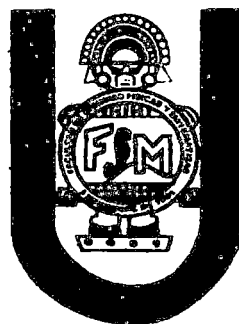




**UNIVERSIDAD NACIONAL  
"PEDRO RUIZ GALLO"**

**FACULTAD DE CIENCIAS FÍSICAS  
Y MATEMÁTICAS**



**ESCUELA PROFESIONAL DE INGENIERÍA EN  
COMPUTACIÓN E INFORMÁTICA**

**"PORTAFOLIO ELECTRÓNICO PARA FACILITAR LA  
INFORMACIÓN DE LA FORMACIÓN ACADÉMICA DE LOS  
ESTUDIANTES DE LA ESCUELA PROFESIONAL DE INGENIERÍA  
EN COMPUTACIÓN E INFORMÁTICA DE LA UNIVERSIDAD  
NACIONAL PEDRO RUIZ GALLO" - LAMBAYEQUE 2014"**

# **TESIS**

**PARA OPTAR EL TÍTULO PROFESIONAL DE  
INGENIERO EN COMPUTACIÓN E INFORMÁTICA**

**PRESENTADO POR:**

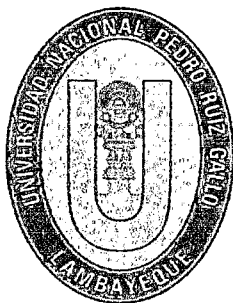
**Bach. SEVERINO SAENZ LUIS JONATAN**

**Bach. REQUEJO DELGADO VICTOR IVAN**

**LAMBAYEQUE - PERÚ**

**2014**

**UNIVERSIDAD NACIONAL**  
**“PEDRO RUIZ GALLO”**  
**FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS**



**“PORTAFOLIO ELECTRÓNICO PARA FACILITAR LA INFORMACIÓN  
DE LA FORMACIÓN ACADÉMICA DE LOS ESTUDIANTES DE LA  
ESCUELA PROFESIONAL DE INGENIERÍA EN COMPUTACIÓN E  
INFORMÁTICA DE LA UNIVERSIDAD NACIONAL PEDRO RUIZ  
GALLO – LAMBAYEQUE 2014”**

**TESIS**

**PARA OPTAR EL TÍTULO DE:**  
**INGENIERO EN COMPUTACIÓN E INFORMÁTICA**

**AUTORES:**

**SEVERINO SAENZ LUIS JONATAN**  
**REQUEJO DELGADO VICTOR IVAN**

**LAMBAYEQUE - PERÚ**

**2014**

Aprobado por:



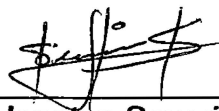
Ing. Oscar Alex Serquén Yparraguirre  
Presidente



Ing. Denny Fuentes Adrianzén  
Secretario



Ing. Franklin E. Terán Santa Cruz  
vocal



Bach. Luis Jonatán Severino Sáenz  
Tesisista



Bach. Víctor Ivan Requejo Delgado  
Tesisista



Ing. Milton German Reyes  
Asesor

## **DEDICATORIA**

A Dios Todopoderoso, por darme la vida y la oportunidad de culminar satisfactoriamente esta carrera y de hacer realidad este anhelado sueño. A mis padres por su constante apoyo aún en los momentos más difíciles. A mis entrañables y queridos compañeros, que me apoyaron y compartieron durante estos cinco años, los mejores momentos de mi formación profesional.

*Luis Jonatan Severino Sáenz*

Con todo mi cariño para mis padres, en especial a mi madre, pilar fundamental en mi vida. A mis hermanos y compañeros que estuvieron siempre conmigo a pesar de las adversidades que nos presenta la vida, porque con su apoyo y aliento hoy he logrado uno de mis más grandes anhelos., a ustedes por siempre mi corazón y mi agradecimiento.

*Victor Iván Requejo Delgado*

## **AGRADECIMIENTO**

Son muchas las personas que han formado parte de mi vida y a las que, en esta ocasión, debo agradecer su amistad, consejos, apoyo, ánimo, comprensión y compañía.

Agradezco también, a mis profesores de la Escuela de Ingeniería en Computación e Informática, que en estos años han aportado su tiempo y dedicación en mi formación como profesional.

*Luis Jonatan Severino Sáenz*

En primer lugar quiero agradecer a Dios por haberme dado la sabiduría, el entendimiento y la fortaleza para poder llegar al final de mi carrera.

Gracias a mi familia, docentes y compañeros por ayudarme cada día a cruzar con firmeza el camino de la superación, por enseñarme la grandeza de la vida y exigirme ser mejor cada día.

*Victor Iván Requejo Delgado*

## **RESUMEN**

El presente trabajo de investigación tiene como objetivo la implementación de un portafolio electrónico para la Escuela Profesional de Ingeniería en Computación e Informática de la Universidad Nacional Pedro Ruiz Gallo, el cual tiene como propósito mejorar la compartición de información entre los docentes y estudiantes. El portafolio electrónico es un conjunto de recursos esenciales para incursionar en los entornos virtuales de educación.

Los pasos metodológicos estuvieron regidos por la metodología programación extrema (XP), concluyendo con una aplicación basada en la necesidad existente de dar solución al inconveniente que se presenta por no contar con una herramienta necesaria para tener la facilidad de compartir los aportes de enseñanza y visualizarlos en cualquier momento.

El proyecto estuvo estructurado en 4 fases, enmarcado como una investigación tecnológica formal. Adicionalmente, para el desarrollo de la aplicación se utilizaron diversas tecnologías como el lenguaje de programación PHP, el sistema de base de datos MySQL, y el servidor web.

De esta manera se pudo concluir que con la implementación del portafolio electrónico se genera un mejor desempeño tanto de los docentes como de los estudiantes en cuanto a la compartición de información.

## **ABSTRACT**

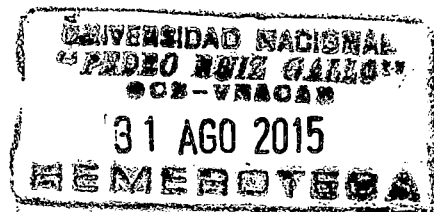
The following research has as objective the implementing an electronic portfolio for vocational school computer engineering and computer science of the Pedro Ruiz Gallo University. This is to improve information sharing between teachers and students.

The methodological steps were governed by the methodology Extreme Programming (XP), concluding with an application based on the existing need for solution to the inconvenience that results from not having a necessary tool to have the ease of sharing educational inputs and display them anytime.

The project was divided into 4 phases, framed as a formal technological investigation. Additionally, for the application development used various technologies like PHP programming language, the system database MySQL and web server.

This way, we can be concluded that with the implementation of electronic portfolio a better performance of teachers and students is generated regarding the sharing of information.

## INDICE GENERAL



DEDICATORIA	2
AGRADECIMIENTO	3
RESUMEN	5
ABSTRACT	6
I. DATOS GENERALES DE LA ORGANIZACIÓN	15
1.1. DESCRIPCIÓN DE LA ORGANIZACIÓN	15
1.2. MISIÓN, VISIÓN Y OBJETIVOS DE LA ORGANIZACIÓN	15
1.2.1. MISIÓN	15
1.2.2. VISIÓN	15
1.3. ORGANIGRAMA DE LA ORGANIZACIÓN	16
II. PROBLEMÁTICA DE LA INVESTIGACIÓN	18
2.1. REALIDAD PROBLEMÁTICA	18
2.1.1. PLANTEAMIENTO DEL PROBLEMA	18
2.2. FORMULACION DEL PROBLEMA	19
2.3. JUSTIFICACIÓN E IMPORTANCIA DE LA INVESTIGACIÓN	19
2.3.1. Justificación Teórica	19
2.3.2. Justificación Práctica	19
2.3.3. Importancia	20
2.4. OBJETIVOS DE LA INVESTIGACIÓN	20
2.4.1. OBJETIVO GENERAL	20
2.4.2. OBJETIVOS ESPECÍFICOS	20
III. MARCO METODOLÓGICO	22
3.1. TIPO DE INVESTIGACIÓN	22
3.2. HIPÓTESIS	22
3.3. VARIABLES	22
3.3.1. VARIABLE INDEPENDIENTE	22
3.3.2. VARIABLE DEPENDIENTE	22
IV. MARCO TEÓRICO	24
4.1. ANTECEDENTES DE LA INVESTIGACIÓN	24
4.1.1. ANTECEDENTE N°1	24



4.1.2.	ANTECEDENTE N°2	26
4.1.3.	ANTECEDENTE N°3	27
4.2.	BASE TEÓRICA	28
4.2.1.	EXTREME PROGRAMMING (XP)	28
4.2.1.1.	Objetivos de XP	29
4.2.1.2.	Los Principios de XP	30
4.2.1.3.	Ventajas y desventajas	31
4.2.1.4.	Ciclo de vida de XP	31
4.2.2.	PORTAFOLIO ELECTRÓNICO	33
4.2.2.1.	El portafolio:	33
4.2.2.2.	El portafolio electrónico (e-P)	34
4.2.3.	LENGUAJES DE PROGRAMACIÓN	36
4.2.3.1.	LARAVEL 4.2	36
4.2.3.2.	BOOSTRAP CSS FRAMEWORK	42
4.2.3.3.	HTML5	45
4.2.4.	BASE DE DATOS	46
4.2.4.1.	VENTAJAS DE LAS BASES DE DATOS	47
4.2.4.2.	SISTEMA DE GESTIÓN DE BASE DE DATOS (SGBD)	47
4.2.4.3.	MYSQL	48
4.2.5.	HERRAMIENTAS	49
4.2.5.1.	WORKBENCH	49
V.	DESARROLLO DE LA PROPUESTA	52
5.1.	ARQUITECTURA DE SOLUCION	52
5.1.1.	REQUISITOS BÁSICOS DE LOS DISPOSITIVOS	53
5.1.1.1.	SERVIDOR WEB	53
5.1.1.2.	PCS:	54
5.1.1.3.	CELULAR:	54
5.1.1.4.	TABLET:	54
5.2.	PLANIFICACIÓN	55
5.2.1.	ITERACIÓN 1	55
5.2.2.	ITERACIÓN 2	63
5.2.3.	ITERACIÓN 3	75
5.2.4.	ITERACIÓN 4	84

5.3.	DISEÑO-----	88
5.3.1.	DISEÑO ITERACIÓN 1-----	88
5.3.2.	DISEÑO ITERACIÓN 2-----	90
5.3.3.	DISEÑO ITERACIÓN 3-----	95
5.3.4.	DISEÑO ITERACIÓN 4-----	99
5.4.	BASE DE DATOS COMPLETA-----	100
VI.	COSTOS Y BENEFICIOS-----	102
6.1.	CRONOGRAMA DE ACTIVIDADES DEL PROYECTO-----	102
6.2.	ANÁLISIS COSTO / BENEFICIO-----	103
6.2.1.	ANÁLISIS DE COSTOS-----	103
6.2.1.1.	RESUMEN DE COSTO DE INVERSION -----	104
6.2.1.2.	COSTO DE OPERACIÓN-----	105
6.2.2.	ANÁLISIS DE BENEFICIOS -----	105
6.2.2.1.	Beneficios Tangibles-----	105
6.2.2.2.	Beneficios Intangibles -----	106
6.2.3.	Cálculo de la recuperación de la inversión -----	107
	CONCLUSIONES -----	109
	RECOMENDACIONES -----	111
	REFERENCIAS BIBLIOGRAFICAS -----	113

## INDICE DE TABLAS

Tabla N° 1: Historia de Usuario N° 1 .....	55
Tabla N° 2: Tarea Diseñar las entidades que abarquen el manejo de usuarios en la base de datos.....	56
Tabla N° 3: Tarea Implementar ventana para iniciar sesión.....	56
Tabla N° 4: Tarea Pruebas en ventana para iniciar sesión .....	57
Tabla N° 5: Historia de Usuario N° 2.....	57
Tabla N° 6: Tarea Implementar ventana para registrar docente .....	58
Tabla N° 7: Tarea Pruebas en ventana para registrar docente .....	58
Tabla N° 8: Historia de Usuario N° 3.....	59
Tabla N° 9: Tarea Implementar ventana para registrar estudiante .....	59
Tabla N° 10: Tarea Pruebas en ventana para registrar estudiante.....	60
Tabla N° 11: Historia de Usuario N° 4.....	61
Tabla N° 12: Tarea Implementar ventana para registrar curso .....	62
Tabla N° 13: Tarea Pruebas en ventana para registrar curso .....	62
Tabla N° 14: Historia de Usuario N° 5.....	63
Tabla N° 15: Tarea Implementar ventana para listar usuarios .....	64
Tabla N° 16: Tarea Implementar ventana para editar usuario.....	64
Tabla N° 17: Tarea Pruebas en ventana para editar usuario .....	65
Tabla N° 18: Historia de Usuario N° 6.....	65
Tabla N° 19: Tarea Implementar ventana para listar cursos .....	66
Tabla N° 20: Tarea Implementar ventana para editar curso.....	66
Tabla N° 21: Tarea Pruebas en ventana para editar curso .....	67
Tabla N° 22: Historia de Usuario N° 7.....	67
Tabla N° 23: Tarea Implementar ventana para editar perfil de usuario .....	68
Tabla N° 24: Tarea Pruebas en ventana para editar perfil de usuario .....	68
Tabla N° 25: Historia de Usuario N° 8.....	69
Tabla N° 26: Tarea Implementar ventana para listar cursos .....	70
Tabla N° 27: Tarea Implementar ventana para inscripción a curso .....	70
Tabla N° 28: Tarea Implementar ventana para listar estudiantes matriculados .....	71
Tabla N° 29: Tarea Pruebas en ventana para inscripción a curso .....	71

Tabla N° 30: Historia de Usuario N° 9.....	72
Tabla N° 31: Tarea Implementar ventana para listar cursos de docente.....	73
Tabla N° 32: Tarea Implementar ventana para matricula de estudiantes.....	74
Tabla N° 33: Tarea Pruebas en ventana para matricula de estudiantes .....	74
Tabla N° 34: Historia de Usuario N° 10.....	75
Tabla N° 35: Tarea Implementar ventana para registrar actividades.....	76
Tabla N° 36: Tarea Pruebas en ventana para registrar actividades .....	76
Tabla N° 37: Historia de Usuario N° 11.....	77
Tabla N° 38: Tarea Implementar ventana para listar actividades de estudiante .....	78
Tabla N° 39: Tarea Implementar ventana para desarrollar actividad.....	78
Tabla N° 40: Tarea Implementar ventana para enviar solución .....	79
Tabla N° 41: Tarea Pruebas en ventana para enviar solución .....	79
Tabla N° 42: Historia de Usuario N° 12.....	80
Tabla N° 43: Tarea Implementar ventana para listar actividades de curso ..	81
Tabla N° 44: Tarea Implementar ventana para mantenimiento de actividades .....	81
Tabla N° 45: Tarea Implementar ventana para listar estudiantes que enviaron actividad .....	82
Tabla N° 46: Tarea Implementar ventana para calificar actividad.....	82
Tabla N° 47: Tarea Pruebas en ventana para calificar actividad .....	83
Tabla N° 48: Historia de Usuario N° 13.....	84
Tabla N° 49: Tarea Implementar ventana para listar actividades enviadas y pendientes .....	85
Tabla N° 50: Historia de Usuario N° 14.....	86
Tabla N° 51: Tarea Implementar ventana para listar aprobados por curso y actividad .....	87

