño CRC	1	1	1

Pruebas Unitarias			
Programación por pares			
Diseño de la Interfaz Gestión de reportes Detallados	1	1	1
Validación de datos de la Interfaz			
Almacenar datos en la BDD			
Ejecución de pruebas Unitarias	1	1	1
Integración			
Pruebas de aceptación			

Fuente: Elaborador por el Autor.

Plan de Entregas

Con la estimación realizada para las Historias de Usuario asignadas a esta iteración se elabora el plan de entregas:

Tabla 2.51: Plan de Entregas – Sexta Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Fecha Entrega
18	Gestión de puntos de acceso	8	12/12/2013
19	Asignar punto de acceso	5	16/12/2013
21	Registrar acceso.	7	25/12/2013
24	Generación de reportes Totales.	3	27/12/2013
25	Generación de reportes Detallados.	3	31/12/2013
Total		26 días	

Fuente: Elaborador por el Autor.

Criterios de Aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.52: Criterios de aceptación – Quinta Iteración.

Nro.	Historia de Usuario	Acción	Criterios de Aceptación
18	Gestión de puntos de acceso	Registrar	Ver Tabla 3.28
		Modificar	Ver Tabla 3.29
		Eliminar	Ver Tabla 3.30
19	Asignar punto de acceso	--	Ver Tabla 3.31
21	Registrar acceso.	--	Ver Tabla 3.35
24	Generación de reportes Totales.	--	Ver Tabla 3.38
25	Generación de reportes Detallados.	--	Ver Tabla 3.39






Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.53: Resultado prueba de aceptación - Sexta Iteración

Gestión de puntos de acceso - Registrar	Resultado
<p>Resultado esperado 1:</p> <p>El sistema muestra interfaz de gestión de puntos de acceso.</p> <p>El sistema registra los datos del punto de acceso.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de usuarios del sistema.</p>	✓
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe el punto de acceso.</p>	✓

Gestión de puntos de acceso – Modificar	Resultado
<p>Resultado esperado 1:</p> <p>El sistema muestra la lista de puntos de accesos registrados en la BDD.</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos del punto de acceso.</p> <p>El sistema actualiza los datos en la tabla de lista de entradas.</p>	
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>	
Gestión de puntos de acceso – Eliminar	
<p>Resultado esperado 1:</p> <p>El sistema muestra listado de puntos de acceso registrados en la BDD.</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos del punto de acceso.</p> <p>El sistema elimina los datos en la tabla de lista de entradas.</p>	
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p> <p>El sistema muestra mensaje de error si el punto de acceso tiene registros de accesos.</p>	
Asignar punto de acceso	
<p>Resultado esperado 1:</p> <p>El sistema muestra interfaz de gestión de puntos de acceso.</p> <p>El sistema lista los puntos de acceso disponibles en la BDD.</p> <p>El sistema guarda en archivo de propiedades la información del punto de acceso seleccionado.</p> <p>El sistema inicia el control en el punto asignado.</p>	

Resultado esperado 2:	
El sistema muestra mensaje de error si el punto de acceso ya existe.	✓
Registrar acceso.	
Resultado esperado 1:	
El usuario del LDICC presenta tarjeta RFID al lector.	
El sistema lee la tarjeta RFID.	
El sistema valida los datos de la tarjeta RFID con los datos registrados en la BDD.	No pasa
El sistema valida el permiso de acceso asignado a la tarjeta RFID.	
El sistema registra acceso.	
Resultado esperado 2:	
El usuario del LDICC presenta tarjeta RFID al lector.	
El sistema lee tarjeta RFID.	No pasa
El sistema valida tarjeta RFID.	
El sistema deniega acceso a tarjeta RFID.	
El sistema registra acceso.	
Generación de reportes Totales.	
Resultado esperado 1:	
El sistema presenta interfaz de gestión de reportes.	
El usuario ingresa intervalos de fechas.	✓
El sistema genera reporte.	
El sistema muestra el reporte.	
Resultado esperado 2:	
El sistema muestra mensaje de error si no se ha ingresado campos requeridos.	✓
El sistema muestra mensaje de error si no existen datos para generar el reporte.	
Generación de reportes Detallados.	
Resultado esperado 1:	
El sistema presenta interfaz de gestión de reportes.	✓
El usuario ingresa intervalos de fechas.	
El sistema genera reporte.	

El sistema muestra el reporte.

Resultado esperado 2:

El sistema muestra mensaje de error si no se ha ingresado campos requeridos.



El sistema muestra mensaje de error si no existen datos para generar el reporte.

Fuente: Elaborador por el Autor.

En la Tabla 2.53 se determina que no pasa la prueba de aceptación la historia de usuario “Registrar Acceso” debido a que el sistema presenta tiempos de respuesta altos (5 segundos) y se tiene que hacer un reajuste en el circuito para tener tiempos de respuesta menores (2 segundos).

2.2.2.1.7. *Séptima Iteración*

El objetivo en esta iteración es corregir los errores encontrados en las pruebas de aceptación fallidas de las anteriores iteraciones.

Las Historias de Usuario para esta iteración son las siguientes:

- Gestión de profesores.
- Gestión de materias.
- Gestión de estudiantes.
- Gestión de otros usuarios.
- Registrar acceso.

A continuación se descompone cada Historia de Usuario en tareas.

Tabla 2.54: *Planning Poker* – Séptima Iteración.

Nombre de tarea	Estimación A	Estimación B	Acuerdo
Gestión de Profesores.			
Pruebas Unitarias	1	1	1
Programación por pares			

Re-Validación de datos de la Interfaz	1	1	1
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Gestión de Materias			
Pruebas Unitarias	1	1	1
Programación por pares			
Re-Validación de datos de la Interfaz	1	1	1
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Gestión de Estudiante.			
Pruebas Unitarias	1	1	1
Programación por pares			
Re-Validación de datos de la Interfaz	1	1	1
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Gestión de usuarios			
Pruebas Unitarias	1	1	1
Programación por pares			
Validación de datos de la Interfaz	1	1	1
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			
Registrar acceso.			
Diseño CRC	1	1	1
Pruebas Unitarias			
Programación por pares			
Re-Validación de datos de la Interfaz	3	2	3
Ejecución de pruebas Unitarias			
Integración	1	1	1
Pruebas de aceptación			

Fuente: Elaborador por el Autor.

Plan de Entregas

Las pruebas de aceptación fallidas que se tuvieron en las iteraciones anteriores debido a que dependían de la funcionalidad que fue implementada en la sexta iteración, se toman en cuenta para esta iteración y el Plan de Entregas es el siguiente:

Tabla 2.55: Plan de Entregas – Séptima Iteración

Nro.	Historia de Usuario	Tiempo Estimado	Fecha Entrega
07	Gestión de profesores	3	06/01/2014
14	Gestión de materias	3	09/01/2014
08	Gestión de estudiantes.	3	14/01/2014
09	Gestión de otros usuarios.	3	17/01/2014
21	Registrar acceso.	5	24/01/2014
Total		17 días	

Fuente: Elaborador por el Autor.

Criterios de Aceptación

Los criterios de aceptación para cada historia de usuario del Plan de Entregas se muestran en la siguiente tabla:

Tabla 2.56: Criterios de aceptación – Séptima Iteración.

Nro.	Historia de Usuario	Acción	Criterios de Aceptación
07	Gestión de profesores	Eliminar	Ver Tabla 3.30
14	Gestión de materias	Eliminar	Ver Tabla 3.24
08	Gestión de estudiantes.	Eliminar	Ver Tabla 3.12
09	Gestión de otros usuarios.	Eliminar	Ver Tabla 3.15
21	Registrar acceso.	---	Ver Tabla 3.35

Fuente: Elaborador por el Autor.

Pruebas de aceptación

Para finalizar las historias de usuario que constan en el Plan de Entregas se

ejecuta las Pruebas de Aceptación. Obteniendo como resultado lo siguiente:

Tabla 2.57: Resultado prueba de aceptación - Séptima Iteración

Gestión de profesores - Eliminar	Resultado
El sistema muestra interfaz de gestión de profesores. El usuario selecciona un registro de un profesor. El sistema muestra mensaje de error si el profesor tiene registros de acceso.	✓
Gestión de materias - Eliminar	
El sistema muestra interfaz de gestión de materias. El usuario selecciona un registro de una materia. El sistema muestra mensaje de error si la materia tiene reservado un laboratorio.	✓
Gestión de estudiantes - Eliminar	
El sistema muestra interfaz de gestión de estudiantes. El usuario selecciona un registro de un estudiante. El sistema muestra mensaje de error si el estudiante tiene registros de acceso.	✓
Gestión de otros usuarios - Eliminar	Resultado
El sistema muestra interfaz de gestión de usuarios. El usuario selecciona un registro de un usuario. El sistema muestra mensaje de error si el usuario tiene registros de acceso.	✓
Registrar acceso.	
Resultado esperado 1: El usuario del LDICC presenta tarjeta RFID al lector. El sistema lee la tarjeta RFID. El sistema identifica tipo de usuario. El sistema valida los datos de la tarjeta RFID con los datos registrados en la BDD. El sistema valida permiso de acceso a la tarjeta RFID.	✓

El sistema registra acceso correcto.

Resultado esperado 2:



El usuario del LDICC presenta tarjeta RFID al lector.

El sistema lee la tarjeta RFID.

El sistema identifica tipo de usuario.

El sistema valida los datos de la tarjeta RFID con los datos registrados en la BDD.

El sistema deniega acceso a tarjeta RFID.

El sistema registra acceso denegado.

Fuente: Elaborador por el Autor.

En el Anexo B se encuentra el cronograma de implementación de cada Historia de Usuario.

2.2.2.2. Plan de Entregas

La metodología XP establece que por cada iteración debe existir un entregable, por esta razón al tener siete iteraciones en el proyecto, se obtiene siete entregables. A continuación, en Tabla 2.58 se resume las fechas de los entregables de cada iteración.

Tabla 2.58: Plan de Entregas

Iteración	Entrega Asignada							Fecha de entrega
	1	2	3	4	5	6	7	
1	X							12/07/2013
2		X						19/08/2013
3			X					18/09/2013
4				X				31/10/2013
5					x			02/12/2013
6						x		31/12/2013
7							X	24/01/2014

Fuente: Elaborador por el Autor.

2.3. METÁFORA DEL SISTEMA

Las metáforas permiten entender de mejor manera lo que se va a implementar, esto es describir la funcionalidad de cada historia de usuario empleando un lenguaje técnico y entendible para el cliente.

2.3.1 MÓDULO DE GESTIÓN DE USUARIOS

Este módulo permite registrar, modificar y eliminar usuarios como profesores, estudiantes, personal administrativo, limpieza, usuarios del sistema y otros. Adicionalmente se gestiona las tarjetas RFID emitidas a las personas, para que puedan acceder en los puntos de acceso de acuerdo a la configuración de los permisos de acceso asignados.

Registro de Profesor

Permite registrar un nuevo profesor con la siguiente información:
Número de cédula, nombres, apellidos y materias que dicta.

Registro de Estudiante

Permite registrar un nuevo estudiante con la siguiente información:
Número de cédula, número único, nombres, apellidos y lista de materias que recibe en el periodo actual.

Registro de Personal Administrativo, Limpieza y Otros

Permite registrar una nueva persona ya sea tipo administrativa, limpieza u otro tipo de persona con la siguiente información:

Número de cédula, nombres y apellidos.

Registro de Usuarios del Sistema

Permite registrar un nuevo usuario del sistema con la siguiente información: Nombre de Usuario y contraseña.

Modificar Usuarios

Permite modificar la información de un usuario tipo estudiante, profesor, personal administrativo, limpieza, usuarios del sistema y otro tipo de personas, seleccionado de una lista.

Eliminar Usuarios

Permite eliminar la información de un usuario tipo estudiante, profesor, personal administrativo, limpieza, usuarios del sistema y otro tipo de personas, seleccionado de una lista.

Gestión de Tarjetas

Permite registrar y actualizar la información de una tarjeta en el sistema y guardar la información del usuario de la tarjeta, en la tarjeta RFID. Además permite asignar o remover permisos de acceso asignados a cada tarjeta.

2.3.2 MÓDULO DE RESERVACIÓN DE LABORATORIOS

Este módulo permite realizar el registro, actualización y consulta de la información de materias y las reservaciones realizadas para cada materia en las distintas salas de computación del LDICC.

Gestión de Materias

Permite registrar, modificar o eliminar una materia. La información de una materia es: Código de la materia, el grupo asignado, el nombre de la materia, horario de la materia y profesor.

Registrar Reserva

Permite registrar la información de una reserva de un laboratorio: La materia, periodo, la sala de computación, el horario, y los requerimientos técnicos en los diferentes sistemas operativos que tenga la sala.

Modificar Reserva

Permite modificar los requerimientos y el horario de la reserva en una sala de computación.

Eliminar Reserva

Permite eliminar una reserva de una sala de computación que pertenece a una materia.

2.3.3 MÓDULO DE GESTIÓN DE ENTRADAS

Este módulo permite registrar o modificar la información de los puntos de acceso en los que se debe realizar un control, tomando en cuenta sus permisos de acceso asignados.

Gestionar Entrada

Permite registrar, modificar una entrada o punto de acceso. La información de un punto de acceso es: La información de un punto de acceso a registrar es: Nombre del punto de acceso, descripción, número de serie del dispositivo RFID y la lista de permisos asignados al punto de acceso

Asignar punto de acceso

Permite especificar en un archivo de propiedades el nombre del punto de acceso en el que se debe realizar el control.

2.3.4 MÓDULO DE CONTROL DE ACCESO

Este módulo permite registrar, modificar y eliminar permisos de acceso, además su principal función es interactuar con el sistema hardware compuesto de un lector RFID que permite capturar los datos de una tarjeta, un micro controlador que permite encender o apagar leds para informar el estado de las salas, un conmutador electrónico que permite la apertura de una chapa eléctrica y finalmente realizar el registro de los accesos en cada puerta.

Gestión de Permisos

Permite registrar, modificar y eliminar un permiso de acceso. Estos permisos de acceso se asignan manualmente a cada tarjeta RFID y a cada punto de acceso creado. La información de un permiso de acceso es: Nombre del permiso y la lista de horarios permitidos para el acceso.

Registro de accesos

Permite registrar un acceso por una puerta determinada realizado por un usuario que posee una tarjeta RFID. La información a registrar es: el código de la persona, el código de la tarjeta, el estado del acceso (concedido o denegado), la fecha y hora de acceso.

Circuito Integrado

El circuito integrado o sistema hardware permite mostrar al usuario un mensaje por medio de un LCD, los estados de las salas y el tipo de acceso por medio de leds. Además permite cerrar un conmutador electrónico mediante señales enviadas por el sistema software.

2.3.5 MÓDULO DE REPORTES

Este módulo permite acceder a reportes resumiendo los eventos de control de acceso en los diferentes puntos de control configurados.

Generar reporte total

Permite generar un reporte resumiendo el total de accesos concedidos y denegados, en un punto de acceso y en un intervalo de fechas especificadas por el usuario.

Generar reporte detallado

Permite generar un reporte detallando el nombre de la persona, la fecha, hora y si fue concedido o denegado el acceso, por una puerta y en un intervalo de fechas especificadas por el usuario.

2.4. TARJETAS CRC

En esta sección se describen las tarjetas CRC (Clase, Responsabilidad y Colaboración) que contiene el nombre de la clase, sus responsabilidades y las clases que colaboran.

