

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA: INGENIERÍA DE SISTEMAS

Tesis previa a la obtención del título de: INGENIERO DE SISTEMAS

TEMA:

**ANÁLISIS, DISEÑO, DESARROLLO E INTEGRACIÓN DEL MÓDULO DE
ADMINISTRACIÓN DE USUARIOS, COMUNICACIÓN, SEGURIDAD Y
AUDITORÍA DEL SISTEMA UPS SCHEDULE PARA LA UNIVERSIDAD
POLITÉCNICA SALESIANA SEDE QUITO**

AUTORES:

**MARCO VINICIO MONTAGUANO VÁSQUEZ
EDISON EDUARDO PACHECO MOSQUERA**

DIRECTOR:

RODRIGO EFRAÍN TUFÍÑO CÁRDENAS

Quito, julio de 2014

**DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO
DEL TRABAJO DE TITULACIÓN**

Nosotros, autorizamos a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, declaramos que los conceptos y análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad de los autores.

Quito, julio de 2014

Marco Vinicio Montaguano Vásquez.
CI: 171771532-8

Edison Eduardo Pacheco Mosquera
CI: 172346819-3

DEDICATORIA

Con todo mi cariño y mi amor para las personas que hicieron todo en la vida para que yo pudiera lograr mis sueños, a mi madre, mi padre y mis hermanos por motivarme y darme la mano cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón y mi agradecimiento.

Edison Eduardo Pacheco Mosquera

A Dios por derramar sus bendiciones sobre mí y darme fuerza para vencer cada obstáculo desde el inicio de mi vida, a mis padres quienes con su amor, entrega y apoyo supieron guiarme y motivarme para alcanzar mis objetivos y sueños, a mis hermanos por sus consejos y su compañía y a todos quienes directa o indirectamente hicieron posible este logro, a ustedes dedico mi esfuerzo y mi agradecimiento.

Marco Vinicio Montaguano Vásquez

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1.....	5
ANÁLISIS Y DISEÑO	5
<i>1.1. Análisis de viabilidad.....</i>	<i>5</i>
1.1.1. Viabilidad operativa.	5
1.1.2. Viabilidad técnica.....	5
1.1.3. Viabilidad económica.....	7
1.1.4. Impacto en la organización.....	9
<i>1.2. Requerimientos.....</i>	<i>10</i>
1.2.1. Alcance.....	10
1.2.2. Definiciones, acrónimos y abreviaturas.	11
1.2.3. Visión general.	11
1.2.4. Descripción general.....	11
1.2.5. Requerimientos específicos.....	13
1.2.6. Requerimientos funcionales.	13
1.2.7. Requisitos de rendimiento.....	15
1.2.8. Restricciones del diseño.	16
1.2.9. Atributos del sistema software.	16
1.2.10. Otros requisitos.	16
<i>1.3. Diseño</i>	<i>16</i>
1.3.1. Diseño conceptual.	17
1.3.2. Mapa navegacional.....	17
1.3.3. Interfaz abstracta.	18
1.3.4. Diseño conceptual de la base de datos.	24
CAPÍTULO 2.....	27
DESARROLLO Y PRUEBAS.....	27
<i>2.1. Herramientas y plataforma</i>	<i>27</i>
2.1.1. Power Architect.....	27
2.1.2. PostgreSQL.	28
2.1.3. NetBeans IDE.....	29
2.1.4. Glassfish.	30

2.1.5. JSF.....	33
2.1.6. PrimeFaces.....	33
2.1.7. Framework X-PRIME.....	34
2.2. Diagrama físico de la base de datos.....	34
2.2.1. Diccionario de datos.....	36
2.3. Codificación.....	38
2.3.1. Framework X-PRIME.....	39
2.3.2. Persistencia de conexión.....	39
2.3.3. Pantalla de gestión de opciones.....	40
2.3.4. Gestión de usuarios.....	42
2.3.5. Gestión de perfiles y permisos.....	43
2.3.6. Gestión de auditoría.....	45
2.3.7. Comunicación.....	47
2.3.8. Reportes.....	51
2.4. Pruebas.....	54
2.4.1. Pruebas funcionales.....	54
2.4.2. Pruebas de carga.....	57
2.4.3. Pruebas de usabilidad.....	60
CAPÍTULO 3.....	63
INTEGRACIÓN.....	63
3.1. Diagrama de despliegue.....	63
CONCLUSIONES.....	68
RECOMENDACIONES.....	70
LISTA DE REFERENCIAS.....	71
ANEXO.....	73

ÍNDICE DE FIGURAS

Figura 1. Caso de uso administrador.....	17
Figura 2. Mapa navegacional funcionamiento general	18
Figura 3. Pantalla de acceso al sistema	19
Figura 4. Pantalla de menú.....	19
Figura 5. Pantalla de administración del usuario logeado.....	20
Figura 6. Pantalla de gestión de usuarios	21
Figura 7. Pantalla de gestión de pantallas	22
Figura 8. Pantalla de gestión de permisos	23
Figura 9. Pantalla de gestión de auditoría	24
Figura 10. Diseño conceptual de la base de datos.....	25
Figura 11. Diagrama físico de la base de datos.....	35
Figura 12. Método forma la tabla del framework X-PRIME.....	39
Figura 13. Constructor de la clase que conecta a la base de datos.....	40
Figura 14. Pantalla gestión de opciones.....	41
Figura 15. Constructor para la pantalla gestión de opciones	41
Figura 16. Pantalla gestión de usuarios.....	42
Figura 17. Constructor para la pantalla de gestión de usuarios.....	43
Figura 18. Pantalla de gestión de perfiles y permisos	44
Figura 19. Constructor para la pantalla de gestión de perfiles y permisos	45
Figura 20. Pantalla de lectura para la gestión de auditoría	46
Figura 21. Constructor para la pantalla de gestión de auditoría.....	47
Figura 22. Configuración de parámetros para salida de correo	48
Figura 23. Botón en la barra de menú para el acceso al correo	48
Figura 24. Pantalla principal para el envío de correos	49
Figura 25. Selección de los destinatarios	49
Figura 26. Redacción del correo	50
Figura 27. Notificación de correo enviado.....	50
Figura 28. Visualización del mensaje entregado.....	50
Figura 29. Filtro de reportes de auditoría.....	51
Figura 30. Filtro de rango de fechas.....	52
Figura 31. Filtro de formato de impresión	52
Figura 32. Reporte en formato pdf de auditoría.....	52

Figura 33. Filtro de reportes de usuario	53
Figura 34. Filtro de formato de impresión	53
Figura 35. Reporte de usuarios.....	54
Figura 36. Gráfico comparativo de rendimiento entre los navegadores Google Chrome, Firefox, Safari, Internet Explorer	59
Figura 37. Resultados gráficos de pruebas de usabilidad	62
Figura 38. Diagrama de despliegue para los módulos integrados.....	63
Figura 39. Pantalla de opciones, configuración de los módulos	64
Figura 40. Pantalla de opciones, configuración del módulo de administración.....	65
Figura 41. Pantalla de opciones, configuración del módulo de gestión académica...	65
Figura 42. Pantalla de opciones, configuración del módulo de sistema	66
Figura 43. Pantalla de perfiles, asignación de permisos	66

ÍNDICE DE TABLAS

Tabla 1. Costos de implementación	7
Tabla 2. Análisis de utilidad neta	8
Tabla 3. Desarrollo VAN / TIR.....	9
Tabla 4. Tabla funcional para gestión de usuarios	14
Tabla 5. Tabla funcional para gestión de perfiles	14
Tabla 6. Tabla funcional para gestión de opciones	14
Tabla 7. Tabla funcional para gestión de permisos.....	14
Tabla 8. Tabla funcional para gestión de auditoría.....	15
Tabla 9. Tabla funcional para auditoría	15
Tabla 10. Tabla funcional para usuarios	15
Tabla 11. Tabla descripción de tablas para el módulo.....	26
Tabla 12. Límites de la base de datos PostgreSQL.....	29
Tabla 13. Características por perfil incluidas en Glassfish	31
Tabla 14. Tabla de acciones disponibles en el sistema	36
Tabla 15. Tabla de auditoría del sistema.....	36
Tabla 16. Tabla de las opciones de menú	37
Tabla 17. Tabla de perfiles existentes para el sistema	37
Tabla 18. Tabla de opciones para cada perfil.....	37
Tabla 19. Tabla de información del usuario.....	38
Tabla 20. Resultados pruebas funcionales	55
Tabla 21. Resultados pruebas funcionales generación de reportes	57
Tabla 22. Resultados pruebas de carga en base a funcionalidades en pantallas de gestión de usuarios, permisos y auditoría.....	58
Tabla 23. Resultados pruebas de usabilidad	61

RESUMEN

Debido al proceso manual que realizan actualmente las diferentes carreras de la Universidad Politécnica Salesiana Sede Quito en la distribución de horarios y espacios físicos, el trabajo pretende solucionar esta problemática mediante una implementación tecnológica que brinde una solución sencilla y ágil, por tanto se describe en este proyecto la construcción de un módulo base que permita la creación de usuarios y su administración de forma segura y auditada de los diferentes módulos que integrarán la aplicación final y funcional que conjuntamente con otras aplicaciones solucionarán el proceso antes mencionado en un sistema web llamado UPS SCHEDULE. Como resultado de este proyecto se muestra su análisis, diseño, implementación e integración de forma detallada. Adicionalmente, este proyecto pretende demostrar las ventajas de la apertura a la interoperabilidad entre sistemas informáticos y la facilidad de uso de los mismos a través de la implementación de tecnología actual y de fácil adaptabilidad al entorno operativo.

ABSTRACT

Due to the manual process currently undertaken by the different races of the Salesiana University in Quito based distribution schedules and physical spaces, this paper aims to solve this problem by implementing a technology that provides a simple and flexible solution therefore described in this project building a base module that allows user creation and administration of secure and audited the different modules that compose the final application form and together with other applications solve the above process in a web system called UPS SCHEDULE. As a result of this project analysis, design, implementation and integration is shown in detail. Additionally, this project aims to demonstrate the benefits of opening the interoperability between computer systems and the ease of use of them through the current technology implementation and easy adaptability to the operating environment.

INTRODUCCIÓN

Planteamiento del problema

La Universidad Politécnica Salesiana es una Institución de Educación Superior que funciona más de dieciséis años en el Ecuador, dispone de tres sedes situadas en las ciudades principales del país (Quito, Guayaquil y Cuenca) y su matriz se localiza en la ciudad de Cuenca. A su vez, las sedes se dividen en campus. La sede Quito cuenta con 4 campus: Campus El Girón (matriz), Campus Kennedy, Campus Sur y Centro de Apoyo Cayambe.

Cada una de las sedes dispone de cierta autonomía administrativa y económica, pero todas se rigen por la “Carta de Navegación”, éste es un documento que contiene el plan estratégico y operativo para toda la universidad.

Todas las sedes disponen de una Dirección Técnica de Administración e Inventario, una de las funciones de este departamento es asignar las aulas y espacios físicos (talleres, espacios abiertos, etc.) a las diferentes carreras que funcionan en la sede.

Así mismo, cada una de las carreras a través de sus respectivos consejos, son las encargadas de generar los horarios para cada período académico; estos horarios generalmente cambian de un período a otro puesto que la oferta académica no siempre es la misma.

Ahora no existe un sistema que permita gestionar la información de estos dos ámbitos, cada una de las carreras trabaja de forma independiente para generar sus horarios y la Dirección Técnica de Administración e Inventario asigna los espacios físicos bajo la demanda de las carreras.

Adicionalmente, el manejo y control de los usuarios es muy deficiente ya que la mayoría de procesos se los realiza a mano o a su vez en programas que no necesitan autenticación, esto genera una grave pérdida de información y tiempo. Las seguridades son casi nulas y la comunicación de todos los involucrados en este proceso se la realiza personalmente o vía telefónica.

Objetivo general

Obtener un módulo funcional para la administración de usuarios, seguridad y auditoría que trabaje de forma integrada para el sistema UPS Schedule de la Universidad Politécnica Salesiana Sede Quito.

Objetivos específicos

- Analizar los procesos de administración de usuarios, comunicación seguridad y auditoría para el sistema.
- Definir los requerimientos del módulo.
- Establecer interfaces de comunicación con los otros módulos del sistema.
- Diseñar el módulo en base a la arquitectura definida para el sistema.
- Programar el módulo utilizando herramientas de software libre.
- Gestionar perfiles de usuario y los permisos a los diferentes módulos y pantallas del sistema.
- Gestionar usuarios, los cuales accederán al sistema.
- Gestionar opciones, módulos y pantallas a las cuales se tendrá acceso.
- Gestionar auditoría, registrar las acciones realizadas en cada una de las pantallas configuradas previamente para llevar auditoría.
- Generar y exportar reportes de las pantallas de usuario, auditoría y perfiles.
- Probar el funcionamiento del módulo y la correcta interacción con el resto del sistema.
- Integrar el módulo al sistema UPS Schedule.

Justificación

Desarrollar este proyecto involucra un aporte innovador puesto que se desarrollará una herramienta informática que dará solución a un problema actual de la sede. La gestión de usuarios involucra una investigación profunda sobre seguridades y a su vez la auditoría de todos los procesos que se realicen en el sistema, para lograr un control de cambios y definir responsabilidades de los usuarios.

Alcance

El desarrollo del módulo gestionará toda la información de usuarios del sistema, pero adicionalmente se desarrollará la infraestructura que trabajará transversalmente con el resto del sistema.

Se realizará la gestión de los usuarios (creación, modificación, eliminación), dentro de una interfaz sencilla pero a la vez muy completa, optimizando tiempo y recursos a quién lo maneje.

Tanto las seguridades como la auditoría serán una característica clave dentro del desarrollo del módulo ya que mediante éstos se controlará todos los procesos realizados por los usuarios, evitando así el ingreso o modificación de datos inservibles en el sistema.

La comunicación entre los usuarios registrados en el sistema, tendrá una interfaz amigable, con la facilidad de enviar correos electrónicos.

La conexión y comunicación con la base de datos serán desarrolladas dentro de este módulo con la capacidad de proveer el servicio a los demás módulos.

Adicionalmente, se generarán reportes en cada pantalla para el análisis gerencial del sistema.

El control de avances, comunicación y versionamiento de código con los demás módulos se lo realizará mediante el sitio web de colaboración “Sourceforge” donde se crearán wikis y/o foros, se compartirán archivos, código, herramientas, etc., brindando así una fuente de consulta y aporte al desarrollo del sistema UPS Schedule.

El presente trabajo consta de tres capítulos, los cuales se desarrollan de la siguiente manera:

En la *introducción* se detalla el ¿por qué? del desarrollo del módulo de administración de usuarios, auditoría y seguridad para el sistema integrado UPS Schedule de la Universidad Politécnica Salesiana sede Quito. Se define el problema que presenta la universidad con el manejo de usuarios, seguridad y auditoría en el sistema actual y la forma en que se lo va a resolver.

En el primer capítulo: *Análisis y diseño*, se describirá el análisis previo realizado para el desarrollo del módulo y su diseño mediante diagramas que reflejarán el funcionamiento del sistema.

En el segundo capítulo: *Desarrollo y pruebas*, se detallan herramientas y plataformas a ser utilizadas para alcanzar los objetivos. Se mostrarán los diccionarios de datos, tanto del módulo como del sistema UPS Schedule al cual pertenece. Así mismo se describirán partes de código que influyen considerablemente dentro del desarrollo del sistema. Para finalizar el capítulo, se mostrarán los resultados de las pruebas realizadas al sistema, específicamente al módulo de administración de usuarios, auditoría y seguridad.

Por último el tercer capítulo: *Integración*, se detallará la forma en que éste interactúa con los demás módulos del sistema, teniendo en cuenta que el módulo presentado, pertenece a un sistema común para otros módulos, y este es la base para el desarrollo del mismo, brindando así la facilidad de integrarse desde el comienzo del proyecto.

CAPÍTULO 1

ANÁLISIS Y DISEÑO

Dentro de este capítulo se detallará el análisis previo realizado para la implementación del módulo en mención y el diseño mediante diagramas que reflejen el funcionamiento del mismo, además se presentara el diseño conceptual y físico de la base de datos.

1.1. Análisis de viabilidad

El objetivo del análisis de viabilidad del módulo de gestión de usuarios, seguridad y auditoría es el estudio de un conjunto concreto de necesidades para proponer una solución a corto plazo, que tenga en cuenta restricciones operativas, técnicas y económicas. Por lo tanto, es necesario definir un problema y varias soluciones para resolverlo.

1.1.1. Viabilidad operativa.

El sistema de gestión de usuarios, seguridad y auditoría se convertirá en una herramienta necesaria para la administración de los diferentes módulos que integrarán el sistema como lo son el módulo administrativo y gestión académica, dando seguridad y auditoría a los mismos con los recursos tecnológicos necesarios y facilidad de uso.

1.1.2. Viabilidad técnica.

▪ Recursos humanos

El análisis, diseño e implementación del sistema, serán realizados por los tesisistas, constando como coordinador de proyecto el tutor del trabajo de titulación, quien supervisará el avance en cada fase del desarrollo.

Los roles de analistas, diseñadores y programadores serán desempeñados por los tesistas.

▪ **Hardware**

La Universidad Politécnica Salesiana, Campus Sur, dispone con la infraestructura necesaria y equipos de última generación, en los cuales se podrá implementar el módulo de usuarios, seguridad y auditoría, ofreciendo las debidas garantías necesarias de funcionalidad.

El hardware disponible para el proyecto tiene las siguientes características:

- ✓ Un Gabinete para Servidores Blades, con los siguientes componentes:
 - Un Enclosure Blade c3000, con conexión eléctrica bifásica
 - Switches de Lan 1GB E2 para Enclosure c3000
 - Sistema de ventiladores (4 instaladas) y fuentes redundantes (2 instaladas)
 - Un módulo On Board Administración

- ✓ Servidores para aplicaciones:
 - Servidor BL460C con el siguiente detalle:
 - Un Procesador Intel Xeon Quad Core, 2.33 GHz 1x4MB Cache
 - 4GB RAM
 - Un Smart Array RAID 1
 - Dos Discos de 146GB 10k SAS HDD

Por consiguiente el módulo de usuarios, seguridad y auditoría se encontrará disponible en equipos de alto rendimiento operando 24 horas los 7 días de la semana, garantizando disponibilidad de información a cada momento.

1.1.3. Viabilidad económica.

Los gastos que tendrá que cubrir la universidad son mínimos, debido a que cuenta con la infraestructura informática para alojar el sistema, además cabe mencionar que el proyecto será desarrollado por los tesisistas por lo que no será considerado como un gasto adicional para la universidad.

Con lo expuesto, el beneficio que dispondrá la universidad con la implementación del sistema son los siguientes:

- Automatización de procesos.
- Seguridad y auditoría de la información.
- Mejor calidad en la prestación de servicios hacia los estudiantes.