## INDICE DE FIGURAS

Figura N° 1: Organigrama de la Facultad de Ciencias Físicas y Matemáticas de la Universidad Nacional Pedro Ruiz Gallo.....	16
Figura N° 2: Arquitectura de solución.....	52
Figura N° 3: Ventana Iniciar Sesión .....	88
Figura N° 4: Ventana Registrar Docente.....	88
Figura N° 5: Ventana Registrar Estudiante.....	89
Figura N° 6: Ventana Registrar Curso .....	89
Figura N° 7: Ventana Listar Usuarios.....	90
Figura N° 8: Ventana Editar Usuario .....	90
Figura N° 9: Ventana Listar Cursos.....	91
Figura N° 10: Ventana Editar Curso.....	91
Figura N° 11: Ventana Editar Perfil de Usuario .....	92
Figura N° 12: Ventana Listar Cursos.....	92
Figura N° 13: Ventana Inscripción a Curso.....	93
Figura N° 14: Ventana Lista de estudiantes matriculados en un curso .....	93
Figura N° 15: Ventana Listar Cursos de Docente .....	94
Figura N° 16: Ventana Matricula de Estudiantes.....	94
Figura N° 17: Ventana Registrar Actividades .....	95
Figura N° 18: Ventana Lista de Actividades de Estudiante .....	95
Figura N° 19: Ventana Desarrollar Actividades .....	96
Figura N° 20: Ventana Enviar Solución de Actividad .....	96
Figura N° 21: Ventana Lista de Actividades de Curso .....	97
Figura N° 22: Ventana Mantenimiento de Actividades.....	97
Figura N° 23: Ventana Lista de Estudiantes que Enviaron Actividad .....	98
Figura N° 24: Ventana Calificar Actividad .....	98
Figura N° 25: Ventana Detalle de Actividades Enviadas y Pendientes. ....	99
Figura N° 26: Ventana Reporte de Aprobados por Curso y Actividad .....	99
Figura N° 27: Base de Datos .....	100

## INDICE DE CUADROS

Cuadro N° 1: Cronograma de actividades del proyecto.....	102
Cuadro N° 2: Costo de útiles de oficina .....	103
Cuadro N° 3: Costo de personal.....	103
Cuadro N° 4: Costo de Servicios.....	104
Cuadro N° 5: Resumen de costo de inversión de la implementación del Portafolio Electrónico y Aplicativo Android .....	104
Cuadro N° 6: Costo de operación de la implementación del Portafolio Electrónico y Aplicativo Android .....	105
Cuadro N° 7: Beneficios tangibles de la implementación del Portafolio Electrónico y Aplicativo Android .....	105
Cuadro N° 8: Beneficios intangibles de la implementación del Portafolio Electrónico y Aplicativo Android .....	106

# **CAPITULO I**

## **DATOS GENERALES DE LA ORGANIZACIÓN**

## **I. DATOS GENERALES DE LA ORGANIZACIÓN**

### **1.1. DESCRIPCIÓN DE LA ORGANIZACIÓN**

**Razón Social:** Universidad Nacional Pedro Ruiz Gallo

**Giro del Negocio:** Empresa que ofrece conocimientos

**Dirección:** Av. Juan XXIII 391 - Lambayeque - Perú

**Teléfono:** 283146 - Lambayeque

**Página Web:** <http://www.unprg.edu.pe>

### **1.2. MISIÓN, VISIÓN Y OBJETIVOS DE LA ORGANIZACIÓN**

#### **1.2.1. MISIÓN**

"Somos una universidad pública que crea, imparte, difunde conocimientos científicos, tecnológicos y humanísticos; forma científicos y profesionales innovadores, éticos, críticos y competitivos, que participan activamente en el desarrollo integral y sustentable de la sociedad"

#### **1.2.2. VISIÓN**

"Somos una universidad líder en la formación humanista, científica y tecnológica en el norte del país".



### 1.3. ORGANIGRAMA DE LA ORGANIZACIÓN

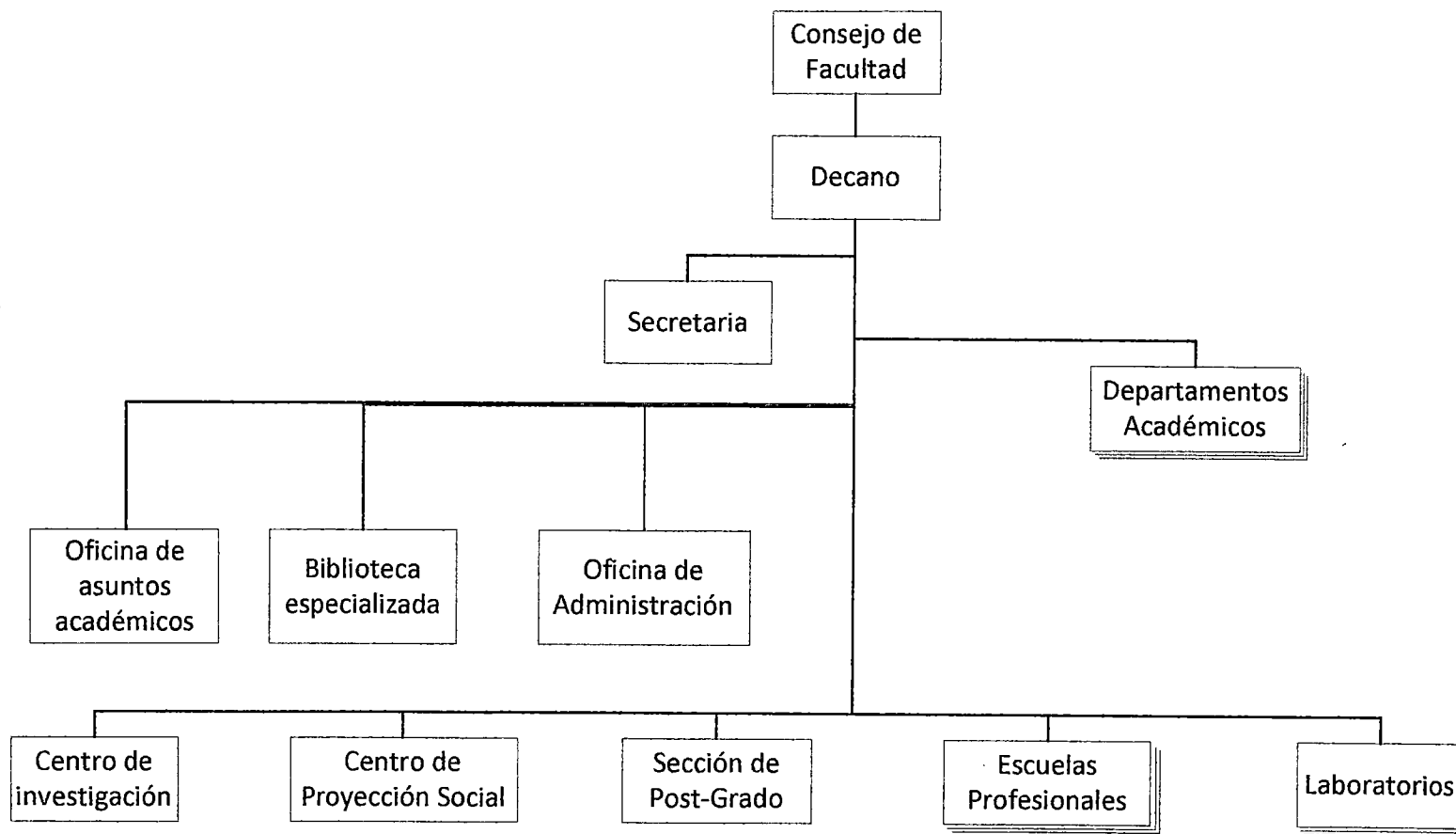


Figura N° 1: Organigrama de la Facultad de Ciencias Físicas y Matemáticas de la Universidad Nacional Pedro Ruiz Gallo

# **CAPITULO II**

## **PROBLEMÁTICA DE LA INVESTIGACIÓN**

## **II. PROBLEMÁTICA DE LA INVESTIGACIÓN**

### **2.1. REALIDAD PROBLEMÁTICA**

#### **2.1.1. PLANTEAMIENTO DEL PROBLEMA**

Los docentes y estudiantes de La Escuela Profesional de Ingeniería en Computación e Informática de La Universidad Nacional Pedro Ruiz Gallo, en la actualidad, los aportes de enseñanza de los docentes se comparten de forma tradicional, es decir haciendo uso de memorias USB los cuales pueden transmitir la información con virus y archivos maliciosos o compartiendo la información en carpetas en red donde muchas veces está limitado la cantidad de estudiantes que pueden visualizar la compartición, haciendo que no exista una retroalimentación de ese aprendizaje por parte de los estudiantes, dando como resultado una comunicación no muy adecuada entre docente – estudiante.

La problemática que viene atravesando la Escuela Profesional de Ingeniería en Computación e Informática es que tanto los estudiantes y docentes no cuentan con la información actualizada al alcance de sus manos ya que para tenerla se tiene que estar perdiendo tiempo en solicitarla.

En tal sentido, la finalidad del presente trabajo es la implementación de un portafolio electrónico que podría ser una solución adecuada al problema, pues facilita mayor capacidad y cobertura mediática de la formación académica.

## **2.2. FORMULACION DEL PROBLEMA**

¿De qué manera la implementación de un portafolio electrónico facilitará la información de la formación académica de los estudiantes de la Escuela Profesional de Ingeniería en Computación e Informática de la Universidad Nacional Pedro Ruiz Gallo?

## **2.3. JUSTIFICACIÓN E IMPORTANCIA DE LA INVESTIGACIÓN**

### **2.3.1. Justificación Teórica**

El desarrollo de la presente investigación, es una herramienta tecnológica que consiste en la aportación de información por parte de los docentes y estudiantes la cual permitirá facilitar la información para la formación académica y socializar el aprendizaje manteniendo una comunicación más fluida, pues justamente los e-Portafolios son una posibilidad mediática que permite demostrar que la enseñanza y aprendizaje guardan la mejor relación cuando se dispone de un espacio colaborativo.

### **2.3.2. Justificación Práctica**

Permitirá en las etapas de desarrollo de investigación integrar en un portafolio electrónico los requerimientos de los docentes y estudiantes en cuanto a la compartición de información académica actualizada y confiable, definiendo el concepto de "recurso de aprendizaje" como elemento básico para presentar resultados beneficiosos en la enseñanza universitaria.

### **2.3.3. Importancia**

Con en el desarrollo de este proyecto de investigación los docentes y estudiantes de la Escuela Profesional de Ingeniería de Computación e Informática de la Universidad Nacional Pedro Ruiz Gallo se pueden favorecer de manera complementaria en el nivel académico, puesto que el portafolio electrónico permitirá contar con información actualizada, confiable y organizada de tal forma que el acceso se realice en forma eficiente y es un reflejo de evolución de un proceso de aprendizaje teniendo en cuenta que los objetivos educacionales son la columna vertebral de la evaluación, siendo el cuerpo docente la base sobre la que se asienta y desarrolla el proceso de aprendizaje por lo que constituye el elemento principal de la calidad de la carrera.

## **2.4. OBJETIVOS DE LA INVESTIGACIÓN**

### **2.4.1. OBJETIVO GENERAL**

Implementar un portafolio electrónico para facilitar la compartición de información de los docentes y estudiantes de la Escuela Profesional de Ingeniería en Computación e Informática de la Universidad Nacional Pedro Ruiz Gallo.

### **2.4.2. OBJETIVOS ESPECÍFICOS**

- Recolectar requerimientos de los docentes y estudiantes
- Diseñar y Modelar una Base de Datos que sirva de soporte para el portafolio electrónico
- Diseñar las interfaces para cada iteración del portafolio electrónico.

# **CAPITULO III**

## **MARCO METODOLÓGICO**

### **III. MARCO METODOLÓGICO**

#### **3.1. TIPO DE INVESTIGACIÓN**

Tecnológica Formal

#### **3.2. HIPÓTESIS**

La implementación de un portafolio electrónico facilitará la compartición de la información académica de los docentes y estudiantes de la Escuela Profesional de Ingeniería en Computación e Informática de la Universidad Nacional Pedro Ruiz Gallo.

#### **3.3. VARIABLES**

##### **3.3.1. VARIABLE INDEPENDIENTE**

Portafolio electrónico.

##### **3.3.2. VARIABLE DEPENDIENTE**

Facilitar la información para la formación académica de los estudiantes de la Escuela Profesional de Ingeniería en Computación e Informática de la Universidad Nacional Pedro Ruiz Gallo.

# **CAPITULO IV**

## **MARCO TEÓRICO**



## **IV. MARCO TEÓRICO**

### **4.1. ANTECEDENTES DE LA INVESTIGACIÓN**

#### **4.1.1. ANTECEDENTE N°1**

**Tipo de Antecedente:** Nacional

**Título:** "Estrategia Didáctica Basada en un Sistema de Red de Computadoras y su Influencia en el Rendimiento Académico de los Alumnos de la Asignatura de Sistemas de Información del VIII Ciclo de Ingeniería de Sistemas e Informática de la UNS".

**Autores:** Díaz Tello, Sixto

**Año:** 2012

**Lugar:** Trujillo - Perú

**RESUMEN:** El propósito de la investigación fue determinar la influencia de la aplicación de la Estrategia Didáctica usando un sistema de red de computadoras conectadas a internet a través de una página web en el rendimiento académico de los alumnos del VIII Ciclo de Ingeniería de Sistemas e Informática de la Universidad Nacional del Santa, en la asignaturas de Sistemas de Información. La población estuvo constituida por 272 alumnos pertenecientes a la escuela de Ingeniería de Sistemas e Informática. La muestra de estudio estuvo representada por 40 alumnos matriculados en la asignatura de Sistemas de Información, la cual fue dividida en dos grupos, experimental con 20 alumnos y grupo control con 20 alumnos. Durante el desarrollo metodológico se administró un test inicial,

test final, así como aplicaciones de instrumentos de enseñanza aprendizaje y evaluaciones periódicas a través de una página Web, que regulaban los progresos experimentados, permitiendo así modificar los planes a medida que los alumnos asimilan nuevos conocimientos en cada clase virtual. Procesados los resultados y haciendo uso de la distribución de la probabilidad normal estándar se validó la hipótesis, lo que nos indica que hubo un incremento significativo en el rendimiento de los alumnos, en la asignatura de sistemas de información, aplicando la estrategia didáctica usando un sistema de red de computadoras conectadas a internet a través de una página Web denominada: Sistema\_de\_Inf., por lo que se recomienda utilizar esta estrategia didáctica usando un sistema de Red de computadoras conectadas a internet a través de una página Web en el aprendizaje de las asignaturas para la especialización de Ingeniería de Sistemas e Informática de la Universidad Nacional del Santa.

#### **4.1.2. ANTECEDENTE N°2**

**Tipo de Antecedente:** Internacional

**Título:** Portafolio en línea en la formación inicial docente.

**Autor:** Mellado Hernandez, Maria Elena

**Año:** 2010

**Lugar:** Temuco - Chile

**RESUMEN:** El objetivo de este estudio es evaluar el grado de satisfacción con respecto al portafolio en línea como herramienta para la evaluación del desempeño inicial docente de los estudiantes de Pedagogía en Educación Básica de la Universidad Católica de Temuco, Chile. En esta investigación se utilizó un grupo experimental que trabajo con el portafolio en línea, y un grupo de control que implemento el portafolio físico. La muestra correspondió a un grupo de 30 estudiantes seleccionados al azar, pertenecientes al cuarto año del Programa. El grado de satisfacción expresado por los usuarios del portafolio se evaluó al final de la experiencia de práctica docente. Para realizar esta medición se construyó un instrumento denominado diferencial semántico. El grupo experimental presento una diferencia significativamente mayor que el grupo de control en la variable grado de satisfacción respecto de la herramienta portafolio. Se observó también, una correlación positiva y significativa entre el grado de satisfacción expresado por los estudiantes y el puntaje obtenido en el portafolio de desempeño inicial docente. Palabras clave: Evaluación del desempeño docente, portafolio, formación de profesores.

#### **4.1.3. ANTECEDENTE N°3**

**Tipo de Antecedente:** Internacional

**Título:** “Propuesta de incorporación del Portafolio virtual apoyado en las NTICs, como estrategia para fortalecer la evaluación en la enseñanza de la educación superior impartida en la PUCE”.

**Autor:** Gârcía Paredes, Yolanda Cecilia

**Año:** 2013

**Lugar:** PUCE - Quito - Ecuador

**RESUMEN:** La sociedad actualmente es impulsada por el avance en el desarrollo y uso generalizado de las Nuevas Tecnologías de Información y Comunicación NTICs. Este panorama conlleva cambios en todos los ámbitos de la actividad humana y que se manifiestan diariamente, a nivel social, educativo, cultural, etc. Las personas continuamente son sujetos de un aprendizaje informal permanente, a través de nuestras relaciones sociales, culturales y de la influencia de los medios de comunicación y tecnología presentes; es por ello que el gran reto de las instituciones educativas de hoy, está en integrar estos canales existentes en los procesos de enseñanza-aprendizaje.