2.4.1 TARJETAS CRC DEL SISTEMA

En la Tabla 2.59 se muestra las tarjetas CRC clasificadas en: clases controladoras y clases entidades.

Tabla 2.59: Tarjetas CRC del Sistema

Clases Controladoras	Clases Entidades
Controlador Acceso	Archivo de Propiedades
Controlador LEDS	Acceso
Controlador Entradas	Circuito LEDS
Controlador Horario de Laboratorio	Entradas
Controlador Materia	Estudiante
Controlador Periodo	Foto
Controlador Permisos	Horario de Permiso
Controlador Persona	Horario de Laboratorio
Controlador Reportes	Laboratorio
Controlador Requerimiento	Materia
Controlador RFID	Periodo
Controlador Laboratorio	Permiso
Controlador Sistema Operativo	Profesor
Controlador Tarjeta	Persona
	Requerimiento
	Reporte Total
	Reporte Detallado
	Sistema Operativo
	Tarjeta
	Tipo de Persona

Clases Controladoras	Clases Entidades
	Usuario

Fuente: Elaborador por el Autor.

En la Tabla 2.60 y Tabla 2.61 se muestra un ejemplo de las tarjetas CRC del sistema.

Tabla 2.60: Tarjeta CRC. Persona

PERSONA	
Responsabilidades	Colaboradores
Devolver la información de Persona	Tipo de Persona

Fuente: Elaborador por el Autor.

Tabla 2.61: Tarjeta CRC. Controlador Persona

CONTROLADOR PERSONA	
Responsabilidades	Colaboradores
Registrar Persona de acuerdo al Tipo de Persona en la BDD.	Tipo de Persona
Actualizar Persona de acuerdo al Tipo de Persona en la BDD.	Persona
Consultar Persona en la BDD de acuerdo al Tipo de Persona.	Foto
Eliminar Persona de la BDD.	Materia
Validar Usuario	Estudiante
	Profesor
	Usuario

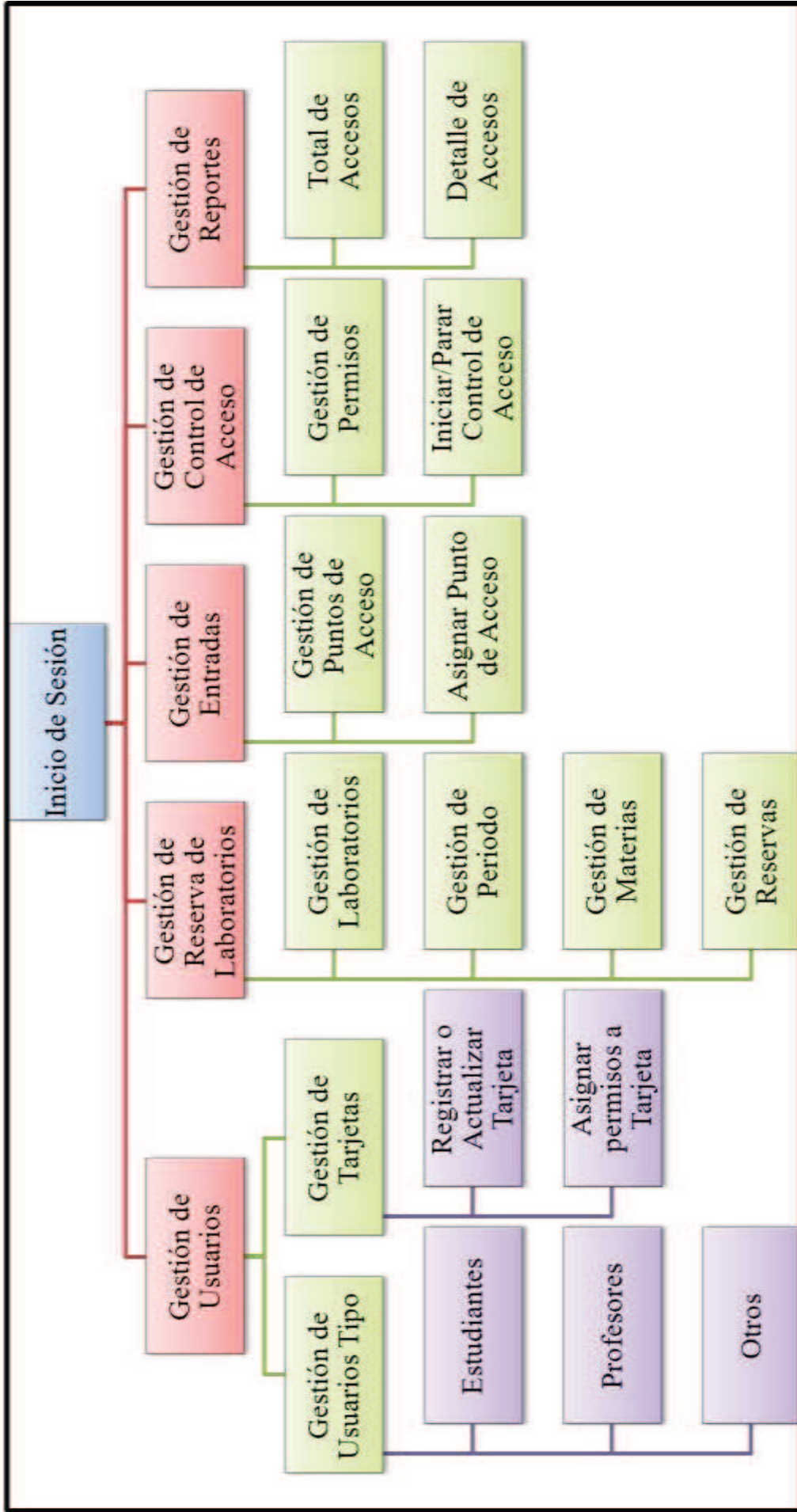
Fuente: Elaborador por el Autor.

Las tarjetas CRC utilizadas para el desarrollo del sistema de control de acceso se encuentran adjuntas en el Anexo C.

2.4.2 ESTRUCTURA DEL SISTEMA DE CONTROL DE ACCESO

En esta sección se presenta un Mapa de Sitio para cada tipo de usuario del sistema:

En la Figura 2.1 se muestra el Mapa del Sitio para el usuario tipo Administrador, el cual tiene acceso a todos los módulos del sistema.



* Otros: Se refiere a usuarios tipo Personal Administrativo, Limpieza e Invitados.

Figura 2.1: Mapa del sitio del tipo usuario Administrador.

2.4.3 ESPECIFICACIÓN DE LA ARQUITECTURA DEL SISTEMA

El diseño de la arquitectura del sistema se ha basado aplicando las mejores prácticas de programación propuestas por el patrón de diseño MVC (Modelo, Vista, Controlador) como se muestra en la Figura 2.2.

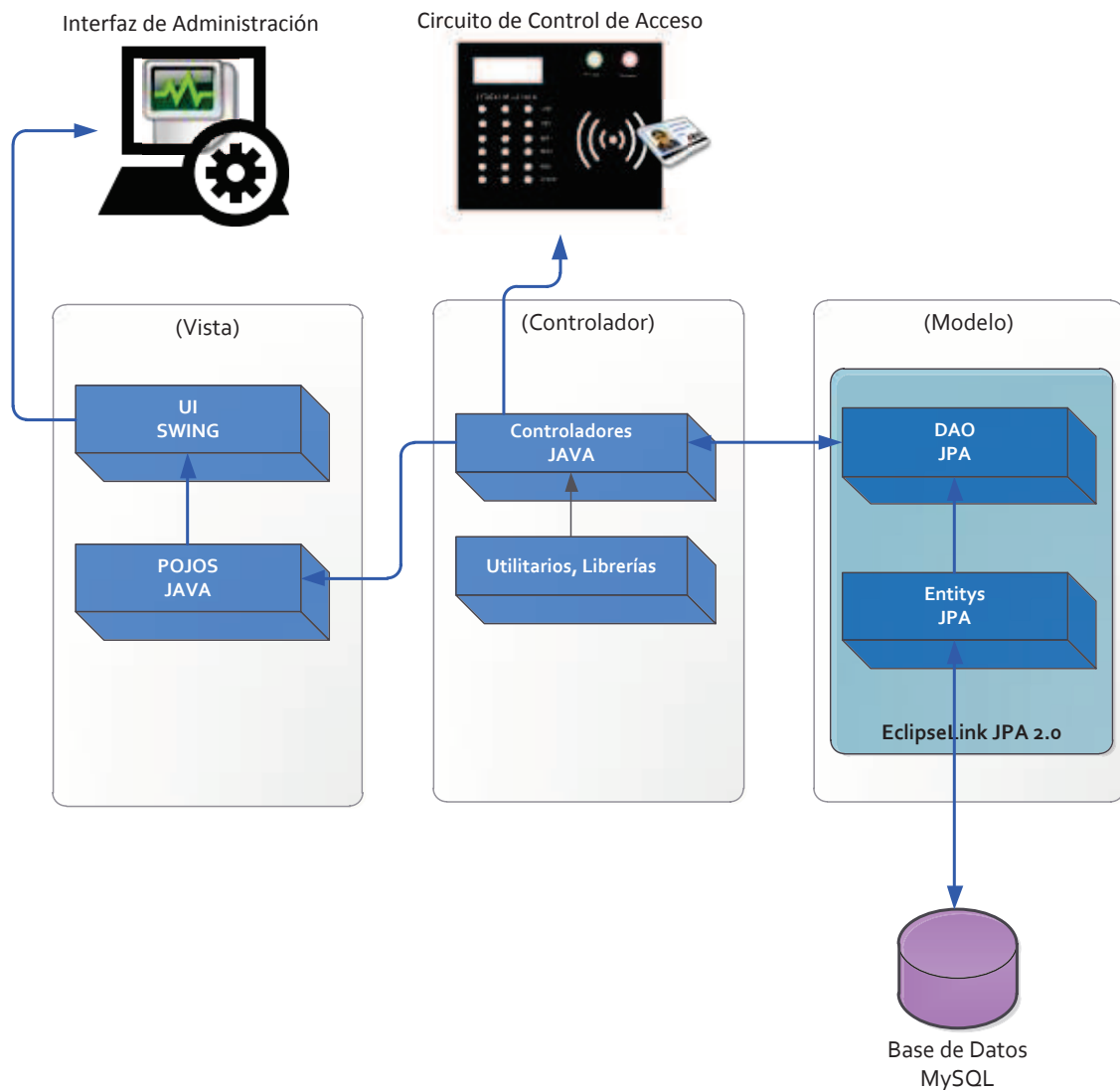


Figura 2.2: Arquitectura del Sistema de Control de Acceso

Fuente: Elaborador por el Autor.

A continuación se describe cada capa del sistema:

Modelo

Esta capa es responsable de acceder a la base de datos que se encuentra en el

Servidor de BDD MySQL. El acceso se lo realiza a través motor de persistencia EclipseLink. En esta capa se encuentran los siguientes elementos:

- **Entitys JPA:** Es una clase (POJO) que representa una tabla de una base de datos y cada instancia de esta clase representa un registro de la tabla. Mediante los Entitys se consigue mapear entre las propiedades de una clase y los campos de un tabla, además se especifica las relaciones que tiene la tabla con otras (uno a uno, uno a muchos, muchos a uno y muchos a muchos).
- **Clases DAO:** Son clases que obedecen el patrón DAO (Data Access Object), se generan a partir de las Clases Entitys y contienen métodos necesarios para registrar, obtener, actualizar y eliminar un Entity.

Controlador

Esta capa se implementa utilizando JAVA como lenguaje de programación junto con NetBeans como IDE de desarrollo. Contiene las reglas de negocio para reaccionar ante eventos capturados en la Interfaz de Administración, como también en el Circuito de Control de Acceso. En esta capa se encuentran los siguientes elementos:

- **Controladores:** Son clases JAVA que contienen las reglas de negocio y actúan ante eventos realizados en la Capa de Presentación (Interfaz de Usuario) e invocan peticiones a la Capa de Modelo. También el Controlador responde ante eventos generados en el Circuito de Control de Acceso (tarjeta RFID detectada) y de acuerdo al resultado de la tarea ejecutada se permite o deniega el acceso.
- **Utilitarios, Librerías:** Los Utilitarios son clases JAVA, que permiten realizar un tratamiento de los datos provenientes de la Capa de Modelo para posteriormente enviar a la Capa de Presentación. Las Librerías proporcionan un conjunto de Clases para dar soporte a las operaciones realizadas por las clases tipo Controlador, por ejemplo: Comunicación con

el Circuito de Control de Acceso (Comunicación Serial), manejo de lector RFID (Smart Card I/O).

Vista

En esta capa se encuentra las interfaces gráficas que se presentan al usuario. El diseño de la interfaz se lo realiza en NetBeans y se emplea SWING de JAVA. En esta capa se encuentran los siguientes elementos:

- **POJOS:** Son clases JAVA simples que no extienden, ni implementan nada en especial. Un POJO únicamente contiene sus atributos privados junto con los métodos getters y setters. En esta capa se utilizan para almacenar los datos que se capturan o muestran al usuario por medio de la interfaz gráfica.
- **UI (SWING):** Son clases que representan una interfaz gráfica y se lo realiza empleando la biblioteca gráfica JAVA Swing. Estas interfaces de usuario se alimentan de los datos almacenados en los objetos POJOS.

Adicionalmente el diseño de la placa del Circuito de Control de Acceso se muestra en la Figura 2.15.

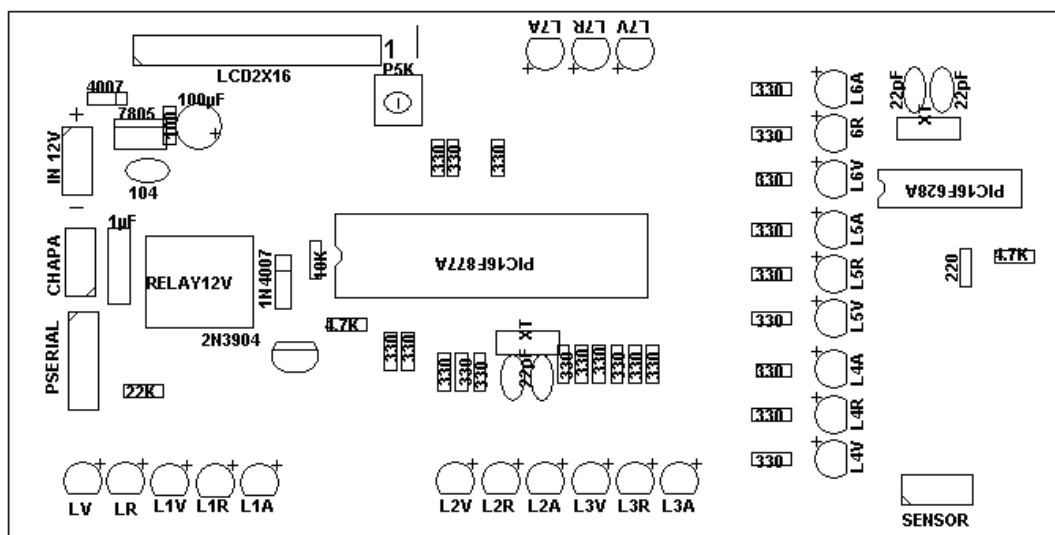


Figura 2.3: Diseño de la placa del Circuito de Control de Acceso

Fuente: Elaborador por el Autor.