Tabla 1. Costos de implementación

Concepto	Valor	Responsable
Software		
- Netbeans	\$ 0	
- PostgreSQL	\$ 0	- Marco Montaguano
- Primefaces	\$ 0	- Eduardo Pacheco
- SQL Power Architect	\$ 0	
Hardware		
- 2 Laptops	\$ 2000	- Marco Montaguano - Eduardo Pacheco
Recursos humanos		
- Desarrolladores	\$ 2000	- Marco Montaguano - Eduardo Pacheco
Gastos indirectos		
- Internet	\$ 225	- Marco Montaguano
- Transporte	\$ 400	- Eduardo Pacheco
- Alimentación	\$700	
Total	\$5325	

Elaborado por: Marco Montaguano y Eduardo Pacheco

Tabla 2. Análisis de utilidad neta

Años de proyección de vida útil del proyecto	1	
Update	0	
Ahorros de costos de inversión	5325	
Gastos de depreciación	0	
Utilidad antes de impuesto		5325
Impuesto del 36,25 %	1930,31	
Utilidad neta		3394,69
Años de proyección de vida útil del proyecto	2	
Update	0	
Ahorros de costos de inversión	3394,69	
Gastos de depreciación	0	
Utilidad antes de impuesto		3394,69
Impuesto del 36,25 %	1230,57	
Utilidad neta		2164,11
Años de proyección de vida útil del proyecto	3	
Update	0	
Ahorros de costos de inversión	2164,11	
Gastos de depreciación	0	
Utilidad antes de impuesto		2164,11
Impuesto del 36,25 %	784,49	
Utilidad neta		1379,62
Años de proyección de vida útil del proyecto	4	
Update	0	
Ahorros de costos de inversión	1379,62	
Gastos de depreciación	0	
Utilidad antes de impuesto		1379,62
Impuesto del 36,25 %	500,11	
Utilidad neta		879,51
Años de proyección de vida útil del proyecto	5	
Update	0	
Ahorros de costos de inversión	879,51	
Gastos de depreciación	0	
Utilidad antes de impuesto		879,51
Impuesto del 36,25 %	318,82	
Utilidad neta		560,69

Elaborado por: Marco Montaguano y Eduardo Pacheco

▪ **Análisis de factibilidad TIR / VAN**

VAN: Es la rentabilidad mínima pretendida por la persona que ha invertido en el proyecto, por de debajo de la cual estará dispuesto a efectuar su inversión.

TIR: Es la tasa de interés que hace que el VAN del proyecto sea igual a cero, es un criterio de rentabilidad, se la obtiene mediante la siguiente fórmula.

$$TIR = \sum_{T=1}^n [FCt/(1 + i)^t] - I_o$$

Dónde:

I_o= Inversión inicial.

FC= Flujo de caja del proyecto

i= Tasa de descuento

t= Tiempo

n= Vida útil del proyecto

Tabla 3. Desarrollo VAN / TIR

Años de proyección de vida útil del proyecto	1	2	3	4	5
Update	0	0	0	0	0
Ahorros de costos de inversión	5325	3394,69	2164,11	1379,62	879,51
Gastos de depreciación	0	0	0	0	0
Utilidad antes de impuesto	5325	3394,69	2164,11	1379,62	879,51
Impuesto del 36,25 %	1930,31	1230,57	784,49	500,11	318,82
Utilidad neta	3394,69	2164,11	1379,62	879,51	560,69
VAN	\$ 3.235,59				
TIR	60%				
C/B	2,39				

Elaborado por: Marco Montaguano y Eduardo Pacheco

1.1.4. Impacto en la organización.

Para lograr la aceptación del sistema, específicamente el manejo del módulo de administración, seguridad y auditoría es necesario capacitar al personal que lo va a manejar, ya que es un sistema nuevo para ellos pero con la facilidad de que sus interfaces son muy amigables e intuitivas, lo cual facilitará el aprendizaje

considerablemente para quienes hayan utilizado hojas de cálculo como Excel u OpenOffice.

Se estima un tiempo muy corto para lograr que los usuarios se adapten al manejo del nuevo sistema.

El nuevo sistema a implementar busca solucionar los problemas que actualmente presentan al momento de realizar sus gestiones, por lo cual dependencias administrativas de cada carrera en la sede sur de la Universidad Politécnica Salesiana están de acuerdo en utilizarlo.

Al ser una tecnología de desarrollo que aún está en constante investigación, en los siguientes años, se podrán realizar mejoras considerables sin dejar de funcionar correctamente.

La Universidad Politécnica Salesiana Sede Quito, al poseer la infraestructura necesaria para el desarrollo e implementación del sistema UPS Schedule del cual forma parte el módulo de administración de usuarios, seguridad y auditoría, no es necesario realizar gastos para ponerla en funcionamiento.

1.2. Requerimientos

En esta sección se describe de forma detallada los requerimientos de software del módulo de administración del sistema Ups Schedule para la Universidad Politécnica Salesiana.

1.2.1. Alcance.

Este sistema será creado para administrar los diferentes módulos que el sistema contará. Específicamente, el sistema permitirá al usuario: administrar los usuarios registrados y el perfil del usuario logeado, administrar las diferentes opciones de menú, administrar perfiles y sus respectivos permisos, generar auditoría de todas las

acciones sobre los módulos que sean habilitados para su auditoría y generar reportes de auditoría y usuarios registrados.

1.2.2. Definiciones, acrónimos y abreviaturas.

- **Ups:** Universidad Politécnica Salesiana.
- **GNU/Linux:** Sistema Operativo libre.
- **Arquitecturas X86_64:** Arquitectura en la cual el sistema operativo trabaja, soportando una cantidad mucho mayor de memoria virtual y memoria física.
- **HTTP:** Hypertext Transfer Protocoló Protocolo de transferencia de hipertexto, usado en cada transacción de la World Wide Web (www).
- **IEEE:** Institute of Electrical and Electronics Engineers, asociación técnico-profesional mundial dedicada a la estandarización tecnológica.

1.2.3. Visión general.

El siguiente subcapítulo expone un resumen de la funcionalidad del sistema. En él se describen los requisitos informales y sirve para establecer un contexto base para la especificación de requerimientos técnicos que se detallan en este capítulo

1.2.4. Descripción general.

▪ Perspectivas del producto

El sistema que se propone es independiente. Requerirá de una base de datos para almacenar la información y un servidor de aplicaciones para su funcionamiento. Los usuarios accederán a través de un navegador web.

▪ Funciones del producto

El sistema se lo desarrollará como una aplicación web, los usuarios accederán a través de un nombre de usuario y clave y dispondrá del siguiente módulo:

✓ **Sistema**

Aquí se gestionará la información de usuarios, perfiles, opciones, permisos, auditoría, y reportes.

▪ **Características del usuario**

Se ha definido para este módulo un usuario:

✓ **Administrador**

Persona técnica que se encargará de administrar el sistema, gestionando la información dentro del módulo de administración.

▪ **Restricciones**

El sistema debe ser instalado en un servidor de aplicaciones, compatible con los gestores de base de datos más comunes del mercado y su acceso deberá ser a través de un navegador web.

▪ **Suposiciones y dependencias**

El sistema operativo sobre el cual correrá el sistema podrá ser GNU/Linux o Microsoft Windows.

▪ **Requisitos para futuras versiones del sistema**

Se podría pensar en generar reportes de forma más detallada de la información almacenada en el sistema, como por ejemplo usuarios creados en un periodo de fecha.

1.2.5. Requerimientos específicos.

▪ Requerimientos de interfaz externo

✓ Interfaces de usuario

Las interfaces de usuario serán web, de fácil uso para la interacción con el usuario y utilizará un menú que permita cambiar entre páginas.

✓ Interfaces de hardware

El sistema debe ser compatible con arquitecturas X86_64. La visualización en los clientes puede variar en resoluciones pero se aconseja manejar una resolución 1024 x 768 píxeles.

✓ Interfaces de software

El sistema permite comunicarse con otros módulos que integran el sistema UPS SCHEDULE como el módulo administrativo y el módulo de gestión académica permitiendo el manejo de perfiles.

✓ Interfaces de comunicación

El sistema debe trabajar sobre el protocolo HTTP, tanto para el acceso de los usuarios a la administración como para reportes.

1.2.6. Requerimientos funcionales.

Para definir los requerimientos funcionales se ha utilizado la plantilla estándar “Anexo de Especificación de Requerimientos de Software IEEE Std. 830-1998” (Borja & Cuji, 2013)

- **Administrador del sistema**

- ✓ **Requerimiento funcional 1.1: gestión de usuarios**

Tabla 4. Tabla funcional para gestión de usuarios

Descripción	Agregar, modificar y eliminar usuarios del sistema
Precondición	Ninguna
Entrada	Datos del usuario(perfil, estado , cédula, nombre, nombre completo, correos principal y secundario, clave y tema)
Proceso	Datos modificados en el sistema
Salida	Mensaje de aceptación / Error

Elaborado por: Marco Montaguano y Eduardo Pacheco

- ✓ **Requerimiento funcional 1.2: gestión de perfiles**

Tabla 5. Tabla funcional para gestión de perfiles

Descripción	Modifica datos del usuario logeado (datos personales, clave, tema)
Precondición	Ninguna
Entrada	Datos del usuario(perfil, estado , cédula, nombre, nombre completo, correos principal y secundario, clave y tema)
Proceso	Datos modificados en el sistema
Salida	Mensaje de aceptación / Error

Elaborado por: Marco Montaguano y Eduardo Pacheco

- ✓ **Requerimiento funcional 1.3: gestión de opciones**

Tabla 6. Tabla funcional para gestión de opciones

Descripción	Agrega, modifica y elimina datos de las opciones del menú.
Precondición	Nombres de paquetes y clases.
Entrada	Datos de opciones (enlace, nombre, paquete, auditada)
Proceso	Datos modificados en el sistema
Salida	Mensaje de aceptación / Error

Elaborado por: Marco Montaguano y Eduardo Pacheco

- ✓ **Requerimiento funcional 1.4: gestión de permisos**

Tabla 7. Tabla funcional para gestión de permisos

Descripción	Agrega, modifica y elimina datos de los permisos habilitados para diferentes pantallas
Precondición	Perfiles, opciones
Entrada	Datos de permisos (nombre de perfil, descripción de perfil, opción, lectura)
Proceso	Datos modificados en el sistema
Salida	Mensaje de aceptación / error

Elaborado por: Marco Montaguano y Eduardo Pacheco

✓ **Requerimiento funcional 1.5: gestión de auditoría**

Tabla 8. Tabla funcional para gestión de auditoría

Descripción	Permite revisar datos agregados, modificados y eliminados de todo el sistema.
Precondición	Ninguna
Entrada	Datos de auditoría (Fecha inicial, Fecha final)
Proceso	Datos modificados en el sistema
Salida	Mensaje de aceptación / Error

Elaborado por: Marco Montaguano y Eduardo Pacheco

▪ **Reporte**

✓ **Requerimiento funcional 2.1: auditoría**

Tabla 9. Tabla funcional para auditoría

Descripción	Permite generar archivo pdf, excel, word con datos auditados.
Precondición	Existan datos en el rango de fechas.
Entrada	Datos reportes (fecha inicial, fecha final, tipo de documento a exportar)
Proceso	Generación de archivo
Salida	Mensaje de aceptación / Error

Elaborado por: Marco Montaguano y Eduardo Pacheco

✓ **Requerimiento funcional 2.2: usuarios**

Tabla 10. Tabla funcional para usuarios

Descripción	Permite generar archivo pdf, excel, word con datos de usuarios existentes.
Precondición	Existan usuarios
Entrada	Datos reportes (tipo de documento a exportar)
Proceso	Generación de archivo
Salida	Mensaje de aceptación / Error

Elaborado por: Marco Montaguano y Eduardo Pacheco

1.2.7. Requisitos de rendimiento.

- El sistema deberá estar disponible 24 horas al día y 7 días a la semana.
- Deberá soportar al menos 20 usuarios concurrentes.

1.2.8. Restricciones del diseño.

Las herramientas y plataformas que utilice para el desarrollo del sistema deberán ser software libre.

1.2.9. Atributos del sistema software.

- **Seguridad**

El sistema debe proveer mecanismos de seguridad para que solo usuarios autorizados ingresen y definir un tiempo límite de inactividad para la sesión de los usuarios. Se desea que las claves de los usuarios se almacenen encriptados con el algoritmo MD5.

1.2.10. Otros requisitos.

- **Ingreso de usuarios**

Todos los usuarios del sistema deben ingresar a través de un nombre de usuario y una clave.

- **Cambio de clave**

Cada usuario cambia su clave sin la necesidad de solicitar ayuda al administrador.

1.3. Diseño

En esta sección se detallará el funcionamiento del módulo de gestión de usuarios, seguridad y auditoría mediante diagramas de casos de uso, luego mediante un mapa navegacional se mostrará cómo interactúan los usuarios con el sistema. Posteriormente en la interfaz abstracta se mostrará un esqueleto de cómo serían las pantallas del módulo para finalmente describir el diseño conceptual de la base de datos a utilizar.

1.3.1. Diseño conceptual.

El diseño conceptual muestra las funciones principales que los usuarios realizan dentro del sistema. Cabe mencionar que existe un solo usuario, el administrador, quién se encargará de manejar el módulo.

- **Caso de uso 1: administrador**

Este caso de uso se refiere a la administración del sistema y el único actor involucrado es el administrador, quien se encarga de configurar el sistema, crear usuarios, perfiles, pantallas, dar permisos de acceso a pantallas. Adicionalmente él mismo podrá generar reportes, los cuales le permitirán monitorear el estado del sistema.

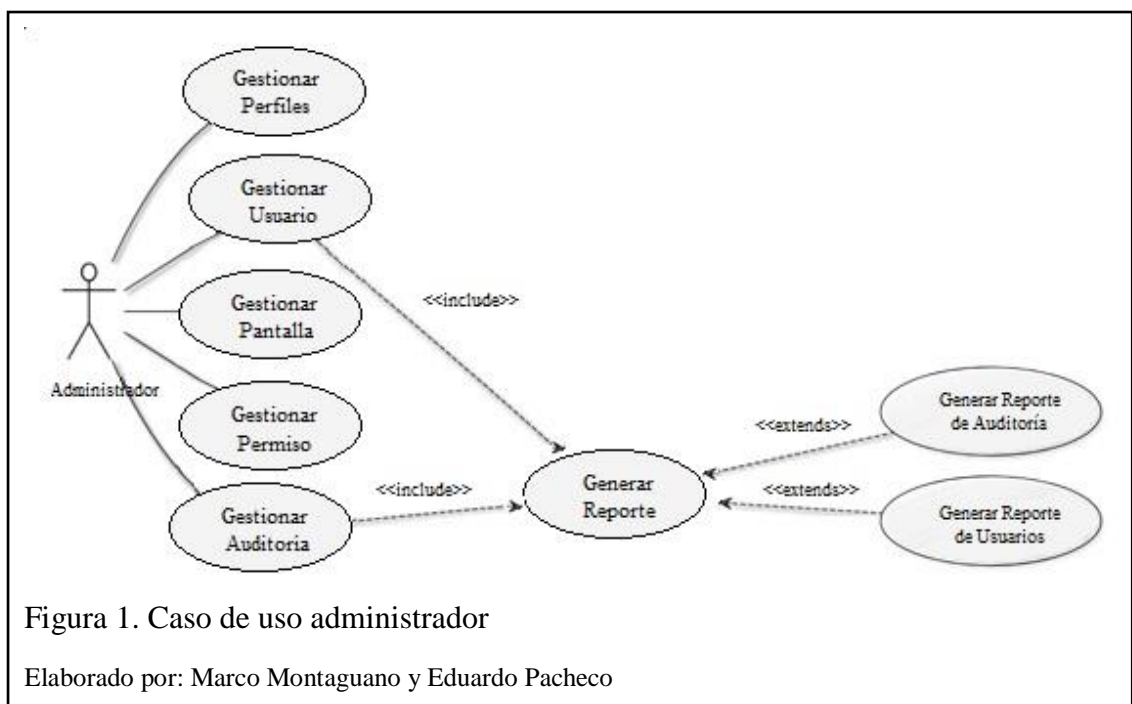


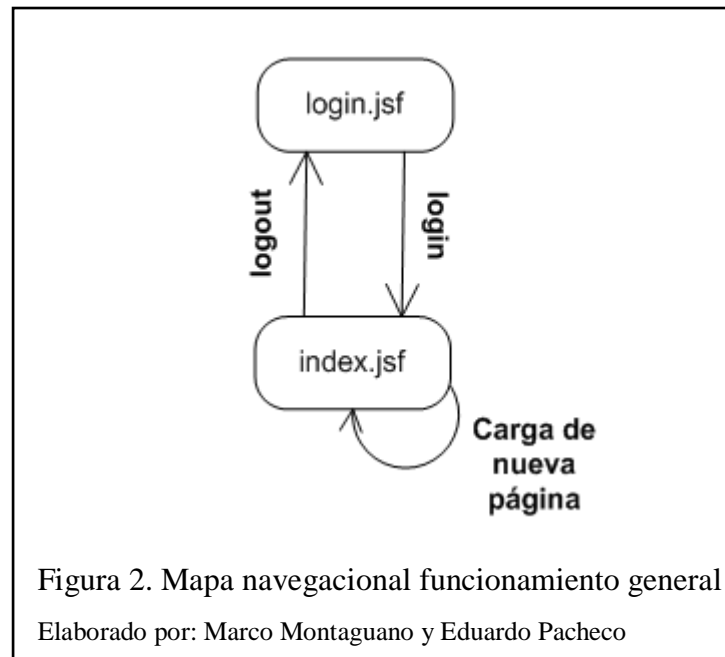
Figura 1. Caso de uso administrador

Elaborado por: Marco Montaguano y Eduardo Pacheco

1.3.2. Mapa navegacional.

En el siguiente diagrama se muestra la interacción del usuario administrador dentro de la aplicación web. Al ingresar al sistema, a todos los usuarios se les despliega la página de login.jsf donde ingresarán las credenciales de acceso al sistema. Luego de pasar el login, el sistema redirige a la página index.jsf, la cual tiene la particularidad de que en ésta se cargará dinámicamente el contenido según la opción seleccionada

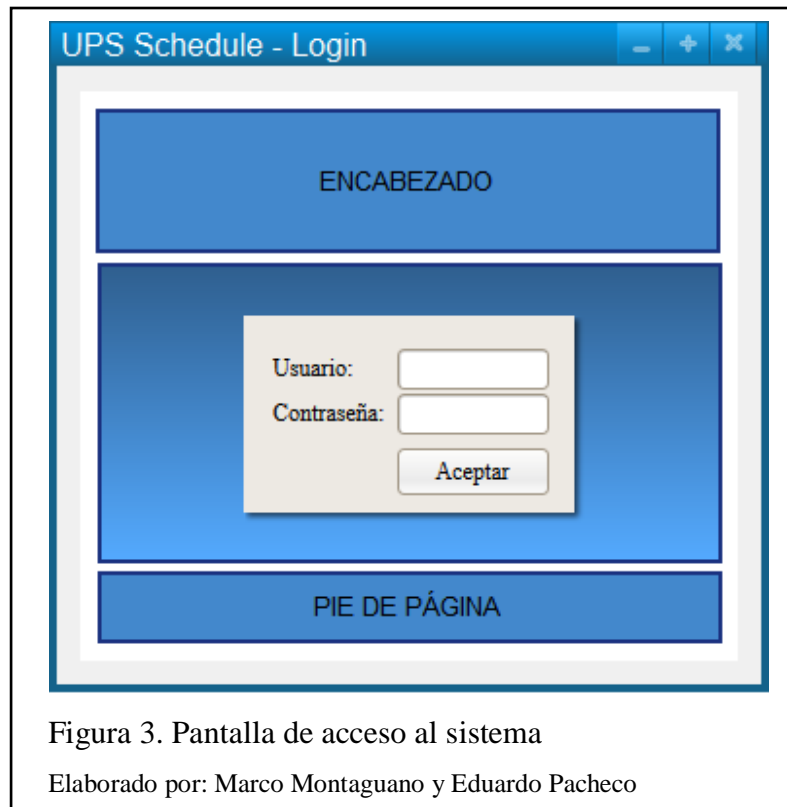
del menú, haciendo innecesaria la creación de más páginas .jsf. Este proceso se detallará en los siguientes capítulos.