## **4.2. BASE TEÓRICA**

### **4.2.1. EXTREME PROGRAMMING (XP)**

Extreme Programming (XP) surge como una nueva manera de encarar proyectos de software, proponiendo una metodología basada esencialmente en la simplicidad y agilidad. Las metodologías de desarrollo de software tradicionales (ciclo de vida en cascada, evolutivo, en espiral, iterativo, etc.) aparecen, comparados con los nuevos métodos propuestos en XP, como pesados y poco eficientes. (Etcheverría Tobón, 2007).

XP es una de las llamadas metodologías ágiles de desarrollo de software más exitosas de los tiempos recientes. La metodología propuesta en XP está diseñada para entregar el software que los clientes necesitan en el momento en que lo necesitan.

XP alienta a los desarrolladores a responder a los requerimientos cambiantes de los clientes, aún en fases tardías del ciclo de vida del desarrollo.

La metodología también enfatiza el trabajo en equipo. Tanto gerentes como clientes y desarrolladores son partes del mismo equipo dedicado a entregar software de calidad.

La metodología XP define cuatro variables para cualquier proyecto de software: costo, tiempo, calidad y alcance.

Además, se especifica que, de estas cuatro variables, sólo tres de ellas podrán ser fijadas arbitrariamente por actores externos al grupo de desarrolladores (clientes y jefes de proyecto). El valor de la variable restante podrá ser establecido por el equipo de desarrollo, en función de los valores de las otras tres. Este mecanismo indica que, por ejemplo, si el cliente establece el

alcance y la calidad, y el jefe de proyecto el precio, el grupo de desarrollo tendrá libertad para determinar el tiempo que durará el proyecto.

El ciclo de vida de un proyecto XP incluye, al igual que las otras metodologías, entender lo que el cliente necesita, estimar el esfuerzo, crear la solución y entregar el producto final al cliente. Sin embargo, XP propone un ciclo de vida dinámico, donde se admite expresamente que, en muchos casos, los clientes no son capaces de especificar sus requerimientos al comienzo de un proyecto.

Por esto, se trata de realizar ciclos de desarrollo cortos (llamados iteraciones), con entregables funcionales al finalizar cada ciclo. En cada iteración se realiza un ciclo completo de análisis, diseño, desarrollo y pruebas, pero utilizando un conjunto de reglas y prácticas que caracterizan a XP (y que serán detalladas más adelante).

#### **4.2.1.1. *Objetivos de XP***

Los objetivos de XP son muy simples: la satisfacción del cliente. Esta metodología trata de dar al cliente el software que él necesita y cuando lo necesita. Por tanto, debemos responder muy rápido a las necesidades del cliente, incluso cuando los cambios sean al final de ciclo de la programación.

El segundo objetivo es potenciar al máximo el trabajo en grupo. Tanto los jefes de proyecto, los clientes y desarrolladores, son parte del equipo y están involucrados en el desarrollo del software. (Calero, 2003)

#### **4.2.1.2. Los Principios de XP**

Hay 5 principios fundamentales y varios mas adicionales.

Los fundamentales son:

- a) Realimentación rápida: con la idea de maximizar el aprendizaje, reduciendo el coste del cambio y de los errores.
- b) Asumir la sencillez: se parte del supuesto de que para todo problema complejo hay siempre una solución sencilla. Se debe trabajar para encontrarla, empezando por aceptar que existe.
- c) Cambio incremental: nada de realizar grandes cambios, estos no funcionan. La diferencia se consigue mediante la sucesión de pequeños cambios que hará la diferencia.
- d) Aceptar el cambio: se optará por las estrategias de desarrollo que permitan posponer las decisiones al máximo, así el cliente no tendrá que optar ni invertir hasta el último momento, con la mayor información posible y el menor costo.
- e) Trabajar con calidad: hacer un trabajo bajo los estándares de calidad posibles, satisfaciendo al máximo al cliente.

#### **4.2.1.3. Ventajas y desventajas**

Las ventajas son: Se adapta al desarrollo de sistemas pequeños y grandes; optimiza el tiempo de desarrollo; permite realizar el desarrollo del sistema en parejas para complementar los conocimientos; el código es sencillo y entendible, además de la poca documentación a elaborar para el desarrollo del sistema.

Las desventajas son: No se tiene la definición del costo y el tiempo de desarrollo; el sistema va creciendo después de cada entrega al cliente y nadie puede decir que el cliente no querrá una función más; se necesita de la presencia constante del usuario, lo cual en la realidad es muy difícil de lograr.

Otra desventaja es la programación en parejas, algunos desarrolladores son celosos del código que escriben y no les es grato que alguien más modifique las funciones que realizó o que su código sea desechado por no cubrir el estándar.

#### **4.2.1.4. Ciclo de vida de XP**

Si bien el ciclo de vida de un proyecto XP es muy dinámico, se puede separar en fases:

##### **✓ Fase de exploración**

En esta fase, el cliente define lo que necesita mediante la redacción de sencillas "historias de



usuarios". Los programadores estiman los tiempos de desarrollo en base a esta información. Debe quedar claro que las estimaciones realizadas en esta fase son primarias (ya que estarán basadas en datos de muy alto nivel), y podrían variar cuando se analicen más en detalle en cada iteración.

Esta fase dura típicamente un par de semanas, y el resultado es una visión general del sistema, y un plazo total estimado.

#### ✓ **Fase de planificación**

La planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación. El resultado de esta fase es un Plan de Entregas, como se detallará en la sección "Reglas y Practicas".

#### ✓ **Fase de iteraciones**

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Como las historias de usuario no tienen suficiente detalle como para permitir su análisis y desarrollo, al principio de cada iteración

se realizan las tareas necesarias de análisis, recabando con el cliente todos los datos que sean necesarios. El cliente, por lo tanto, también debe participar activamente durante esta fase del ciclo.

Las iteraciones son también utilizadas para medir el progreso del proyecto. Una iteración terminada sin errores es una medida clara de avance.

#### ✓ **Fase de puesta en producción**

Si bien al final de cada iteración se entregan módulos funcionales y sin errores, puede ser deseable por parte del cliente no poner el sistema en producción hasta tanto no se tenga la funcionalidad completa.

En esta fase no se realizan más desarrollos funcionales, pero pueden ser necesarias tareas de ajuste.

## **4.2.2. PORTAFOLIO ELETRÓNICO**

### **4.2.2.1. *El portafolio:***

Un Portafolio es una colección sistemática y organizada de evidencias utilizadas por el profesor y los estudiantes para supervisar la evolución del conocimiento, las habilidades y las actividades de estos últimos en una materia determinada, además de que sirve como base para examinar el esfuerzo, los progresos, los procesos y los logros, así como satisfacer las exigencias de

responsabilidad habitualmente alcanzadas por procedimientos de prueba más formales.

#### **4.2.2.2. *El portafolio electrónico (e-P)***

El portafolio electrónico se le puede definir como una herramienta útil donde se recopila información digital que demuestre evidencias del desarrollo o del resultado del aprendizaje y las habilidades. El proceso de realización de un portafolio electrónico (redacción, grabación, etc.,) por lo general requiere una síntesis de ideas y reflexión de los logros obtenidos. Existen distintos tipos de portafolios, que se pueden definir por sus efectos (como la presentación, aplicación, evaluación y planificación del desarrollo personal), el diseño pedagógico, el nivel de la estructura, la duración y otros factores.

También como "...una colección de materiales digitalizados que incluyen demostraciones, recursos y logros que representan a un individuo, un grupo o una institución. Este texto puede comprender texto, gráficos o elementos multimedia que se pueden consultar en Internet o en otro medio electrónico. Un eP es más que una simple colección – también sirve como herramienta para administrar, organizar y controlar los trabajos creados con diferentes aplicaciones. Los e-portafolios incluyen reflexión personal y por lo regular promueven el intercambio de ideas y retroalimentación".

De igual forma “un portafolio electrónico, también conocido como e-portafolio o portafolio digital, es una colección personal de información que describe o documenta los logros y aprendizajes de los usuarios, usualmente en la web.

Estas pruebas electrónicas pueden incluir entradas de texto, archivos electrónicos tales como Microsoft Word y Adobe PDF, imágenes, multimedia, entradas de blog e hipervínculos. Los e-portafolios son a la vez una demostración de las habilidades del usuario y de las plataformas de libre expresión. Algunas aplicaciones del portafolio electrónico permite diversos grados de acceso al público, de modo que el mismo portafolio podría ser utilizado para múltiples fines

### **4.2.3. LENGUAJES DE PROGRAMACIÓN**

Un lenguaje de programación se refiere a cualquier lenguaje artificial que pueda ser empleado para definir una secuencia de instrucciones para su procesamiento por una computadora u ordenador. Por lo general, se encuentra formado por un conjunto de símbolos y reglas de tipos semánticas y sintácticas, que permiten a los programadores definir de manera precisa acerca de qué datos debe operar una computadora, cómo estos datos deben ser almacenados o transmitidos y qué acciones debe tomar ante diferentes eventos.

#### **4.2.3.1. LARAVEL 4.2**

Laravel es un framework para aplicaciones web con sintaxis expresiva y elegante. Creemos que el desarrollo debe ser una experiencia agradable y creativa para que sea verdaderamente enriquecedora. Laravel busca eliminar el sufrimiento del desarrollo facilitando las tareas comunes utilizadas en la mayoría de los proyectos web, como la autenticación, enrutamiento, sesiones y almacenamiento en caché.

Laravel pretende hacer que el proceso de desarrollo sea agradable para el desarrollador sin sacrificar funcionalidades de la aplicación. Desarrolladores felices hacen mejor código. Con este fin, hemos intentando combinar lo mejor que hemos visto en otros frameworks

web, incluyendo frameworks de otros lenguajes, como Ruby on Rails, ASP.NET MVC y Sinatra.

Laravel es accesible, pero potente, ofreciendo herramientas poderosas necesarias para aplicaciones de gran envergadura. Un espléndido contenedor de inversión de control, sistema de migraciones expresivo, y un soporte para pruebas unitarias estrechamente integrado, te entregan las herramientas que necesitas para construir cualquier aplicación que te hayan encargado.

Gran parte de Laravel está formado por dependencias, especialmente de Sinfonía, esto implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias.

#### **Características:**

- ✓ Sistema de ruteo, también RESTful.
- ✓ Blade, Motor de plantillas.
- ✓ Peticiones Fluent.
- ✓ Eloquent ORM.
- ✓ Basado en Composer.
- ✓ Soporte para el caché.
- ✓ Soporte para MVC.
- ✓ Usa componentes de Symfony.

## Patrón MVC

Laravel, propone en el desarrollo usar 'Routes with Closures', en lugar de un MVC tradicional con el objetivo de hacer el código más claro. Aun así permite el uso de MVC tradicional.

```
<?php
Route::get('usuario/listar', function() {
    //punto de entrada de la petición HTTP ...});
```

### a. Modelo

Laravel incluye un sistema de mapeo de datos relacional llamado Eloquent ORM que facilita la creación de modelos. Este ORM se funda en patrón active record y su funcionamiento es muy sencillo. Es opcional el uso de Eloquent, pues también dispone de otros recursos que nos facilitan interactuar con los datos, o específicamente la creación de modelos.

La forma de crear Modelos en Laravel usando Eloquent ORM, es tan simple como:

```
class Libro extends Eloquent {
    //defiendo el nombre de la tabla con la info de los
    libros
    public static $table = 'tb_libros';
}
```

Incluso el código del modelo puede ser más simple aún, en el caso que el nombre de la tabla coincida con el nombre de la clase. Pues Laravel

al igual que Ruby on Rails, usa el paradigma de programación donde se favorece "la convención sobre la configuración". Y si ahora necesitamos disponer un listado, en la ruta: `http://mi-aplicacion.com/libro/listar` Entonces, sólo bastaría crear la Ruta e interactuar con el modelo 'Libro' anteriormente creado, del modo siguiente:

```
Route::get('libro/listar', function() {  
    $libros = Libro::all();  
    return View::make('mi_vista', $libros);  
    //Muestra los datos  
});
```

#### **b. Vista**

Laravel incluye de paquete un sistema de procesamiento de plantillas llamado Blade. Este sistema de plantillas favorece un código mucho más limpio en las Vistas, además de incluir un sistema de Caché que lo hace mucho más rápido. El sistema Blade de Laravel, permite una sintaxis mucho más reducida en su escritura.<sup>4</sup> Por ejemplo, en vez pintar la vista usando el código PHP:

```
<php echo $mi_nombre; ?>
```

En Blade se escribiría:

```
{{ $mi_nombre }}
```

Lo cual no es una gran ventaja sobre todo cuando siempre es posible usar una expresión resumida



en PHP. No obstante, lo que sí es una gran ventaja, es el modo en que Blade maneja las plantillas.

## **Plantillas**

Las plantillas en Blade son archivos de texto plano que contiene todo el HTML de la página con etiquetas que representan elementos o zonas a incluir en la plantilla, o vistas parciales como se conocen en otros frameworks PHP. Sin embargo, en Blade estos elementos incrustados se organizan en un sólo archivo. Esta es una idea muy interesante de Laravel que mejora la organización de las vistas y su rendimiento. Sobre todo cuando las vistas pueden llegar a ser muy complejas incluso con elementos anidados. En el render de una Vista completa en Lavarel se usan dos archivos: la plantilla definiendo el HTML global y las zonas a incluir. Un sólo archivo, la Vista, con los elementos (partial views).

Un ejemplo típico para una plantilla HTML5 de Blade:

```
<!DOCTYPE html>

<html lang="es">

    <head>

        <meta charset="UTF-8">

        <title>@yield('titulo')</title>
```

```
</head>

<body>

    @yield('navegacion')

</body>

</html>
```

### c. Controlador

Los controladores contienen la lógica de la aplicación y permiten organizar el código en clases sin tener que escribirlo todo en las rutas. Todos los controladores deben extenderse de la clase BaseController.

Un ejemplo de un controlador en Laravel:

```
class UserController extends BaseController {
    public function mostrarPerfil($id)
    {
        $user = User::find($id);
        return View::make('user.profile',
            array('user' => $user));
    }
}
```

#### **4.2.3.2. *BOOSTRAP CSS FRAMEWORK***

Bootstrap es un libre y de código abierto colección de herramientas para la creación de sitios web y aplicaciones web. Contiene HTML - y CSS plantillas de diseño basados en la tipografía, formas, botones, navegación y otros componentes de la interfaz, así como opcionales de JavaScript extensiones.

##### **4.2.3.2.1. Características**

Bootstrap es compatible con las últimas versiones del Google Chrome, Firefox, Internet Explorer, Opera y Safari navegadores, aunque algunos de estos navegadores no son compatibles con todas las plataformas.

Desde la versión 2.0 también es compatible con el diseño web sensible. Esto significa que el diseño de las páginas web ajusta dinámicamente, teniendo en cuenta las características del dispositivo utilizado (de escritorio, tableta, teléfono móvil).

Desde la versión 3.0, Bootstrap adoptó un primer móvil filosofía de diseño, haciendo hincapié en el diseño de respuesta por defecto.

Manos a la Obra es de código abierto y está disponible en GitHub. Se anima a los

desarrolladores a participar en el proyecto y hacer sus propias contribuciones a la plataforma.

#### **4.2.3.2.2. Estructura y Función**

Bootstrap es modular y consiste esencialmente en una serie de hojas de estilo menos que implementan los distintos componentes de la caja de herramientas. Una hoja de estilo llamado **BOOTSTRAP.LESS** incluye las hojas de estilo de componentes. Los desarrolladores pueden adaptar el Bootstrap propio archivo, seleccionar los componentes que desean utilizar en su proyecto.

Los ajustes son posibles de forma limitada a través de una hoja de estilos de configuración central.

El uso de un lenguaje menos de estilos permite el uso de variables, funciones y operadores, selectores anidados, así como los llamados mixins.

Desde la versión 2.0, la configuración de Bootstrap también tiene una opción especial "Personalizar" en la documentación. Por otra parte, el desarrollador elige en un formulario los componentes deseados y ajusta, si es necesario, los valores de las distintas opciones a sus necesidades. El paquete generado posteriormente ya incluye la hoja de estilos CSS pre-construidos.

Sistema de red y el diseño de respuesta viene de serie con un diseño 1170 píxeles de ancho,

cuadrícula. Alternativamente, el desarrollador puede utilizar un diseño de anchura variable. Para ambos casos, el conjunto de herramientas tiene cuatro variaciones de hacer uso de diferentes resoluciones y tipos de dispositivos: teléfonos móviles, retrato y el paisaje, tabletas y PCs con baja y alta resolución. Cada variación ajusta el ancho de las columnas.