Modelo Entidad Relación



Figura 2.4: Modelo Entidad Relación

Diagrama de Componentes

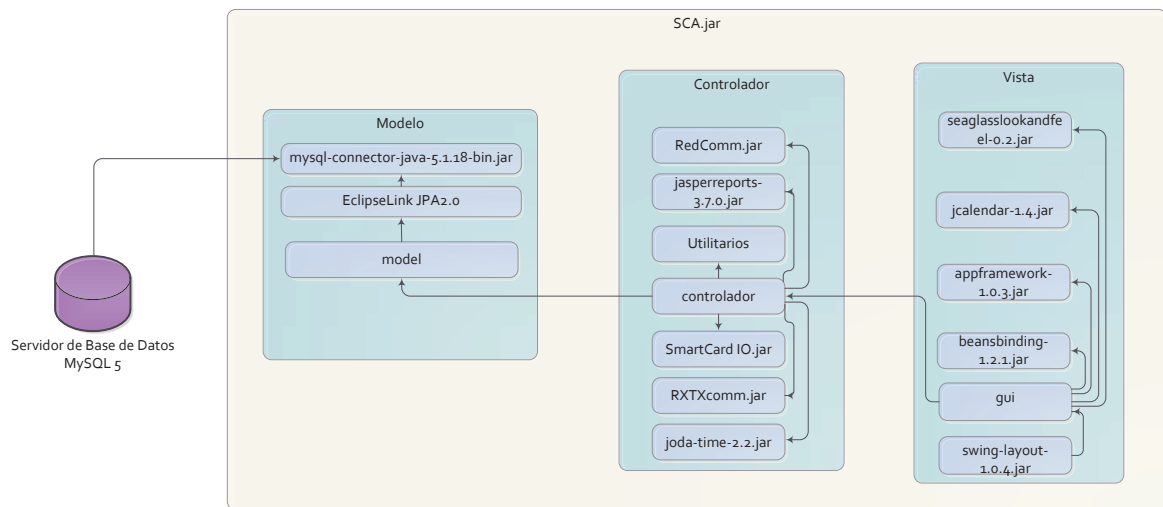


Figura 2.5: Diagrama de Componentes

Fuente: Elaborador por el Autor.

El sistema informático corresponde a una aplicación cliente/servidor en donde:

Servidor

En el lado del servidor se tiene al gestor de base datos MySql 5, en donde se encuentra el esquema de la base de datos utilizada por la aplicación.

Cliente

En el lado del cliente se tiene a la aplicación SCA.jar, misma que tiene las siguientes capas lógicas:

En la capa de modelo se tiene los siguientes componentes:

Tabla 2.62: Componentes de la capa de modelo.

Componente	Descripción
mysql-connector-java-5.1.18-bin.jar	Permite establecer una conexión entre el servidor de base de datos y la aplicación.
EclipseLink JPA2.0.jar	Maneja la persistencia en la aplicación, se encarga de mapear una clase tipo entity de la

Componente	Descripción
	aplicación con una tabla de la base de datos.
model	Conjunto de clases tipo Entity, mismas que representan a una tabla de la base de datos.

Fuente: Elaborador por el Autor.

En la capa de controlador se tiene los siguientes componentes:

Tabla 2.63: Componentes de la capa de controlador.

Componente	Descripción
RedComm.jar RXTXcomm.jar	Permiten establecer una comunicación serial entre la aplicación y el circuito de control.
jasperreports-3.7.0.jar	Permite generar reportes a partir de una plantilla diseñada previamente.
Utilitarios	Contiene clases que dan soporte para el tratamiento de datos.
Controlador	Contiene clases que se encargan de procesar las reglas del negocio establecidas para el sistema informático, esta acción se lleva a cabo cuando se origina eventos en la vista.
SmartCard IO.jar	Permite establecer una comunicación con el lector RFID y con las tarjetas que el lector detecta en su proximidad.
joda-time-2.2.jar	Librería que brinda un conjunto de funciones para el tratamiento de fechas.

Fuente: Elaborador por el Autor.

En la capa de vista se tiene los siguientes componentes:

Tabla 2.64: Componentes de la capa de vista.

Componente	Descripción
seaglasslookandfeel-0.2.jar	Permite cambiar a un tema la apariencia de la vista.
jcalendar-1.4.jar	Permite incrustar en la vista calendarios dinámicos para seleccionar fechas.
appframework-1.0.3.jar	Es un Framework que da soporte para el desarrollo de aplicaciones basadas en Swing.
beansbinding-1.2.1.jar	Permite ligar un componente de la vista con una variable que sea compatible, de tal manera si se cambia el valor de la variable, el componente de la vista se refresca y muestra el valor actual de la variable.
gui	Contiene clases que representan las interfaces de usuario.
swing-layout-1.0.4.jar	Librerías que dan soporte para la creación de las interfaces de usuario.

Fuente: Elaborador por el Autor.

Diagrama de Paquetes

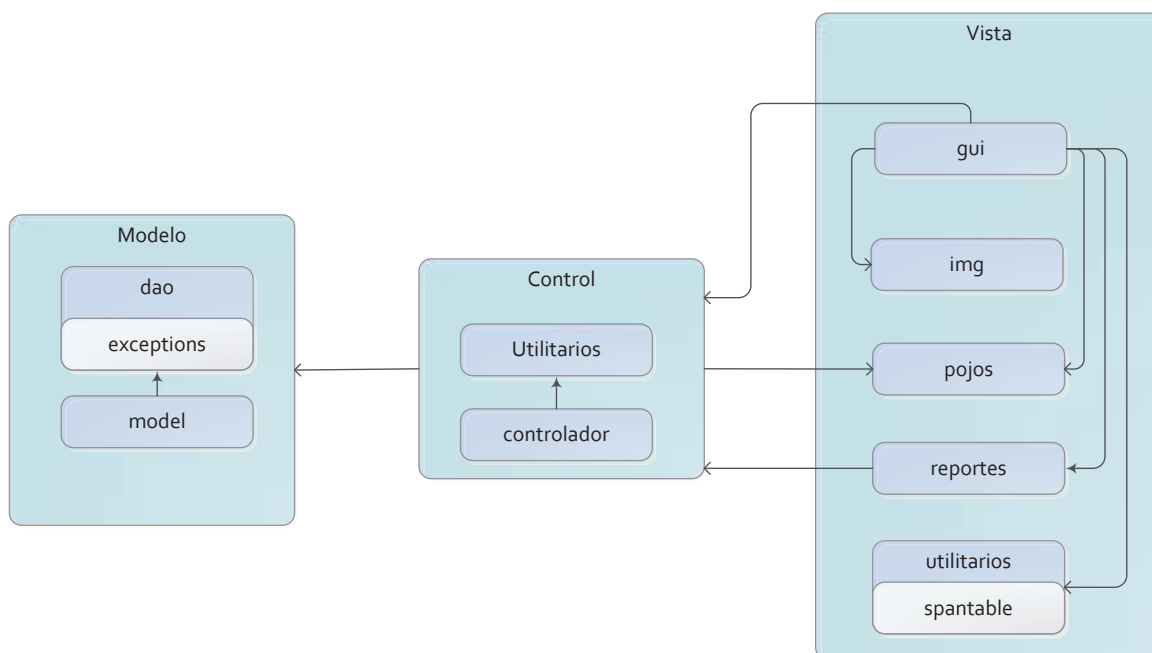


Figura 2.6: Diagrama de Paquetes

Fuente: Elaborador por el Autor.

En la capa modelo existe los siguientes paquetes:

Tabla 2.65: Paquetes de la capa de modelo.

Paquete	Descripción
dao	Contiene clases que permiten realizar operaciones de insertar, actualizar, eliminar y buscar registros sobre una tabla de la base de datos representada por una clase modelo.
exceptions	Contiene clases que manejan las excepciones cuando se ejecuta alguna operación en un objeto DAO.
model	Contiene clases tipo Entities, mismas que representan a una tabla de la base de datos.

Fuente: Elaborador por el Autor.

En la capa de control existen los siguientes paquetes:

Tabla 2.66: Paquetes de la capa de controlador.

Paquete	Descripción
controlador	Contiene clases que se encargan de procesar las reglas del negocio establecidas para el sistema informático, esta acción se lleva a cabo cuando se origina eventos en la vista.
utilitarios	Contiene clases que dan soporte para el tratamiento de datos.

Fuente: Elaborador por el Autor.

En la capa de vista existen los siguientes paquetes:

Tabla 2.67: Paquetes de la capa de vista.

Paquete	Descripción
gui	Contiene clases que representan las interfaces de usuario.
Img	Contiene imágenes utilizadas en las interfaces de usuario.
pojos	Contiene clases que almacenan los datos capturados por medio de las interfaces de usuario.
reportes	Contiene las plantillas utilizadas para generar los reportes con la herramienta iReports, a su vez contiene las clases que generan la información requerida.
utilitarios	Contiene clases que permiten mejorar la presentación de las interfaces de usuario.
spantable	Contiene clases externas e internas que permite en una tabla, dividir celdas o unir celdas.

Fuente: Elaborador por el Autor.

CAPÍTULO 3: DESARROLLO Y PRUEBAS

3.1. PROGRAMACIÓN

Antes de realizar la programación del sistema de control de acceso, se define los estándares de programación a utilizarse durante el desarrollo.

La programación del sistema, se lo realiza siguiendo la planificación del plan de entrega definido en el capítulo 2. En el plan de entrega se tiene cada iteración con un conjunto de historias que se deben implementar.

3.1.1 ESTÁNDARES DE IMPLEMENTACIÓN

En esta sección se describe los estándares empleados para la implementación de cada una de las capas del sistema:

Cada capa del sistema se encuentra en carpetas llamadas modelo, vista y controlador respectivamente.

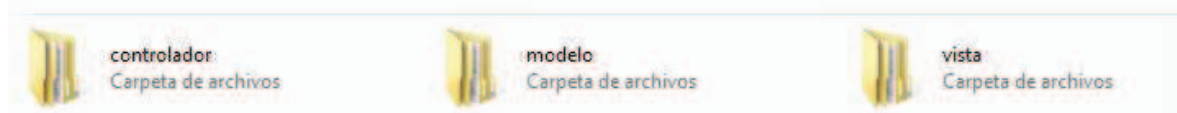


Figura 3.1: Organización de las capas del sistema en carpetas.

Fuente: Elaborador por el Autor.

Estándares para aplicativo

Se describe los estándares utilizados para la implementación del modelo, vista y controlador.

Nomenclatura en la capa VISTA

La nomenclatura empleada para implementar la capa de presentación es la siguiente.

Tabla 3.1: Nomenclatura utilizada en la implementación de la capa de vista.

Objeto	Nomenclatura	Ejemplo
JButton	jButtonVariable	jButtonAceptar
JComboBox	jComboBoxVariable	jComboBoxPeriodo
JTextField	jTextFieldVariable	jTextFieldNombre
JLabel	jLabelVariable	jLabelNombre
JPanel	jPanelVariable	jPanelHorarios
JTabbedPane	jTabbedPaneVariable	jTabbedPaneHorariosLaboratorio
JList	jListVariable	jListHorariosDisponibles
JRadioButton	jRadioButtonVariable	jRadioButtonLAN
JPasswordField	jPasswordFieldVariable	jPasswordFieldUsuario
JSplitPane	jSplitPaneVariable	jSplitPaneMenu
JMenuBar	jMenuBarVariable	jMenuBarMenuPrincipal
JTable	jTableVariable	jTablePersonal

Fuente: Elaborador por el Autor.

Nomenclatura en la capa CONTROLADOR

Para realizar la implementación de la capa controlador se han empleado las convenciones especificadas por JAVA².

Tabla 3.2: Estándares de nomenclatura.

COMPONENTE	CONVENCIÓN
CLASES	<p>Los nombres de las clases deben ser sustantivos y con la primera letra mayúscula, en caso de mezclar varios sustantivos la primera letra de cada palabra debe ir en mayúscula y sin espacios.</p> <p>Utilizar palabras completas, evitar abreviaciones y acrónimos.</p>

² <http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

COMPONENTE	CONVENCIÓN
VARIABLES	<p>Los nombres de las variables deben ir en minúsculas, en caso de tener nombre compuesto, la primera letra a partir de la segunda palabra se deben emplear mayúsculas.</p> <p>Los nombres deben ser descriptivos de acuerdo al valor que contienen.</p> <p>Evitar nombres de una sola letra, a menos que se traten de variables auxiliares como i, j, k, etc. Utilizadas para sentencias if, for y while.</p>
MÉTODOS	<p>Los nombres de los métodos deben ser verbos y se escriben en minúsculas, en caso de tener nombre compuesto, la primera letra a partir de la segunda palabra se deben emplear mayúsculas.</p>
CONSTANTES	<p>Los nombres de las constantes deben estar en mayúsculas y separadas con “_” en caso de tener nombres compuestos.</p>

Fuente: <http://www.oracle.com>.

Nomenclatura utilizada para la capa de Modelo

En la capa de modelo se ha empleado la nomenclatura propuesta por el IDE de desarrollo Netbeans.

Tabla 3.3: Nomenclatura para la capa de modelo

COMPONENTE	CONVENCIÓN
Entitys	<p>El nombre de la clase debe tener la primera letra mayúscula, si es un nombre compuesto, cada palabra debe empezar con letra mayúscula.</p> <p>El nombre de las variables debe ser en letra minúscula, si un nombre compuesto, cada palabra a partir de la segunda debe empezar con letra mayúscula.</p> <p>Se debe implementar los métodos getters and setters para todas las variables de la clase.</p>
Controladores JPA	<p>El nombre de la clase debe ser el nombre de la clase</p>

COMPONENTE	CONVENCIÓN
	entidad seguido de JpaController, ejemplo: Nombre de la clase entidad: Persona Nombre del controlador: PersonaJpaController

Fuente: Elaborador por el Autor.

Estándar para el diseño de la Base de Datos

La nomenclatura empleada para el diseño de la base de datos, ha sido una recopilación de la nomenclatura propuesta por MySQL Workbench y nomenclaturas publicadas en la web³.

Tabla 3.4: Nomenclatura para diseño de la base de datos.

COMPONENTE	CONVENCIÓN
Base de Datos	El nombre de la base de datos debe estar en mayúsculas.
Tablas	El nombre de la tabla debe ser descriptivo, en singular y en mayúsculas. Las tablas identifican una entidad del sistema con un nombre completo. Si la entidad o proceso que genera la tabla tienen más de una palabra se deberá colocar cada una de ellas en singular y deben ser separadas por un "_".
Campos	Los nombres de los campos deben ser descriptivos, en singular y en minúscula. Para nombres de campos compuestos emplear la primera letra mayúscula a partir de la segunda palabra. Para campos que sean para claves foráneas anteponer el nombre de la tabla en mayúscula luego el separador "_" seguido del nombre del campo.
Constraints	El nombre de la tabla debe estar en mayúscula. <i>1. Índices con llaves primarias: PK + Nombre de la tabla</i> <i>2. Índices con llaves foráneas o relacionadas:</i>

³ [Estándar de nomenclatura para base de datos](#) - Assembla

COMPONENTE	CONVENCIÓN
	<i>FK_NombreTablaOrigen_NombreTablaReferenciada</i> 3. <i>Índices Unique: UQ_NemónicoTabla_NombreUnique</i> 4. <i>Default: DF_NemónicoTabla_NombreColumna</i> 5. <i>Check: CK_NemónicoTabla_NombreCheck</i>
Triggers	Nomenclatura para los Triggers : 6. <i>TR_NOMBRETABLA_D (Cuando se realiza una eliminación en la tabla).</i> 7. <i>TR_NOMBRETABLA_U (Cuando se realiza una actualización en la tabla).</i> 8. <i>TR_NOMBRETABLA_I (Cuando se realiza una inserción en la tabla).</i> 9. <i>TR_NOMBRETABLA_IU (Cuando se realiza una inserción o actualización en la tabla).</i>
Vistas	Nomenclatura para las Vistas: Para el nombre de las Vistas utilizar verbos en Infinitivo. VW_YYY.... VW : Identificador para las Vistas YYY : Nombre de la Vista en Mayúsculas

Fuente: Elaborador por el Autor.