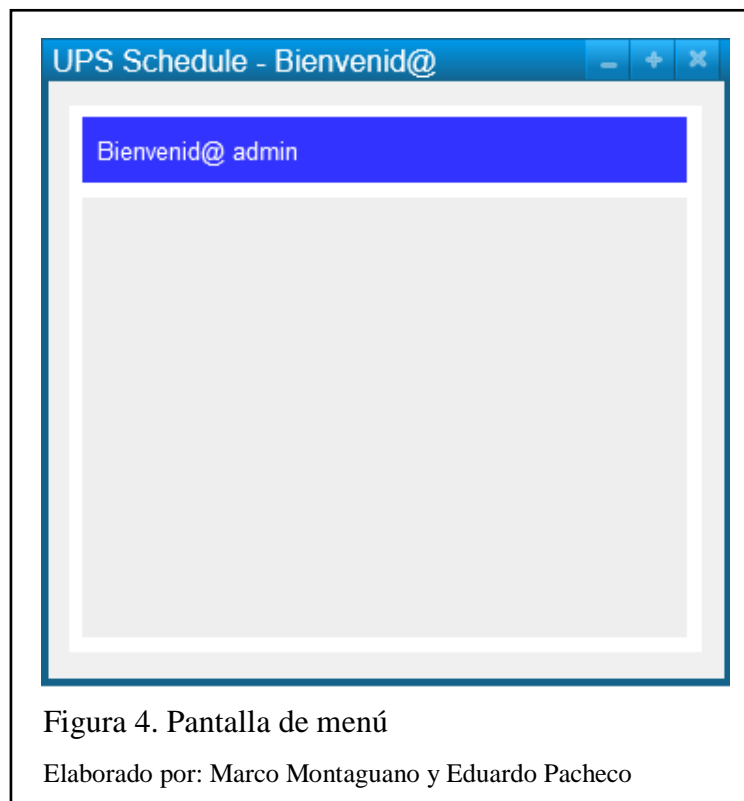
1.3.3. Interfaz abstracta.

El sistema, dentro del cual se encuentra el módulo de gestión de usuarios, seguridad y auditoría dispone de dos páginas JSF dónde se cargarán las diferentes pantallas y dentro de estas los controles que permitirán al usuario manejar el sistema de una forma amigable, rápida e intuitiva.

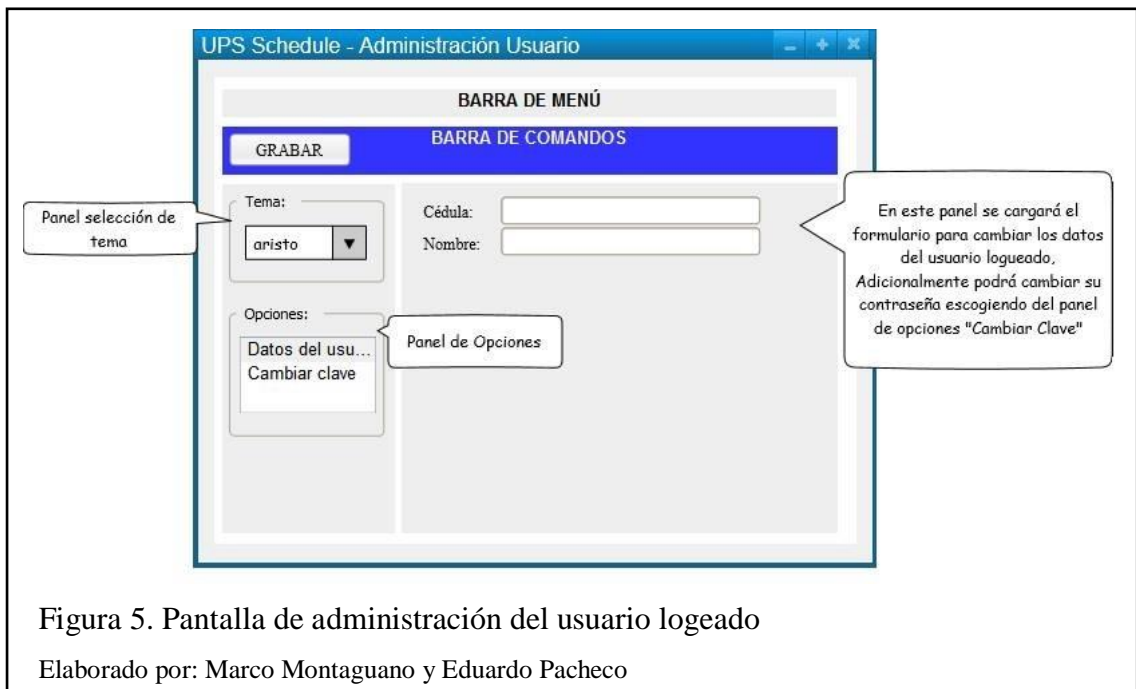
En la siguiente figura se muestra la pantalla de acceso al sistema, constará de 3 paneles encabezado, central y pie de página, de los cuales el panel central consta con el formulario de ingreso.



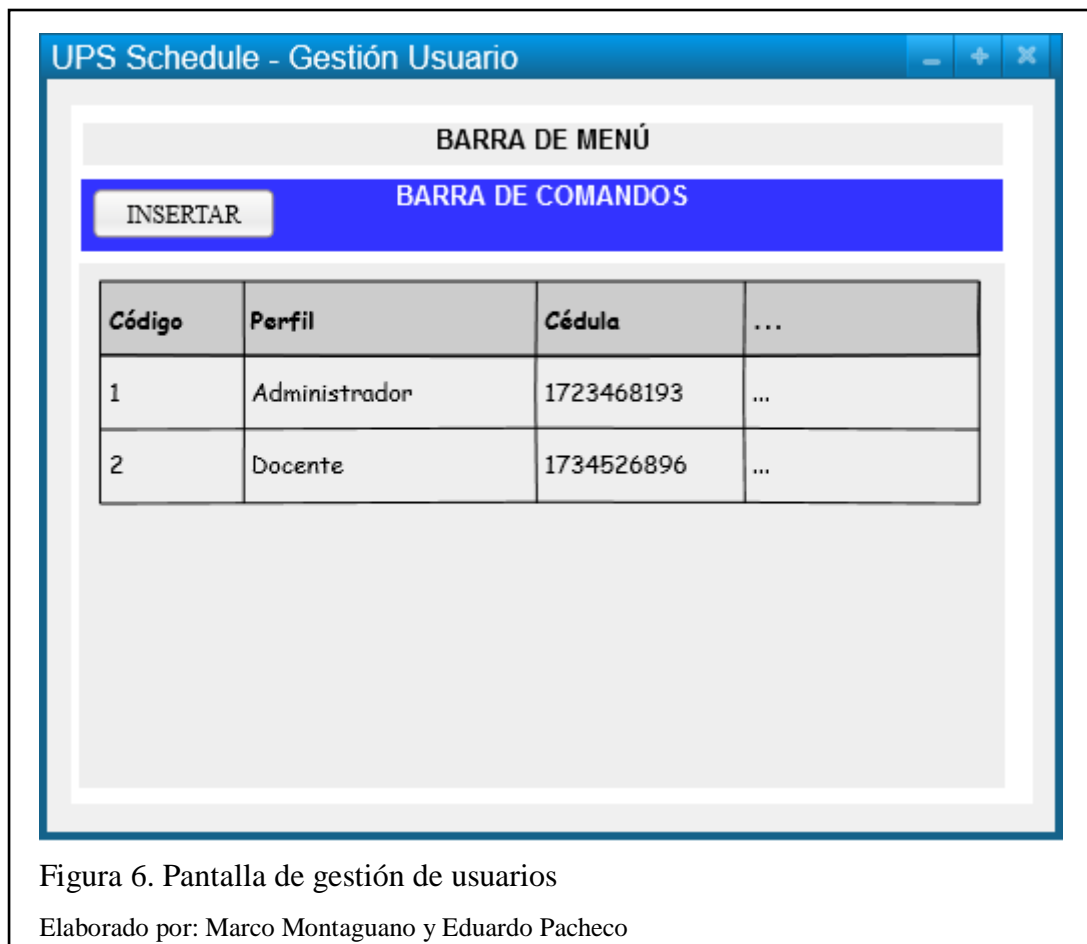
Una vez realizado el ingreso, se desplegará una pantalla de bienvenida donde en la parte superior constará el menú de opciones.



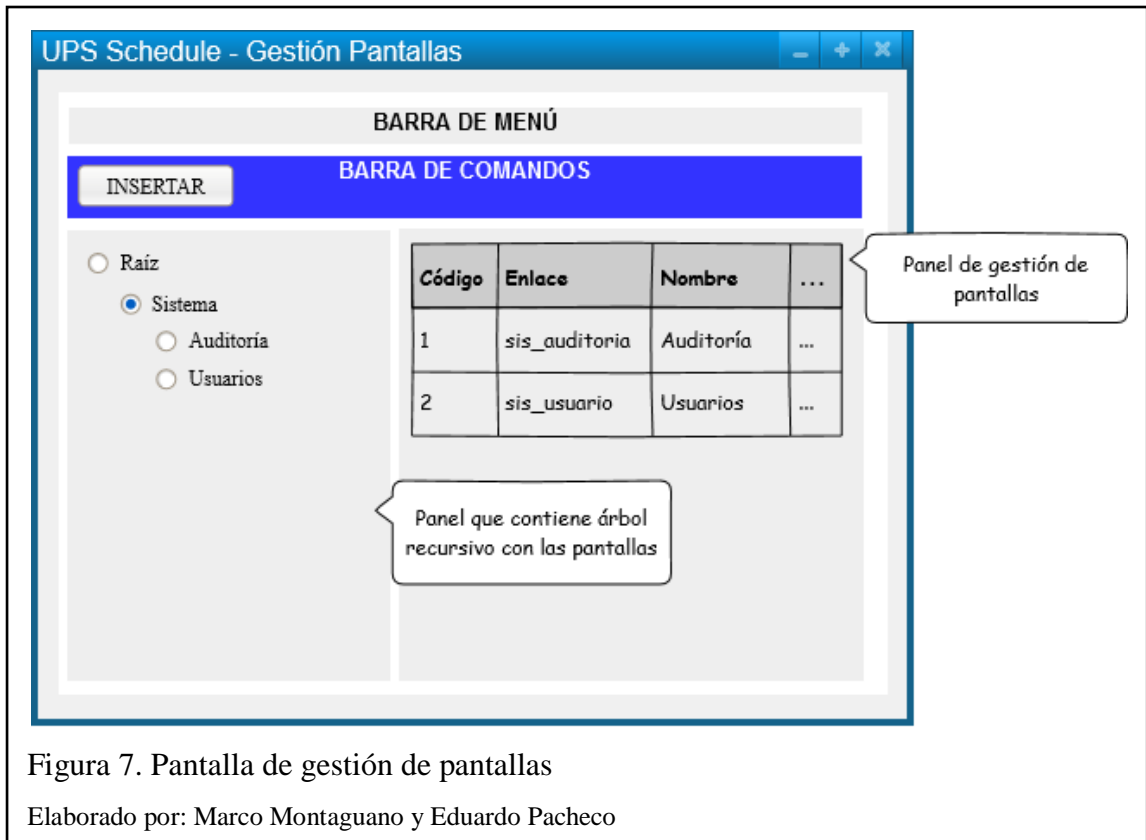
Cada vez que el usuario seleccione una opción del menú, en el panel inferior se cargará la pantalla correspondiente. Un ejemplo de esto se muestra en la figura siguiente.



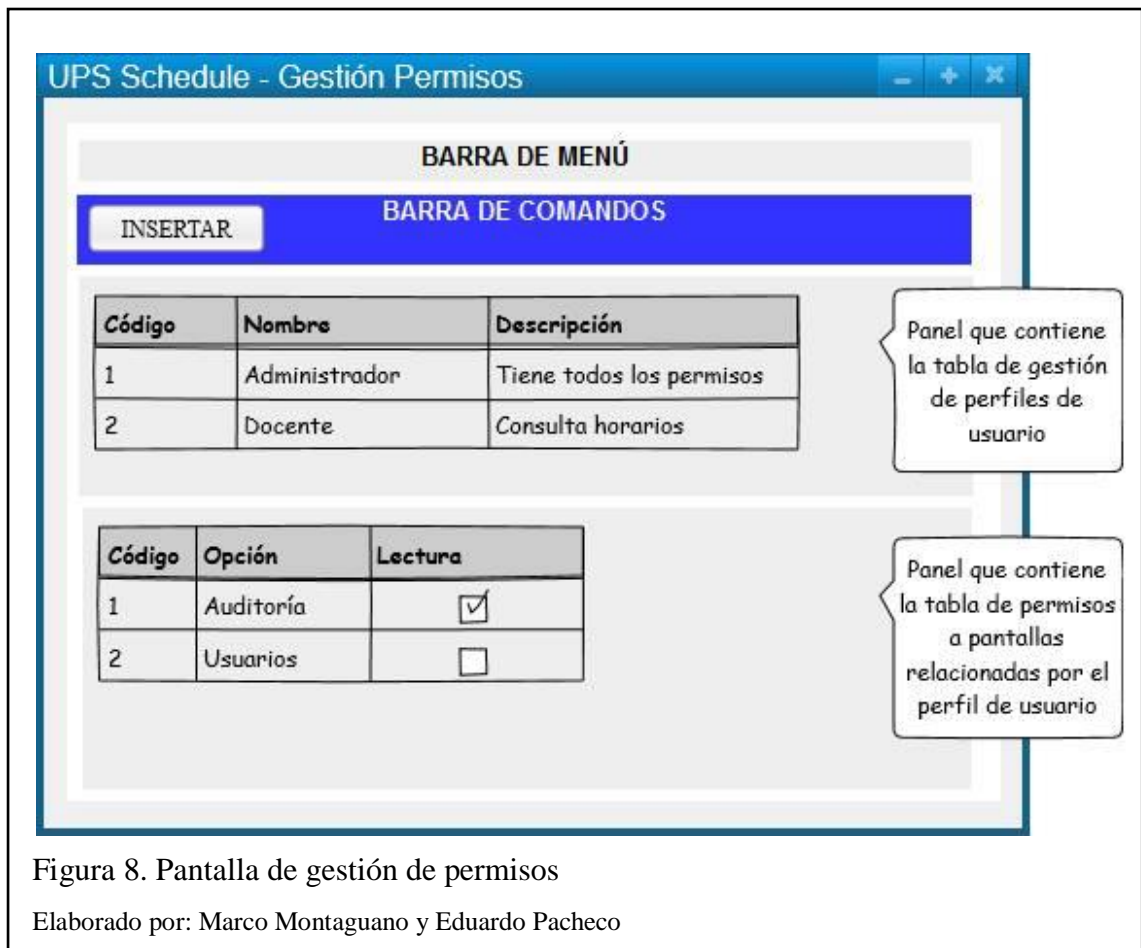
Una parte del módulo consta la gestión de usuarios, en esta pantalla se podrán registrar, actualizar datos, y eliminar usuarios mediante una interfaz de usuario amigable y similar a hojas de cálculo.



A continuación, la gestión de pantallas se manejará mediante la interfaz mostrada en el gráfico siguiente. Mediante un árbol recursivo se mostrará la jerarquía ente las páginas configuradas. De igual manera que los usuarios, se podrá registrar, actualizar y eliminar.



Para la gestión de permisos, el formato de pantalla a manejar se muestra en la siguiente figura. Las transacciones a realizar serán ingreso, actualización y eliminación de permisos y perfiles.



Una parte muy importante dentro del módulo es la gestión de auditoría. La figura a continuación muestra la forma en la cual se visualizarán las acciones realizadas dentro del sistema, siempre y cuando la pantalla sea configurada para manejar esta opción.

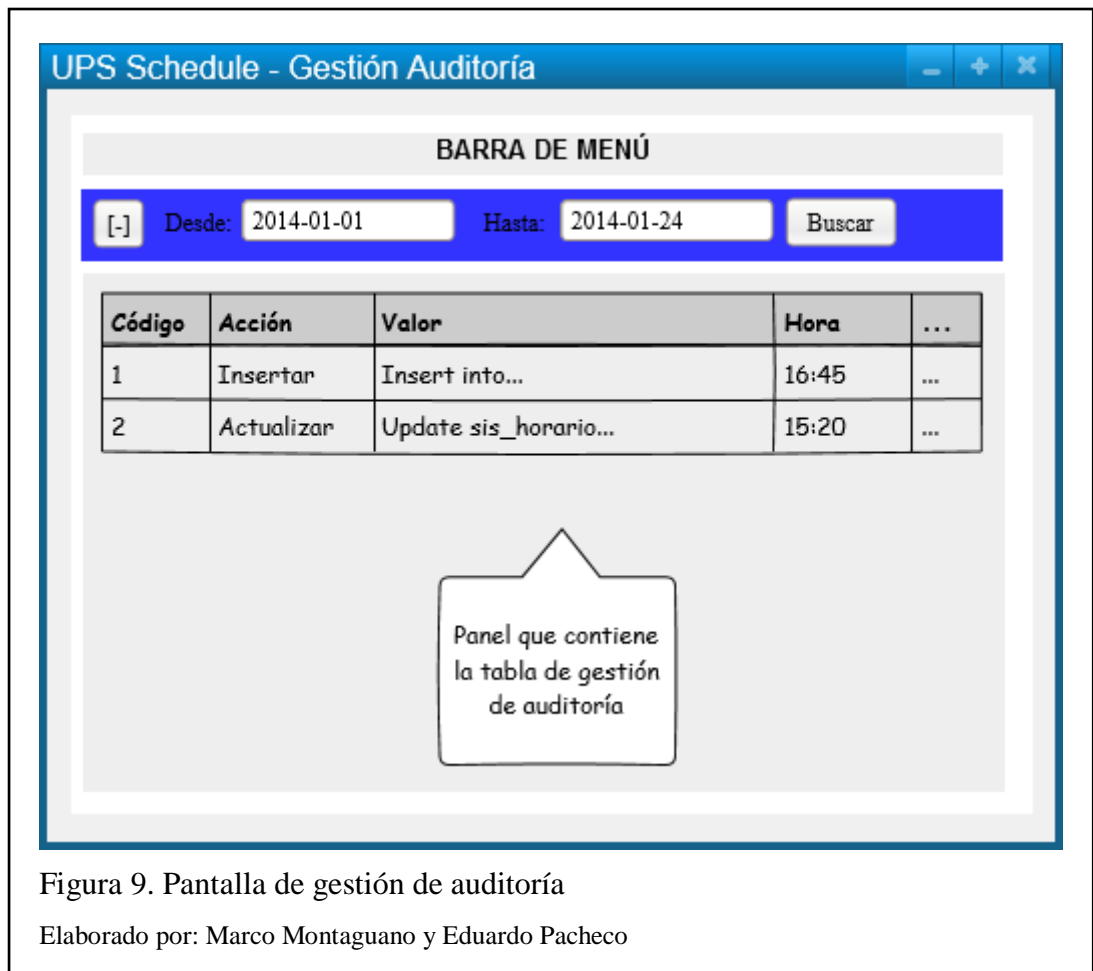


Figura 9. Pantalla de gestión de auditoría

Elaborado por: Marco Montaguano y Eduardo Pacheco

1.3.4. Diseño conceptual de la base de datos.

La base de datos fue diseñada conjuntamente con el resto del sistema UPS Schedule, dividiéndolo en módulos funcionales, a continuación se muestra el diseño conceptual donde cada módulo está diferenciado por colores.