#### ✓ **El CSS**

Bootstrap proporciona un conjunto de hojas de estilo que proporcionan definiciones de estilo básicas para todos los componentes HTML clave. Estos proporcionan un uniforme, aspecto moderno para formatear texto, tablas y elementos de formulario.

#### ✓ **Reutilizables componentes**

Además de los elementos HTML regulares, Bootstrap contiene otros elementos de la interfaz de uso común. Estos incluyen botones con funciones avanzadas (por ejemplo, la agrupación de botones o teclas opción desplegable con, hacer y listas de navegación, pestañas horizontales y verticales, la navegación, la ruta de navegación, la paginación, etc.), etiquetas, capacidades tipográficas avanzadas, miniaturas, mensajes de advertencia y una barra de progreso. Los componentes se

implementan como clases CSS, que deben aplicarse a ciertos elementos HTML en una página.

#### ✓ Componentes de JavaScript

Bootstrap viene con varios componentes de JavaScript en la forma de jQuery plugins. Proporcionan elementos adicionales de interfaz de usuario, tales como cuadros de diálogo, información sobre herramientas y carruseles. También amplían la funcionalidad de algunos elementos de la interfaz existentes, incluyendo por ejemplo una función de autocompletado de campos de entrada. En la versión 2.0, los siguientes plugins de JavaScript son compatibles: Modal, desplegable, Scrollspy, Tab, Tooltip, Popover, Alerta, Button, Collapse, Carrusel y TYPEAHEAD.

#### 4.2.3.3. HTML5

HTML5, en conjunto con CSS3, define los nuevos estándares de desarrollo web, tanto para equipos de escritorio, como para dispositivos móviles y teléfonos celulares. Se abren las puertas para nuevas opciones de diseño y dinámica de información. Se trata de un conjunto de etiquetas que sirven para dar formato a diferentes tipos de contenidos, sobre todo texto.

El HTML solamente formatea y muestra contenidos, y no tiene la capacidad de interactuar con variables, bases de datos, etc. Aún los sitios web más complejos, que utilizan grandes sistema de bases de datos y realizan complejas tareas, envían al navegador (Internet Explorer, Chrome, etc.) puro código HTML.

Hoja de estilo en cascada (CCS).

CSS tiene una sintaxis muy sencilla, que usa unas cuantas palabras clave tomadas del inglés para especificar los nombres de varias propiedades de estilo. Una hoja de estilo se compone de una lista de reglas. Cada regla o conjunto de reglas consiste en uno o más selectores y un bloque de declaración (o "bloque de estilo") con los estilos a aplicar para los elementos del documento que cumplan con el selector que les precede.

#### **4.2.4. BASE DE DATOS**

Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.

Una base de datos proporciona a los usuarios el acceso a datos, que pueden visualizar, ingresar o actualizar, en concordancia con los derechos de acceso que se les hayan otorgado. Se convierte más útil a medida que la cantidad de datos almacenados crece.

La principal ventaja de utilizar bases de datos es que múltiples usuarios pueden acceder a ellas al mismo tiempo.

#### **4.2.4.1. VENTAJAS DE LAS BASES DE DATOS**

La utilización de bases de datos como plataforma para el desarrollo de sistemas de aplicación en las organizaciones se ha incrementado notablemente en los últimos años y esto se debe a las ventajas que ofrece su utilización, algunas de las cuales se comentarán a continuación: (Date, 2001)

- a. Compactación.
- b. Velocidad.
- c. Menos trabajo laborioso.
- d. Actualidad.

#### **4.2.4.2. SISTEMA DE GESTIÓN DE BASE DE DATOS (SGBD)**

Es un conjunto de programas que permite a los usuarios crear y mantener una base de datos. Por tanto, el SGBD es un sistema de Software de propósito general que facilita el proceso de definir, construir y manipular bases de datos para diversas aplicaciones. (Elmasri, R. y Navathe, S., 2000).



#### **4.2.4.3. MYSQL**

MySQL es un sistema gestor de bases de datos (SGBD, DBMS por sus siglas en inglés) muy conocido y ampliamente usado por su simplicidad y notable rendimiento. MySQL está disponible para múltiples plataformas.

##### **Características:**

- ✓ Está desarrollado en C/C++.
- ✓ Se distribuyen ejecutables para cerca de diecinueve plataformas diferentes.
- ✓ La API se encuentra disponible en C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.
- ✓ Está optimizado para equipos de múltiples procesadores.
- ✓ Es muy destacable su velocidad de respuesta.
- ✓ Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
- ✓ Cuenta con un rico conjunto de tipos de datos.
- ✓ Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD a cada caso concreto.
- ✓ Su administración se basa en usuarios y privilegios.

## **4.2.5. HERRAMIENTAS**

### **4.2.5.1. WORKBENCH**

MySQL Workbench es una herramienta unificada de base de datos visuales para arquitectos, desarrolladores y DBAs. MySQL Workbench proporciona el modelado de datos, desarrollo de SQL y herramientas completas de administración de configuración del servidor, administración de usuarios, y mucho más. MySQL Workbench está disponible para ejecutar en Windows, Linux y Mac OS.

#### **Ventajas:**

- ✓ Libre, distribuida bajo la licencia GPL
- ✓ Multiplataforma disponible para Windows, GNU/Linux. Mac.
- ✓ Permite crear diagramas E-R.
- ✓ Importar archivos SQL.
- ✓ Permite generar los scripts SQL a partir del modelo creado.
- ✓ Ofrece soporte completo a las características de MySQL 5.

Diseño: MySQL Workbench permite a un desarrollador DBA o arquitecto de datos diseñar visualmente y gestionar bases de datos.

Desarrolla: MySQL Workbench proporciona herramientas visuales para crear, ejecutar y optimizar consultas SQL.

Administrar: MySQL Workbench proporciona una representación visual de la consola para administrar fácilmente entornos de MySQL y obtener una mejor visibilidad en base de datos.

# **CAPITULO V**

## **DESARROLLO DE LA PROPUESTA**

## V. DESARROLLO DE LA PROPUESTA

### 5.1. ARQUITECTURA DE SOLUCION

El sistema es una arquitectura cliente servidor, el servidor es donde se aloja la aplicación. Los clientes son todos los usuarios que accederán al sistema a través de cualquier navegador web o aplicativo Android.

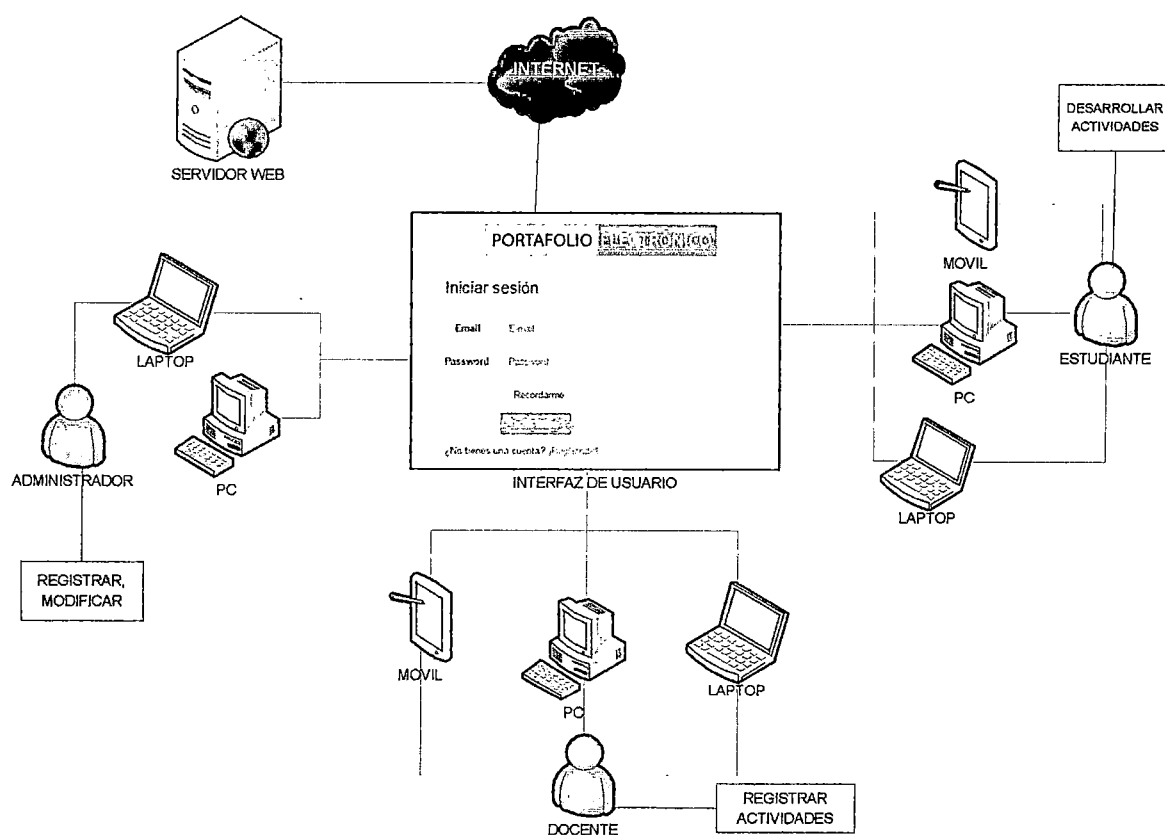


Figura Nº 2: Arquitectura de solución

## **5.1.1. REQUISITOS BÁSICOS DE LOS DISPOSITIVOS**

### **5.1.1.1. SERVIDOR WEB**

Este servidor corre bajo CENTOS 6, en el cual se instalaron los siguientes servidores:

#### **✓ APACHE**

Servidor web, con Apache 2.0 con mod\_rewrite habilitado.

#### **✓ PHP**

- Requerido PHP mayor o igual a la V.5.3.7
- Extensión PHP MCrypt

#### **✓ MYSQL O MARIADB**

Se necesita un gestor de base de datos relacional:

- Mysql mayor o igual a la V. 5.6.24
- MariaDB mayor o igual a la V.10.0.17

### **HARDWARE NECESARIO:**

1. RAM: 512MB
2. Procesador: 1.5GHZ
3. Espacio de disco: 10GB

#### **5.1.1.2. PCS:**

Para el uso de la aplicación web, se necesita ordenadores con navegadores web instalados: Chrome, Firefox, Opera o Safari, IE8+.

Compatible con sistemas operativos Windows, OSX y Linux.

#### **5.1.1.3. CELULAR:**

Smartphones de gama media en adelante con los siguientes requisitos:

- Sistema operativo Android 4.0.3 o superior.
- Espacio de almacenamiento libre de al menos 20MB.
- Conexión a internet mediante Wifi o Datos Móviles.

#### **5.1.1.4. TABLET:**

Smartphones de gama media en adelante con los siguientes requisitos:

- Sistema operativo Android 4.0.3 o superior.
- Espacio de almacenamiento libre de al menos 20MB
- Resolución de pantalla de al menos 800x480
- Conexión a internet mediante Wifi o Datos Móviles

## 5.2. PLANIFICACIÓN

### 5.2.1. ITERACIÓN 1

Consta de 4 Historias de Usuario:

- ❖ **Iniciar Sesión:** 2 puntos – 3 tareas
- ❖ **Registrar Docente:** 3 puntos - 2 tareas
- ❖ **Registrar Estudiante:** 3 puntos - 2 tareas
- ❖ **Registrar Curso:** 4 puntos - 2 tareas

Historia de Usuario	
Número: 1	Usuario: Administrador de Web
Nombre historia: Iniciar Sesión	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  Para poder ingresar a la web, los usuarios tendrán que ingresar su correo electrónico y password, el usuario tendrá acceso a las opciones del menú dependiendo de los permisos que éste tenga asignados.	
<b>Observaciones:</b>  ➤ Si el correo electrónico o password son incorrectos, se mostrará un mensaje de error y se solicitará nuevamente el ingreso de datos.	

Tabla N° 1: Historia de Usuario N° 1



Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 01, Iniciar Sesión
<b>Nombre tarea:</b> Diseñar las entidades que abarquen el manejo de usuarios en la base de datos	
<b>Tipo de tarea :</b> Diseño	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 08/12/2014	<b>Fecha fin:</b> 10/12/2014
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se modelarán las entidades que se usarán para el manejo de usuarios del sistema. Éstas se incluirán en el Modelo Relacional de la base de datos.	

Tabla N° 2: Tarea Diseñar las entidades que abarquen el manejo de usuarios en la base de datos

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 01, Iniciar Sesión
<b>Nombre tarea:</b> Implementar ventana para iniciar sesión	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 11/12/2014	<b>Fecha fin:</b> 13/12/2014
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita iniciar sesión, la ventana contará con dos controles, uno para ingresar el Email del usuario y el otro control para ingresar su password, si el usuario no está registrado, habrá una opción para permitirle registrarse, además se realizará el cargado de las opciones de sistema de acuerdo al tipo de usuario que ha iniciado sesión.	

Tabla N° 3: Tarea Implementar ventana para iniciar sesión

Tarea	
<b>Número de tarea: 3</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 01, Iniciar Sesión
<b>Nombre tarea:</b> Pruebas en ventana para iniciar sesión	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 14/12/2014	<b>Fecha fin:</b> 15/12/2014
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se harán las pruebas correspondientes para iniciar sesión y se realizarán los cambios que sean necesarios. Si hay presencia de errores en la verificación de datos, se harán las correcciones correspondientes.	

Tabla N° 4: Tarea Pruebas en ventana para iniciar sesión

Historia de Usuario	
<b>Número: 2</b>	<b>Usuario: Administrador de Web</b>
<b>Nombre historia:</b> Registrar Docente	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b>  El administrador, luego de identificar los docentes encargados a dictar los cursos procede a ingresar sus datos personales y confirma el registro.	
<b>Observaciones:</b>  ➤ Solo el administrador es el responsable del registro de los docentes, esto evitará a que personas ajenas o sin atribuciones se registren como docentes. ➤ El correo electrónico debe ser único para cada docente, no debe repetirse con algún correo registrado por un anterior usuario.	

Tabla N° 5: Historia de Usuario N° 2

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 02, Registrar Docente
<b>Nombre tarea:</b> Implementar ventana para registrar docente	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 2.5</b>
<b>Fecha inicio:</b> 16/12/2014	<b>Fecha fin:</b> 19/12/2014
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita registrar un docente, pedirá sus datos personales y se tendrá en cuenta la validación de correo electrónico como dato único para cada docente.	

Tabla N° 6: Tarea Implementar ventana para registrar docente

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 02, Registrar Docente
<b>Nombre tarea:</b> Pruebas en ventana para registrar docente	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 20/12/2014	<b>Fecha fin:</b> 21/12/2014
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se harán las pruebas correspondientes al ingreso de datos del docente y a su registro, si hay presencia de errores en el registro de datos, se harán las correcciones correspondientes.	

Tabla N° 7: Tarea Pruebas en ventana para registrar docente

Historia de Usuario	
<b>Número: 3</b>	<b>Usuario: Administrador de Web y Estudiante</b>
<b>Nombre historia:</b> Registrar Estudiante	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b>  El administrador, luego de identificar que los estudiantes pertenecen a la EPICl procede a ingresar sus datos personales y confirma el registro. El estudiante también puede registrarse directamente por la web, ingresa sus datos personales y confirma su registro.	
<b>Observaciones:</b>  ➤ Tanto el administrador como el estudiante pueden realizar dicho registro. ➤ El correo electrónico debe ser único para cada estudiante, no debe repetirse con algún correo registrado por un anterior usuario.	

Tabla N° 8: Historia de Usuario N° 3

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 03, Registrar Estudiante
<b>Nombre tarea:</b> Implementar ventana para registrar estudiante	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2.5
<b>Fecha inicio:</b> 22/12/2014	<b>Fecha fin:</b> 25/12/2014
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita registrar un estudiante, pedirá sus datos personales y se tendrá en cuenta la validación de correo electrónico como dato único para cada estudiante.	

Tabla N° 9: Tarea Implementar ventana para registrar estudiante

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 03, Registrar Estudiante
<b>Nombre tarea:</b> Pruebas en ventana para registrar estudiante	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 26/12/2014	<b>Fecha fin:</b> 27/12/2014
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se harán las pruebas correspondientes al ingreso de datos del estudiante y a su registro, si hay presencia de errores en el registro de datos, se harán las correcciones correspondientes.	