3.1.2 CODIFICACIÓN DE HISTORIAS DE USUARIO

Para ejemplificar la codificación de las historias de usuario de cada iteración, se toma como referencia la historia de usuario “Gestión de profesores”. Las tareas a realizar se muestran en la Figura 3.2.

<input type="checkbox"/> Gestión de Profesores.	10,5 días	mié 19/06/13	mié 03/07/13
Diseño CRC	0,5 días	mié 19/06/13	mié 19/06/13
Pruebas Unitarias	2,5 días	mié 19/06/13	vie 21/06/13
<input type="checkbox"/> Programación por pares	6 días	lun 24/06/13	lun 01/07/13
Diseño de la Interfaz Gestión de Profesores	2 días	lun 24/06/13	mar 25/06/13
Validación de datos de la Interfaz	2 días	mié 26/06/13	jue 27/06/13
Almacenar datos en la BDD	1 día	vie 28/06/13	vie 28/06/13
Ejecución de pruebas Unitarias	1 día	lun 01/07/13	lun 01/07/13
Integración	1 día	mar 02/07/13	mar 02/07/13
Pruebas de aceptación	0,5 días	mié 03/07/13	mié 03/07/13

Figura 3.2: Tareas de la Historia de Usuario “Gestión de profesores”.

Fuente: Elaborador por el Autor.

Diseño CRC

El equipo de desarrollo identifica las Tarjetas CRC a utilizar en la implementación

de la historia de usuario. Para este ejemplo las tarjetas identificadas son:

- Clases Entidades: Persona, Foto, Materia
- Clase Controladora: Controlador Persona.

Programación

La programación de cada Historia de Usuario se realiza ejecutando dos actividades: la codificación y refactorización del software.

Codificación

La codificación se la realiza utilizando NetBeans como IDE de desarrollo y se sigue el patrón de diseño MVC. Se empieza implementando la capa de modelo utilizando las herramientas de NetBeans para generar automáticamente las clases Entidades.

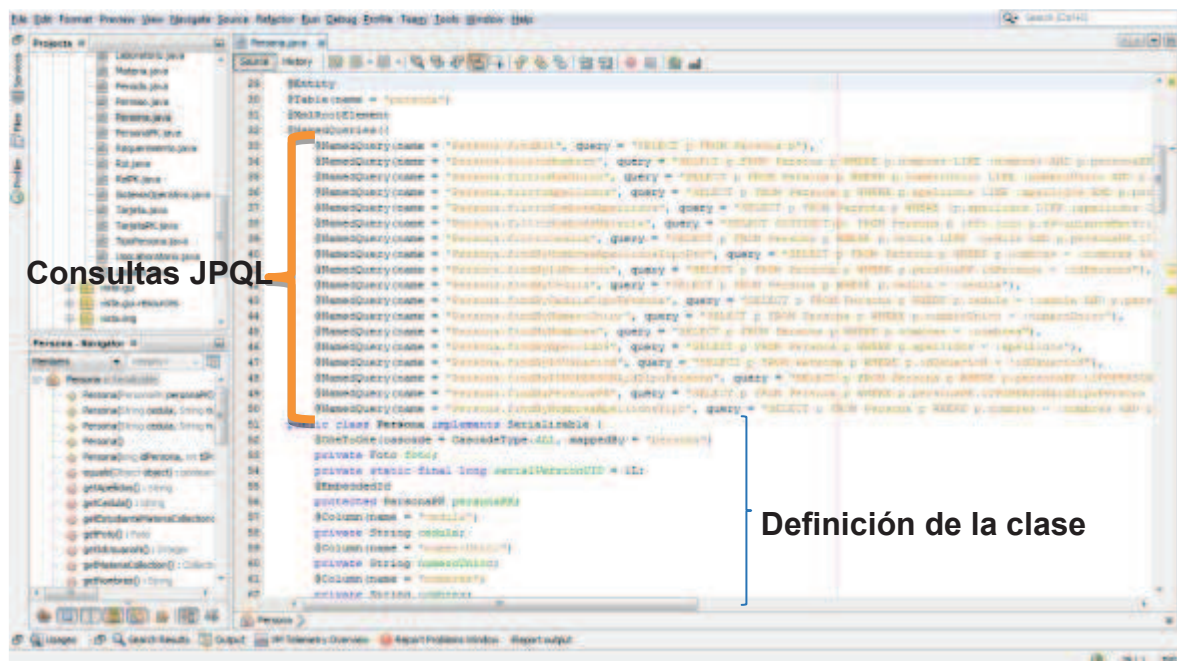


Figura 3.3: Persona.java

Fuente: Elaborador por el Autor.

En la Figura 3.3 se ilustra la estructura que tiene una clase Entidad generada con la herramienta. La clase `Persona.java` contiene un conjunto de consultas en lenguaje JPQL, luego sigue la definición de la clase en donde se encuentran los atributos relacionados con los campos de la tabla de la base de datos y finalmente los métodos getters y setters. Después de implementar la Entidad se crea la clase

DAO por cada Entidad, que igualmente se genera automáticamente con la herramienta de desarrollo como se muestra en la Figura 3.4.

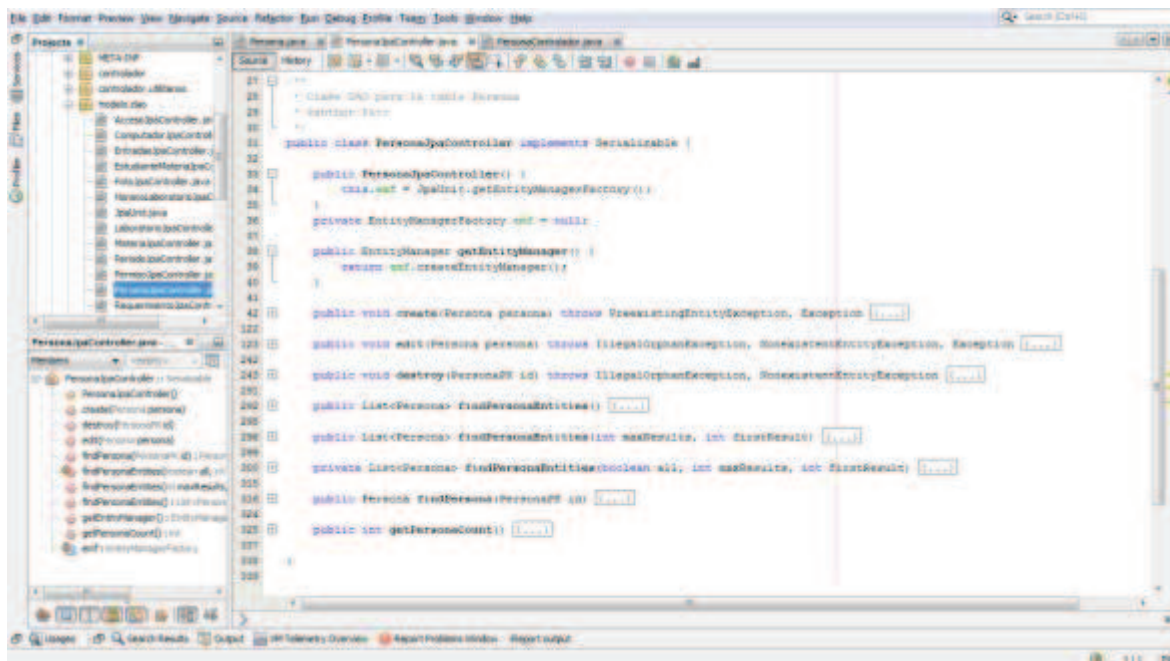


Figura 3.4: Clase DAO – PersonaJpaController.java

Fuente: Elaborador por el Autor.

Una vez implementada la capa de modelo se implementa la capa de controlador, para este caso se ilustra la clase controladora “PersonaControlador.java”:

```

package controlador;

/**
 * Clase tipo controlador para relizar operaciones de lectura y escritura
 * sobre la tabla de Persona de la base de datos
 * @author Pato
 */
public class PersonaControlador {
    PersonaJpaController personaJpaController;
    Persona persona ;
    EntityManager entityManager;
    Query consulta;
    public String mensaje;

    /**
     * Crea un nuevo objeto tipo controlador persona
     */
    public PersonaControlador () {
        mensaje="";
        personaJpaController = new PersonaJpaController ();
        entityManager= personaJpaController.getEntityManager ();
    }

```

```

/**
 * Obtiene la lista de personal filtrando por el tipo de persona
 (profesor, estudiante, etc)
 * @return List<PersonaPOJO>
 */
public List<PersonaPOJO> getAllPersonal(int idTipoPersona){
    List<PersonaPOJO> personas = new ArrayList();

consulta=entityManager.createNamedQuery("Persona.findByTIPOPERSONAidTipoP
ersona");
    consulta.setParameter("tIPOPERSONAidTipoPersona", idTipoPersona);
    List<Persona> listaProfesores = consulta.getResultList();
    for (Persona profesor : listaProfesores) {
        PersonaPOJO personaPOJO =null;
        if(profesor.getFoto() !=null){
            personaPOJO = new
PersonaPOJO(profesor.getPersonaPK().getIdPersona(), profesor.getCedula(),
profesor.getNombres(), profesor.getApellidos(), new
FotoPOJO(profesor.getFoto().getFoto(),
profesor.getFoto().getFotoPK().getPERSONAidPersona(),
profesor.getFoto().getFotoPK().getPERSONATIPOPERSONAidTipoPersona()));
        }
        else{
            personaPOJO= new
PersonaPOJO(profesor.getPersonaPK().getIdPersona(), profesor.getCedula(),
profesor.getNombres(), profesor.getApellidos());
        }
        personaPOJO.setIdTipoPersona(idTipoPersona);
        personas.add(personaPOJO);
    }
    return personas;
}

/**
 * Busca una persona por la cédula
 * @param cedula Cedula de la persona
 * @return objeto tipo Persona o null
 */
public Persona getPersonaXCedula(String cedula, int tipoPersona){
    consulta =
entityManager.createNamedQuery("Persona.findByCedulaTipoPersona");
    consulta.setParameter("cedula", cedula);
    consulta.setParameter("tIPOPERSONAidTipoPersona", tipoPersona);

    if(!consulta.getResultList().isEmpty()){
        Persona persona = (Persona) consulta.getSingleResult();
        return persona;
    }
    return null;
}

long idPersona;
/**
 * Registra en la BDD una persona
 * @param personaP objeto que se debe almacenar en la base de datos
 * @return true si el registro fué correcto o false si hubo errores
 */
public boolean registrarPersona(PersonaPOJO personaP){

```



```

if(getPersonaXCedula(personaP.getCedula(), personaP.getIdTipoPersona()) != null){
    mensaje = "Yá existe el registro";
    return false;
}
TipoPersona tipoPersona = new
TipoPersona(personaP.getIdTipoPersona());
Persona persona = null;
persona = new
Persona(personaP.getCedula(), personaP.getNumeroUnico(), personaP.getNombres(), personaP.getApellidos(), tipoPersona);

try {
    EntityManager
entityManager=personaJpaController.getEntityManager();
entityManager.getTransaction().begin();

    personaJpaController.create(persona);
    if(personaP.getFoto() !=null){
        persona =
getPersonaXCedula(personaP.getCedula(), personaP.getIdTipoPersona());
        Foto foto = new Foto(new
FotoPK(persona.getPersonaPK().getIdPersona(),
persona.getPersonaPK().getTIPOPERSONAidTipoPersona()),
personaP.getFoto().getNombre(), personaP.getFoto().getFoto());
        //foto.setPersona(persona);
        persona.setFoto(foto);
        entityManager.persist(foto);
        entityManager.getTransaction().commit();
    }

    idPersona = persona.getPersonaPK().getIdPersona();
    System.out.println("id persona:
"+persona.getPersonaPK().getIdPersona());
    return true;
} catch (PreexistingEntityException ex) {
    mensaje = "Yá existe el registro";

//Logger.getLogger(PersonaControlador.class.getName()).log(Level.SEVERE,
null, ex);
} catch (Exception ex) {
    Throwable t = getLastThrowable(ex); //fetching Internal
Exception
    SQLException exxx = (SQLException) t; //casting Throwable
object to SQL Exception
    if(exxx.getSQLState().compareTo("23000")==0) // Integrity
constraint violation
    {
        mensaje = "Ya existe un registro con el mismo número
único, no puede repetirse";
    }else
        mensaje = "Se produjo un error desconocido";

//Logger.getLogger(PersonaControlador.class.getName()).log(Level.SEVERE,
null, ex);
}
System.err.println("No se registro persona");
return false;

```

```

}
/**
 * Registra en la BDD un profesor
 * @param personaP objeto que se debe almacenar en la base de datos
 * @return true si el registro fue correcto o false si hubo errores
 */
public boolean registrarProfesor(PersonaPOJO personaP) {
    EntityManager
entityManager=personaJpaController.getEntityManager();
entityManager.getTransaction().begin();

    Persona persona ;

    if(registrarPersona(personaP)){

        if(personaP.getMateriasProfesor().isEmpty()){
            return true;
        }
        mensaje = "";
        //entityManager.

        persona=entityManager.find(Persona.class, new
PersonaPK(idPersona, personaP.getIdTipoPersona()));
        Collection<Persona> profesor= new ArrayList<Persona>();
        profesor.add(persona);
        Collection<MateriaPOJO> materiaPOJOs =
personaP.getMateriasProfesor();
        Collection<Materia> materias = new ArrayList<Materia>();
        for (MateriaPOJO materiaPOJO : materiaPOJOs) {
            Materia materia = entityManager.find(Materia.class,
materiaPOJO.getIdMateria());
            materia.setPersonaCollection(profesor);
            materias.add(materia);
        }
        persona.setMateriaCollection(materias);
        entityManager.merge(persona);

    }else{
        return false;
    }
    try {
        entityManager.getTransaction().commit();
        return true;
    } catch (Exception ex) {
        Throwable t = getLastThrowable(ex); //fetching Internal
Exception
        SQLException exxx = (SQLException) t; //casting Throwable
object to SQL Exception
        if(exxx.getSQLState().compareTo("23000")==0) // Integrity
constraint violation
        {
            mensaje = "Ya existe un registro con el mismo número
único, no puede repetirse";
        }else
            mensaje = "Se produjo un error desconocido";

        //Logger.getLogger(PersonaControlador.class.getName()).log(Level.SEVERE,
null, ex);
        entityManager.getTransaction().rollback();
    }
}

```

```

    }
    System.err.println("No se registro persona");
    return false;
}

private Throwable getLastThrowable(Exception e) {
    Throwable t = null;
    for(t = e.getCause(); t.getCause() != null; t = t.getCause());
    return t;
}
}

```

Y por último se implementa la capa de vista, que viene a ser la interfaz de usuario. Para crear las interfaces de usuario se utiliza SWING de java.