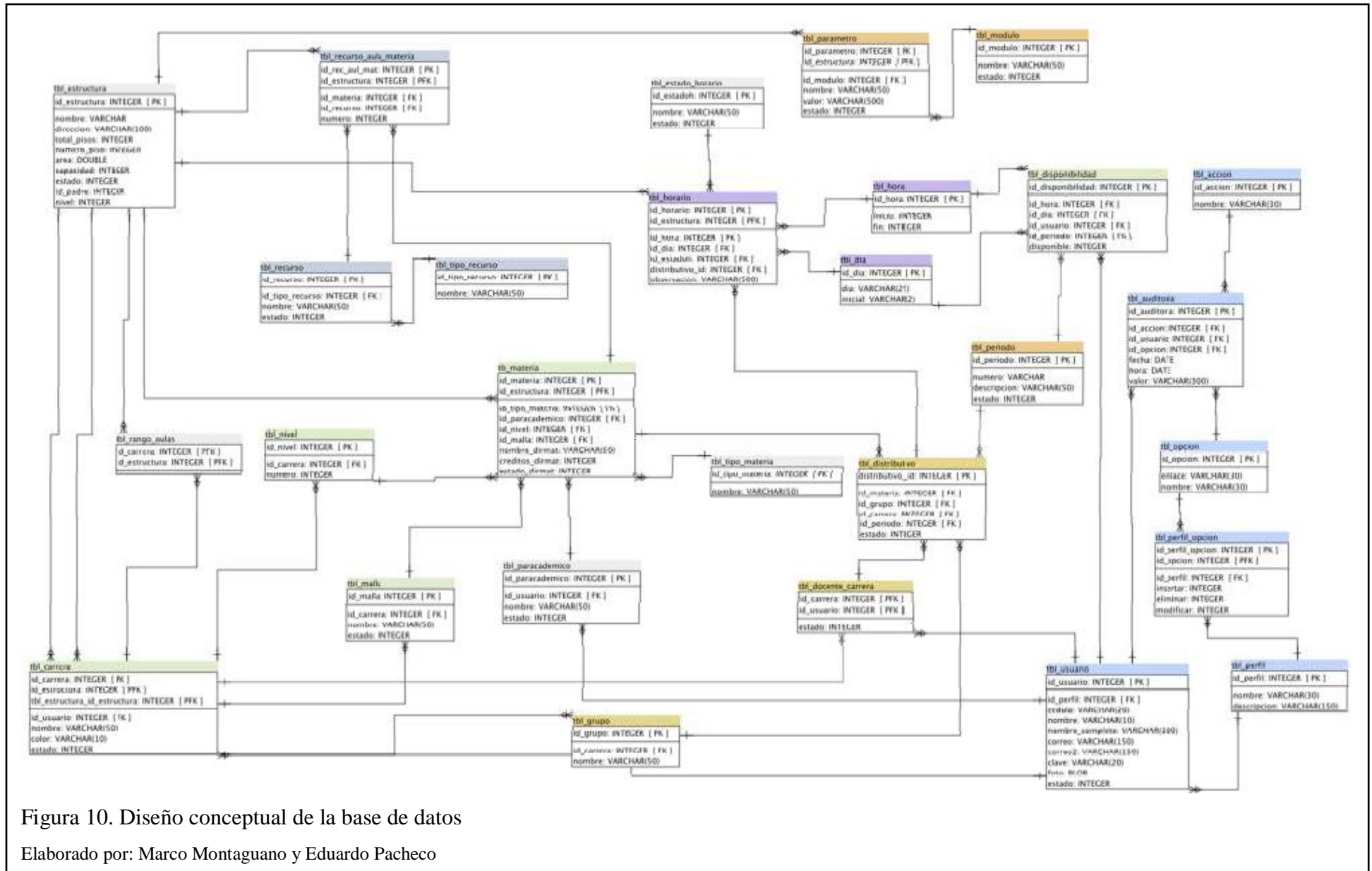


Figura 10. Diseño conceptual de la base de datos

Elaborado por: Marco Montaguano y Eduardo Pacheco

El módulo de administración de usuarios, seguridad y auditoría usa las tablas descritas a continuación:

Tabla 11. Tabla descripción de tablas para el módulo

Tabla	Descripción
tbl_accion	Posibles acciones que se pueden realizar para el manejo de la auditoría
tbl_auditoria	Tabla para la gestión de auditoría
tbl_opcion	Tabla que maneja la gestión de pantallas del sistema
tbl_perfil	Tabla para la gestión de perfiles de usuario
tbl_perfil_opcion	Tabla para la relación entre los perfiles de usuario y las pantallas del sistema
tbl_usuario	Tabla para la gestión de usuarios

Elaborado por: Marco Montaguano y Eduardo Pacheco

CAPÍTULO 2

DESARROLLO Y PRUEBAS

En este capítulo se hace mención a las herramientas de desarrollo y la plataforma en la cual se desenvolverá el módulo perteneciente al sistema UPS Schedule. Se describirán las partes de código más importantes utilizadas. Finalmente se mostrarán los resultados de las pruebas realizadas al módulo de administración de usuarios, seguridad y auditoría.

2.1. Herramientas y plataforma

Las herramientas utilizadas para el desarrollo del módulo son libres, lo cual facilita tanto la instalación como la distribución de las mismas. A continuación se describen cada una de ellas:

2.1.1. Power Architect.

Es una herramienta libre de modelado de datos SQL, fue creado por diseñadores de almacenamiento de datos y tiene muchas características únicas dirigidas específicamente para una arquitectura de almacenamiento de datos. Permite a los usuarios realizar ingeniería inversa de bases de datos existentes, realizar perfiles de datos en bases de datos fuente y auto-generar metadatos ETL.

Entre sus mayores funcionalidades están:

- Accede a las bases de datos de origen a través de JDBC.
- Se conecta a múltiples bases de datos fuente al mismo tiempo.
- Compara modelos de datos y estructuras de base de datos e identifica las discrepancias.
- Recuerda el origen de cada columna.
- Genera informes de mapeo visual de origen a destino.
- Ingeniería inversa a PostgreSQL, Oracle, MS SQL Server y más.

- Guarda origen la estructura de datos de instantáneas en el proyecto, para que pueda trabajar de forma remota.
- Todos los datos de proyecto se almacena en un formato XML.
- Modelado esquema OLAP: Cubos, medidas, dimensiones, jerarquías y niveles.
- Descarga gratuita (Open Source, v.3 GPL licencia) (Community Edition).

Otra característica clave del Arquitecto Poder SQL que lo diferencia de otras herramientas de modelado de datos, es que recuerda el origen de cada columna y es capaz de generar automáticamente las asignaciones de datos de origen a destino. (SQL Power Architect, 2014)

2.1.2. PostgreSQL.

PostgreSQL es un potente sistema de base de datos, de código abierto objeto-relacional. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad e integridad de datos en todos los principales sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows. Es totalmente compatible con ACID, tiene soporte completo para claves foráneas, uniones, vistas, disparadores y procedimientos almacenados (en varios idiomas). Incluye más de SQL: 2008 tipos de datos, incluyendo INTEGER, numéricos, booleanos, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. También es compatible con el almacenamiento de grandes objetos binarios, como imágenes, sonidos o vídeo. Cuenta con interfaces de programación nativas para C / C + +, Java, Net, Perl, Python, Ruby, Tcl, ODBC, entre otros, y una extensa documentación.

Una base de datos de clase empresarial, PostgreSQL cuenta con características avanzadas tales como multi-versión Control de concurrencia (MVCC), punto en el tiempo de recuperación, tablespaces, replicación asincrónica, transacciones anidadas (puntos de retorno), copias de seguridad en caliente, un sofisticado consultor planificador / optimizador, y un registro para la tolerancia a fallos. Es compatible con los juegos de caracteres

internacionales, codificaciones de caracteres multibyte, Unicode, y es consciente de la configuración regional.

Es altamente escalable tanto en la enorme cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede interactuar. Hay sistemas de PostgreSQL activos en entornos de producción que manejan más de 4 terabytes de datos. Algunos de los límites generales de PostgreSQL se incluyen en la siguiente tabla. (PostgreSQL, 2012)

Tabla 12. Límites de la base de datos PostgreSQL

Límite	Valor
Tamaño máximo de la base	Ilimitado
Tamaño máximo de las tablas	32 TB
Tamaño máximo de fila	1.6 TB
Tamaño máximo de campo	1 GB
Número máximo de filas por tabla	Ilimitado
Máximo de columnas por tabla	250 - 1600 dependiendo del tipo de columna
Máximo de índices por tabla	Ilimitado

Fuente: (PostgreSQL, 2012)

2.1.3. NetBeans IDE.

NetBeans IDE es un entorno de desarrollo, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE.

NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

También está disponible NetBeans Platform; una base modular y extensible usada como estructura de integración para crear grandes aplicaciones de escritorio. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente

en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones.

Ambos productos son de código abierto y gratuito para uso tanto comercial como no comercial. El código fuente está disponible para su reutilización de acuerdo con la Common Development and Distribution License(CDDL) v1.0 and the GNU General Public License (GPL) v2.

Además, NetBeans IDE posee un portal de la comunidad de código abierto dedicado a construir un IDE de primera clase. Netbeans.org permite a usuarios de más de 160 países de todo el mundo estar en contacto con los recursos y las personas alrededor de NetBeans. Es posible descargar desde aquí las últimas versiones de NetBeans, acceder a la documentación de ayuda en línea, profundizar su conocimiento personal de Java, estar al corriente de las últimas noticias, unirse a una lista de distribución, contribuir código, conocer las personas implicadas en el proyecto, conocer gente, y mucho más. (NetBeans, 2013)

2.1.4. Glassfish.

El término Glassfish, traducido al español sería algo parecido como “Pez de Cristal”, es el nombre de un pez que realmente existe y vive en el agua dulce; su cuerpo es transparente, por lo que sus huesos son visibles. El nombre fue elegido debido a la transparencia que los creadores querían darle al proyecto, que utiliza una licencia open source, concretamente la licencia Common Development and Distribution License (CDDL) v1.0 y la GNU Public License (GPL) v2.

GlassFish es un servidor de aplicaciones desarrollado por Sun Microsystems que implementa las tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación. La versión comercial es denominada Sun GlassFish Enterprise Server. Soporta las últimas versiones de tecnologías como: JSP, Servlets, EJBs, Java API para Servicios Web (JAX-WS), Arquitectura Java para enlaces XML (JAXB), Metadatos de

Servicios Web para la Plataforma Java 1.0, y muchas otras tecnologías. (Glassfish, 2012)

Glassfish ha sido desarrollado a partir de un tiempo de ejecución modular y flexible basado en el estándar OSGi. Permite a las organizaciones crear e implementar aplicaciones web con un perfil web Java EE ligero y sacar partido fácilmente de la potencia de toda la plataforma Java EE para aplicaciones de empresa. Es una plataforma flexible que aplica solamente lo necesario para responder a las necesidades de la empresa.

Puesto que Glassfish emplea una arquitectura de micro-kernel basada en OSGi, se puede utilizar el Update Center (Centro de actualización) para actualizar dinámicamente hasta la plataforma completa de Java EE.

Glassfish además, de ser un servidor de aplicaciones, es una comunidad de usuarios donde existen partners que contribuyen agregándole más características importantes a Glassfish. Además, ingenieros y beta testers desarrollan código y prueban las versiones liberadas para eliminar todo fallo que se encuentre. La comunidad fue lanzada en el año 2005 en java.net. Al igual que el pez original, la Comunidad Glassfish es transparente en cuanto a términos de entrega de código fuente, discusiones de ingeniería, agendas, datos de descarga, etc. Se puede tener acceso a todo esto, y formar parte de todo el proceso detrás de la comunidad Glassfish. (Proyecto GlassFish, 2012)

Algunas características soportadas por Glassfish se detallan a continuación:

Tabla 13. Características por perfil incluidas en Glassfish

Función	Incluidas en Perfil completa	Incluidas en Perfil web
Plataforma de Java, Enterprise Edition 6	X	X
EJB 3.1 Lite (compatible sólo con el contenedor local)	X	X
JTS/JTA 1.1	X	X

Continúa...

Tabla 13. Características por perfil incluidas en Glassfish (continuación...)

Microkernel basado en OSGi	X	X
Tecnologías web de Java (Servlet 3.0, JSP 2.2, JSF 2.0, JSTL 1.2, el 2.2)	X	X
Grizzly/Comet	X	X
JPA 2.0, (EclipseLink), agrupación de conexiones JDBC	X	X
JDBC 4.0	X	X
JavaDB 10.5.3	X	X
Consola de administración web y CLI	X	X
Re implementación rápida (mantenimiento del estado de la sesión durante la ejecución de la re implementación)	X	X
OMG CORBA APIs 3.0	X	X
Debugging Support for Other Languages 1.0	X	X
DependencyInjectionfor Java 1.0	X	X
ManagedBeans 1.0	X	X
Interceptors 1.1	X	X
Contexts and Dependency Injection for Java 1.0 (CDI, Web Beans)	X	X
BeanValidation 1.0	X	X
JAX-RS 1.1 (Jersey)	X	X
API completas para EJB 3.1 (compatibles con contenedores remotos)	X	
Java EE Connector Architecture 1.6	X	
JAX-WS 2.2	X	
JAXB 2.2	X	
Implementing Java Web Services 1.3	X	
Web Services Metadata for the Java Platform 2.1	X	
JMS 1.1	X	
JavaMail 1.4	X	
JACC 1.4	X	
Java Authentication Service Provider Interface for Containers 1.1	X	
Java EE Application Deployment 1.2	X	
J2EE Management 1.1	X	
JAX-RPC 1.1	X	
JAXR 1.0	X	
CORBA 3.0.0	X	
Metro	X	

Fuente: (Proyecto GlassFish, 2012)

2.1.5. JSF.

Java Server Faces es un framework o ambiente de trabajo, destinado a facilitar la construcción y desarrollo de interfaces para aplicaciones basadas en entornos web. JSF tiene la función de proveer un grupo muy extenso de herramientas pre desarrollado para facilitar la creación de interfaces; herramientas como: botones, cajas de texto, tablas de datos, etc. Similar a los objetos empleados para la generación de aplicaciones de escritorio como swing.

Uno de los puntos fuertes hacia cual se dirige la arquitectura JSF es la generación de interfaces que pueden ser consumidas por diferentes tipos de dispositivos clientes.

De esta manera, la misma interface puede verse desde una PC de escritorio, Tablet, PDA, celulares etc. Sin tener que crear nuevas interfaces ni desarrollar diferentes procesos para cada uno de los ambientes antes mencionados.

Además, muchos de los IDE's actuales utilizados para el desarrollo, incorporan muchos de los controles empleados por el estándar de manera que el proceso de generación de una interfaz de aplicación puede ser llevada a cabo utilizando ambientes visuales, sin tener que escribir todo el código que se requiere para generar los controles. (Loor, 2013)

2.1.6. PrimeFaces.

“PrimeFaces es una librería de componentes visuales para Java Server Faces (JSF) de código abierto que cuenta con una gran cantidad de componentes que facilitan la creación de las aplicaciones web.” (Guamantica & Tupiza, 2013)

Algunas características son:

- Un interesante conjunto de componentes (editor HTML, autocompletado, gráficas, etc.)

- Sin dependencias, ni configuraciones, además de ser muy ligero.
- Soporte para Ajax, basándose en el estándar JSF 2.0 Ajax API.
- Múltiples temas de apariencia, listos para usar.
- Excelente documentación con ejemplos de código.
- Soporte para interfaces de usuario sobre dispositivos móviles, nos provee de un kit para este menester.
- Amplia difusión del framework, con lo cual existe una comunidad que respalda al proyecto. (Características de PrimeFaces, 2014)

2.1.7. Framework X-PRIME.

X-PRIME es un framework desarrollado en lenguaje JSF por Eduardo Pacheco (Tesista), basado principalmente en la reutilización de código y software libre, es una implementación y adaptación de PrimeFaces con el objetivo de permitir al desarrollador facilitar su trabajo al momento de crear una aplicación web.

X-PRIME tiene como principal característica la fácil adaptación a cualquier giro de negocio, el único requisito es que en la base de datos se deben incluir varias tablas de configuración, estas son básicamente dónde se almacenarán, usuarios, permisos, accesos a pantallas, auditoría, etc.

2.2. Diagrama físico de la base de datos

2.2.1. Diccionario de datos.

El siguiente diccionario de datos hace referencia exclusivamente al módulo de gestión de usuarios, seguridad y auditoría del sistema UPS Schedule y a las tablas de configuración del framework X-PRIME.