Tabla N° 10: Tarea Pruebas en ventana para registrar estudiante

Historia de Usuario	
<b>Número: 4</b>	<b>Usuario: Administrador de Web</b>
<b>Nombre historia:</b> Registrar Curso	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  El administrador, luego de haber registrado a los docentes y estudiantes procede a registrar un curso, para lo cual ingresa los datos correspondientes, elige al docente encargado de dictarlo, el número de estudiantes que están matriculados y las fechas de inicio y fin del curso, luego confirma el registro.	
<b>Observaciones:</b>  <ul style="list-style-type: none"> <li>➤ Solo el administrador es el responsable del registro de los cursos, para abrir un curso es necesario haber registrado antes a los docentes y estudiantes.</li> <li>➤ Cada curso se registra con estado disponible el cual es un estado por defecto.</li> </ul>	

Tabla N° 11: Historia de Usuario N° 4

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 04, Registrar Curso.
<b>Nombre tarea:</b> Implementar ventana para registrar curso	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 3.5</b>
<b>Fecha inicio:</b> 28/12/2014	<b>Fecha fin:</b> 31/12/2014
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz.	
<b>Descripción:</b> Se diseñará y programará una ventana que permita registrar un curso, la ventana contará con dos pestañas las cuales serán "Últimos cursos" y "Registrar curso", la primera pestaña permitirá listar todos los cursos registrados por el administrador y la segunda pestaña para registrar el curso, para registrar un curso se pedirá sus respectivos datos y se tendrá en cuenta la validación de que si no existen docentes registrados, el registro no se podrá realizar.	

Tabla N° 12: Tarea Implementar ventana para registrar curso

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 04, Registrar Curso.
<b>Nombre tarea:</b> Pruebas en ventana para registrar curso	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 01/01/2015	<b>Fecha fin:</b> 02/01/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se harán las pruebas correspondientes al ingreso de datos del curso y a su registro, si hay presencia de errores en el registro de datos, se harán las correcciones correspondientes.	

Tabla N° 13: Tarea Pruebas en ventana para registrar curso

### 5.2.2. ITERACIÓN 2

Consta de 5 Historia de Usuario:

- ❖ **Editar Usuario:** 2 puntos – 3 tareas
- ❖ **Editar Curso:** 2 puntos – 3 tareas
- ❖ **Editar Perfil de Usuario:** 2 puntos – 2 tareas
- ❖ **Inscripción a Curso:** 4 puntos – 4 tareas
- ❖ **Matrícula de Estudiantes:** 4 puntos – 3 tareas

Historia de Usuario	
Número: 5	Usuario: Administrador de Web
Nombre historia: Editar Usuario	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 2	Iteración asignada: 2
Programador responsable: Victor Iván Requejo Delgado	
<b>Descripción:</b>  El administrador, lista los docentes y estudiantes registrados, cada uno muestra en la lista sus datos principales y tienen a un costado la opción "Editar", al hacerle click, cada dato del usuario se carga en su respectivo control de la nueva ventana, si el administrador lo desea puede modificar los datos del usuario que crea conveniente.	
<b>Observaciones:</b>  ➤ El nombre de usuario es el nombre de bienvenida del usuario al acceder al sistema. ➤ La modificación de contraseña del usuario es opcional, si el administrador llena el campo contraseña, ésta reemplazará a la existente, si no llena este campo, la contraseña será la misma que se creó en el registro inicial.	

Tabla N° 14: Historia de Usuario N° 5



Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 05, Editar Usuario
<b>Nombre tarea:</b> Implementar ventana para listar usuarios	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 03/01/2015	<b>Fecha fin:</b> 05/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar usuarios, la ventana mostrará los principales datos de los usuarios registrados y contará con la opción "Editar" para abrir una nueva ventana donde se ingrese los datos nuevos del usuario a modificar, también contará con una opción para buscar al usuario según sea necesario.	

Tabla N° 15: Tarea Implementar ventana para listar usuarios

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 05, Editar Usuario
<b>Nombre tarea:</b> Implementar ventana para editar usuario	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 06/01/2015	<b>Fecha fin:</b> 08/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita editar un usuario, pedirá sus datos personales y se tendrá en cuenta la validación de correo electrónico como dato único para cada usuario.	

Tabla N° 16: Tarea Implementar ventana para editar usuario

<b>Tarea</b>	
<b>Número de tarea: 3</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 05, Editar Usuario
<b>Nombre tarea:</b> Pruebas en ventana para editar usuario	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 09/01/2015	<b>Fecha fin:</b> 10/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se harán las pruebas correspondientes al ingreso de datos del usuario y a guardar los cambios, si hay presencia de errores al guardar los cambios, se harán las correcciones correspondientes.	

Tabla N° 17: Tarea Pruebas en ventana para editar usuario

<b>Historia de Usuario</b>	
<b>Número: 6</b>	<b>Usuario: Administrador de Web</b>
<b>Nombre historia:</b> Editar Curso	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  El administrador, lista los cursos registrados, la lista muestra los cursos disponibles y finalizados, cada curso muestra en la lista sus datos principales y tienen a un costado la opción "Editar", al hacerle click, cada dato del curso se carga en su respectivo control de la nueva ventana, si el administrador lo desea puede modificar los datos del curso que crea conveniente.	
<b>Observaciones:</b>  <ul style="list-style-type: none"> <li>➤ Cada curso presenta un estado "Disponible" y "No disponible", si el estado es "No disponible", la ventana mostrará en la lista como curso finalizado.</li> <li>➤ La ventana presenta una opción llamada "Eliminar Curso" para utilizarla según sea necesario.</li> </ul>	

Tabla N° 18: Historia de Usuario N° 6

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 06, Editar Curso
<b>Nombre tarea:</b> Implementar ventana para listar cursos	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 11/01/2015	<b>Fecha fin:</b> 13/01/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar cursos, la ventana mostrará los principales datos de los cursos registrados y contará con la opción "Editar" para abrir una nueva ventana donde se ingrese los datos nuevos del curso a modificar.	

Tabla N° 19: Tarea Implementar ventana para listar cursos

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 06, Editar Curso
<b>Nombre tarea:</b> Implementar ventana para editar curso	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 14/01/2015	<b>Fecha fin:</b> 16/01/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita editar un curso, pedirá los datos correspondientes a modificar y se tendrá en cuenta la validación de que la fecha de inicio no puede ser mayor a la fecha de fin del curso y viceversa, La ventana también contará con una opción llamada "Eliminar Curso" que se utilizará cuando sea necesario.	

Tabla N° 20: Tarea Implementar ventana para editar curso

Tarea	
Número de tarea: 3	Historia de Usuario ( Nro. y Nombre ): Historia 06, Editar Curso
Nombre tarea: Pruebas en ventana para editar curso	
Tipo de tarea : Corrección	Puntos estimados: 0.5
Fecha inicio: 17/01/2015	Fecha fin: 18/01/2015
Programador responsable: Luis Jonatan Severino Sáenz	
Descripción: Se harán las pruebas correspondientes al ingreso de datos del curso y a guardar los cambios, si hay presencia de errores al guardar los cambios, se harán las correcciones correspondientes.	

Tabla N° 21: Tarea Pruebas en ventana para editar curso

Historia de Usuario	
Número: 7	Usuario: Administrador de Web, Docente y Estudiante
Nombre historia: Editar Perfil de Usuario	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 2	Iteración asignada: 2
Programador responsable: Victor Iván Requejo Delgado	
<p><b>Descripción:</b></p> <p>El administrador, docente y estudiante pueden modificar sus datos personales si lo desean, para ello tienen que haber iniciado sesión y en la pestaña "Ajustes", ubicar la opción "Editar Perfil", al hacerle click, cada dato del usuario se carga en su respectivo control de la nueva ventana, ingresará los nuevos datos que desea modificar y guardará los cambios de perfil.</p>	
<p><b>Observaciones:</b></p> <ul style="list-style-type: none"> <li>➤ El correo electrónico debe ser único para cada usuario, no debe repetirse con algún correo registrado por un anterior usuario.</li> <li>➤ El nombre de usuario es el nombre de bienvenida del usuario al acceder al sistema.</li> </ul>	

Tabla N° 22: Historia de Usuario N° 7

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 07, Editar Perfil de Usuario
<b>Nombre tarea:</b> Implementar ventana para editar perfil de usuario	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1.5</b>
<b>Fecha inicio:</b> 19/01/2015	<b>Fecha fin:</b> 21/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita editar el perfil de un usuario, pedirá sus datos personales y se tendrá en cuenta la validación de correo electrónico como dato único para cada usuario.	

Tabla N° 23: Tarea Implementar ventana para editar perfil de usuario

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 07, Editar Perfil de Usuario
<b>Nombre tarea:</b> Pruebas en ventana para editar perfil de usuario	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 22/01/2015	<b>Fecha fin:</b> 23/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se harán las pruebas correspondientes al ingreso de datos personales del usuario y a guardar los cambios, si hay presencia de errores al guardar los cambios, se harán las correcciones correspondientes.	

Tabla N° 24: Tarea Pruebas en ventana para editar perfil de usuario

Historia de Usuario	
<b>Número: 8</b>	<b>Usuario: Estudiante</b>
<b>Nombre historia:</b> Inscripción a Curso	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b>  El estudiante, al acceder al sistema, lista todos los cursos disponibles en los cuales se puede inscribir según corresponda a su matrícula, elige un curso con un click sobre su nombre, cada curso muestra su descripción así como su fecha de inicio y fin. Para inscribirse al curso deberá de hacer click en la opción "Inscribirme" que se ubica al costado del curso seleccionado.	
<b>Observaciones:</b>  <ul style="list-style-type: none"> <li>➤ Haber verificado que los cursos en los cuales se va a inscribir correspondan a los cursos que se matriculó en el Sistema de Matrícula de la Universidad.</li> <li>➤ Luego de haberse inscrito, la inscripción queda como pendiente para ser evaluada por el docente encargado de dictarlo y previa verificación de estudiantes matriculados aceptará al estudiante a su curso.</li> </ul>	

Tabla N° 25: Historia de Usuario N° 8

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 08, Inscripción a Curso
<b>Nombre tarea:</b> Implementar ventana para listar cursos	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 24/01/2015	<b>Fecha fin:</b> 25/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita mostrar la lista de todos los cursos registrados por el administrador, cada curso estará acompañado de su descripción y fechas de inicio y de fin del curso. Cada nombre de curso es un enlace a la ventana de inscripción del estudiante.	

Tabla N° 26: Tarea Implementar ventana para listar cursos

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 08, Inscripción a Curso
<b>Nombre tarea:</b> Implementar ventana para inscripción a curso	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 26/01/2015	<b>Fecha fin:</b> 28/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita Inscribirse en un curso, la ventana contará con dos pestañas las cuales serán "Detalles" y "Estudiantes matriculados", la primera pestaña muestra el nombre, descripción, fecha de inicio y fin del curso que se eligió para inscribirse y la segunda pestaña para mostrar los estudiantes matriculados en el curso, para inscribirse se hará click en la opción "Inscribirme" que se ubica al costado del curso elegido.	

Tabla N° 27: Tarea Implementar ventana para inscripción a curso

Tarea	
<b>Número de tarea: 3</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 08, Inscripción a Curso
<b>Nombre tarea:</b> Implementar ventana para listar estudiantes matriculados	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 29/01/2015	<b>Fecha fin:</b> 30/01/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará la ventana que permita listar los estudiantes matriculados, en la pestaña "Estudiantes matriculados", se listará todos los estudiantes matriculados en el curso seleccionado mostrando sus datos personales. El estudiante se mostrará en la lista de los estudiantes matriculados después de que el docente apruebe su inscripción en el curso.	

Tabla N° 28: Tarea Implementar ventana para listar estudiantes matriculados

Tarea	
<b>Número de tarea: 4</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 08, Inscripción a Curso
<b>Nombre tarea:</b> Pruebas en ventana para inscripción a curso	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 31/02/2015	<b>Fecha fin:</b> 01/02/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se harán las pruebas correspondientes para la inscripción del estudiante a un curso, si hay presencia de errores al momento de la inscripción, se harán las correcciones correspondientes.	

Tabla N° 29: Tarea Pruebas en ventana para inscripción a curso



Historia de Usuario	
<b>Número: 9</b>	<b>Usuario: Docente</b>
<b>Nombre historia:</b> Matricula de Estudiantes	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  El docente, lista todos los cursos disponibles y finalizados que están bajo su cargo para dictar., selecciona el curso haciendo click sobre su nombre, en la pestaña "Detalles" muestra su descripción y fechas de inicio y fin, en la pestaña "Estudiantes" se muestra todos los estudiantes que se inscribieron al curso seleccionado. La matrícula se hace al hacer click en la opción "Aprobar" que se muestra al costado de cada estudiante inscrito.	
<b>Observaciones:</b>  <ul style="list-style-type: none"> <li>➤ Los estudiantes han tenido que haberse inscrito en los cursos.</li> <li>➤ El docente, antes de matricular al estudiante al curso, verifica que estén matriculados en el Sistema de Matrículas de la Universidad</li> <li>➤ Al agregar al estudiante al curso, éste dejará de aparecer en la lista de inscripciones pendientes y formara parte de los estudiantes matriculados.</li> </ul>	

Tabla N° 30: Historia de Usuario N° 9

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 09, Matrícula de Estudiantes
<b>Nombre tarea:</b> Implementar ventana para listar cursos de docente	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1.5</b>
<b>Fecha inicio:</b> 02/02/2015	<b>Fecha fin:</b> 03/02/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar los cursos de docente, la ventana mostrará la lista de todos los cursos que han sido asignados al docente, cada curso estará acompañado de su descripción y sus fechas de inicio y fin. Cada nombre de curso es un enlace a la nueva ventana que contará con las pestañas "Detalles", "Estudiantes" y "Actividades".	

Tabla N° 31: Tarea Implementar ventana para listar cursos de docente

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 09, Matrícula de Estudiantes
<b>Nombre tarea:</b> Implementar ventana para matricula de estudiantes	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 04/02/2015	<b>Fecha fin:</b> 07/02/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita matricular a los estudiantes al curso seleccionado, en la pestaña "Estudiantes" la ventana contará con dos listas, una para mostrar los estudiantes matriculados y la otra lista para mostrar los estudiantes con inscripciones pendientes. Para matricular un estudiante, el docente se fijará en la lista de inscripciones pendientes, allí están todos los estudiantes mostrando sus datos personales, el docente hará click en la opción "Aprobar" que se ubica al costado de cada estudiante.	

Tabla N° 32: Tarea Implementar ventana para matricula de estudiantes

Tarea	
<b>Número de tarea: 3</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 09, Matrícula de Estudiantes
<b>Nombre tarea:</b> Pruebas en ventana para matricula de estudiantes	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 08/02/2015	<b>Fecha fin:</b> 09/02/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se harán las pruebas correspondientes para la matrícula de los estudiantes a un curso, si hay presencia de errores al momento de la matrícula, se harán las correcciones correspondientes.	

Tabla N° 33: Tarea Pruebas en ventana para matricula de estudiantes

### 5.2.3. ITERACIÓN 3

Consta de 3 Historia de Usuario:

- ❖ **Registrar Actividades:** 3 puntos – 2 tareas
- ❖ **Desarrollar Actividades:** 4 puntos – 4 tareas
- ❖ **Revisar Actividades:** 4 puntos – 5 tareas

Historia de Usuario	
Número: 10	Usuario: Docente
Nombre historia: Registrar Actividades	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Programador responsable: Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  El docente, de la lista de cursos disponibles que están a su cargo para dictar, ha elegido uno de ellos, ingresa a la pestaña "Actividades" y hace click en la opción "Nueva actividad" con la cual abre la ventana correspondiente al registro de una actividad, ingresa las instrucciones, selecciona el tipo, adjunta archivos de ser necesario y registra la actividad dando click a la opción "Guardar".	
<b>Observaciones:</b>  ➤ La actividad tiene fecha de inicio y fecha de fin que se refiere que hasta ese día la actividad será válida. Pasada esa fecha el docente puede revisarlo si lo desea pero con otro criterio de evaluación.	

Tabla N° 34: Historia de Usuario N° 10

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 10, Registrar Actividades
<b>Nombre tarea:</b> Implementar ventana para registrar actividades	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 2.5</b>
<b>Fecha inicio:</b> 10/02/2015	<b>Fecha fin:</b> 12/02/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita registrar actividades al curso seleccionado, en la pestaña "Actividades", la ventana contará con la opción "Nueva actividad" con la cual abre la ventana correspondiente al registro de una actividad, selecciona el tipo de actividad que desea agregar (tarea, practica, examen), si hay archivos adjuntos por enviar junto a la actividad se agrega dando click en el botón que se ubica al costado. Se ingresa el resto de datos que requiere la actividad y se procede a guardarla.	