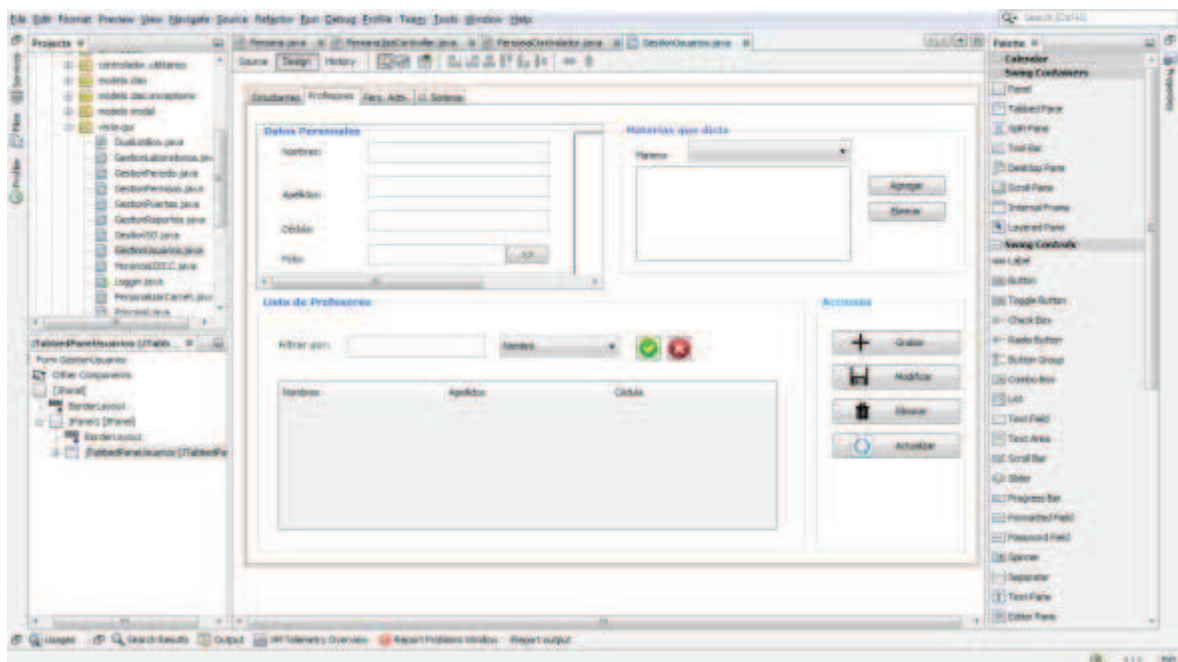


Figura 3.5: Interfaz Gráfica - Gestión de Usuarios

Fuente: Elaborador por el Autor.

En la Figura 3.5 se tiene la interfaz gráfica para realizar el registro de Profesores, luego se crea un evento sobre el botón “Grabar” en donde se invoca al controlador para que realice la operación de registro.

```

private void jButtonNuevoActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    if(!camposProfesorValidos()){
        return;
    }
    //Se verifica si el profesor ya existe

```

```

List<PersonaPOJO> profesorPOJOs = new ArrayList<PersonaPOJO>();
profesorPOJOs.addAll(profesorList);

int index =
controladorPersona.getIndexofProfesor(jTextFieldNombres.getText(),
jTextFieldApellidos.getText(),profesorPOJOs);
System.out.println("index de persona: "+index);
if(index!=-1){

//System.err.println("index"+profesorList.indexOf(auxMateria)+" GR:
"+auxMateria.getGrupo()+" Com: "+profesorList.contains(auxMateria));
JOptionPane.showMessageDialog(this, "El profesor ya existe.
Si desea modificar el registro, presione el botón Modificar");

jTableProfesor.getSelectionModel().setSelectionInterval(index, index);
jTableProfesorMouseClicked(null);
return;
}
/**
 * Se crea un objeto tipo Profesor para almacenar los datos
 * ingresados por el usuario
 * para su posterior registro en la base de datos
 */
PersonaPOJO profesor = new PersonaPOJO();
profesor.setNombres(jTextFieldNombres.getText());
profesor.setApellidos(jTextFieldApellidos.getText());
if(jTextFieldCedula.getText().trim().compareTo("")==0){
profesor.setCedula(null);
}else{
profesor.setCedula(jTextFieldCedula.getText());
}
profesor.setIdTipoPersona(1);
profesor.setMateriasProfesor(materiaListDest);
try {
File file = new File(jTextFieldFotoProfe.getText());
profesor.setFoto(new
FotoPOJO(imageTool.readImageOldWay(file)));
profesor.getFoto().setNombre(file.getName());
} catch (FileNotFoundException ex) {

//Logger.getLogger(GestionUsuarios.class.getName()).log(Level.SEVERE,
null, ex);
} catch (IOException ex) {

//Logger.getLogger(GestionUsuarios.class.getName()).log(Level.SEVERE,
null, ex);
}
if(controladorPersona.registrarProfesor(profesor)){
limpiarCamposProfesor();
controladorPersona = new PersonaControlador();
//profesorList.add(profesor);
observableList.clear();
observableList.addAll(controladorPersona.getAllProfesores());
jTableProfesor.repaint();
jScrollPane5.repaint();
System.out.println("Registro Correcto");
}else{
JOptionPane.showMessageDialog(this, "No es posible guardar el
registro. \nEl usuario a crear ya existe. Revise el campo cédula, no
puede duplicarse!!", "Error de registro",JOptionPane.ERROR_MESSAGE);
}

```

```

    }
}

```

Refactorización

La refactorización del software en XP se la logra gracias a la programación en parejas, una persona se encarga de programar y la otra revisa el código desarrollado. Logrando así que el código sea más fácil de entender y eliminar código duplicado, a continuación se muestra un ejemplo:

A continuación en la Figura 3.6 se puede ver que el código desarrollado tiene código repetido:

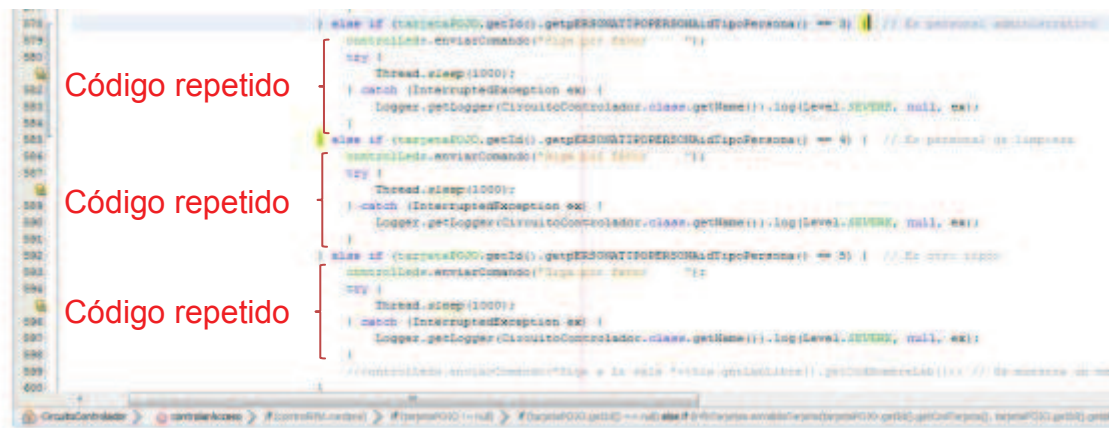


Figura 3.6: Refactorización del Software.

Fuente: Elaborador por el Autor.

En la refactorización del software se elimina el código repetido y en su lugar se crea una función, como se muestra a continuación:

```

* Envía un mensaje de notificación y se muestra en el panel LCD
*/
private void enviarNotificacionAPuerta() {
    controlLeds.enviarComando("Siga por favor      ");
    try {
        Thread.sleep(1000);
    } catch (InterruptedException ex) {
        Logger.getLogger(CircuitoControlador.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

La función creada debe ser reemplazar el código repetido, de esta manera el código queda legible.

```

} else if (tarjetaPOJO.getId().getPERSONATIPOPERSONAidTipoPersona() == 3) { // Es personal administrativo
    enviarNotificacionAPuerta();
} else if (tarjetaPOJO.getId().getPERSONATIPOPERSONAidTipoPersona() == 4) { // Es personal de limpieza
    enviarNotificacionAPuerta();
} else if (tarjetaPOJO.getId().getPERSONATIPOPERSONAidTipoPersona() == 5) { // Es otro tipo
    enviarNotificacionAPuerta();
}

```

Pruebas de Aceptación

La Historia de Usuario finaliza con la revisión de la Prueba de Aceptación definida por el cliente previamente. Si luego de verificar la prueba el cliente indica que se debe realizar cambios, el equipo de desarrollo implementa los nuevos cambios, de lo contrario la historia de usuario satisface la prueba de aceptación y se da por terminada. Las pruebas de aceptación para cada historia de usuario se detallan en la sección 3.4.

3.2. INTEGRACIÓN

En la Figura 3.7 se muestra el modelo propuesto como una solución al problema de control de acceso. En la implementación se ha utilizado tarjetas RFID junto con un lector de tarjetas RFID que se encuentra embebido en el circuito de control de acceso y mediante el uso del Sistema permite controlar el acceso al laboratorio del LDICC y a su vez restringe el acceso al personal no autorizado.

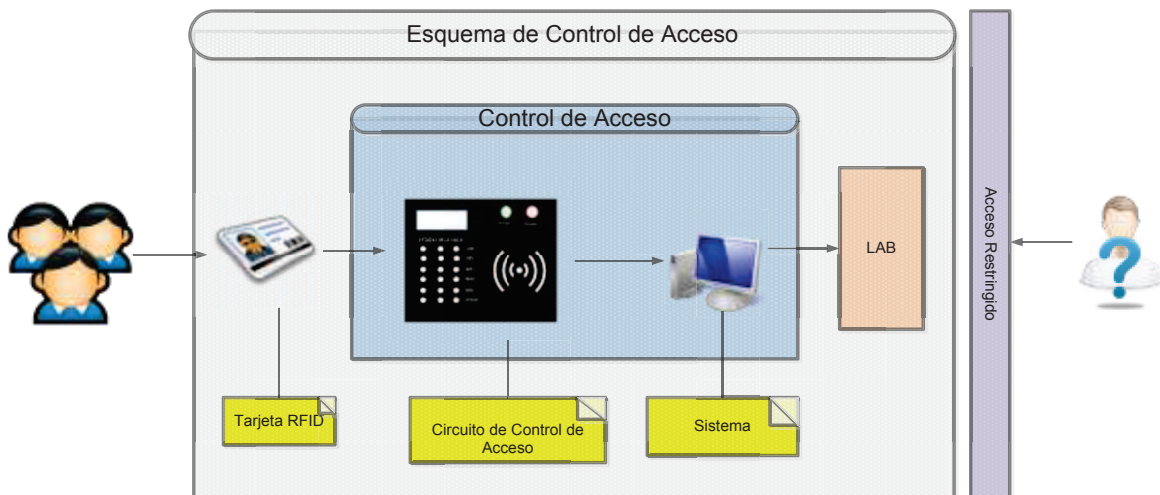


Figura 3.7: Esquema de Control de Acceso

Fuente: Elaborador por el Autor.

Los pasos a seguir para llevar a cabo el control de acceso, mediante el esquema propuesto son:

1. El usuario presenta la tarjeta RFID al Circuito de Control de Acceso.
2. El Circuito de Control de Acceso lee la tarjeta RFID y envía los datos al Sistema (cable usb).
3. El Sistema valida los datos de la credencial con la información almacenada en la base de datos.
4. Si la validación es exitosa se permite el acceso mediante el Circuito de Control con la apertura de la puerta (cable rs232), de lo contrario se restringe.

Para la comunicación del sistema con el lector RFID se emplea la librería SmartCard I/O de java, de igual manera para la comunicación con el Circuito de Control se emplea las librerías RedComm y RXTXcomm, mismas que se especificaron en el capítulo 2, en la sección 4, en el diagrama de componentes.

3.3. IMPLANTACIÓN DEL CASO DE ESTUDIO

La implantación del sistema de control de acceso no se la realizará en su totalidad, debido a que se necesitaría emitir tarjetas RFID a todos los usuarios del LDICC, por tal motivo se realiza una demostración funcional del sistema de control de acceso, en la sala de servidores del LDICC con el Ing. Patricio Proaño.

3.3.1 REQUISITOS DE INSATALACIÓN

Para realizar la instalación del sistema de control de acceso se tiene los siguientes requerimientos de hardware y software.

Hardware

- Computador: Los requerimientos mínimos del computador donde se va instalar el sistema son:
 - Procesador Intel core o superior.
 - Memoria RAM de 1GB o más.
 - Puerto usb

- Puerto serial
- Puerto Ethernet
- Servidor
 - Intel core o superior o superior
 - Memoria RAM 512MB o más
 - Puerto Ethernet
- Circuito de Control de Acceso: elaborado para exclusivamente para el presente proyecto.

Software

- Computador
 - Sistema Operativo: Windows 7
 - JRE 7.0 o superior
- Servidor
 - Sistema Operativo: Sistema Operativo que soporte el motor de base de datos.
 - Motor de base de datos: MySQL 5.0 o superior

3.4. PRUEBAS

3.4.1 PRUEBAS UNITARIAS

En la metodología XP, las pruebas unitarias son utilizadas por los programadores para verificar si el código implementado cumple con el comportamiento esperado en forma aislada antes de ser integrado en el sistema.

Las Pruebas Unitarias se implementan utilizando la plataforma JUnit, de esta manera se genera automáticamente las pruebas unitarias a partir de una clase existente. Las pruebas generadas son para las clases tipo controlador descritas en el diseño de tarjetas CRC. En la Tabla 3.5 se indica las pruebas unitarias de cada clase controladora con los métodos a validar.

Tabla 3.5: Pruebas Unitarias

Clases	Pruebas Unitarias	Métodos
Controladoras		
Controlador Acceso	Controlador Acceso Test	Registrar, Actualizar, Consultar, Eliminar y Validar
Controlador LEDS	Controlador LEDS Test	Actualizar, Imprimir, Borrar, Validar.
Controlador Entradas	Controlador Entradas Test	Asignar, Registrar, Actualizar, Consultar, Eliminar.
Controlador Horario de Laboratorio	Controlador Horario de Laboratorio Test	Registrar, Actualizar, Consultar, Eliminar.
Controlador Materia	Controlador Materia Test	Registrar, Actualizar, Consultar, Eliminar y Validar
Controlador Periodo	Controlador Periodo Test	Registrar, Actualizar, Consultar, Eliminar.
Controlador Permisos	Controlador Permisos Test	Registrar, Actualizar, Consultar, Eliminar.
Controlador Persona	Controlador Persona Test	Generar
Controlador Reportes	Controlador Reportes Test	Generar
Controlador Requerimiento	Controlador Requerimiento Test	Registrar, Actualizar, Consultar, Eliminar
Controlador RFID	Controlador RFID Test	Conectar, Desconectar, Autenticar, Guardar.
Controlador Laboratorio	Controlador Laboratorio Test	Registrar, Actualizar, Consultar, Eliminar.
Controlador Sistema Operativo	Controlador Sistema Operativo Test	Registrar, Actualizar, Consultar, Eliminar y Validar
Controlador Tarjeta	Controlador Tarjeta Test	Registrar, Actualizar, Consultar, Eliminar y Validar

Fuente: Elaborador por el Autor.

A continuación se ilustra la implementación y ejecución de una prueba unitaria para la historia de usuario “Gestión de usuarios – Registro de Profesores”