Tabla 14. Tabla de acciones disponibles en el sistema

tbl_accion				
Nombre Columna	Tipo de Dato	¿No Nulo?	Clave Primaria	Descripción
id_accion	Integer	Si	Si	Clave primaria de la tabla
nombre	character varying(30)	Si	No	Es el nombre de la acción que se registra en la auditoría
Restricciones				
Nombre		Tipo		Definición
pk_codigo_audacc		Clave Primaria		(id_accion)

Elaborado por: Marco Montaguano y Eduardo Pacheco

Tabla 15. Tabla de auditoría del sistema

tbl_auditoria				
Nombre Columna	Tipo de Dato	¿No Nulo?	Clave Primaria	Descripción
id_auditoria	Integer	Si	Si	Clave primaria de la tabla
id_accion	Integer	Si	No	Clave de la acción que se ejecutó
id_usuario	Integer	Si	No	Clave del usuario que ejecutó la acción
id_opcion	Integer	Si	No	Clave de la opción donde se ejecutó la acción
valor	character varying(300)	Si	No	Almacena los cambios realizados en cada una de las opciones, llevando un registro
hora	character varying(10)	No	No	Hora en la que se realizó la acción
fecha	character varying(15)	No	No	Fecha en la que se realizó la acción
Restricciones				
Nombre		Tipo		Definición
pk_codigo_audadi		Clave Primaria		(id_auditoria)

Elaborado por: Marco Montaguano y Eduardo Pacheco

Tabla 16. Tabla de las opciones de menú

tbl_opcion				
Nombre Columna	Tipo de Dato	¿No Nulo?	Clave Primaria	Descripción
id_opcion	Integer	Si	Si	Clave primaria de la tabla
id_opcion_2	Integer	No	No	
enlace	character varying(30)	Si	No	Nombre de la opción con el que se encuentra programada
nombre	character varying(30)	Si	No	Nombre visual de la opción en el menú
paquete	character varying(60)	No	No	Nombre del paquete en el que se encuentra programado
auditada	boolean	Si	No	Si la opción es auditada o no
Restricciones				
Nombre	Tipo		Definición	
pk_codigo_sisopc	Clave Primaria		(id_opcion)	

Elaborado por: Marco Montaguano y Eduardo Pacheco

Tabla 17. Tabla de perfiles existentes para el sistema

tbl_perfil				
Nombre Columna	Tipo de Dato	¿No Nulo?	Clave Primaria	Descripción
id_perfil	Integer	Si	Si	Clave primaria de la tabla
nombre	character varying(30)	Si	No	Nombre del perfil
descripción	character varying(150)	No	No	Descripción del perfil
Restricciones				
Nombre	Tipo		Definición	
tb_sis_perfil_codigo_sisper_pk	Clave Primaria		(id_perfil)	

Elaborado por: Marco Montaguano y Eduardo Pacheco

Tabla 18. Tabla de opciones para cada perfil

tbl_perfil_opcion				
Nombre Columna	Tipo de Dato	¿No Nulo?	Clave Primaria	Descripción
id_perfil_opcion	Integer	Si	Si	Clave primaria de la tabla
id_opcion	Integer	Si	No	Clave de la opción habilitada para el perfil
id_perfil	Integer	Si	No	Clave del perfil
lectura	Boolean	Si	No	Configura el permiso en la opción para insertar
Restricciones				
Nombre	Tipo		Definición	
pk_codigo_sisopc	Clave Primaria		(id_perfil_opcion)	
tbl_opcion_tbl_perfil_opcion_fk	Clave Foranea		(id_opcion) REFERENCES tbl_opcion (id_opcion)	
tbl_perfil_tbl_perfil_opcion_fk	Clave Foranea		(id_perfil) REFERENCES tbl_perfil (id_perfil)	

Elaborado por: Marco Montaguano y Eduardo Pacheco

Tabla 19. Tabla de información del usuario

tbl_usuario				
Nombre Columna	Tipo de Dato	¿No Nulo?	Clave Primaria	Descripción
id_usuario	Integer	Si	Si	Clave primaria de la tabla
id_perfil	Integer	Si	No	Clave del perfil.
estado_sesion	Integer	Si	No	Estado en el que esta la sesión
cedula	character varying(20)	Si	No	Numero de cedula
nombre	character varying(10)	Si	No	Nombre con el que se realiza el login
nombre_completo	character varying(100)	Si	No	Nombre real del usuario, servirá para generación de reportes y como información complementaria
correo	character varying(150)	No	No	Correo institucional del usuario
correo_2	character varying(150)	No	No	Correo personal del usuario
clave	character varying(100)	Si	No	Clave con la que se realiza el login
tema	character varying(50)	Si	No	Tema con el cual se cargara la interfaz web
foto	character varying(50)	No	No	Foto del usuario
Restricciones				
Nombre	Tipo	Definición		
tb_sis_usuario_codigo_sisusu_pk	Clave Primaria	(id_usuario)		
tbl_perfil_tbl_usuario_fk	Clave Foranea	(id_perfil)	REFERENCES	tbl_perfil (id_perfil)

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3. Codificación

A continuación se hará mención a las partes más importantes del código desarrollado, se hará una explicación del framework utilizado, la forma cómo se realiza la conexión a la base de datos, cómo se realiza la gestión de usuarios, la gestión de permisos de acceso mediante perfiles de usuarios, cómo se gestiona y se realiza la auditoría. Por último, se explicará cómo se obtienen los reportes en las diferentes pantallas desarrolladas.

2.3.1. Framework X-PRIME.

Ya que el framework fue desarrollado por uno de los tesisistas, a continuación se muestra parte del código que hace posible la implementación del mismo dentro del desarrollo del tema.

- **Clase pf_tabla:**

Esta clase es la más importante del framework ya que sirve como capa de persistencia entre la base de datos y el desarrollo, es una implementación del componente DataTable de la librería PrimeFaces que permite hacer uso de todas sus opciones, además de algunas otras funciones adaptadas para la interacción con la base de datos. En la gráfica se muestra las opciones de configuración que tiene este componente:

```
public void setTabla(String tabla, String campoPrimaria) {  
    this.tabla = tabla;  
    this.campoPrimaria = campoPrimaria;  
    this.sql = "SELECT * FROM " + tabla;  
    formar_columnas();  
}
```

Figura 12. Método forma la tabla del framework X-PRIME

Elaborado por: Marco Montaguano y Eduardo Pacheco

El método formar_columnas() se encarga de hacer una consulta en la base de datos de la tabla que se va a construir, toma cada campo y lo agrega a un arreglo de columnas las cuales se van a mostrar junto al arreglo de filas en forma de una tabla similar a una hoja de cálculo de Excel, teniendo varias opciones: insertar, eliminar, modificar la celdas que el usuario desee.

2.3.2. Persistencia de conexión.

Una de las principales implementaciones que el framework facilita es la capa de persistencia. Junto a la clase pf_tabla, la clase cla_conexion interactúa con la base de datos directamente mediante un pool de conexiones JSF. En la gráfica se muestra

cómo se logra la conexión a la base de datos, mapeando los pool de conexiones dentro del entorno y mediante ResultSet realiza las acciones.

```
public cla_conexion() {  
    //obtiene el EntityManagerFactory de la unidad de persistencia prj_distribucion_horariosPU  
    fabrica = Persistence.createEntityManagerFactory(unidad_persistencia);  
    //obtiene el EntityManager del EntityManagerFactory  
    manejador = fabrica.createEntityManager();  
    manejador.setFlushMode(FlushModeType.COMMIT);  
    try {  
        //crea Context para la conexion  
        Context lcon_ctx = new InitialContext();  
        //obtiene el DataSource mediante un lookup al pool de conexiones  
        DataSource ldso_das = (DataSource) lcon_ctx.lookup("pool_postgres");  
        //obtiene y configura la conexión desde el DataSource  
        conexion_postgres = ldso_das.getConnection();  
        conexion_postgres.setAutoCommit(false);  
        conexion_postgres.setTransactionIsolation(Connection.TRANSACTION_READ_COMMITTED);  
        LOGGER.log(Level.INFO, "Conectado a pool_postgres.");  
    } catch (Exception ex) {  
        LOGGER.log(Level.SEVERE, "Ocurrio un error. Causa--> {0}", ex);  
        sis_soporte.obtener_instancia_soporte().crear_error(ex.getMessage(), "Clase Conexión");  
    }  
}
```

Figura 13. Constructor de la clase que conecta a la base de datos

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3.3. Pantalla de gestión de opciones.

Esta pantalla es exclusiva del framework, en ella se configuran las pantallas del sistema. Consta de dos paneles separados verticalmente, en el primero se encuentra un menú en forma de árbol recursivo y en el segundo una pantalla simple con la tabla de opciones en la que se realizará la configuración. Las dos pantallas están relacionadas a la misma tabla de opciones en la base de datos pero se presenta en formato distinto (menú y tabla). A continuación se muestra la pantalla de opciones.

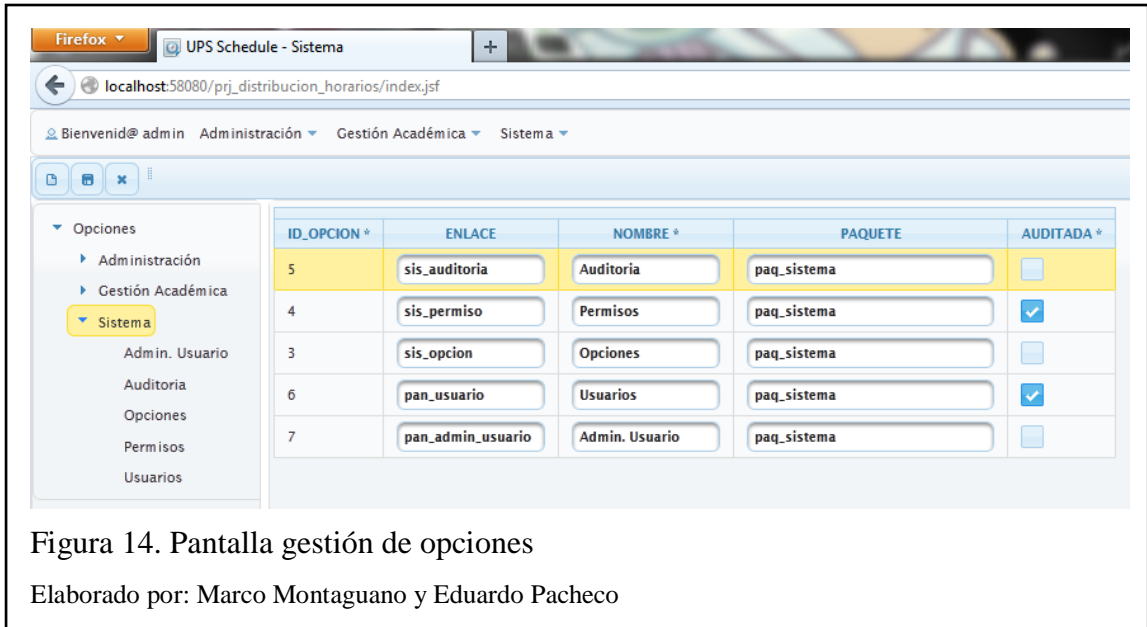


Figura 14. Pantalla gestión de opciones

Elaborado por: Marco Montaguano y Eduardo Pacheco

El código para la creación de esta pantalla se muestra a continuación.

```

public sis_opcion() {
    LOGGER.log(Level.INFO, "sis_opcion cargando");
    //configuracion del menu en forma de arbol recursivo
    arb_arbol.setId("arb_arbol");
    arb_arbol.configurar_arbol("tbl_opcion", "id_opcion", "nombre", "id_opcion_2", "Opciones");
    arb_arbol.onSelect("seleccionar_arbol", "tab_tabla1");
    arb_arbol.dibujar();

    //configuracion de la tabla de opciones
    tab_tabla.setId("tab_tabla1");
    tab_tabla.setTabla("tbl_opcion", "id_opcion");
    tab_tabla.setCampoPadre("id_opcion_2");
    tab_tabla.setFocus();
    tab_tabla.dibujar();
    pf_panel_tabla pat_panell = new pf_panel_tabla();
    pat_panell.setPanelTabla(tab_tabla);

    //panel division en forma vertical (arbol | tabla)
    div_division.setId("div_division");
    div_division.dividir2(arb_arbol, pat_panell, "21%", "V"); //arbol y divi

    gru_pantalla.getChildren().add(bar_botones); //agrega barra de botones
    gru_pantalla.getChildren().add(div_division); // agrega panel vertical

    //configura ajax en los botones insertar, guardar, eliminar
    bar_botones.getBot_insertar().setUpdate("tab_tabla1");
    bar_botones.getBot_guardar().setUpdate("tab_tabla1,arb_arbol");
    bar_botones.getBot_eliminar().setUpdate("tab_tabla1");

    //agrega la pantalla al panel correspondiente bajo el menu de opciones
    sis_soporte.obtener_instancia_soporte().agregar_pantalla(gru_pantalla);
}

```

Figura 15. Constructor para la pantalla gestión de opciones

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3.4. Gestión de usuarios.

La gestión de usuarios es una pantalla simple donde se permite el ingreso, modificación y eliminación de las cuentas de usuarios que ingresan al sistema. La siguiente figura muestra la pantalla del sistema.

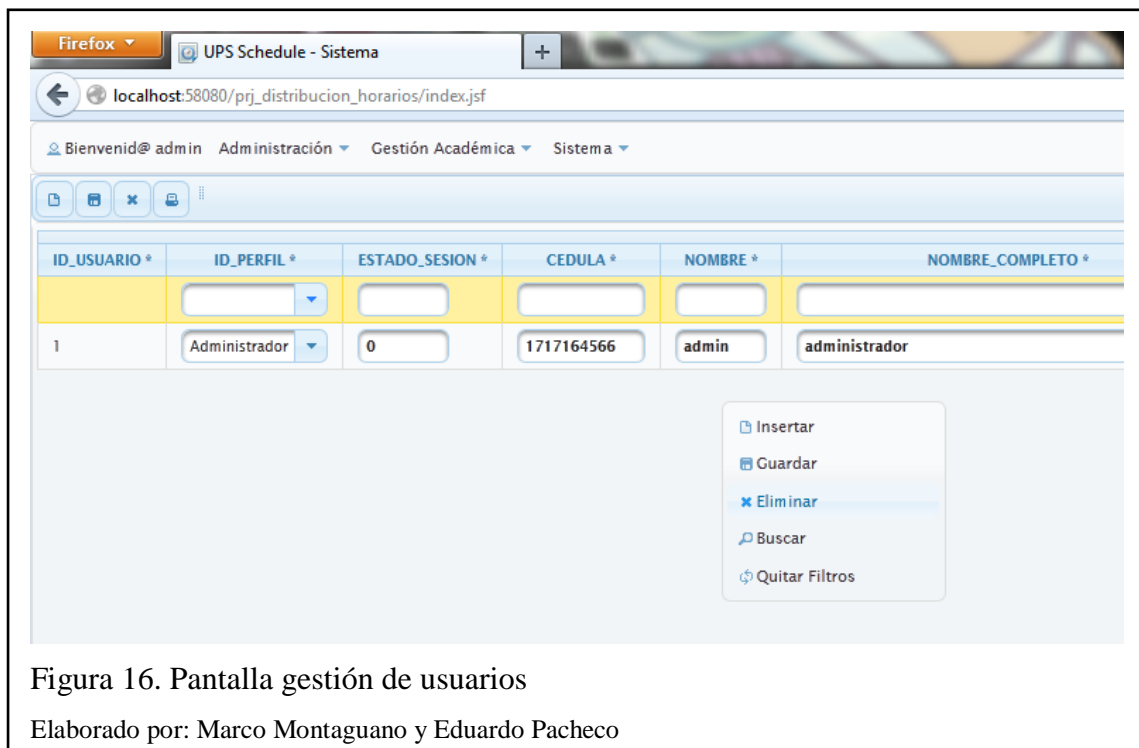


Figura 16. Pantalla gestión de usuarios

Elaborado por: Marco Montaguano y Eduardo Pacheco

La creación y configuración de la pantalla mediante el framework X-PRIME es sencilla, en la figura siguiente se muestra el código utilizado.

```

public pan_usuario() {
    LOGGER.log(Level.INFO, "pan_usuario cargando");
    tab_tabla.setId("tab_tabla"); //se configura el ID
    tab_tabla.setTabla("tbl_usuario", "id_usuario"); //se usa el metodo setTabla del framework
    tab_tabla.getColumna("id_perfil").configurar_combo("tbl_perfil", "id_perfil", "nombre", "");
    tab_tabla.setFocus();
    tab_tabla.dibujar(); //metodo del framework que construye la tabla.

    bar_botones.agregarReporte(); //agrega el boton de reportes en la barra de botones

    pf_panel_tabla pat_panel = new pf_panel_tabla(); //panel donde se carga la tabla
    pat_panel.setPanelTabla(tab_tabla); //se agrega la tabla al panel

    div_division.setId("div_division"); //se crea una division de pantalla simple
    div_division.dividir1(pat_panel);

    gru_pantalla.getChildren().add(bar_botones); //se agrega a la pantalla la barra de botones
    gru_pantalla.getChildren().add(div_division); //se agrga a la pantalla la division pantalla

    //se configura AJAX para cada boton, indicando el componente que va a actualizar
    //luego de realizar la accion asignada
    bar_botones.getBot_insertar().setUpdate("tab_tabla"); //boton insertar
    bar_botones.getBot_guardar().setUpdate("tab_tabla"); //boton guardar
    bar_botones.getBot_eliminar().setUpdate("tab_tabla"); //boton eliminar

    //se agrega la pantalla al panel correspondiente bajo la barra de menu
    sis_soporte.obtener_instancia_soporte().agregar_pantalla(gru_pantalla);
}

```

Figura 17. Constructor para la pantalla de gestión de usuarios

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3.5. Gestión de perfiles y permisos.

La gestión de perfiles y permisos se la realiza mediante una pantalla doble, en la parte superior se encuentran los perfiles de usuario y en la parte inferior las pantallas a las cuales tiene acceso. A continuación se muestra la pantalla del sistema.

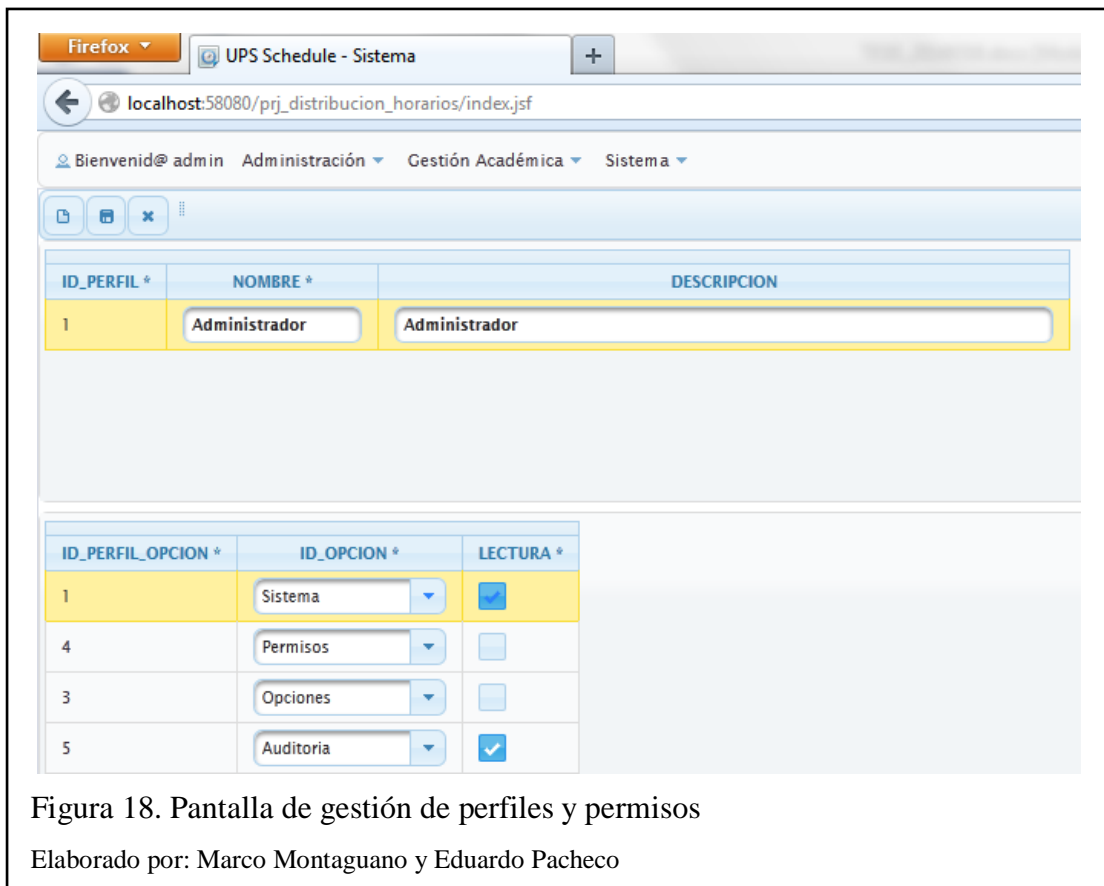


Figura 18. Pantalla de gestión de perfiles y permisos

Elaborado por: Marco Montaguano y Eduardo Pacheco

La pantalla se configura mediante el framework X-PRIME como se muestra en el siguiente gráfico, se usan dos tablas relacionadas mediante un código foráneo. Tanto en Perfiles como en Permisos (pantalla superior e inferior respectivamente) se pueden realizar acciones tanto de inserción como de modificación y eliminación de registros.

```

public sis_permiso() {
    LOGGER.log(Level.INFO, "sis_permiso cargando");
    //configuracion de la tabla de perfiles
    tab_tabla_pri.setId("tab_tabla_pri");
    tab_tabla_pri.setTabla("tbl_perfil", "id_perfil");
    tab_tabla_pri.onSelect("seleccionar_tabla_pri", "tab_tabla_sec");
    tab_tabla_pri.agregarRelacion(tab_tabla_sec); //agrega la relacion con la tabla de permisos
    tab_tabla_pri.setFocus();
    tab_tabla_pri.dibujar();
    pf_panel_tabla pat_panel_pri = new pf_panel_tabla();
    pat_panel_pri.setPanelTabla(tab_tabla_pri);

    //configuracion de la tabla de permisos
    tab_tabla_sec.setId("tab_tabla_sec");
    tab_tabla_sec.setTabla("tbl_perfil_opcion", "id_perfil_opcion");
    tab_tabla_sec.setCampoForanea("id_perfil");
    tab_tabla_sec.getColumna("id_opcion").configurar_combo("tbl_opcion", "id_opcion", "nombre", "");
    tab_tabla_sec.setRows(20);
    tab_tabla_sec.dibujar();
    pf_panel_tabla pat_panel_sec = new pf_panel_tabla();
    pat_panel_sec.setPanelTabla(tab_tabla_sec);

    //panel de division horizontal
    div_division.setId("div_division");
    div_division.dividir2(pat_panel_pri, pat_panel_sec, "40%", "H");

    gru_pantalla.getChildren().add(bar_botones);
    gru_pantalla.getChildren().add(div_division);

    //ajax para los botones insertar, guardar, eliminar
    bar_botones.getBot_insertar().setUpdate("tab_tabla_pri,tab_tabla_sec");
    bar_botones.getBot_guardar().setUpdate("tab_tabla_pri,tab_tabla_sec");
    bar_botones.getBot_eliminar().setUpdate("tab_tabla_pri,tab_tabla_sec");

    //se agrega la pantalla al panel correspondiente bajo el menu de opciones
    sis_soporte.obtener_instancia_soporte().agregar_pantalla(gru_pantalla);
}

```

Figura 19. Constructor para la pantalla de gestión de perfiles y permisos

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3.6. Gestión de auditoría.