Tabla N° 35: Tarea Implementar ventana para registrar actividades

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 10, Registrar Actividades
<b>Nombre tarea:</b> Pruebas en ventana para registrar actividades	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 13/02/2015	<b>Fecha fin:</b> 14/02/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se harán las pruebas correspondientes al ingreso de datos de la actividad y a su registro, si hay presencia de errores en el registro de datos, se harán las correcciones correspondientes.	

Tabla N° 36: Tarea Pruebas en ventana para registrar actividades

Historia de Usuario	
<b>Número: 11</b>	<b>Usuario: Estudiante</b>
<b>Nombre historia:</b> Desarrollar Actividades	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b>  El estudiante, ingresa a la pestaña "Actividades" y lista todas las actividades registradas por los docentes para cada curso. Para desarrollar cada actividad, el estudiante debe hacer click sobre el nombre de la actividad, cada actividad muestra sus instrucciones y los archivos adjuntos en el caso se hayan agregado. Se desarrolla la actividad y la solución se envía haciendo click en la opción "Enviar solución", en la ventana de enviar solución se adjunta el archivo, se escribe un mensaje y se envía al docente el cual revisará y pondrá la nota correspondiente.	
<b>Observaciones:</b>  ➤ La actividad tiene fecha de inicio y fecha de fin que se refiere que hasta ese día la actividad será válida. Pasada esa fecha el docente puede revisarlo si lo desea pero con otro criterio de evaluación.	

Tabla N° 37: Historia de Usuario N° 11

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 11, Desarrollar Actividades
<b>Nombre tarea:</b> Implementar ventana para listar actividades de estudiante	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 15/02/2015	<b>Fecha fin:</b> 16/02/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar todas las actividades registradas por los docentes para cada curso, cada actividad estará acompañada del nombre del curso al cual pertenece, vigencia y estado, el estado de la actividad es "Activo" ya que aún no se ha enviado su solución al docente. Cada nombre de la actividad será un enlace que abrirá una nueva ventana en la cual mostrará detalles de la actividad y tendrá la opción para enviar su desarrollo.	

Tabla N° 38: Tarea Implementar ventana para listar actividades de estudiante

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 11, Desarrollar Actividades
<b>Nombre tarea:</b> Implementar ventana para desarrollar actividad	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 17/02/2015	<b>Fecha fin:</b> 18/02/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita desarrollar actividades, la ventana mostrará el nombre de la actividad seleccionada, las instrucciones de desarrollo y los archivos que el docente adjunte a la actividad. La ventana contará con la opción "Enviar solución", que abrirá una nueva ventana para el envío correspondiente de la actividad desarrollada.	

Tabla N° 39: Tarea Implementar ventana para desarrollar actividad

Tarea	
<b>Número de tarea: 3</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 11, Desarrollar Actividades
<b>Nombre tarea:</b> Implementar ventana para enviar solución	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1.5</b>
<b>Fecha inicio:</b> 19/02/2015	<b>Fecha fin:</b> 21/02/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita enviar la solución de la actividad seleccionada, la ventana contará con un control en el cual el estudiante escribirá un mensaje al docente, también contará con una opción con la que se adjuntará los archivos de desarrollo y la opción "Enviar ", para enviar la solución al docente.	

Tabla N° 40: Tarea Implementar ventana para enviar solución

Tarea	
<b>Número de tarea: 4</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 11, Desarrollar Actividades
<b>Nombre tarea:</b> Pruebas en ventana para enviar solución	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 22/02/2015	<b>Fecha fin:</b> 23/02/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se harán las pruebas correspondientes para enviar la solución de la actividad al docente, si hay presencia de errores al momento del envío, se harán las correcciones correspondientes.	

Tabla N° 41: Tarea Pruebas en ventana para enviar solución



Historia de Usuario	
<b>Número: 12</b>	<b>Usuario: Docente</b>
<b>Nombre historia:</b> Revisar Actividades	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  El Docente, para revisar una actividad hace click en el botón "Revisar actividad", que abre la ventana en donde listan todos los estudiantes que hasta ese momento han enviado su solución, cada estudiante al costado tiene la opción "Ver y calificar", que abre otra ventana donde muestra el mensaje y el archivo adjunto que envió el estudiante y que será descargado por el docente para la revisión respectiva, la ventana tiene un campo para ingresar la nota que será registrada según el criterio del docente.	
<b>Observaciones:</b>  <ul style="list-style-type: none"> <li>➤ Luego de haber finalizado la fecha máxima para enviar las actividades, el docente finaliza la actividad pero puede revisarlo si lo desea asumiendo otro criterio de evaluación.</li> <li>➤ El docente puede eliminar la actividad si lo desea.</li> <li>➤ El estado de la actividad es "Pendiente" si aún no lo ha revisado, "Finalizado" si fue cancelada y "Revisado" si es que ya puso nota por la actividad.</li> </ul>	

Tabla N° 42: Historia de Usuario N° 12

Tarea	
<b>Número de tarea: 1</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 12, Revisar Actividades
<b>Nombre tarea:</b> Implementar ventana para listar actividades de curso	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 24/02/2015	<b>Fecha fin:</b> 25/02/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar todas las actividades del curso seleccionado, cada actividad mostrará sus detalles como son: tipo, vigencia y estado. Cada nombre de la actividad será un enlace que abrirá una nueva ventana en la cual se mostrará detalles de la actividad y tendrá la opción para revisar la actividad.	

Tabla N° 43: Tarea Implementar ventana para listar actividades de curso

Tarea	
<b>Número de tarea: 2</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 12; Revisar Actividades
<b>Nombre tarea:</b> Implementar ventana para mantenimiento de actividades	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1.5</b>
<b>Fecha inicio:</b> 26/03/2015	<b>Fecha fin:</b> 01/03/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita el mantenimiento de actividades, la ventana mostrará el nombre de la actividad seleccionada, las instrucciones de desarrollo y los archivos que el docente adjuntó a la actividad. La ventana contará con las opciones "Finalizar tarea", "Revisar actividad" y "Eliminar actividad" para ser usadas por el docente cuando sea necesario.	

Tabla N° 44: Tarea Implementar ventana para mantenimiento de actividades

Tarea	
<b>Número de tarea: 3</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 12, Revisar Actividades
<b>Nombre tarea:</b> Implementar ventana para listar estudiantes que enviaron actividad	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 02/03/2015	<b>Fecha fin:</b> 03/03/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar los estudiantes que enviaron la actividad, la ventana mostrará el nombre de la actividad y el listado de los estudiantes que enviaron la solución hasta ese momento, la ventana mostrará la fecha de envío y tendrá a un costado la opción "Ver y calificar", que abrirá una nueva ventana para la respectiva calificación del docente.	

Tabla N° 45: Tarea Implementar ventana para listar estudiantes que enviaron actividad

Tarea	
<b>Número de tarea: 4</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 12, Revisar Actividades
<b>Nombre tarea:</b> Implementar ventana para calificar actividad	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 04/03/2015	<b>Fecha fin:</b> 06/03/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se diseñará y programará una ventana que permita calificar la actividad, la ventana mostrará el mensaje enviado por el estudiante y el archivo adjunto del desarrollo de la actividad, el cual será descargado por el docente para la revisión respectiva, la ventana contará con un control para ingresar la nota según el criterio del docente.	

Tabla N° 46: Tarea Implementar ventana para calificar actividad

Tarea	
<b>Número de tarea: 5</b>	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 12, Revisar Actividades
<b>Nombre tarea:</b> Pruebas en ventana para calificar actividad	
<b>Tipo de tarea :</b> Corrección	<b>Puntos estimados: 0.5</b>
<b>Fecha inicio:</b> 07/03/2015	<b>Fecha fin:</b> 09/03/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b> Se harán las pruebas correspondientes para calificar la actividad, si hay presencia de errores al momento de calificar, se harán las correcciones correspondientes.	

Tabla N° 47: Tarea Pruebas en ventana para calificar actividad

#### 5.2.4. ITERACIÓN 4

Consta de 2 Historia de Usuario:

❖ **Detalle de Actividades Enviadas y Pendientes:**

0.5 puntos – 1 tarea

❖ **Reporte de Aprobados por Curso y Actividad:**

2 puntos – 1 tarea

Historia de Usuario	
Número: 13	Usuario: Estudiante
Nombre historia: Detalle de Actividades Enviadas y Pendientes	
Prioridad en negocio: Baja	Riesgo en desarrollo: Bajo
Puntos estimados: 0.5	Iteración asignada: 4
Programador responsable: Victor Iván Requejo Delgado	
<b>Descripción:</b>  El estudiante, ingresa a la pestaña "Actividades" y lista todas las actividades registradas de los respectivos cursos que se matriculó, las fechas de vigencia por cada actividad y su estado de cada una.	
<b>Observaciones:</b>  ➤ Finalizada la fecha máxima para enviar las actividades, el docente cancela la actividad pero puede revisarlo si lo desea.  ➤ El docente luego de la revisión respectiva de la actividad enviada por el estudiante, procede a evaluar la actividad y coloca la nota correspondiente.	

Tabla N° 48: Historia de Usuario N° 13

Tarea	
<b>Número de tarea:</b> 1	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 13, Detalle de actividades enviadas y pendientes
<b>Nombre tarea:</b> Implementar ventana para listar actividades enviadas y pendientes	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 10/03/2015	<b>Fecha fin:</b> 11/03/2015
<b>Programador responsable:</b> Victor Iván Requejo Delgado	
<b>Descripción:</b> Se diseñará y programará una ventana que permita listar todas las actividades registradas por los docentes para cada curso, cada actividad estará acompañada del nombre del curso al cual pertenece, vigencia y estado, el estado de la actividad es "Activo" si el estudiante ya tiene nota o si aún no lo ha enviado, "Finalizado" si fue cancelada o está fuera de la vigencia, si la actividad ya ha sido calificada, mostrará la nota correspondiente.	

Tabla N° 49: Tarea Implementar ventana para listar actividades enviadas y pendientes

Historia de Usuario	
<b>Número: 14</b>	<b>Usuario: Docente</b>
<b>Nombre historia:</b> Reporte de Aprobados por Curso y Actividad	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<b>Descripción:</b>  El Docente, selecciona un curso y muestra los estudiantes matriculados, selecciona una actividad y muestra las notas de los estudiantes aprobados y desaprobados que ha revisado hasta ese momento.	
<b>Observaciones:</b>  ➤ Si el docente no tiene actividades registradas para algún curso, la lista muestra vacía. ➤ Permite exportar a Excel los nombres de los estudiantes con su respectiva nota de actividad.	

Tabla N° 50: Historia de Usuario N° 14

Tarea	
<b>Número de tarea:</b> 1	<b>Historia de Usuario ( Nro. y Nombre ):</b> Historia 14, Reporte de aprobados por curso y actividad
<b>Nombre tarea:</b> Implementar ventana para listar aprobados por curso y actividad	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 12/03/2015	<b>Fecha fin:</b> 14/03/2015
<b>Programador responsable:</b> Luis Jonatan Severino Sáenz	
<p><b>Descripción:</b> Se diseñará y programará una ventana que permita listar los estudiantes aprobados por curso y actividad, la ventana contará con el combo "Curso", que mostrará todos los cursos que el docente tiene a su cargo para dictar, al elegir un curso mostrará la lista de estudiantes matriculados, también contará con el combo "Actividad" que mostrará todas las actividades registradas por cada curso, al elegir una actividad se mostrará las notas de los estudiantes aprobados y desaprobados que el docente ha revisado hasta ese momento.</p> <p>La ventana muestra un gráfico estadístico que indica que el color celeste es para los estudiantes que están aprobados, el color rojo para los estudiantes desaprobados y el color anaranjado es para los estudiantes que aún no se les ha revisado su actividad y están sin nota.</p> <p>La ventara también contará con una opción para exportar a Excel los nombres de los estudiantes con su respectiva nota de actividad.</p>	

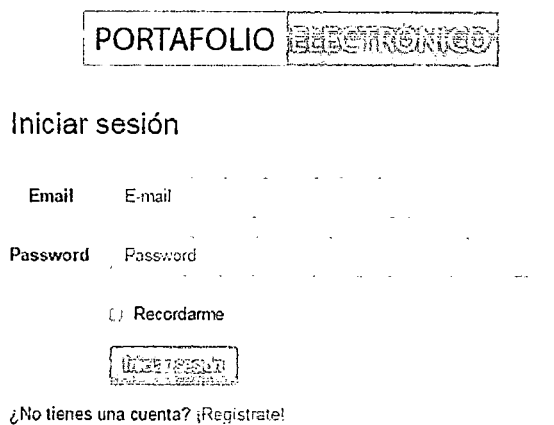
Tabla N° 51: Tarea Implementar ventana para listar aprobados por curso y actividad



### 5.3. DISEÑO

#### 5.3.1. DISEÑO ITERACIÓN 1

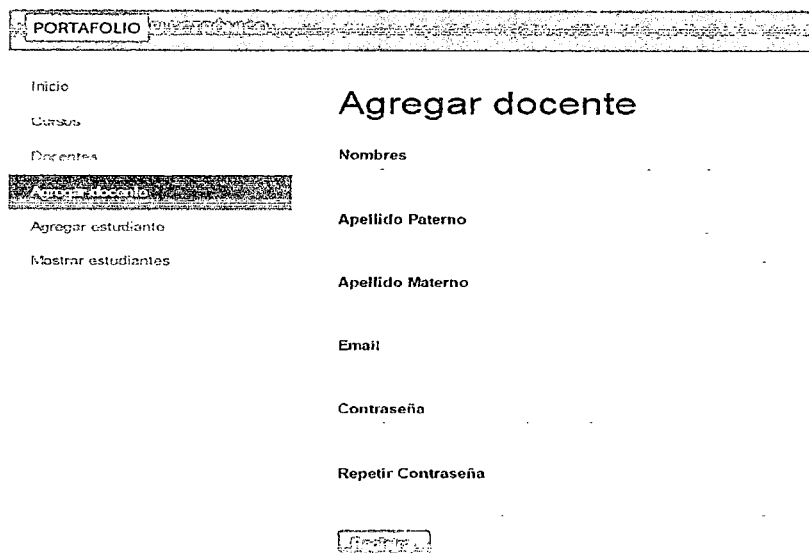
##### ❖ INICIAR SESIÓN



The screenshot shows the login interface for 'PORTAFOLIO ELECTRONICO'. It features a header with the site name. Below it, the text 'Iniciar sesión' is centered. There are two input fields: 'Email' (with a sub-label 'E-mail') and 'Password'. A checkbox labeled 'Recordarme' is positioned below the password field. A 'LOGUEARSE' button is centered below the checkbox. At the bottom, a link reads '¿No tienes una cuenta? ¡Regístrate!'.

Figura N° 3: Ventana Iniciar Sesión

##### ❖ REGISTRAR DOCENTE



The screenshot displays the registration page for teachers. The header 'PORTAFOLIO ELECTRONICO' is at the top. On the left, a sidebar menu includes links for 'Inicio', 'Cursos', 'Docentes', 'Agregar docente' (which is highlighted), 'Agregar estudiante', and 'Mostrar estudiantes'. The main content area is titled 'Agregar docente'. It contains several form fields: 'Nombres', 'Apellido Paterno', 'Apellido Materno', 'Email', 'Contraseña', and 'Repetir Contraseña'. A 'REGISTRAR' button is located at the bottom of the form.