```

package controlador;

import java.util.ArrayList;
import java.util.List;
import modelo.model.Persona;
import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;
import vista.pojos.MateriaPOJO;
import vista.pojos.PersonaPOJO;
import vista.pojos.UsuarioPOJO;

/**
 *
 * @author Pato
 */
public class PersonaControladorTest {
    PersonaPOJO personaPOJO;
    public PersonaControladorTest () {
        personaPOJO= new PersonaPOJO(Long.parseLong("19"), "0458766565",
"Miriam", "Carrasco");
    }

    @BeforeClass
    public static void setUpClass () {
    }

    @AfterClass
    public static void tearDownClass () {
    }

    @Before
    public void setUp () {
    }

    @After
    public void tearDown () {
    }

    /**
     * Test of getAllPersonal method, of class PersonaControlador.
     */
    @Test
    public void testGetAllPersonal () {
        System.out.println("getAllPersonal");
        int idTipoPersona = 3;
        personaPOJO.setIdTipoPersona(idTipoPersona);
        PersonaControlador instance = new PersonaControlador();
    }
}

```

```

        List expectedResult = new ArrayList();
        expectedResult.add(personaPOJO);
        List result = instance.getAllPersonal(idTipoPersona);
        assertEquals(expectedResult, result);
    }

    /**
     * Test of getPersonaXCedula method, of class PersonaControlador.
     */
    @Test
    public void testGetPersonaXCedula() {
        System.out.println("getPersonaXCedula");
        String cedula = "0458766565";
        int tipoPersona = 3;
        PersonaControlador instance = new PersonaControlador();
        Persona expectedResult = new Persona(19, tipoPersona);
        Persona result = instance.getPersonaXCedula(cedula, tipoPersona);
        assertEquals(expectedResult, result);
    }