La gestión de Auditoría es una pantalla simple de sólo lectura, en esta se registran todas las acciones (insertar, modificar, eliminar) realizadas en las diferentes pantallas del sistema. Esta pantalla contiene un filtro de fechas tomando en cuenta que las acciones de auditoría por día son elevadas dependiendo de las pantallas sobre las cuales se está realizando el control, siempre se presenta la auditoría del día. En la figura siguiente se muestra la pantalla.

Desde: 2014-01-01 Hasta: 2014-03-19

ID_AUDITORIA *	ID_ACCION *	ID_USUARIO *	ID_OPCION *	VALOR *
13	Guardar	administrador	Usuarios	DELETE FROM tbl_usuario WHERE id_usuario="2"
14	Guardar	administrador	Usuarios	DELETE FROM tbl_usuario WHERE id_usuario="3"
15	Guardar	administrador	Usuarios	DELETE FROM tbl_usuario WHERE id_usuario="4"
16	Guardar	administrador	Permisos	DELETE FROM tbl_perfil WHERE id_perfil="2"
17		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 10, 1, false)
18		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 1, false)
19		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 12, 1, false)
20		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 11, 1, false)
21		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 8, 1, false)
22		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 13, 1, false)
23		administrador	Permisos	INSERT INTO tbl_perfil_opcion(ID_PERFIL_OPCION, ID_OPC 14, 1, false)

Figura 20. Pantalla de lectura para la gestión de auditoría

Elaborado por: Marco Montaguano y Eduardo Pacheco

La codificación de la pantalla se muestra en la siguiente figura.


```

public sis_auditoria() {
    LOGGER.log(Level.INFO, "sis_auditoria cargando");
    //configuracion de la tabla de auditoria
    tab_tabla.setId("tab_tabla");
    tab_tabla.setTabla("tbl_auditoria", "id_auditoria");
    tab_tabla.getColumna("id_accion").configurar_combo("tbl_accion", "id_accion", "nombre", "");
    tab_tabla.getColumna("id_usuario").configurar_combo("tbl_usuario", "id_usuario", "nombre_completo", "");
    tab_tabla.getColumna("id_opcion").configurar_combo("tbl_opcion", "id_opcion", "nombre", "");
    tab_tabla.setCondicionBuscar("fecha='" + sis_soporte.obtener_instancia_soporte().obtener_fecha_actual() + "'");
    tab_tabla.setLectura(true); //se configura para que sea solo de lectura
    tab_tabla.setFocus();
    tab_tabla.dibujar(); //construye la tabla con las configuradas asignadas

    bar_botones.agregarReporte(); //agrega el boton de reportes a la barra de botones

    //panel donde se aloja la tabla
    pf_panel_tabla pat_panel = new pf_panel_tabla();
    pat_panel.setPanelTabla(tab_tabla);

    //division simple para la pantalla
    div_division.setId("div_division");
    div_division.dividir1(pat_panel);

    gru_pantalla.getChildren().add(bar_botones); //agrega la barra de botones al panel
    gru_pantalla.getChildren().add(div_division); //agrega la pantalla al panel

    bar_botones.getBot_insertar().setRendered(false); //oculta boton insertar
    bar_botones.getBot_guardar().setRendered(false); //oculta boton guardar
    bar_botones.getBot_eliminar().setRendered(false); //oculta boton eliminar

    bar_botones.agregarFiltro(); //agrega el panel de filtro por rango de fechas
    bar_botones.getBot_filtrar().setUpdate("tab_tabla"); //ajax para el boton filtro

    //agrega la pantalla al panel correspondiente bajo el menu de opciones
    sis_soporte.obtener_instancia_soporte().agregar_pantalla(gru_pantalla);
}

```

Figura 21. Constructor para la pantalla de gestión de auditoría

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3.7. Comunicación.

Dentro del sistema se ha incluido una herramienta que le permite al usuario logeado comunicarse directamente con otros usuarios mediante el envío de correos electrónicos a cuentas personales o institucionales ingresados cuando se registra un usuario nuevo dentro del sistema.

Los parámetros de configuración para el envío de salida se las hacen en las siguientes líneas de código dentro del paquete “paq_sistema” y en la clase “EnviarCorreo”:

```

/**
 *
 * @author xEdushx
 */
public class EnviarCorreo {

    private Properties propiedades = new Properties();
    private String host = "mail.mplus.ec";
    private String puerto = "587";
    private String correo = "helpdesk@mplus.ec";
    private String clave = "";
}

```

Figura 22. Configuración de parámetros para salida de correo

Elaborado por: Marco Montaguano y Eduardo Pacheco

La interfaz para envío de correos permite al usuario seleccionar de una lista de usuarios el o los destinatarios deseados. El campo “Destinatarios” está bloqueado para que solo se envíen correos a quienes estén registrados en el sistema, evitando así el mal uso de la herramienta. Posee un editor de texto HTML, permitiendo al usuario redactar su mensaje con el formato que desee.

A continuación se muestra la interfaz y sus características:



Figura 23. Botón en la barra de menú para el acceso al correo

Elaborado por: Marco Montaguano y Eduardo Pacheco

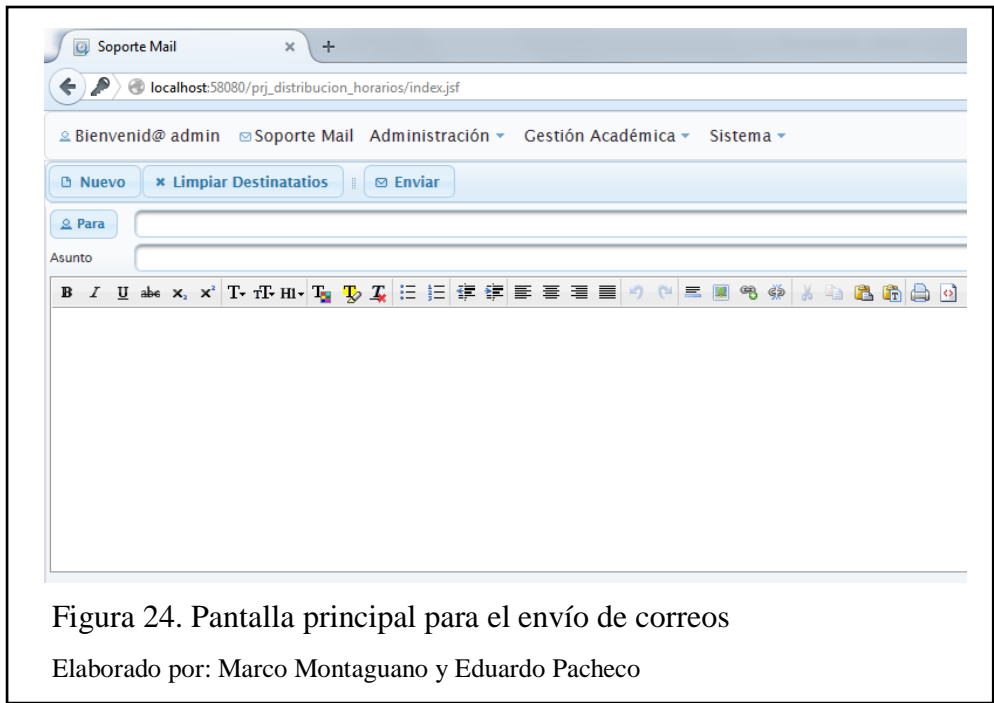


Figura 24. Pantalla principal para el envío de correos

Elaborado por: Marco Montaguano y Eduardo Pacheco

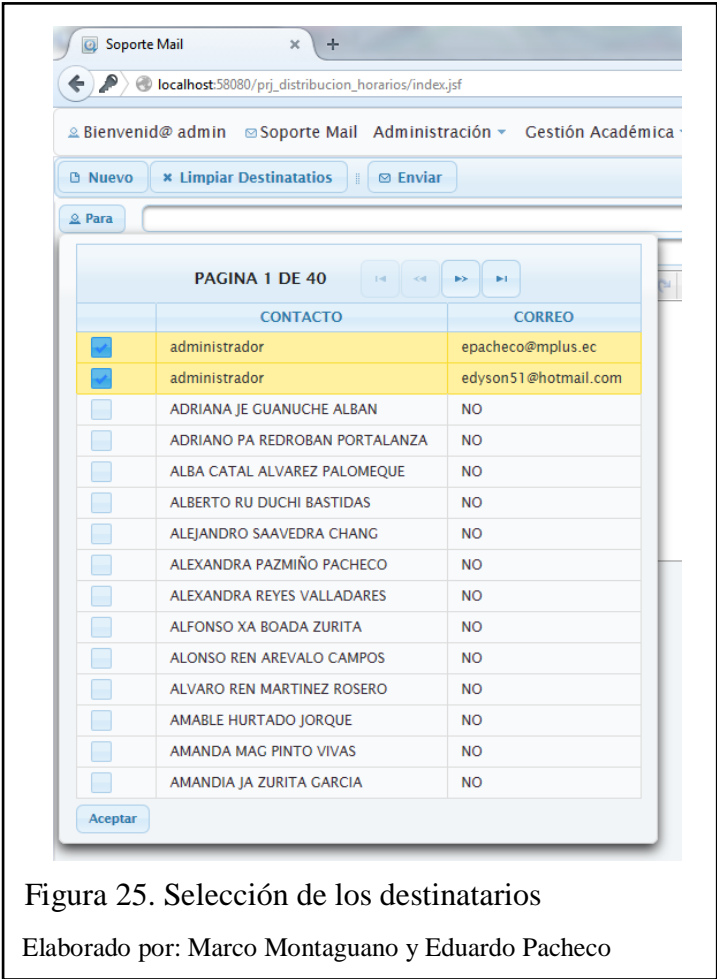


Figura 25. Selección de los destinatarios

Elaborado por: Marco Montaguano y Eduardo Pacheco

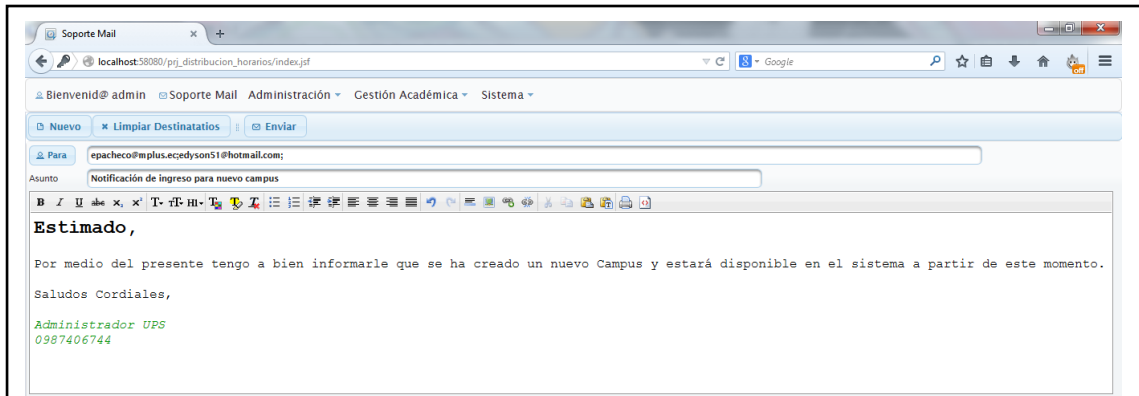


Figura 26. Redacción del correo

Elaborado por: Marco Montaguano y Eduardo Pacheco

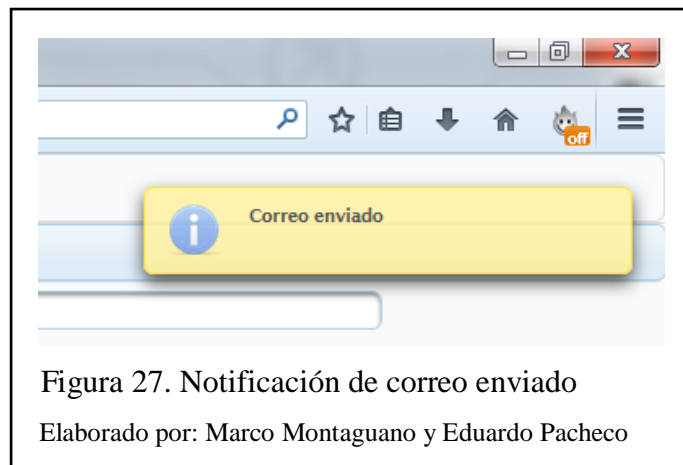


Figura 27. Notificación de correo enviado

Elaborado por: Marco Montaguano y Eduardo Pacheco

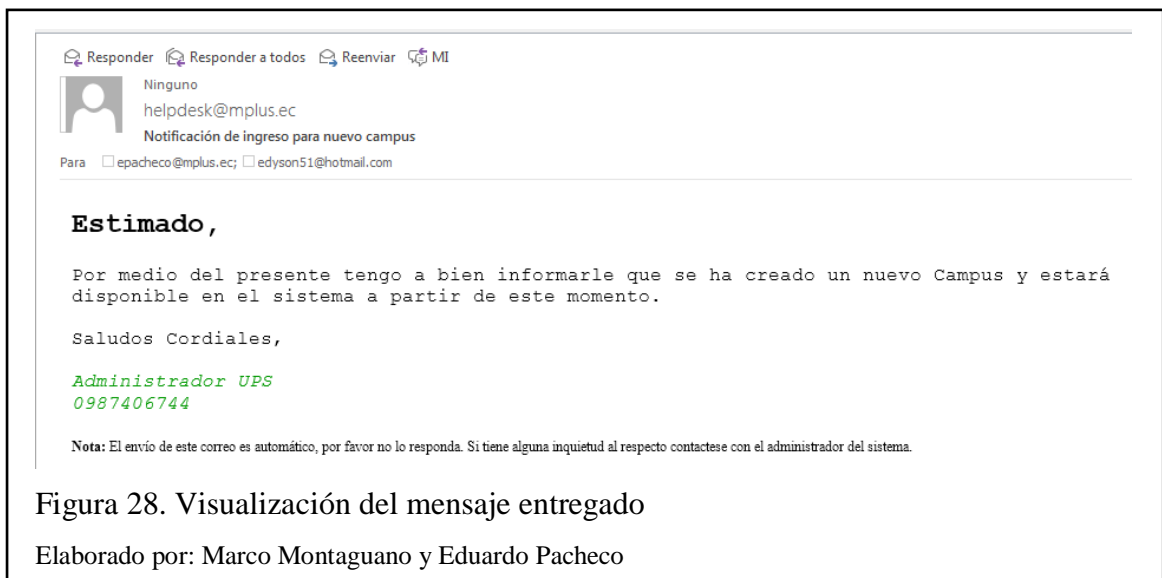


Figura 28. Visualización del mensaje entregado

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.3.8. Reportes.

Los reportes que genera el módulo de gestión de usuarios, auditoría y seguridad son específicamente dos: reporte de auditoría y reporte de usuarios. Estos están disponibles para descarga en formatos PDF, Word y Excel. Cada uno tiene diferentes filtros, los cuales dependiendo de la pantalla se irán presentando hasta la selección del formato de impresión.

▪ Reporte de auditoría

En este reporte se muestra las acciones realizadas en un lapso de tiempo, luego de seleccionar los filtros: reportes de auditoría, rango de fechas, formato de impresión.

A continuación se presentan los filtros y finalmente el reporte generado en formato PDF.