Figura N° 4: Ventana Registrar Docente

## ❖ REGISTRAR ESTUDIANTE

PORTAFOLIO REGISTRACION	
Inicio	<h3>Agregar estudiante</h3> <p><b>Nombres</b></p> <p><b>Apellido Paterno</b></p> <p><b>Apellido Materno</b></p> <p><b>Email</b></p> <p><b>Contraseña</b></p> <p><b>Repetir Contraseña</b></p> <p><input type="button" value="Registrar"/></p>
Cursos	
Docentes	
Agregar docente	
<b>Agregar estudiante</b>	
Mostrar estudiantes	

Figura N° 5: Ventana Registrar Estudiante

## ❖ REGISTRAR CURSO

PORTAFOLIO REGISTRACION	
Inicio	Inicio Cursos
<b>Cursos</b>	<h3>Cursos</h3>
Docentes	<p>Ultimos cursos    <b>Abrir curso</b></p> <p><a href="#">Ingresar detalles del curso...</a></p>
Agregar docente	<p><b>Nombre</b></p> <p>Nombre del curso</p>
Agregar estudiante	<p><b>Descripción</b></p> <p>Detalles del curso...</p>
Mostrar estudiantes	<p><b>Profesor:</b> Carlos Velazco    <b>N° Alumnos</b></p> <p><b>Fecha de inicio:</b> 2015/04/16    <b>Fecha de fin:</b> 2015/04/16</p> <p><input type="button" value="Registrar"/></p>

Figura N° 6: Ventana Registrar Curso

## 5.3.2. DISEÑO ITERACIÓN 2

### ❖ EDITAR USUARIO

PORTAFOLIO				[ Luis Severino ]
Inicio	Estudiantes			Buscar alumnos
Cursos				
Docentes				
Agregar docente				
Agregar estudiante				
	Nombres	Apellidos	Email	Editar
	Victor Ivan	Requejo Delgado	vired301@hotmail.com	
	Marco	Polo Polo	marco@gmail.com	
	Luis	Severino Severino	luis@gmail.com	
	Omar	Panta Ramirez	omar@hotmail.com	
	David	Torres Castro	davidxd@gmail.com	
	Carla	Rivera I. Iolocho	carla@hotmail.com	
	Diego		ddiestras@gmail.com	
	Ana Perez		anaperez@hotmail.com	

Figura N° 7: Ventana Listar Usuarios

PORTAFOLIO ELECTRONICO	
Inicio	Editar estudiante
Cursos	
Docentes	
Agregar docente	
Agregar estudiante	
Mostrar estudiantes	
	<b>Nombre</b>
	Victor Ivan
	<b>Apellido Paterno</b>
	Requejo
	<b>Apellido Materno</b>
	Delgado
	<b>Nombre de usuario</b>
	vrequijo
	<b>Email</b>
	vired@hotmail.com
	<b>Contraseña</b>

Figura N° 8: Ventana Editar Usuario

## ✦ EDITAR CURSO

PORTAFOLIO ELECTRONICO						[Luis Severino]
Inicio	Cursos					
Cursos	Ultimos cursos	Agregar curso				
Docentes	Nombre	Profesor	Vigencia	Estado	Editar	
Agregar docente	Programacion III	Carlos Valdivia	2015-04-20 - 2015-08-16	En curso		
Agregar estudiante	Arquitectura de los Computadores	Luis Reyes	2015-04-20 - 2015-08-16	En curso		
Mostrar estudiantes	Redes II	Gilberto Carrón	2015-04-20 - 2015-08-16	En curso		
	Algoritmos y Estructura de Datos II	Carlos Valdivia	2015-04-20 - 2015-08-16	En curso		
	Programacion III	Franklin Teron	2015-04-20 - 2015-08-16	En curso		
	Programacion II	Franklin Teron	2015-04-20 - 2015-08-16	En curso		
	Ingeniería de Software	Denny Fuentes	2015-04-20 - 2015-08-16	En curso		
	Programación Avanzada	Censuelo Del Castillo	2015-04-20 - 2015-08-16	En curso		
	Redes I	Alex Serquén	2015-04-20 - 2015-08-16	En curso		
	Redes II	Jessie Bravo	2015-04-20 - 2015-08-16	Finalizado		

Figura N° 9: Ventana Listar Cursos

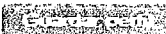
PORTAFOLIO ELECTRONICO	
Inicio Cursos Docentes Agregar docente Agregar estudiante Mostrar estudiantes	<h2>Editar Curso</h2> <p><b>Nombre</b> Programacion III</p> <p><b>Descripción</b>          El curso de Programación III forma parte del grupo de asignaturas pertenecientes a la categoría Científico-Profesional, tal como lo establece el Plan Curricular vigente. Los contenidos de la asignatura están orientados al desarrollo de aplicaciones Vwindows basadas en formularios haciendo uso de herramientas de software actuales, tales como Microsoft Visual Studio 2008 y Microsoft SQL Server 2005. La naturaleza del curso       </p> <p><b>N° Alumnos</b> 20</p> <p><b>Fecha de inicio</b>      <b>Fecha de fin</b>          2015-04-20                      2015-08-16       </p> <p><b>Estado</b>          Disponible       </p> <p>  <span>Eliminar Curso [X]</span> </p>

Figura N° 10: Ventana Editar Curso

❖ EDITAR PERFIL DE USUARIO

PORTAFOLIO

[Luis Severino]

Inicio

Cursos

Docentes

Agregar docente

Agregar estudiante

Modificar estudiantes

Editar perfil

Nombre

Luis Severino

Apellido Paterno

Apellido Materno

Nombre de usuario

Email

lj.severino@gmail.com

Guardar

Figura N° 11: Ventana Editar Perfil de Usuario

❖ INSCRIPCION A CURSO

PORTAFOLIO

[Victor Ivan Requejo]

Inscrip

Mis cursos

Actividades

Panel de control

Alumno : Victor Ivan Requejo Delgado

Bienvenido... @v-requejo

Cursos disponibles

Nombre	Descripción	Inicio	Finaliza
Programación III	El curso de Programación III forma parte del grupo de asignaturas pertenecientes a la categoría Científico-Profesional, tal como lo establece el Plan Curricular vigente. Los contenidos de la asignatura están orientados al desarrollo de aplicaciones Windows basadas en formularios haciendo uso de herramientas de software actuales, tales como Microsoft Visual Studio 2008 y Microsoft SQL Server 2005. La naturaleza del curso permitirá potenciar la capacidad creativa y lógica del estudiante, así como su destreza para resolver problemas de manera práctica y eficaz que proporcionen soluciones de software seguro y de calidad para ambientes mono y multi-usuario.	2015-03-20	2015-05-16
Redes I	El curso de Redes I pertenece a la línea de Redes Informáticas de la Escuela Profesional de Computación e Informática, cuyo objetivo fundamental es implementar el Diseño Físico y Lógico de Redes aprendiendo en el curso prerrequisito de Teleprocesos.	2015-03-20	2015-05-16
Programación Avanzada	El curso de Programación Avanzada es de naturaleza científica profesional, cuyo propósito general es que el estudiante desarrolle aplicaciones web, utilizando herramientas y tecnologías de programación óptimas. El curso está estructurado en 2 unidades temáticas organizadas de la siguiente manera: Unidad I: PHP Unidad II: ASP.Net	2015-03-20	2015-05-16

Figura N° 12: Ventana Listar Cursos

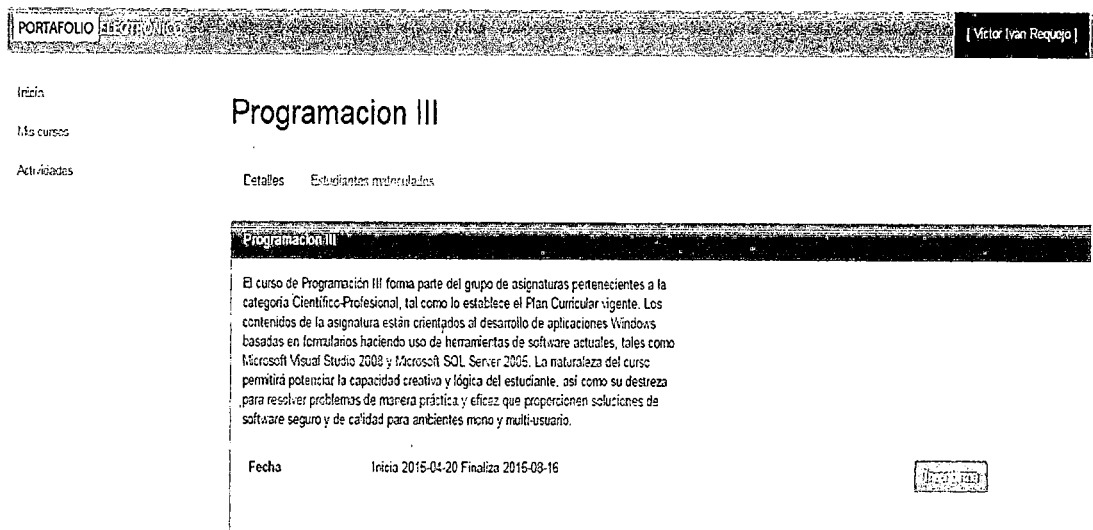


Figura Nº 13: Ventana Inscripción a Curso

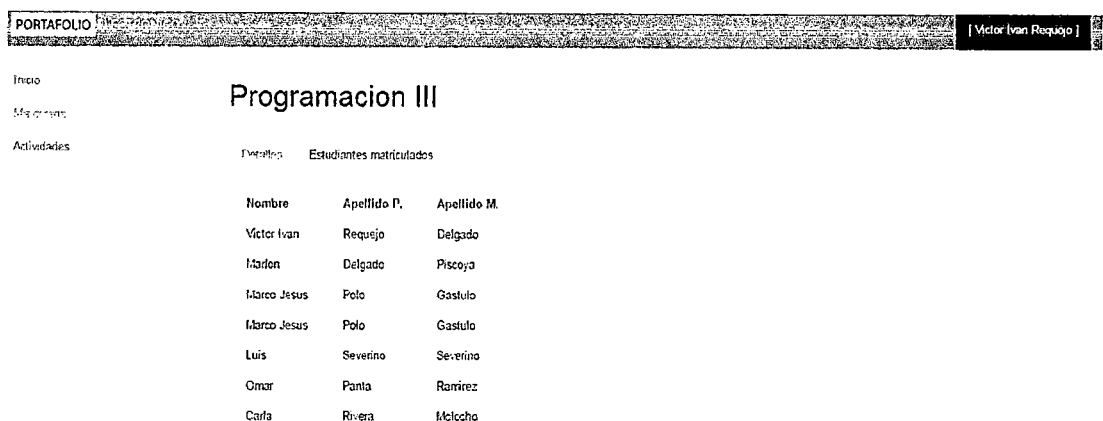


Figura Nº 14: Ventana Lista de estudiantes matriculados en un curso

## ❖ MATRICULA DE ESTUDIANTES

PORTAFOLIO				[ Carlos Valdivia ]
Inicio	Inicio	Cursos		
Estadísticas				
Cursos				
Últimos cursos				
Nombre	Descripción	Vigencia	Estado	
Programación III	El curso de Programación III forma parte del grupo de asigna...	2015-04-20 - 2015-08-16	En curso	
Algoritmos y Estructura de Datos II	En el presente curso se prepara al estudiante en las técnica...	2015-04-20 - 2015-08-16	En curso	
Diseño Asistido por Computadora	El diseño asistido por computadora es un proceso conocido po...	2015-04-20 - 2015-08-16	Finalizado	

Figura N° 15: Ventana Listar Cursos de Docente

PORTAFOLIO

Inicio Cursos Programación III

Inicio

Cursos

Estadísticas

Programación III

Detalles Estudiantes Actividades

Matriculados

Nombre	Apellido P.	Apellido M.	Fecha Aceptación
Victor Ivan	Requeje	Delgado	2015-04-16 21:49:48
Luis	Severino	Severino	2015-04-16 21:50:26

Inscripciones pendientes

Nombre	Apellido P.	Apellido M.	Fecha Inscripción	Aprobar
Marlon	Delgado	Piscocoy	2015-04-17 04:06:23	<input type="checkbox"/>
Marco	Polo	Polo	2015-04-17 04:06:56	<input type="checkbox"/>
Marco	Polo	Polo	2015-04-17 04:07:14	<input type="checkbox"/>
Omar	Panta	Ramirez	2015-04-17 04:08:34	<input type="checkbox"/>
Carla	Rivera	Moloch	2015-04-17 04:10:09	<input type="checkbox"/>

Figura N° 16: Ventana Matricula de Estudiantes

### 5.3.3. DISEÑO ITERACIÓN 3


#### ❖ REGISTRAR ACTIVIDADES


Nueva actividad

**Nombre**  
Registro de boletos de viaje en modo consola

**Instrucciones**  
Implementar registros (Structure) para:  
- Un boleto de viaje  
- Una lista de boletos  
**Condiciones:**  
- No puede existir boletos con número repetido.  
- Todos los datos son PRIVATE  
**Reportes:**  
- Listado de todos los boletos  
- Listado de boletos por destino  
- Buscar un boleto en particular por número  
\*Máximo 5000 Caracteres

**Tipo**  
Tarea

Agregar archivos adjuntos  
Ejercicio1\_Console.pptx 

**Fecha de inicio**  
2015/04/16 

**Fecha de fin**  
2015/04/16



 

Figura N° 17: Ventana Registrar Actividades

#### ❖ DESARROLLAR ACTIVIDADES

PORTAFOLIO

DESARROLLO

[ Victor Ivan Requejo ]

Inicio

Actividades

Mis cursos

Actividades

Todas las actividades

Actividad	Curso	Vigencia	Estado	Mi Nota
Registro de boletos de viaje en modo consola	Programacion III	2015-04-16 - 2015-04-20	Activo	
EXAMEN DE PROGRAMACIÓN III	Programacion III	2015-04-16 - 2015-04-16	Activo	
Técnicas de estimación de costos de software en el Perú	Ingeniería de Software	2015-04-17 - 2015-04-24	Activo	

Figura N° 18: Ventana Lista de Actividades de Estudiante



Inicio

Inicio Cursos Programación III Registro de boletos de viaje en modo consola

Los cursos

Actividades

## Registro de boletos de viaje en modo consola

Atención: Se revisará hasta la fecha máxima.



Implementar registros (Structure) para: - Un boleto de viaje - Una lista de boletos Condiciones: - No puede existir boletos con número repetido. - Todos los datos son PRIVATE Reports. - Listado de todos los boletos - Listado de boletos por destino - Buscar un boleto en particular por número

### Archivos adjuntos



Ejercicio Boletos de viaje

Figura N° 19: Ventana Desarrollar Actividades

Enviar solución

Mensaje:

Estimado Profesor, le envio mi trabajo para su revisión correspondiente, espero lo revise y me haga saber mi nota lo más antes posible, muchas gracias, que tenga un buen día.

Atte.

Victor Requejo

Máximo 5000 Caracteres.

Agregue archivos:

Ejercicio Boletos de viaje.rar

Choose file

Choose file

Cerrar

Enviar

Figura N° 20: Ventana Enviar Solución de Actividad

## ❖ REVISAR ACTIVIDADES

PORTAFOLIO

[Inicio](#)
[Cursos](#)
[Programación III](#)

---

[Inicio](#)
[Cursos](#)

# Programacion III

[Detalles](#)
[Estudiantes](#)
[Actividades](#)

Nombre	Tipo	Vigencia	Estado	Estadísticas
Registro de boletos de viaje en modo consola	tarea	2015-04-16 - 2015-04-20	Activo	
EXAMEN DE PROGRAMACIÓN III	examen	2015-04-16 - 2015-04-16	Activo	

**Figura N° 21: Ventana Lista de Actividades de Curso**

PORTAFOLIO

INTEGRANTE

[ Carlos Vaktiria ]

[Inicio](#)
[Cursos](#)
[Programación III](#)
[Registro de boletos de viaje en modo consola](#)

## Registro de boletos de viaje en modo consola

Atención: Se revisará hasta la fecha máxima.

PROYECTO

PRÁCTICA Nº 1

Implementar registros (Structure) para:

- Un boleto de viaje
- Una lista de boletos

Condiciones:

- No puede existir boletos con número repetido.
- Todos los datos son PRIVATE

Reportes:

- Listado de todos los boletos
- Listado de boletos por destino
- Buscar un boleto en particular por número

### Archivos adjuntos

[Boleto de Viaje.pdf](#)

[Eliminar actividad \[X\]](#)

**Figura N° 22: Ventana Mantenimiento de Actividades**

PORTAFOLIO

BOLETINES

[ Carlos Vaktiva ]

Inicio

Cursos

Programación II

Registro de boletos de viaje en modo consola

Revisar

Cursos

Estadísticas

Registro de boletos de viaje en modo consola

Estudiantes que enviaron hasta este momento...