    /**
     * Test of registrarProfesor method, of class PersonaControlador.
     */
    @Test
    public void testRegistrarProfesor() {
        System.out.println("registrarProfesor");
        PersonaPOJO personaP = new PersonaPOJO();
        personaP.setIdTipoPersona(2);
        personaP.setApellidos("Laso");
        personaP.setNombres("Gabriela");
        personaP.setCedula("1789457125");
        personaP.setMateriasProfesor(new ArrayList<MateriaPOJO>());
        PersonaControlador instance = new PersonaControlador();
        boolean expectedResult = false;
        boolean result = instance.registrarProfesor(personaP);
        assertEquals(expectedResult, result);
    }
}
}

```

Una vez implementada la prueba unitaria se ejecuta y el resultado indica si el método es correcto o si existen errores, como se muestra en la Figura 3.8.



Figura 3.8: Resultado de prueba unitaria – PersonaControladorTest.java

Fuente: Elaborador por el Autor.

3.4.2 PRUEBAS DE ACEPTACIÓN

Las pruebas de aceptación son definidas por el cliente junto con el equipo de desarrollo y se utilizan para validar los requerimientos del cliente con la implementación de una historia de usuario.

Para elaborar las pruebas de aceptación existe un formato propuesto por Kent Beck como se muestra en la Tabla 3.6

Tabla 3.6: Formato de Prueba de Aceptación⁴

Número de prueba:
Historia de Usuario:
Precondiciones:
Entrada:
Resultado esperado 1:
Resultado esperado 2:

Fuente: Elaborador por el Autor.

En donde:

Número de prueba: Identificador de la prueba de aceptación.

Historia de Usuario: Nombre de la historia de usuario para la que se define la prueba.

Precondiciones: Condiciones que debe cumplir antes de ejecutar la funcionalidad de la historia de usuario.

Entradas: Son los pasos que tiene que hacer el usuario para realizar la funcionalidad de la historia de usuario.

Resultado esperado: Respuesta que se espera tras ejecutar la funcionalidad de la historia de usuario.

3.4.2.1. Pruebas de Aceptación del Sistema

En esta sección se detalla las pruebas de aceptación diseñadas para el desarrollo

⁴ KENT BECK Martin. Planning Extreme Programming. Adison Wedley 2000

del sistema.

Tabla 3.7: Prueba de aceptación - Registro de Profesores

Número de prueba: 1
Historia de Usuario: Gestión de Profesores – Registro
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña Profesor</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón Guardar</p>
<p>Resultado esperado 1:</p> <p>El sistema registra los datos del profesor en la base de datos.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de profesores.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe el profesor.</p>

Fuente: Elaborador por el Autor.

Tabla 3.8: Prueba de aceptación - Modificar Profesor

Número de prueba: 2
Historia de Usuario: Gestión de profesor – Modificar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña Profesor.</p> <p>El usuario selecciona un registro de la tabla de profesores.</p>

<p>El usuario modifica los campos en el formulario.</p> <p>El usuario presiona botón Modificar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos del profesor.</p> <p>El sistema actualiza los datos en la tabla de lista de profesores.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>

Fuente: Elaborador por el Autor.

Tabla 3.9: Prueba de aceptación - Eliminar Profesor.

<p>Número de prueba: 3</p>
<p>Historia de Usuario: Gestión de profesores – Eliminar.</p>
<p>Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña Profesor.</p> <p>El usuario selecciona un registro de la tabla de profesores.</p> <p>El usuario presiona botón Eliminar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos del profesor.</p> <p>El sistema elimina los datos en la tabla de lista de profesores.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p> <p>El sistema muestra mensaje de error si el profesor tiene registros de acceso.</p>

Fuente: Elaborador por el Autor.

Tabla 3.10: Prueba de aceptación - Registro de Estudiante

Número de prueba: 4
Historia de Usuario: Gestión de estudiantes – Registro.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada: El usuario selecciona el menú Gestión de Usuario. El usuario selecciona pestaña Estudiante. El usuario selecciona un periodo. El usuario ingresa todos los campos requeridos en el formulario. El usuario presiona botón Guardar
Resultado esperado 1: El sistema registra los datos del estudiante. El sistema muestra el nuevo registro en la tabla de lista de estudiantes.
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si ya existe el estudiante.

Fuente: Elaborador por el Autor.

Tabla 3.11: Prueba de aceptación - Modificar Estudiante

Número de prueba: 5
Historia de Usuario: Gestión de estudiantes – Modificar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada: El usuario selecciona el menú Gestión de Usuario. El usuario selecciona pestaña Estudiante. El usuario selecciona un periodo. El usuario selecciona un registro de la tabla de estudiantes. El usuario modifica los campos en el formulario. El usuario presiona botón Modificar.
Resultado esperado 1:

El sistema llena el formulario con los datos del registro seleccionado.

El sistema modifica los datos del estudiante.

El sistema actualiza los datos en la tabla de lista de estudiantes.

Resultado esperado 2:

El sistema muestra mensaje de error si no se ha ingresado campos requeridos.

El sistema muestra mensaje de error si no existe registro seleccionado.

Fuente: Elaborador por el Autor.

Tabla 3.12: Prueba de aceptación - Eliminar Estudiante

Número de prueba: 6

Historia de Usuario: Gestión de estudiantes – Eliminar.

Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.

Entrada:

El usuario selecciona el menú Gestión de Usuario.

El usuario selecciona pestaña Estudiante.

El usuario selecciona un registro de la tabla de estudiantes.

El usuario presiona botón Eliminar.

Resultado esperado 1:

El sistema llena el formulario con los datos del registro seleccionado.

El sistema elimina los datos del estudiante.

El sistema elimina los datos en la tabla de lista de estudiantes.

Resultado esperado 2:

El sistema muestra mensaje de error si no existe registro seleccionado.

El sistema muestra mensaje de error si el estudiante tiene registros de acceso.

Fuente: Elaborador por el Autor.

Tabla 3.13: Prueba de aceptación - Registro de Otros

Número de prueba: 7

Historia de Usuario: Gestión de otros usuarios – Registro.

<p>Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña Pers. Adm.</p> <p>El usuario selecciona el tipo de persona a registrar (Personal Administrativo, Personal de Limpieza, Otros).</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón Guardar.</p>
<p>Resultado esperado 1:</p> <p>El sistema actualiza la tabla de lista de personas de acuerdo al tipo seleccionado.</p> <p>El sistema registra los datos del tipo de persona seleccionado.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de personas de acuerdo al tipo de persona seleccionado.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe la persona.</p>

Fuente: Elaborador por el Autor.

Tabla 3.14: Prueba de aceptación - Modificar Otros

<p>Número de prueba: 8</p>
<p>Historia de Usuario: Gestión de otros usuarios – Modificar.</p>
<p>Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña Pers. Adm.</p> <p>El usuario selecciona el tipo de persona a registrar (Personal Administrativo, Personal de Limpieza, Otros).</p> <p>El usuario selecciona un registro de la tabla de personal.</p>

<p>El usuario modifica los campos en el formulario.</p> <p>El usuario presiona botón Modificar.</p>
<p>Resultado esperado 1:</p> <p>El sistema actualiza la tabla de lista de personas de acuerdo al tipo seleccionado.</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos de la persona.</p> <p>El sistema actualiza los datos en la tabla de lista de personal.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>

Fuente: Elaborador por el Autor.

Tabla 3.15: Prueba de aceptación - Eliminar Otro

Número de prueba: 9
Historia de Usuario: Gestión de otros usuarios – Eliminar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña Profesor.</p> <p>El usuario selecciona el tipo de persona a registrar (Personal Administrativo, Personal de Limpieza, Otros).</p> <p>El usuario selecciona un registro de la tabla de personal.</p> <p>El usuario presiona botón Eliminar.</p>
<p>Resultado esperado 1:</p> <p>El sistema actualiza la tabla de lista de personas de acuerdo al tipo seleccionado.</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos de la persona.</p> <p>El sistema elimina los datos en la tabla de lista de personal.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>

El sistema muestra mensaje de error si el profesor tiene registros de acceso.

Fuente: Elaborador por el Autor.

Tabla 3.16: Prueba de aceptación - Registro de Usuario del Sistema.

Número de prueba: 10
Historia de Usuario: Gestión de usuarios del sistema – Registro.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada: El usuario selecciona el menú Gestión de Usuario. El usuario selecciona pestaña U. Sistema. El usuario ingresa todos los campos requeridos en el formulario. El usuario presiona botón Guardar.
Resultado esperado 1: El sistema registra los datos del usuario. El sistema muestra el nuevo registro en la tabla de lista de usuarios del sistema.
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si ya existe usuario.

Fuente: Elaborador por el Autor.

Tabla 3.17: Prueba de aceptación - Modificar Usuarios del Sistema

Número de prueba: 11
Historia de Usuario: Gestión de usuarios del sistema – Modificar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada: El usuario selecciona el menú Gestión de Usuario. El usuario selecciona pestaña U. Sistema. El usuario selecciona un registro de la tabla de usuarios del sistema. El usuario modifica los campos en el formulario.

El usuario presiona botón Modificar.
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos del profesor.</p> <p>El sistema actualiza los datos en la tabla de lista de usuarios del sistema.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>

Fuente: Elaborador por el Autor.

Tabla 3.18: Prueba de aceptación - Eliminar Usuario del Sistema

Número de prueba: 12
Historia de Usuario: Gestión de usuarios del sistema – Eliminar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Usuario.</p> <p>El usuario selecciona pestaña U. Sistema.</p> <p>El usuario selecciona un registro de la tabla de usuarios del sistema.</p> <p>El usuario presiona botón Eliminar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos del usuario del sistema.</p> <p>El sistema elimina los datos en la tabla de lista de usuarios del sistema.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p> <p>El sistema muestra mensaje de error si el registro a eliminar es el usuario administrador.</p>

Fuente: Elaborador por el Autor.

Tabla 3.19: Prueba de aceptación - Configuración de Tarjeta RFID

Número de prueba: 13
Historia de Usuario: Configuración de Tarjeta RFID.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña. Tarjeta RFID configurada.
Entrada: El usuario selecciona menú Gestión de Carnets. El usuario selecciona pestaña “Crear/Editar”. El usuario selecciona lector RFID. El usuario presiona botón “ON”, se conecta con el lector seleccionado. El usuario presenta una tarjeta RFID nueva al lector RFID.
Resultado esperado 1: El sistema se conecta con el lector RFID. El sistema configura las zonas de la memoria a utilizar de la criptomemoria de la tarjeta RFID.
Resultado esperado 2: El sistema se conecta con el lector RFID. El sistema no puede configurar la tarjeta RFID. El sistema muestra mensaje de error.
Resultado esperado 3: El sistema no se conecta con el lector RFID. El sistema muestra mensaje de error.

Fuente: Elaborador por el Autor.

Tabla 3.20: Prueba de aceptación - Registro de Tarjetas

Número de prueba: 14
Historia de Usuario: Registro de Tarjetas
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada:

<p>El usuario selecciona menú Gestión de Carnets.</p> <p>El usuario selecciona pestaña “Crear/Editar”.</p> <p>El usuario selecciona lector RFID.</p> <p>El usuario presiona botón “ON”, se conecta con el lector seleccionado.</p> <p>El usuario presenta una tarjeta RFID al lector RFID.</p> <p>El usuario presiona botón “>>” y selecciona una persona de la lista desplegada.</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón “Guardar”.</p>
<p>Resultado esperado 1:</p> <p>El sistema se conecta con el lector RFID.</p> <p>El sistema detecta tarjeta RFID configurada.</p> <p>El sistema lista las personas de acuerdo al tipo de persona.</p> <p>El sistema registra los datos de la tarjeta en la BDD y en la criptomemoria de la tarjeta RFID.</p>
<p>Resultado esperado 2:</p> <p>El sistema se conecta con el lector RFID.</p> <p>El sistema detecta tarjeta RFID sin configurar.</p> <p>El sistema muestra mensaje de error.</p>
<p>Resultado esperado 3:</p> <p>El sistema no se conecta con el lector RFID.</p> <p>El sistema muestra mensaje de error.</p>

Fuente: Elaborador por el Autor.

Tabla 3.21: Prueba de aceptación - Asignación de permisos a Tarjeta RFID

Número de prueba: 15
Historia de Usuario: Asignación de permisos a Tarjeta RFID
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada:
<p>El usuario selecciona menú Gestión de Carnets.</p> <p>El usuario selecciona pestaña “Modificar”.</p>

<p>El usuario selecciona una persona de la lista desplegada presionando el botón “>>”.</p> <p>El usuario ingresa todos los datos requeridos en el formulario.</p> <p>El usuario presiona el botón Guardar.</p>
<p>Resultado esperado 1:</p> <p>El sistema muestra la lista de personas por tipo de persona.</p> <p>El sistema registra los datos de los permisos para una tarjeta.</p>
<p>Resultado esperado 2:</p> <p>El sistema despliega mensaje de error si los datos requeridos no se han ingresado o son inválidos.</p>

Fuente: Elaborador por el Autor.

Tabla 3.22: Prueba de aceptación - Registrar Materia.

Número de prueba: 16
Historia de Usuario: Gestión de materias – Registrar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reservas.</p> <p>El usuario selecciona pestaña Adm. Materias.</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón Guardar.</p>
<p>Resultado esperado 1:</p> <p>El sistema registra los datos de la materia.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de materias.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe materia.</p>

Fuente: Elaborador por el Autor.

Tabla 3.23: Prueba de aceptación - Modificar Materia

Número de prueba: 17
Historia de Usuario: Gestión de materias – Modificar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reservas.</p> <p>El usuario selecciona pestaña Adm. Materias.</p> <p>El usuario selecciona un registro de la tabla de materias.</p> <p>El usuario modifica los campos en el formulario.</p> <p>El usuario presiona botón Modificar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos de la materia.</p> <p>El sistema actualiza los datos en la tabla de lista de materias.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe materia.</p>

Fuente: Elaborador por el Autor.

Tabla 3.24: Prueba de aceptación – Eliminar materia

Número de prueba: 18
Historia de Usuario: Gestión de materias – Eliminar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reservas.</p> <p>El usuario selecciona pestaña Adm. Materias.</p> <p>El usuario selecciona un registro de la tabla de materias.</p>

El usuario presiona botón Eliminar.
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos de la materia.</p> <p>El sistema elimina los datos en la tabla de lista de materias.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p> <p>El sistema muestra mensaje de error si la materia tiene reservado un laboratorio.</p>

Fuente: Elaborador por el Autor.

Tabla 3.25: Prueba de aceptación - Registrar reservación de un laboratorio

Número de prueba: 19
Historia de Usuario: Registrar reservación de un laboratorio.
<p>Precondiciones:</p> <p>El usuario debe ingresar al sistema mediante usuario y contraseña.</p> <p>Materia Registrada.</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reservas.</p> <p>El usuario selecciona pestaña Horarios.</p> <p>EL usuario selecciona un Periodo, Laboratorio y Materia.</p> <p>El usuario presiona botón Reservar.</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón Guardar.</p>
<p>Resultado esperado 1:</p> <p>El sistema despliega lista de periodos, laboratorios y materias.</p> <p>El sistema registra los datos de la reserva.</p> <p>El sistema muestra el nuevo registro en la tabla de horarios de reservas.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si ya existe cruce de horarios en la reserva.</p>

Fuente: Elaborador por el Autor.

Tabla 3.26: Prueba de aceptación - Modificar reservación de un laboratorio

Número de prueba: 20
Historia de Usuario: Modificar reservación de un laboratorio.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reservas.</p> <p>El usuario selecciona pestaña Horarios.</p> <p>El usuario selecciona un Periodo y Laboratorio</p> <p>El usuario selecciona un registro de la tabla de horarios.</p> <p>El usuario presiona botón Modificar.</p> <p>El usuario modifica los campos en el formulario.</p> <p>El usuario presiona botón Guardar.</p>
<p>Resultado esperado 1:</p> <p>El sistema despliega lista de periodos y laboratorios.</p> <p>El sistema despliega formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos de la reserva.</p> <p>El sistema actualiza los datos en la tabla de horarios.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si hay cruce de horarios.</p>

Fuente: Elaborador por el Autor.

Tabla 3.27: Prueba de aceptación - Eliminar reservación de un laboratorio

Número de prueba: 21
Historia de Usuario: Eliminar reservación de un laboratorio
Precondiciones: El usuario debe ingresar al sistema mediante usuario y

contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reservas.</p> <p>El usuario selecciona pestaña Horarios.</p> <p>El usuario selecciona un Periodo y Laboratorio</p> <p>El usuario selecciona un registro de la tabla de horarios.</p> <p>El usuario presiona botón Eliminar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos de la reserva.</p> <p>El sistema elimina los datos en la tabla de horarios.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>

Fuente: Elaborador por el Autor.

Tabla 3.28: Prueba de aceptación - Registrar punto de acceso

Número de prueba: 22
Historia de Usuario: Gestión de puntos de acceso – Registrar.
<p>Precondiciones:</p> <p>El usuario debe ingresar al sistema mediante usuario y contraseña.</p> <p>Entrada registrada</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Puertas.</p> <p>El usuario selecciona pestaña “Crear/Editar Entradas”.</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón Guardar.</p>
<p>Resultado esperado 1:</p> <p>El sistema registra los datos del punto de acceso.</p> <p>El sistema muestra el nuevo registro en la tabla de lista de usuarios del sistema.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p>

El sistema muestra mensaje de error si ya existe el punto de acceso.

Fuente: Elaborador por el Autor.

Tabla 3.29: Prueba de aceptación - Modificar punto de acceso

Número de prueba: 23
Historia de Usuario: Gestión de puntos de acceso – Modificar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña. El punto de acceso debe estar registrado en el sistema.
Entrada: El usuario selecciona el menú Gestión de Entradas. El usuario selecciona pestaña “Crear/Editar Entradas”. El usuario selecciona un registro de la tabla de lista de puertas. El usuario modifica los campos en el formulario. El usuario presiona botón Modificar.
Resultado esperado 1: El sistema llena el formulario con los datos del registro seleccionado. El sistema modifica los datos del punto de acceso. El sistema actualiza los datos en la tabla de lista de entradas.
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si no existe registro seleccionado.

Fuente: Elaborador por el Autor.

Tabla 3.30: Prueba de aceptación - Eliminar punto de acceso

Número de prueba: 24
Historia de Usuario: Gestión de puntos de acceso – Eliminar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña. El punto de acceso debe estar registrado en el sistema
Entrada: El usuario selecciona el menú Gestión de Puertas.

<p>El usuario selecciona pestaña “Crear/Editar Entradas”.</p> <p>El usuario selecciona un registro de la tabla de lista de entradas</p> <p>El usuario presiona botón Eliminar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos del punto de acceso.</p> <p>El sistema elimina los datos en la tabla de lista de entradas.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p> <p>El sistema muestra mensaje de error si el punto de acceso tiene registros de accesos.</p>

Fuente: Elaborador por el Autor.

Tabla 3.31: Prueba de aceptación - Asignar punto de acceso

Número de prueba: 25
Historia de Usuario: Asignar punto de acceso
<p>Precondiciones:</p> <p>El usuario debe ingresar al sistema mediante usuario y contraseña.</p> <p>El punto de acceso debe estar registrado en el sistema</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Puertas.</p> <p>El usuario selecciona pestaña “Punto de Acceso”.</p> <p>El usuario selecciona Entrada.</p> <p>El usuario presiona botón Agregar.</p>
<p>Resultado esperado 1:</p> <p>El sistema lista los puntos de acceso disponibles.</p> <p>El sistema guarda en archivo de propiedades la información del punto de acceso seleccionado.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si el punto de acceso ya existe.</p>

Fuente: Elaborador por el Autor.

Tabla 3.32: Prueba de aceptación - Registrar permiso de acceso

Número de prueba: 26
Historia de Usuario: Gestión de permisos de acceso - Registrar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada: El usuario selecciona el menú Gestión de Entradas. El usuario selecciona pestaña Gestión de Permisos. El usuario ingresa todos los campos requeridos en el formulario. El usuario presiona botón Guardar.
Resultado esperado 1: El sistema registra los datos del Permiso. El sistema muestra el nuevo registro en la tabla de lista de permisos.
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si ya existe el permiso de acceso.