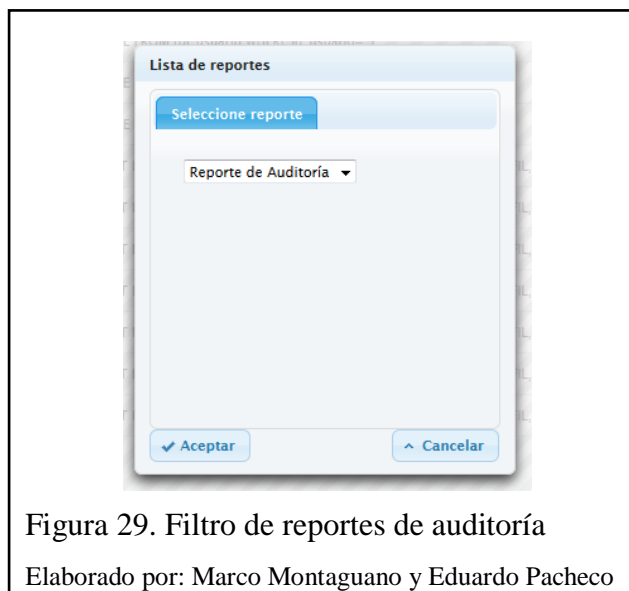


Figura 29. Filtro de reportes de auditoría

Elaborado por: Marco Montaguano y Eduardo Pacheco



Figura 30. Filtro de rango de fechas

Elaborado por: Marco Montaguano y Eduardo Pacheco

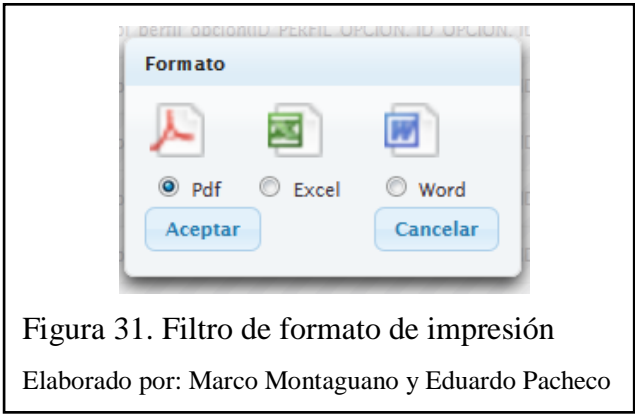


Figura 31. Filtro de formato de impresión

Elaborado por: Marco Montaguano y Eduardo Pacheco

Reporte de Auditoría

UPS Schedule

PERFIL	OPCIÓN	ACCIÓN	FECHA	HORA	VALOR
administrador					
Administrador	Usuarios	Guardar	2014-02-01	10:31:32	DELETE FROM tbl_usuario WHERE id_usuario="2"
Administrador	Usuarios	Guardar	2014-02-01	10:31:32	DELETE FROM tbl_usuario WHERE id_usuario="3"
Administrador	Usuarios	Guardar	2014-02-01	10:31:32	DELETE FROM tbl_usuario WHERE id_usuario="4"
Administrador	Permisos	Guardar	2014-02-01	11:00:44	DELETE FROM tbl_perfil WHERE id_perfil="2"

miércoles 19 marzo 2014
Página 1 de 1

Figura 32. Reporte en formato pdf de auditoría

Elaborado por: Marco Montaguano y Eduardo Pacheco

- **Reporte de usuarios**

En este reporte se muestran los usuarios gestionados, luego de seleccionar los filtros: reportes de usuario, formato de impresión.

A continuación se presentan los filtros y finalmente el reporte generado en formato PDF.

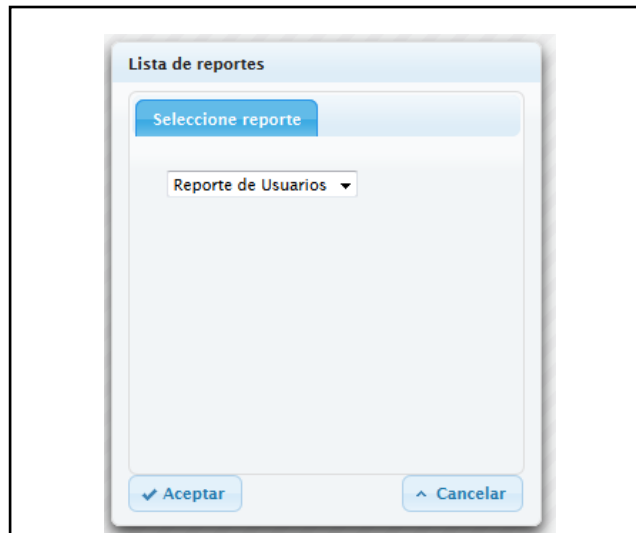


Figura 33. Filtro de reportes de usuario

Elaborado por: Marco Montaguano y Eduardo Pacheco

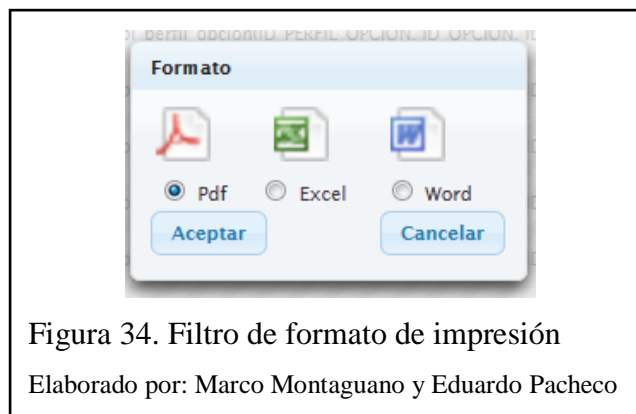


Figura 34. Filtro de formato de impresión

Elaborado por: Marco Montaguano y Eduardo Pacheco

Reporte de Usuarios							UPS Schedule
ID	CÉDULA	USUARIO	NOMBRES	CORREO	CORREO 2	TEMA	
1	1717164566	admin	administrador	a@gmail.com		cupertino	

Figura 35. Reporte de usuarios

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.4. Pruebas

A continuación, se hará mención a los resultados obtenidos en cada una de las pruebas realizadas al módulo. Estas pruebas se realizaron en tres fases: funcionales, carga y usabilidad.

2.4.1. Pruebas funcionales.

Estas pruebas se basan en el cumplimiento de requerimientos funcionales y específicos, a continuación se presenta una tabla en la que se describe cada uno de ellos.

▪ **Gestión de usuarios, perfiles, permisos y auditoría.**

Tabla 20. Resultados pruebas funcionales

No.	Nombre de la prueba	Req.	Pasos	OK	Observaciones
1	Crear usuarios.	1.1	1. Acceder al sistema como administrador. 2. Ingresar a la página de usuarios mediante el menú de opciones. 3. Dar clic en el ícono INSERTAR. (Opcional clic derecho - Insertar). 4. Llenar los campos obligatorios. 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar).	Sí	Antes de guardar, se pueden crear varios registros en la misma transacción antes de guardarla.
2	Modificar usuarios.	1.1	1. Acceder al sistema como administrador. 2. Ingresar a la página de usuarios mediante el menú de opciones. 3. Seleccionar la fila del usuario a modificar. 4. Modificar los datos que se desee. 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar).	Sí	Se pueden modificar varios registros en una sola transacción antes de guardarla.
3	Eliminar usuarios.	1.1	1. Acceder al sistema como administrador. 2. Ingresar a la página de usuarios mediante el menú de opciones. 3. Seleccionar la fila del usuario a eliminar. 4. Dar clic en el ícono ELIMINAR. (Opcional clic derecho – eliminar sobre la fila). 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar).	Sí	Se pueden eliminar varios registros en una sola transacción antes de guardarla.
4	Modificar datos del usuario logeado.	1.2	1. Acceder al sistema. (Cualquier usuario). 2. Ingresar a la página de administración del usuario mediante el menú de opciones. 3. Cambiar los datos del usuario presentados en un formulario. 4. Dar clic en el ícono GUARDAR.	Sí	EL usuario tiene la opción de modificar sus datos personales así como la clave de acceso y el tema visual que utiliza la aplicación.
5	Crear opción.	1.3	1. Acceder al sistema como administrador. 2. Ingresar a la página de opciones mediante el menú de opciones. 3. Seleccionar del árbol de opciones el nivel donde se va a crear la opción, colocarse en la tabla y dar clic en el ícono INSERTAR. (Opcional clic derecho - Insertar). 4. Llenar los campos obligatorios. 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar).	Sí	Antes de guardar, se pueden crear varios registros en la misma transacción antes de guardarla.

Continúa...

Tabla 20. Resultados pruebas funcionales (continuación...)

6	Modificar opción.	1.3	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de opciones mediante el menú de opciones. 3. Seleccionar la fila de la opción a modificar. 4. Modificar los datos que se desee. 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar). 	Sí	Se pueden modificar varios registros en una sola transacción antes de guardarla.
7	Eliminar opción.	1.3	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de opciones mediante el menú de opciones. 3. Seleccionar la fila de la opción a eliminar. 4. Dar clic en el ícono ELIMINAR. (Opcional clic derecho – eliminar sobre la fila). 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar). 	Sí	Se pueden eliminar varios registros en una sola transacción antes de guardarla.
8	Crear perfil– permisos.	1.4	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de perfiles mediante el menú de opciones. 3. Colocarse en la tabla Perfiles (para insertar un perfil) o Permisos (para insertar un permiso al perfil seleccionado), dar clic en el ícono INSERTAR. (Opcional clic derecho - Insertar). 4. Llenar los campos obligatorios. 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar). 	Sí	Antes de guardar, se pueden crear varios registros en la misma transacción antes de guardarla. Tener en cuenta que las tablas son relacionadas de la siguiente forma: un perfil tiene varios permisos.
9	Modificar perfil – permisos.	1.4	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de perfiles mediante el menú de opciones. 3. Seleccionar la fila del perfil o permiso a modificar. 4. Modificar los datos que se desee. 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar). 	Sí	Se pueden modificar varios registros en una sola transacción antes de guardarla.
10	Eliminar perfil – permisos.	1.4	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de perfiles mediante el menú de opciones. 3. Seleccionar la fila del perfil o permiso a eliminar. 4. Dar clic en el ícono ELIMINAR. (Opcional clic derecho – eliminar sobre la fila). 5. Dar clic en el ícono GUARDAR. (Opcional clic derecho - Guardar). 	Sí	Se pueden eliminar varios registros en una sola transacción antes de guardarla.
11	Gestionar auditoría.	1.5	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de auditoría mediante el menú de opciones. 3. Ingresar en los filtros, las fechas de inicio y fin para visualizar la auditoría. 	Sí	Al ingresar a la pantalla se muestran los datos de auditoría del día actual.

Elaborado por: Marco Montaguano y Eduardo Pacheco

▪ **Reportes**

Tabla 21. Resultados pruebas funcionales generación de reportes

No.	Nombre de la prueba	Req.	Pasos	OK	Observaciones
1	Generar reporte de auditoría.	2.1	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de auditoría mediante el menú de opciones. 3. Dar clic en el ícono REPORTEES. 4. Seleccionar los filtros. 5. Escoger el formato de impresión del reporte. 	Sí	<p>La generación del reporte toma aproximadamente 8 segundos con filtros que despliegan 50 registros.</p> <p>La opción de FORMATO se mantiene en pantalla, permitiendo al usuario generar el reporte en otros formatos. Cuando hace esto, el tiempo se reduce a 3 segundos ya que el reporte está creado previamente en la primera impresión.</p>
2	Generar reporte de usuarios.	2.2	<ol style="list-style-type: none"> 1. Acceder al sistema como administrador. 2. Ingresar a la página de auditoría mediante el menú de opciones. 3. Dar clic en el ícono REPORTEES. 4. Seleccionar los filtros. 5. Escoger el formato de impresión del reporte. 	Sí	<p>La generación del reporte toma aproximadamente 10 segundos desplegando 50 registros.</p> <p>La opción de FORMATO se mantiene en pantalla, permitiendo al usuario generar el reporte en otros formatos. Cuando hace esto, el tiempo se reduce a 3 segundos ya que el reporte está creado previamente en la primera impresión.</p>

Elaborado por: Marco Montaguano y Eduardo Pacheco

2.4.2. Pruebas de carga.

Las pruebas de carga se la realizaron bajo el criterio de tiempo de respuesta en realizar una transacción sobre cada pantalla del módulo. Los resultados se muestran en la siguiente tabla, los tiempos son promedios obtenidos en cada navegador Google Chrome (GC), Firefox (FRX), Safari (SFR), Internet Explorer (IE) luego de diez pruebas sobre cada uno.

Tabla 22. Resultados pruebas de carga en base a funcionalidades en pantallas de gestión de usuarios, permisos y auditoría

No.	Nombre de la prueba	Tiempo (segundos)				O K	Observaciones
		GC	FRX	SFR	IE		
1	Ingresar al sistema.	1050	850	1070	1120	Sí	
2	Ingresar a la pantalla de gestión de perfiles - permisos.	450	400	420	350	Sí	La pantalla inicia con el perfil de administrador y todos sus permisos, en total se muestran 50 registros.
3	Transacciones sobre la pantalla de perfiles - permisos.	300	220	280	350	Sí	Creación de un perfil con permisos de acceso a las pantallas del administrador. Modificación y eliminación del perfil. (Tiempo promedio en estas operaciones).
4	Ingresar a la pantalla de gestión de opciones	250	250	230	280	Sí	Se despliegan los tres módulos del sistema.
5	Transacciones sobre la pantalla de opciones.	200	160	200	220	Sí	
6	Ingresar a la pantalla de Usuarios.	250	150	160	180	Sí	Se despliegan cinco usuarios.
7	Transacciones sobre la pantalla de usuarios.	200	180	190	230	Sí	Se crea un usuario, se modifican los datos y se elimina el usuario. (Tiempo promedio en estas operaciones).
8	Generar el reporte de usuarios.	4240	4150	4200	4460	Sí	Se despliega el reporte en formato PDF, cuando se selecciona otro formato (sin cerrar el filtro de formato) se demora 1 segundo.
9	Ingresar a la pantalla de gestión de auditoría.	1200	1130	1180	1220	Sí	Se despliega la información auditada en el día (30 registros).
10	Filtrar la pantalla de gestión de auditoría.	1100	1040	1080	1130	Sí	Cuando el filtro despliega 30 registros.
11	Generar el reporte de Auditoría.	4020	3400	3520	4210	Sí	Cuando los filtros despliegan 30 registros.
12	Cambiar el tema del usuario logeado.	1020	850	830	1210	Sí	El tema se cambia en tiempo real mientras el usuario lo selecciona de un combo box.
13	Cambiar la clave del usuario logeado.	1050	820	820	1130	Sí	
PROMEDIO		1179	1046	1091	1238		

Elaborado por: Marco Montaguano y Eduardo Pacheco

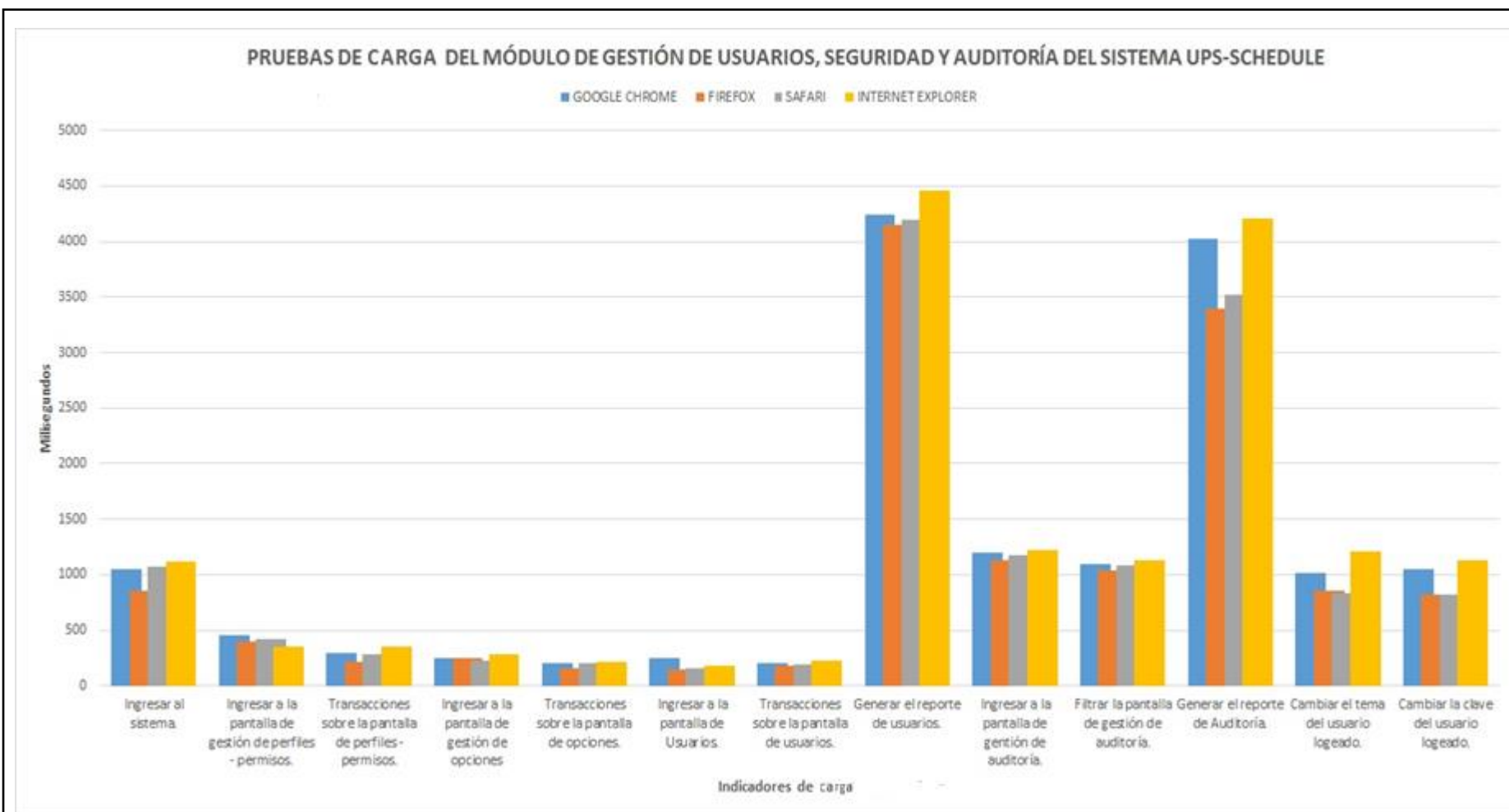


Figura 36. Gráfico comparativo de rendimiento entre los navegadores Google Chrome, Firefox, Safari, Internet Explorer

Elaborado por: Marco Montaguano y Eduardo Pacheco

Los tiempos en las transacciones realizadas son muy bajos debido a la comunicación con la capa de persistencia que maneja el framework X-PRIME. Las variaciones dependen de cuántos datos se muestran en la pantalla y la carga del servidor web. El número máximo de usuarios recurrentes que permite la aplicación (funcionando en óptimas condiciones) es de 10, pasado esto el sistema baja el rendimiento en un 5% por cada usuario adicional debido al consumo de recursos en el servidor de aplicaciones.

2.4.3. Pruebas de usabilidad.

Estas pruebas se realizaron para ver el nivel de complejidad que tiene el manejo del módulo y cuán intuitivo es realizar las transacciones en cada pantalla de gestión, dirigida a usuarios con perfiles administrativos.

La muestra fue de 10 usuarios, los cuales usaron la aplicación en cada uno de sus dispositivos, de la siguiente manera: 6 usuarios con notebooks (navegadores Firefox, Chrome, Internet Explorer), 2 usuarios Android (navegadores Firefox, Chrome), 2 usuarios IOS (navegador Safari).

Se utilizó un test desarrollado por la empresa *INTERGRAPHIC DESIGNS S.A.*, el cual está disponible en versión libre para su descarga en “InterGraphicDesign” (InterGraphic Design Agency, 2014)

El test tiene los siguientes indicadores, dentro de los cuales se consideran varios criterios. Ver Anexo (Formato Test de Usabilidad):

- Página de inicio.
- Orientación a tareas y funcionalidad del sitio.
- Navegabilidad y arquitectura de la información.
- Formularios y entrada de datos.
- Confianza y credibilidad.
- Calidad del contenido y escritura.
- Diagramación y diseño gráfico.