Estudiante	Fecha	Estado	Revisar
Victor Ivan, Requejo Delgado	2015-04-17 01:19:11	Pendiente	<div></div>

Figura N° 23: Ventana Lista de Estudiantes que Enviaron Actividad

Victor Ivan Requejo

**Mensaje:**

Estimado Profesor, le envio mi trabajo para su revisión correspondiente, espero lo revise y me haga saber mi nota lo más antes posible. muchas gracias. que tenga un buen día.  
Atte: Víctor Requejo



Ejercicio Boletos de viaje

**Nota:** 0.00

Calificar Luego



Figura N° 24: Ventana Calificar Actividad

5.3.4. DISEÑO ITERACIÓN 4

❖ DETALLE DE ACTIVIDADES ENVIADAS Y PENDIENTES

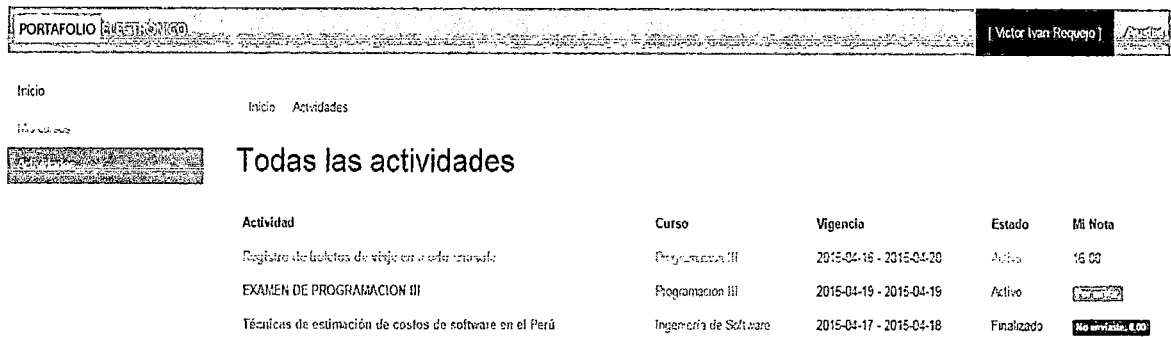


Figura N° 25: Ventana Detalle de Actividades Enviadas y Pendientes.

❖ REPORTE DE APROBADOS POR CURSO Y ACTIVIDAD

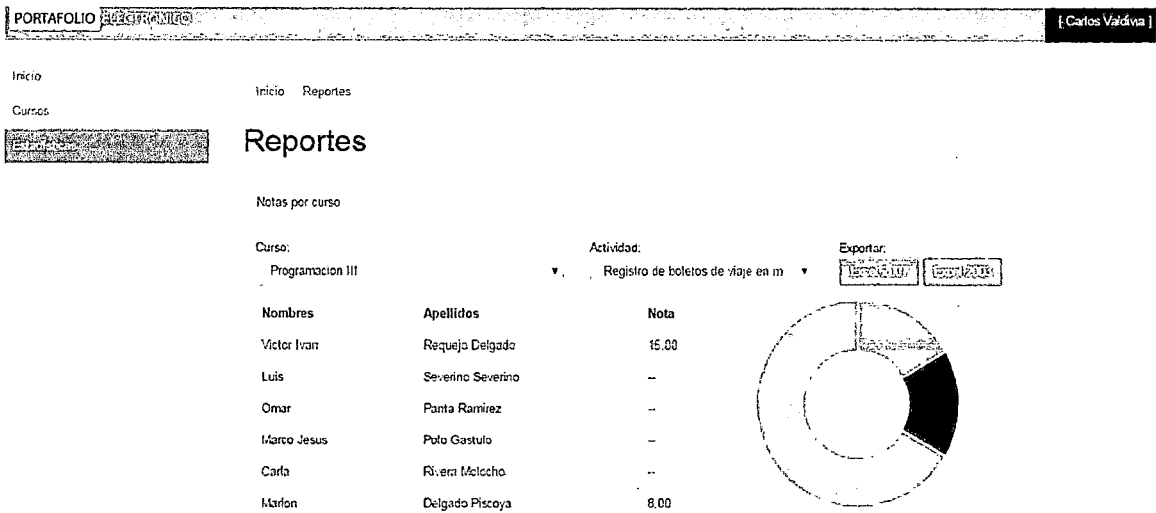


Figura N° 26: Ventana Reporte de Aprobados por Curso y Actividad

## 5.4. BASE DE DATOS COMPLETA

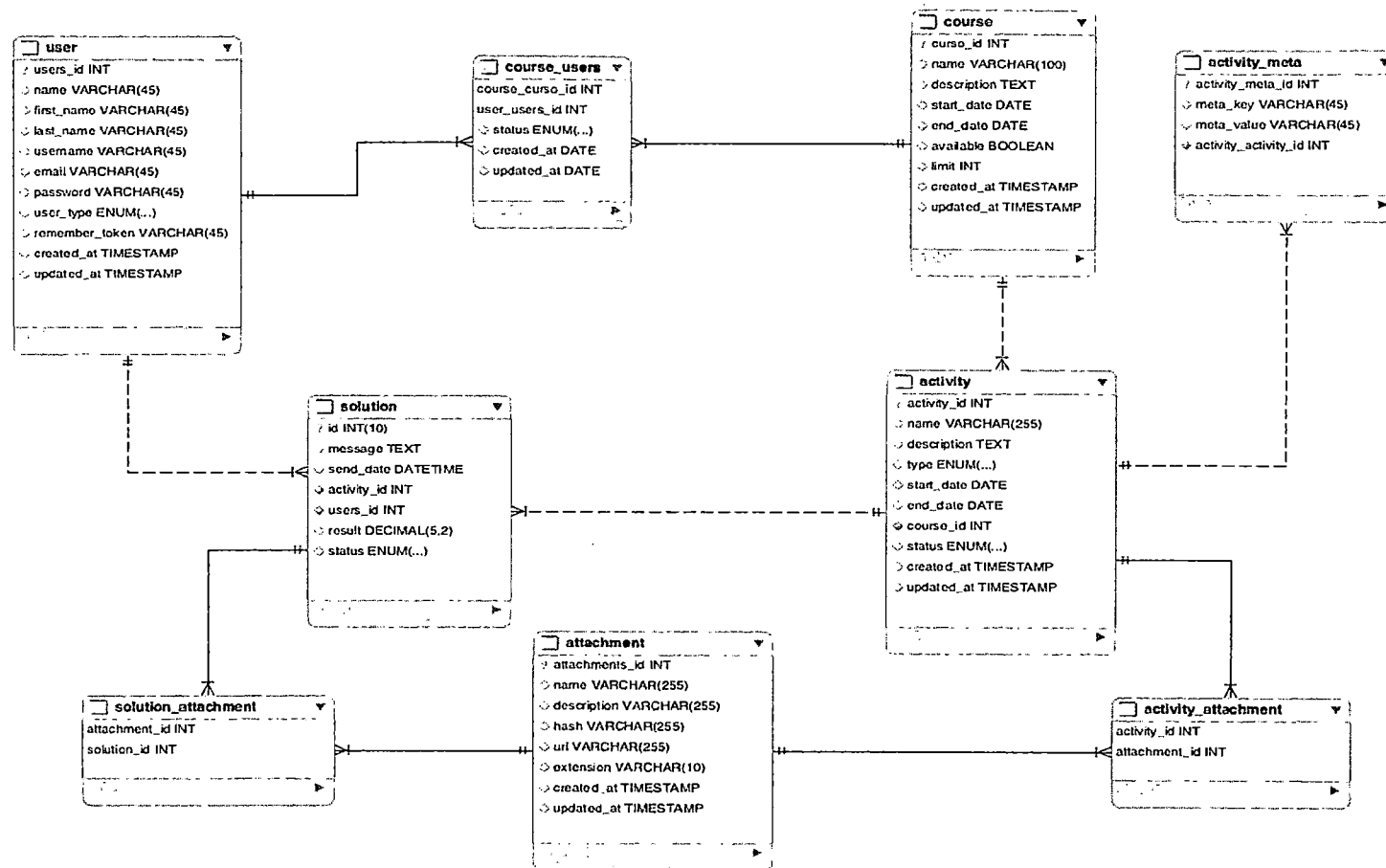


Figura N° 27: Base de Datos

# **CAPITULO VI**

## **COSTOS Y BENEFICIOS**

## VI. COSTOS Y BENEFICIOS

### 6.1. CRONOGRAMA DE ACTIVIDADES DEL PROYECTO

Cronograma de actividades	Octubre				Noviembre				Diciembre				Enero				Febrero				Marzo				Abril				Mayo				
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
Recopilación y estudio de material bibliográfico	■	■	■	■	■	■	■	■																									
Elaboración del esquema de trabajo de Investigación Científica.						■	■	■	■																								
Elaboración del proyecto de Tesis									■	■	■	■	■	■																			
Implementación del modelo										■	■	■	■	■	■	■	■	■	■	■	■												
Pruebas y análisis del modelo desarrollado																				■	■	■	■										
Evaluación y descripción de resultados																					■	■	■	■									
Elaboración del informe de tesis																						■	■	■									
Presentación del informe de tesis																									■								
Sustentación de tesis																											■						

Cuadro N° 1: Cronograma de actividades del proyecto

## 6.2. ANÁLISIS COSTO / BENEFICIO

### 6.2.1. ANALISIS DE COSTOS

#### ❖ COSTO DE UTILES DE OFICINA

ARTICULOS	PRECIO
Paquete de hojas bond X 4 meses	S/. 30.00
Recargas de cartuchos X 4 meses	S/. 30.00
Folders	S/. 10.00
Lapiceros	S/. 10.00
<b>TOTAL</b>	<b>S/. 80.00</b>

Cuadro N° 2: Costo de útiles de oficina

#### ❖ COSTO DE PERSONAL

ESPECIALISTAS	PRECIO
Analista	S/. 1,500.00
Diseñador	S/. 1,000.00
Programador	S/. 3,500.00
<b>TOTAL</b>	<b>S/. 6,000.00</b>

Cuadro N° 3: Costo de personal



❖ **COSTO DE SERVICIOS**

<b>SERVICIOS</b>	<b>PRECIO</b>
Luz X 4 meses	S/. 240.00
Internet X 4 meses	S/. 200.00
Teléfono X 4 meses	S/. 180.00
Movilidad X 4 meses	S/. 500.00
Alimentación X 4 meses	S/. 1,400.00
<b>TOTAL</b>	<b>S/. 2,520.00</b>

Cuadro N° 4: Costo de Servicios

**6.2.1.1. RESUMEN DE COSTO DE INVERSION**

<b>RESUMEN DE COSTOS DE INVERSIÓN</b>	
<b>Descripción</b>	<b>Sub Total (S/.)</b>
Útiles de oficina	S/. 80.00
Personal especial de desarrollo	S/. 6,000.00
Servicios	S/. 2,520.00
<b>TOTAL</b>	<b>S/. 8.600.00</b>

Cuadro N° 5: Resumen de costo de inversión de la implementación del Portafolio Electrónico y Aplicativo Android

### 6.2.1.2. COSTO DE OPERACIÓN

COSTO DE OPERACIÓN		
Descripción	Valor (S/.)	Estimaciones
Costo de servidor	S/. 480.00	Pago anual por concepto de servicios de dominio, hosting e instalación.  S/. 40 * 12 = 480
Mantenimiento y actualizaciones	S/. 1,000.00	Valor estimado anualmente o si el sistema lo requiere.
<b>TOTAL</b>	<b>S/. 1,480.00</b>	

Cuadro N° 6: Costo de operación de la implementación del Portafolio Electrónico y Aplicativo Android

### 6.2.2. ANÁLISIS DE BENEFICIOS

#### 6.2.2.1. Beneficios Tangibles

Descripción	Costo ahorrado
Ahorro en costo por la implementación del Portafolio Electrónico y Aplicativo Android por terceras empresas.	S/. 10,000.00
<b>TOTAL</b>	<b>S/. 10,000.00</b>

Cuadro N° 7: Beneficios tangibles de la implementación del Portafolio Electrónico y Aplicativo Android

#### 6.2.2.2. Beneficios Intangibles

BENEFICIOS INTANGIBLES
<ul style="list-style-type: none"><li>▪ Mayor integración de la información académica.</li><li>▪ Reducción de pérdida de información por presencia de virus y archivos maliciosos.</li><li>▪ Contar con información organizada, actualizada y confiable para disponer de ella en cualquier momento.</li><li>▪ Seguridad de la información, evitando la manipulación o modificación de los datos por personas indebidas y/o ajenas a la escuela profesional.</li><li>▪ Mayor comodidad de los usuarios.</li><li>▪ Automatizar y llevar un mejor control de los procesos académicos</li></ul>

Cuadro N° 8: Beneficios intangibles de la implementación del Portafolio Electrónico y Aplicativo Android

### 6.2.3. Cálculo de la recuperación de la inversión

➤ **Valor Presente Neto (VPN):**

$$VPN = \sum_{i=0}^{i=n} (Bi - Cj) / (1 + i/100)^n$$

✓ Vida Útil del sistema (n) = 4 años

✓ Tasa de interés anual (i) = 12%

$$\begin{aligned} VPN = & [(0 - 8600)/(1 + 0.12)^0] + [(10000 - 1480)/(1 + 0.12)^1] + \\ & [(10000 - 1480)/(1 + 0.12)^2] + [(10000 - 1480)/(1 + 0.12)^3] + \\ & [(10000 - 1480)/(1 + 0.12)^4] \end{aligned}$$

$$\begin{aligned} VPN = & (-8600)/(1.12)^0 + (8520)/(1.12)^1 + (8520)/(1.12)^2 \\ & + (8520)/(1.12)^3 + (8520)/(1.12)^4 \end{aligned}$$

$$VPN = -8600 + 7607.14 + 6792.09 + 6064.37 + 5414.61$$

$$VPN = 17278.21$$

$$VPN > 0 (Factible)$$

➤ **Tasa Interna de Retorno (TIR):**

$$TIR = (Inv. Total) / (Beneficios netos)$$

$$TIR = 8600 / (10000 - 1480)$$

$$TIR = 8600 / 8520$$

$$TIR = 1.01 \text{ años}$$

NOTA: El tiempo de recuperación de la inversión para la implementación del Portafolio Electrónico y Aplicativo Android será de 1 año y 4 días aprox.

# **CAPITULO VII**

## **CONCLUSIONES**

## **CONCLUSIONES**

- I. Los datos recolectados, permitieron el desarrollo de cada una de las historias de usuario pertenecientes a cada iteración, así mismo los datos recolectados permitieron definir los requerimientos funcionales y no funcionales.
- II. Se desarrolló el portafolio electrónico, con el gestor de base de datos MySQL 5.6.24 el cual permitió diseñar un sistema seguro.
- III. Como en toda empresa se hace necesario seguir los estándares de desarrollo de sistemas los cuales ayudan a llevar de manera más organizada la información; poder especificar los contenidos que se necesitan visualizar en el sistema y lograr que los beneficiarios se acoplen sin mayor dificultad en su manejo.

# **CAPITULO VIII**

## **RECOMENDACIONES**

## **RECOMENDACIONES**

- I. Se recomienda realizar mantenimientos de la parte informativa del sistema, para que de esta manera el sistema se mantenga actualizado con respecto a la información de la institución.
- II. Se recomienda mantener un área de servidor para la confiabilidad y seguridad de la información de la base de datos. Así mismo el encriptado del protocolo HTTP de la capa de seguridad.
- III. Se recomienda hacer una conexión con la base de datos del Sistema de Matrículas de la Universidad Nacional Pedro Ruiz Gallo para de esta manera optimizar el tiempo de los procesos de registro de docentes, estudiantes, cursos y matrícula.
- IV. Complementar el portafolio electrónico, agregando las herramientas como exámenes online, videoconferencias, etc.



# **CAPITULO IX**

## **REFERENCIAS BIBLIOGRÁFICAS**

## REFERENCIAS BIBLIOGRAFICAS

### 1. BIBLIOGRAFIA

Berzal, F y Cubero, J. (2005). *Desarrollo Profesional de Aplicaciones Web con ASP. NET*. Editorial ikor Consulting.

Calero, M. (2003). *Una explicación de la programación extrema*. Madrid.

Date, C. (2001). *"Introducción a los Sistemas de Base de Datos"*. España: Editorial Pearson Educación.

Elmasri, R. y Navathe, S. (2000). *Sistemas de Bases de Datos. Conceptos Fundamentales*. México: Editorial Addison- Wesley Iberoamericana.

Etcheverría Tobón, L. M. (2007). *"Caso práctico de la metodología XP al desarrollo de software"*. Pereira: Universidad Tecnológica de Pereira.

González Seco, J. A. (2012). *"Introducción a visual C#"*.

Quintana, G. y Marqués, M. (2008). *"Aprende SQL"*. España.

Xhafa, S. C. (2008). *"Aplicaciones Distribuidas en Java con tecnología RMI"*. España: Editorial Delta.