Fuente: Elaborador por el Autor.

Tabla 3.33: Prueba de aceptación - Modificar permiso de acceso

Número de prueba: 27
Historia de Usuario: Gestión de permisos de acceso – Modificar.
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.

El permiso de acceso debe estar registrado en el sistema.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Entradas.</p> <p>El usuario selecciona pestaña Gestión de Permisos.</p> <p>El usuario selecciona un registro de la tabla de permisos.</p> <p>El usuario modifica los campos en el formulario.</p> <p>El usuario presiona botón Modificar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema modifica los datos del permiso.</p> <p>El sistema actualiza los datos en la tabla de lista de permisos.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si no existe registro seleccionado.</p>

Fuente: Elaborador por el Autor.

Tabla 3.34: Prueba de aceptación - Eliminar permiso de acceso

Número de prueba: 28
Historia de Usuario: Gestión de permisos de acceso – Eliminar.
<p>Precondiciones:</p> <p>El usuario debe ingresar al sistema mediante usuario y contraseña.</p> <p>El punto de acceso debe estar registrado en el sistema.</p>
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Entradas.</p> <p>El usuario selecciona pestaña Gestión Permisos.</p> <p>El usuario selecciona un registro de la tabla de usuarios del sistema.</p> <p>El usuario presiona botón Eliminar.</p>
<p>Resultado esperado 1:</p> <p>El sistema llena el formulario con los datos del registro seleccionado.</p> <p>El sistema elimina los datos del permiso de acceso.</p> <p>El sistema elimina los datos en la tabla de lista de permisos de acceso.</p>

Resultado esperado 2:

El sistema muestra mensaje de error si no existe registro seleccionado.

Fuente: Elaborador por el Autor.

Tabla 3.35: Prueba de aceptación - Registrar acceso

Número de prueba: 29
Historia de Usuario: Registrar acceso
<p>Precondiciones:</p> <p>El usuario debe ingresar al sistema mediante usuario y contraseña. Punto de acceso registrado. Tarjeta RFID registrada.</p>
<p>Entrada:</p> <p>El usuario presenta tarjeta RFID en punto de acceso.</p>
<p>Resultado esperado 1:</p> <p>El sistema lee tarjeta RFID. El sistema valida tarjeta RFID. El sistema valida permiso de acceso a la tarjeta RFID. El sistema registra acceso.</p>
<p>Resultado esperado 2:</p> <p>El sistema lee tarjeta RFID. El sistema valida tarjeta RFID. El sistema deniega acceso a tarjeta RFID. El sistema registra acceso.</p>
<p>Resultado esperado 3:</p> <p>El sistema no puede leer tarjeta RFID. El sistema deniega acceso a tarjeta RFID.</p>
<p>Observaciones</p> <p>En estas pruebas se identificó que el tiempo de apertura de la puerta (Microcontrolador) fue grande (2segundos), lo que provocaba un mayor tiempo de espera para leer otra tarjeta (4segundos).</p>

Solución

En el microcontrolador se reajusto el tiempo de espera a medio segundo para la apertura, obteniendo así una espera de dos segundos para la lectura de la otra tarjeta.

Fuente: Elaborador por el Autor.

Tabla 3.36: Prueba de aceptación - Elaboración de circuito con leds

Número de prueba: 30
Historia de Usuario: Elaboración de circuito con leds
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña. El sistema establece comunicación serial con el circuito.
Entrada: El sistema ejecuta evento de permiso de acceso concedido. El sistema ejecuta evento de permiso de acceso denegado. El sistema ejecuta evento de actualización de horarios de laboratorios.
Resultado esperado 1: El circuito recibe comando de permiso de acceso concedido, abre cerradura eléctrica y enciende led verde. El circuito recibe comando de permiso de acceso denegado, enciende led rojo. El circuito recibe comando para actualizar leds de estado de uso de laboratorios, cambia estado de los leds.

Fuente: Elaborador por el Autor.

Tabla 3.37: Prueba de aceptación - Imprimir mensaje en pantalla LCD

Número de prueba: 31
Historia de Usuario: Imprimir mensaje en pantalla LCD
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.

El sistema establece comunicación serial con el circuito.
Entrada: El sistema ejecuta evento de envío de mensaje al circuito.
Resultado esperado 1: El circuito imprime mensaje en pantalla LCD.

Fuente: Elaborador por el Autor.

Tabla 3.38: Prueba de aceptación - Generación de reportes Totales

Número de prueba: 32
Historia de Usuario: Generación de reportes Totales
Precondiciones: El usuario debe ingresar al sistema mediante usuario y contraseña.
Entrada: El usuario selecciona el menú Gestión de Reportes. El usuario selecciona pestaña Reportes. El usuario ingresa todos los campos requeridos en el formulario. El usuario presiona botón "Ver Reporte".
Resultado esperado 1: El sistema genera reporte. El sistema muestra el reporte.
Resultado esperado 2: El sistema muestra mensaje de error si no se ha ingresado campos requeridos. El sistema muestra mensaje de error si no existen datos para generar el reporte.

Fuente: Elaborador por el Autor.

Tabla 3.39: Prueba de aceptación - Generación de reportes Detallados

Número de prueba: 33
Historia de Usuario: Generación de reportes Detallados
Precondiciones:

El usuario debe ingresar al sistema mediante usuario y contraseña.
<p>Entrada:</p> <p>El usuario selecciona el menú Gestión de Reportes.</p> <p>El usuario selecciona pestaña Reportes.</p> <p>El usuario ingresa todos los campos requeridos en el formulario.</p> <p>El usuario presiona botón “Ver Reporte”.</p>
<p>Resultado esperado 1:</p> <p>El sistema genera reporte.</p> <p>El sistema muestra el reporte.</p>
<p>Resultado esperado 2:</p> <p>El sistema muestra mensaje de error si no se ha ingresado campos requeridos.</p> <p>El sistema muestra mensaje de error si no existen datos para generar el reporte.</p>

Fuente: Elaborador por el Autor.

Resultados de las Pruebas de Aceptación

A continuación se resume los resultados de las pruebas de aceptación generadas por cada historia de usuario.

Tabla 3.40: Resultados de pruebas de aceptación

Resultado de Pruebas de Aceptación		
Historia de Usuario	Denegada	Aprobada
Gestión de profesores – Registro.		✓
Gestión de profesores – Modificar.		✓
Gestión de profesores – Eliminar.		✓
Gestión de estudiantes – Registro.		✓
Gestión de estudiantes – Modificar.		✓
Gestión de estudiantes – Eliminar.		✓
Gestión de otros usuarios – Registro.		✓
Gestión de otros usuarios – Modificar.		✓
Gestión de otros usuarios – Eliminar.		✓
Gestión de usuarios del sistema – Registro.		✓

Gestión de usuarios del sistema – Modificar.	✓
Gestión de usuarios del sistema – Eliminar.	✓
Configuración de Tarjeta RFID.	✓
Registro de Tarjetas.	✓
Asignación de permisos a Tarjeta RFID.	✓
Gestión de materias - Registrar.	✓
Gestión de materias – Modificar.	✓
Gestión de materias – Eliminar.	✓
Registrar reservación de un laboratorio.	✓
Modificar reservación de un laboratorio.	✓
Eliminar reservación de un laboratorio.	✓
Gestión de puntos de acceso - Registrar.	✓
Gestión de puntos de acceso -Modificar.	✓
Gestión de puntos de acceso -Eliminar.	✓
Asignar punto de acceso	✓
Gestión de permisos de acceso – Registrar.	✓
Gestión de permisos de acceso – Modificar.	✓
Gestión de permisos de acceso – Eliminar.	✓
Registrar acceso.	✓
Elaboración de circuito con leds.	✓
Imprimir mensaje en pantalla LCD.	✓
Generación de reportes Totales.	✓
Generación de reportes Detallados.	✓

Fuente: Elaborador por el Autor.

3.5. ANÁLISIS DE RESULTADOS

Instalación del Sistema de Control de acceso

Como primer paso para la instalación del Sistema de Control de Acceso se

necesita configurar el Circuito de Control de Acceso con la chapa eléctrica, para ello es necesario que una persona con conocimientos en electricidad lleve a cabo el proceso, que se describe en el Manual de Instalación y Configuración del Circuito de Control de Acceso adjunto en el Anexo D.

Una vez realizada la configuración del circuito, para proceder con la instalación del Sistema de Control de Acceso se recomienda que sea una persona que tenga conocimientos en informática relacionados a configuración de servidores de base de datos e instalación de programas informáticos; la instalación del Sistema de Control de Acceso se detalla en el Manual de Instalación y Configuración del Circuito de Control de Acceso adjunto en el Anexo D.

Implementación de Sistema de Control de Acceso

Para implementar el Sistema de Control de Acceso en un punto de acceso se necesita de un Circuito de Control, chapa eléctrica, tarjeta RFID y un computador, y por cada punto de acceso adicional se necesitaría de un Circuito de Control y una chapa eléctrica.

Aceptación de los usuarios

En una puerta modelo se configuró e instaló el Sistema de Control de Acceso, misma en la que se realizó una demostración del sistema al Ing. Patricio Proaño y del cual se recibió la aceptación de la funcionalidad del sistema informático. Adicionalmente se probó el sistema con estudiantes de la facultad mismos que supieron manifestar su aceptación con el Sistema de Control de Acceso.

Análisis de aplicativo

El módulo de gestión de usuarios, el LDICC tendrá almacenada y organizada la información de todos sus usuarios como profesores, estudiantes, personal administrativo y otros.

Mediante el módulo de reservación de laboratorios se ha logrado que la reservación sea dinámica y además permite informar a los usuarios del LDICC la disponibilidad de las aulas en tiempo real.

Con el módulo de gestión de entradas se puede escalar el número de puntos de

acceso que se necesita controlar, por ejemplo, adicionalmente al punto de acceso principal del LDICC, se lo puede realizar el control de acceso en cada una de las puertas de las salas de computación o implementar en los puntos de acceso del segundo piso, en donde solo profesores y personal administrativo tienen acceso.

El módulo de Control de Acceso permite que en cada punto de acceso se lleve a cabo un control de acuerdo a los permisos de acceso que sean asignados, logrando así en áreas exclusivas la entrada a solo el personal autorizado.

Con el módulo de reportes permite al Jefe del LDICC disponer de forma inmediata la información de los accesos de los usuarios en un intervalo de fechas requerido, permitiendo así tomar decisiones de manera oportuna, por ejemplo en caso de alguna pérdida dentro del laboratorio del LDICC se puede saber que personas fueron las que estuvieron presentes en ese momento.

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- EL sistema diseñado para el Control de Acceso mejora la seguridad en las instalaciones del LDICC, puesto que la estructura de la aplicación permite hacer un seguimiento de uso de las instalaciones del laboratorio por parte de usuarios autorizados. El sistema incluye datos de identificación de usuarios para acceso y guarda información histórica de uso del laboratorio los cuales pueden ser vistos por el administrador del sistema mediante el módulo de reportes.
- El sistema producto de este trabajo hace uso de tarjetas RFID, las cuales almacenan datos del usuario de manera segura y que son usados en el proceso de validación para el acceso al LDICC. Los datos almacenados en una tarjeta inteligente están protegidos por sofisticados mecanismos de seguridad mediante el uso de cálculos avanzados de criptografía.
- El desarrollo de la aplicación empleando el patrón de diseño MVC, ha logrado tener un código organizado, permitiendo así detectar o corregir un error de manera rápida, a su vez permite la optimización de tiempo en la etapa de mantenimiento.
- El primer prototipo de la capa de modelo fue realizado empleando JDBC mediante conexiones nativas, el cual supuso el empleo de una cantidad alta de tiempo y esfuerzo en la etapa de desarrollo, sin embargo en un segundo prototipo se utilizó JPA verificando que el uso esta API disminuye el tiempo y esfuerzo en el desarrollo con respecto al uso de JDBC. Además JPA permite autogenerar código fuente de la capa de persistencia, siendo esta otra importante y útil característica de esta API.

4.2. RECOMENDACIONES

- La aplicación desarrollada en este trabajo posee arquitectura cliente/servidor. Sería de utilidad que en un trabajo futuros se realice la migración del aplicativo a una plataforma que permita acceso web, esto facilitaría el uso del sistema a los usuarios del sistema.
- La fase de mantenimiento es parte del ciclo de vida del software, para facilitar y optimizar recursos en las tareas de esta fase es necesario el uso de un patrón de diseño como MVC, puesto que esto permite producir código en forma organizada. De la misma forma es de utilidad que la implementación de la capa de persistencia se lo realice con JPA porque permite optimizar el tiempo de desarrollo, auto generando código fuente para esta capa.
- La utilización de tarjetas con criptomemorias en sistemas de control de acceso brinda mayor seguridad en la protección de los datos de identificación y evita que estos sean clonados o capturados sin la debida autorización. Es recomendable verificar y analizar las propiedades y limitaciones del modelo de criptomemoria a emplear y tomar en cuenta estos parámetros para el análisis del sistema de control.
- Las especificaciones de cada historia de usuario debería ser realizada definiendo grupos según niveles de complejidad, lo cual permitirá asignar un periodo máximo variante según el proyecto pero preferentemente de una semana para su implementación, si este tiempo es superado, es recomendable dividir la complejidad de la historia de usuario. Atacar un problema grande dividiéndolo en pequeñas partes es siempre una buena estrategia.

- Una metodología de desarrollo permite crear software de calidad y disminuir riesgos en el proyecto. Cada una se ajusta a necesidades específicas, así que es recomendable analizar las características y naturaleza del sistema a desarrollar antes de decidir cual utilizar a la hora de enfrentar un proyecto de desarrollo.
- En la fase de desarrollo, en donde se implementa el código del software es recomendable seguir los estándares de programación y sobre todo las mejores prácticas, ya que se logra que el código sea legible para cualquier programador.

BIBLIOGRAFÍA

- [1] PROMATCO, «Soluciones en seguridad y accesos,» 2005. [En línea]. <http://www.promatco.com.ec/acc.html>. [Último acceso: 2013 07 10].
- [2] E. Menchán, «Sistema de control de acceso,» 30 04 2007. [En línea]. <http://dspace.ups.edu.ec/bitstream/123456789/2290/15/UPS-GT000025.pdf>. [Último acceso: 2013 08 16].
- [3] J. L. Transcore, «Shrouds of Time The history of RFID,» 01 10 2001. [En línea]. <https://www.transcore.com/sites/default/files/History%20of%20RFID%20White%20Paper.pdf>. [Último acceso: 13 08 2014].
- [4] LIBERA NETWORKS, Emergence of standards., Madrid, 2010.
- [5] «Revista Consumer,» 01 Marzo 2007. [En línea]. <http://revista.consumer.es/web/es/20070301/pdf/internet.pdf>.
- [6] Kifer, «Introducción a los Sistemas RFID,» [En línea]. <http://www.kifer.es/Recursos/Pdf/RFID.pdf>. [Último acceso: 26 10 2013].
- [7] «Análisis comparativo de las tecnologías RFID, HID y AWID,» 01 2009. [En línea]. <http://bibdigital.epn.edu.ec/bitstream/15000/1085/1/CD-1923.pdf>. [Último acceso: 14 01 2014].
- [8] A. Urueña, A. Cimorra y F. Cañas, «La tecnología RFID: Usos y oportunidades,» 2009. [En línea]. <http://www.jesusamieiro.com/wp-content/uploads/2011/06/La-tecnolog%C3%ADa-RFID.-Usos-y-oportunidades.pdf>.
- [9] «NewTech Solutions Group,» [En línea]. <http://www.newtech.net/productos-y-servicios/tarjetas-inteligentes/>.
- [10] J. Highsmith, «Agile Software Development Ecosystems,» Addison-Wesley, 2002.
- [11] Universidad de Murcia, «Metodologías de desarrollo de software,» [En línea]. <http://www.um.es/docencia/barzana/IAGP/lagp2.html>. [Último acceso:

- 16 03 2014].
- [12] P. Letelier, «Métodologías ágiles para el desarrollo de software,» [En línea]. http://www.cyta.com.ar/ta0502/b_v5n2a1.htm. [Último acceso: 13 04 2013].
- [13] M. Calero Solís, «Una explicación de la programación XP,» [En línea]. <http://www.willydev.net/descargas/prev/ExplicaXP.pdf>. [Último acceso: 12 05 2013].
- [14] PMO Informática, «Los 5 valores de la programación extrema (XP),» 26 11 2012. [En línea]. <http://www.pmoinformatica.com/2012/11/los-5-valores-de-la-programacion.html>. [Último acceso: 10 09 2013].
- [15] S. Priolo, Metodos Agiles, Creative Andina Corp, 2009.
- [16] Oracle, «Java Platform Standard Edition,» [En línea]. <http://docs.oracle.com/javase/8/docs/>. [Último acceso: 16 09 2013].
- [17] E. Bascón Pantoja, «El patrón de diseño MVC,» 10 2004. [En línea]. <http://www.ucbcba.edu.bo/Publicaciones/revistas/actanova/documentos/v2n4/v2.n4.bascon.pdf>. [Último acceso: 18 09 2013].
- [18] «NetBeans IDE,» [En línea]. <https://netbeans.org/features/index.html>. [Último acceso: 19 10 2013].
- [19] «MySQL,» [En línea]. Available: <http://misclassblog.com/database-design-and-development/mysql/>. [Último acceso: 16 10 2013].
- [20] «UltraMAGiCard,» [En línea].: <http://www.ultramagicard.es/tecnologia/banda-magnetica/>.
- [21] «DATABAC,» [En línea]. http://www.databac.es/productos/tarjetas/tecnologias/wiegand/tarjetas_wiegand.php.
- [22] «akrocard,» [En línea]. Available: <http://www.akrocard.com/aplicaciones.php>.
- [23] «SEGURIDAD,» [En línea]. <http://www.seguridad.com/detalles.asp?id=3413&idsubfamilia=218>.
- [24] [En línea]. http://1.bp.blogspot.com/--riwR7m3E9c/T8TR1e7_y4I/AAAAAAAAAGg/bzRy1x2WANc/s1600/Burn_down_chart.png.

- [25] J. A. A. Sánchez, Sistema de Control de Acceso con RFID.
- [26] J. Highsmith, «Using Adaptive Software Development to meet the challenges of a high-speed,» 2000.
- [27] A. Cockburn, «Crystal Clear,» 2004.
- [28] J. Gonza, «JGC Estudio,» [En línea].
<http://jgcprogramacion.blogspot.com/search/label/METODOLOGIA>.
- [29] A. CORREA ESPINAL y C. E. ÁLVAREZ L, «ICESI,» [En línea].
http://elearning2.icesi.edu.co/revistas/index.php/estudios_gerenciales/article/view/372/html.
- [30] L. Cosentino, «RNDS,» [En línea].
http://www.rnds.com.ar/articulos/046/RNDS_156W.pdf.
- [31] «La tecnología RFID: usos y oportunidades,» [En línea].
www.ametic.es/DescargarDocumento.aspx?idd=2885. [Último acceso: 13 02 2014].
- [32] TRIPOD, «Fases de la Programación Extrema,» [En línea].
<http://programacionextrema.tripod.com/fases.htm>. [Último acceso: 15 09 2013].

GLOSARIO

CRIPTOMEMORIA

Es una memoria tipo EEPROM con diferentes niveles de seguridad como: zonas de memoria con libre acceso, zonas de memoria protegidas mediante contraseña, zonas de memoria con encriptación y códigos de autenticación.

EEPROM

Por sus siglas en inglés, Electrically Erasable Programmable Read-Only Memory (Memoria Programable Borrable de Solo Lectura), es un tipo de memoria no volátil que puede ser programada, borrada y reprogramada eléctricamente.

WORM

Por sus siglas en inglés Write Once Read Many es una memoria que permite una única escritura, pero varias lecturas. Los datos una vez escritos ya no pueden ser sobre escritos.

SCA

Sistema de Control de Acceso es una medida de seguridad física, cuyo objetivo es proteger recursos específicos de entes no autorizados.

LDICC

El Laboratorio del Departamento de Informática y Ciencias de la Computación está encargado de la administración y gestión de las salas de computación y los servidores de la Facultad de Sistemas de la EPN.

RFID

Por sus siglas en inglés Radio Frequency IDentification hace referencia a la tecnología que permite saber la identidad de una etiqueta de forma remota empleando un lector que establece una comunicación por ondas de radio.

IFF

Por sus siglas en inglés Identification, Friend or Foe es un sistema de identificación por radio frecuencia inventado por León Theremin en la Segunda Guerra Mundial para distinguir aviones amigos o enemigos.

TRANCEPTOR

Es un dispositivo electrónico que permite al mismo tiempo el envío y recepción de información, empleando elementos comunes del circuito en ambas funciones.

XP

Extrem Programing o Programación Extrema hace referencia a la metodología de desarrollo ágil de software. Se basa principalmente en la simplicidad y agilidad. Promueve trabajo en equipo, la capacitación de los desarrolladores y se centra en fomentar las relaciones interpersonales como parte fundamental para el éxito del proyecto.

Tarjeta RFID

Es un dispositivo en forma de carnet que contiene un chip en donde se almacena los datos de identificación de una persona y genera una señal de radiofrecuencia con dichos datos.

Lector RFID

Es el dispositivo que se emplea para establecer una comunicación por radio frecuencia con una tarjeta RFID para realizar operaciones de lectura o escritura.

Entity

Es una clase que se utiliza para mapear dicha clase con una tabla de una base de datos, se emplea generalmente en persistencia.

DAO

Por sus siglas en inglés Data Acces Object es un Objeto de Acceso a Datos que suministra una interfaz común para la manipulación de datos almacenados en una base de datos.

JPA

El API de persistencia de java, es un framework de la plataforma JEE y provee un mapeo objeto relacional mediante POJOs para interactuar con una base de datos.

JPQL

Por sus siglas en inglés Java Persistence Query Language es un lenguaje de consulta que manipula objetos tipo Entities en lugar de las tablas de la base de datos.

CRC

Hace referencia a las tarjetas CRC (Clase Responsabilidad y Colaboración) que se generan en la metodología XP en la fase de diseño.

MVC

Es el patrón de diseño que separa el modelo, la vista y el controlador, que se emplea en el desarrollo de software.

IDE

Por sus siglas en inglés Integrated Development Environment es el entorno de desarrollo integrado que contiene un conjunto de herramientas para la creación de software.

ANEXOS

ANEXO A: Especificación Técnica de Tarjeta RFID.

CD-ROM: \TESIS\ANEXOS\Anexo A - Especificación Técnica de Tarjeta RFID.

ANEXO B: Cronograma de Implementación de Historias de Usuario (Anexo Digital).

CD-ROM: \TESIS\ANEXOS\Anexo B - Cronograma de Implementación de Historias de Usuario.

ANEXO C: Tarjetas CRC del Sistema.

CD-ROM: \TESIS\ANEXOS\Anexo C - Tarjetas CRC del Sistema.

ANEXO D: Manual de Instalación y Configuración.

CD-ROM: \TESIS\ANEXOS\Anexo D - Manual de Instalación y Configuración.

ANEXO E: Javadoc – Manual de Programador.

CD-ROM: \TESIS\ANEXOS\Anexo E - Javadoc – Manual de Programador.