- Búsquedas.
- Ayuda, retroalimentación y recuperación de errores.

Los resultados obtenidos se muestran a continuación, cabe mencionar que los valores mostrados son un valor promedio entre quienes participaron del test.

Tabla 23. Resultados pruebas de usabilidad

Indicadores	Calificación Neta	# Preguntas	Calificación
Página de Inicio	14	20	85%
Orientación a Tareas y Funcionalidad del Sitio	30	44	84%
Navegabilidad y Arq. De la Información	20	29	84%
Formularios y entrada de datos	19	23	91%
Confianza y Credibilidad	11	13	92%
Calidad del Contenido y Escritura	18	23	89%
Diagramación y Diseño Gráfico	38	38	100%
Búsquedas	-5	20	38%
Ayuda, Retroalimentación & Recuperación de Errores	19	37	76%
Calificación Final		247	82%

Elaborado por: Marco Montaguano y Eduardo Pacheco

La siguiente figura muestra el porcentaje que abarca la aplicación según los indicadores calificados en la prueba, el rango más óptimo es 100% y el mínimo 0%. La línea azul indica el nivel donde se encuentra la aplicación.

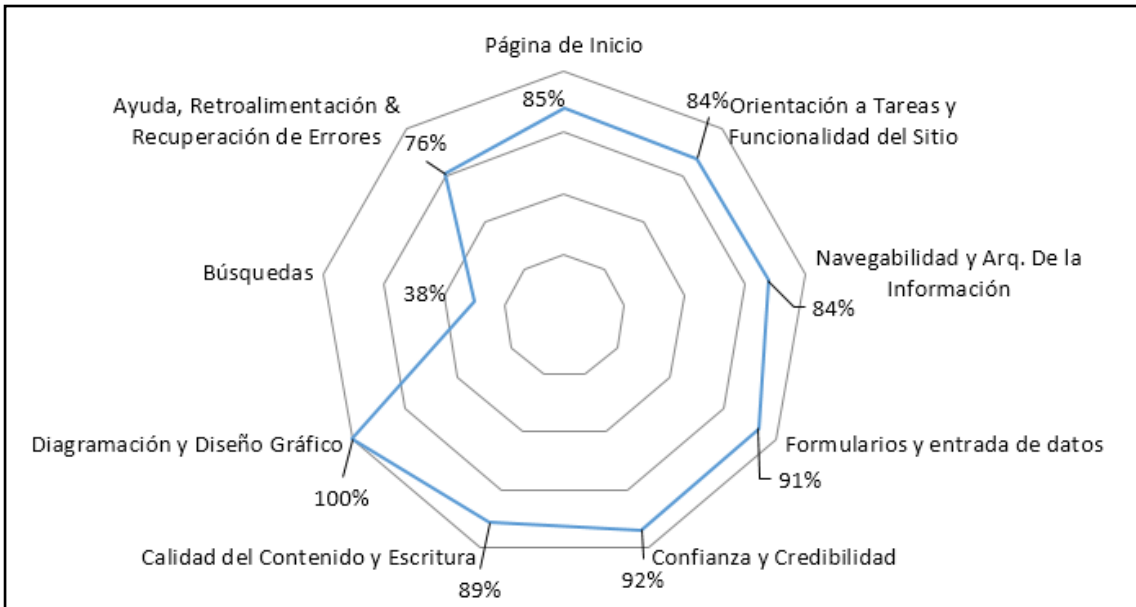


Figura 37. Resultados gráficos de pruebas de usabilidad

Elaborado por: Marco Montaguano y Eduardo Pacheco

CAPÍTULO 3

INTEGRACIÓN

En este capítulo se detallará la forma como se logró la integración de los tres módulos que conforman la parte administrativa del sistema UPS Schedule. Se incluirá un diagrama de despliegue para la arquitectura de la aplicación integrada.

3.1. Diagrama de despliegue

A continuación se muestra el diagrama de despliegue de la arquitectura sobre la cual está desarrollada la solución integrada. Debido a esta arquitectura junto con el framework X-PRIME, la integración no es más que “copiar & pegar” los paquetes que contienen las clases .JAVA correspondientes a cada módulo, luego de realizar ligeros cambios en ciertas partes de código, el sistema queda integrado físicamente.

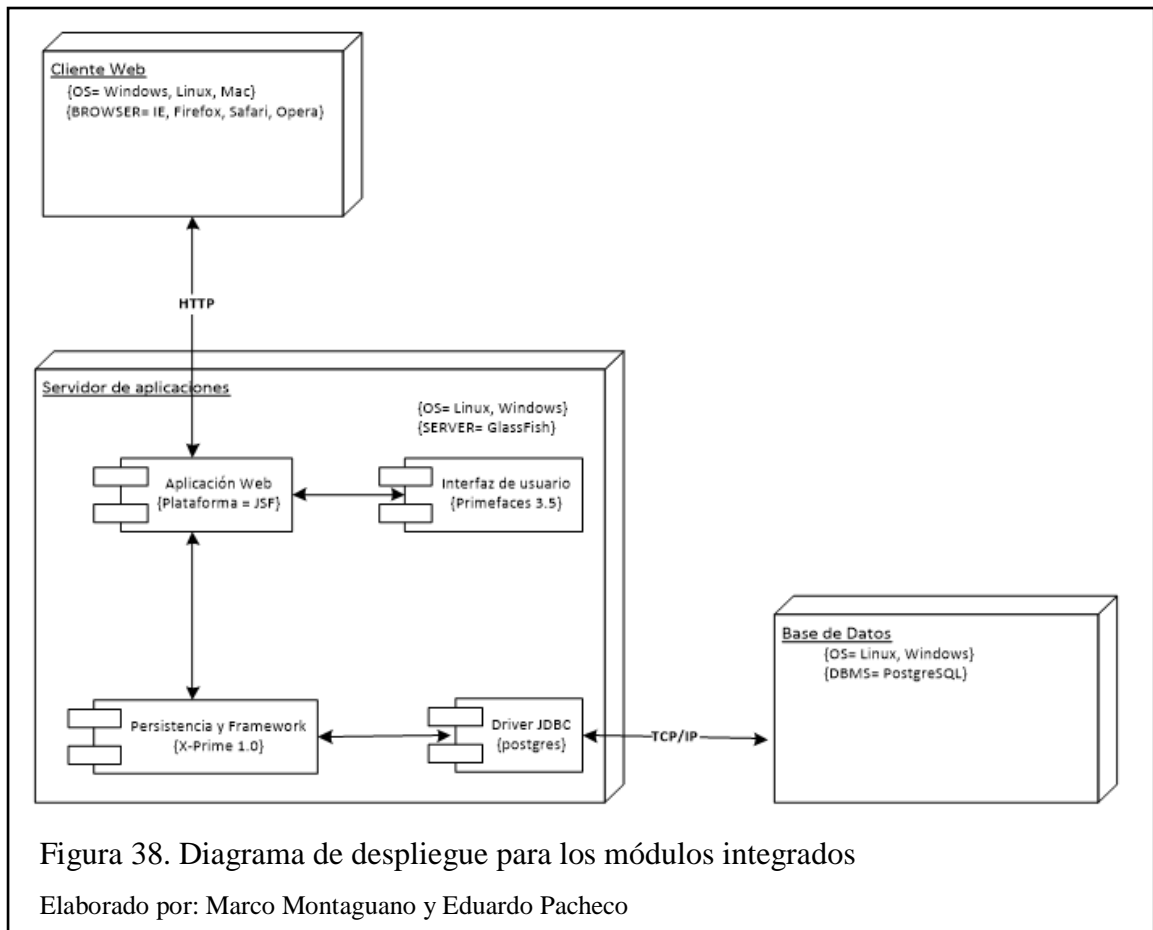


Figura 38. Diagrama de despliegue para los módulos integrados

Elaborado por: Marco Montaguano y Eduardo Pacheco

Para lograr la integración lógica – funcional, es necesario igualar la base de datos; esto se realiza de la siguiente manera:

- Ejecutar la aplicación integrada.
- Ingresar como administrador.
- Dirigirse a la pantalla de “Opciones”.
- Configurar los módulos y las páginas de cada uno.
- Dirigirse a la pantalla de “Permisos”.
- Dar permisos de acceso a los módulos y pantallas correspondientes a cada perfil.



Figura 39. Pantalla de opciones, configuración de los módulos

Elaborado por: Marco Montaguano y Eduardo Pacheco

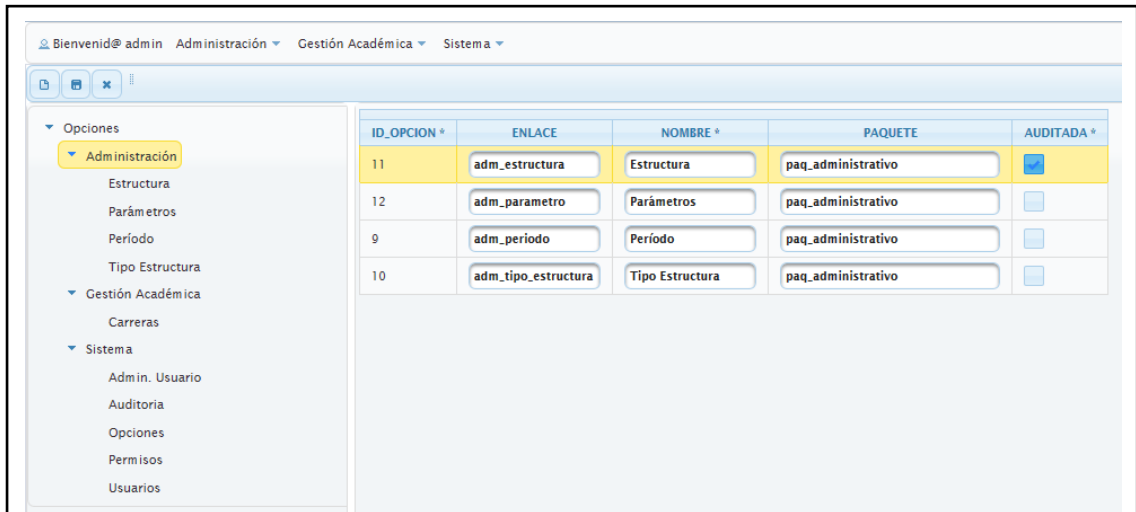


Figura 40. Pantalla de opciones, configuración del módulo de administración

Elaborado por: Marco Montaguano y Eduardo Pacheco

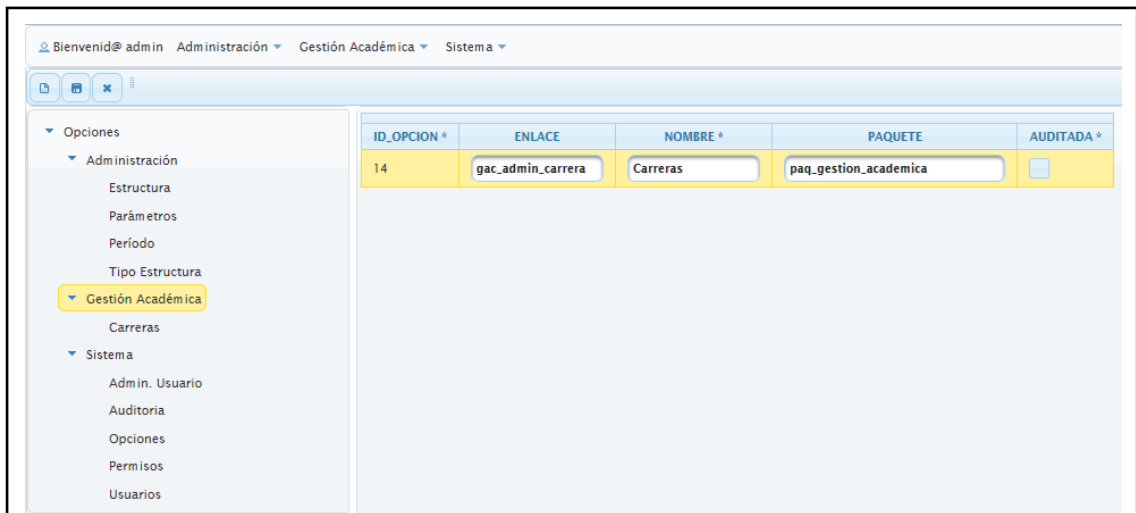


Figura 41. Pantalla de opciones, configuración del módulo de gestión académica

Elaborado por: Marco Montaguano y Eduardo Pacheco

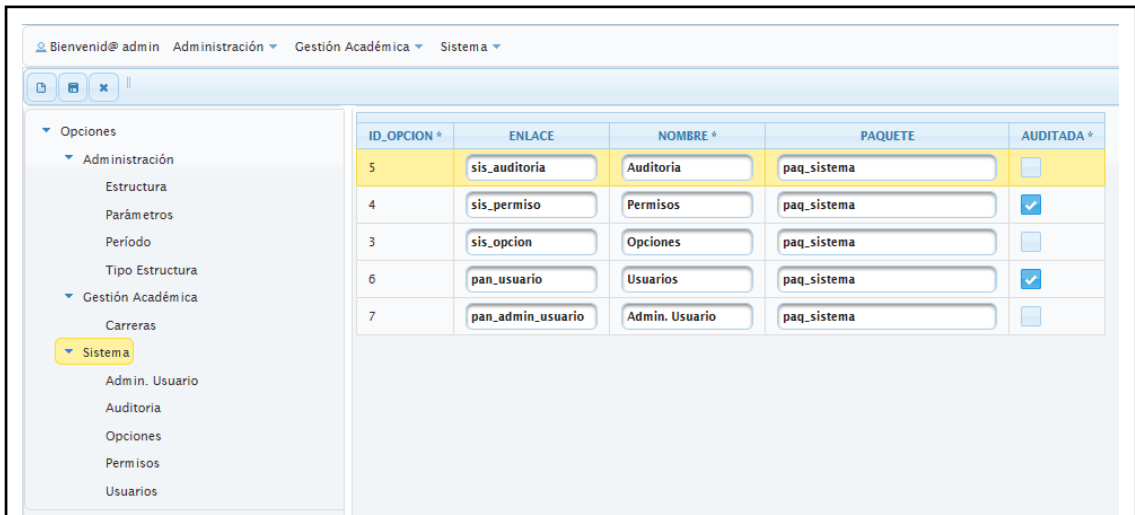


Figura 42. Pantalla de opciones, configuración del módulo de sistema

Elaborado por: Marco Montaguano y Eduardo Pacheco

- Dirigirse a la pantalla de “Permisos”.
- Dar permisos de acceso a los módulos y pantallas correspondientes a cada perfil.

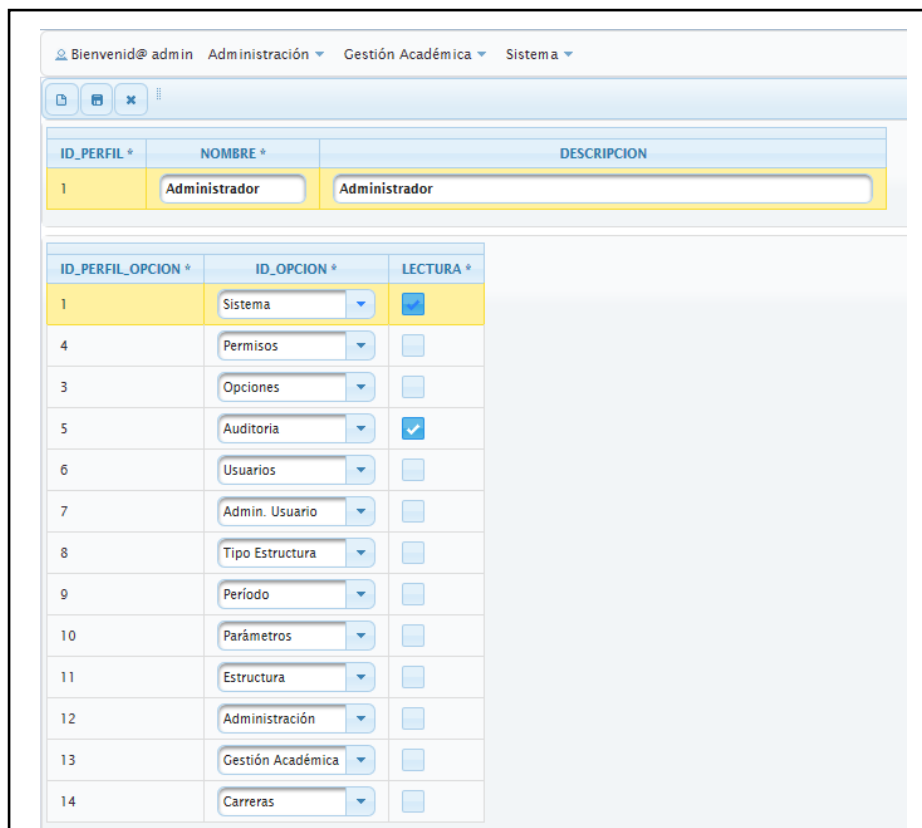


Figura 43. Pantalla de perfiles, asignación de permisos

Elaborado por: Marco Montaguano y Eduardo Pacheco

Uno de los principales factores para lograr la integración fueron las reuniones técnicas y de coordinación realizadas con el equipo de desarrollo de los tres módulos que intervienen en el sistema UPS-SCHEDULE. En estas reuniones se realizaron las siguientes actividades:

- Definición de estándares para la programación.
- Selección de herramientas de desarrollo (IDE).
- Selección del motor de la base de datos.
- Construcción del diagrama conceptual de la base de datos.
- Recolección de datos para cargarlos en la base.
- Capacitaciones sobre el uso y la programación sobre el framework X-PRIME.

CONCLUSIONES

1. Se ha obtenido un módulo funcional del sistema UPS-SCHEDULE para la gestión de usuarios, comunicación, seguridad y auditoría que cumple con los requerimientos planteados por la Universidad Politécnica Salesiana, mediante la utilización de software libre y tecnología de punta.
2. Con la implementación de la librería X-PRIME y la codificación estandarizada, se logró de manera ágil y sin complicación la integración de los tres módulos involucrados en el sistema UPS-SCHEDULE.
3. El trabajo realizado en Sourceforge.net es de gran ayuda para el desarrollo de sistemas colaborativos, permitiendo a los miembros de cada módulo interactuar entre sí para resolver dudas, publicar avances, tener control de versiones o realizar comentarios de una forma ordenada; teniendo un historial de las acciones que se han ido presentando durante el desarrollo del sistema.
4. Luego de haber realizado las pruebas de carga, se puede considerar al navegador Mozilla Firefox versión 29.0.1 como el más óptimo para trabajar con el sistema ya que tarda en promedio 133 milisegundos menos que Google Chrome versión 35.0.1916.114 m, 45 milisegundos menos que Safari versión 7.0 y 192 milisegundos menos que Internet Explorer versión 11 en realizar cada una de las acciones sobre las pantallas del módulo. Internet Explorer, por el contrario, no es recomendable ya que el FrontEnd del sistema no trabaja adecuadamente presentando el valor más alto en consumo de tiempo con un valor de 1238 milisegundos.
5. Las pruebas de usabilidad muestran que el módulo de gestión de usuarios, comunicación, seguridad y auditoría es intuitivo y fácil de utilizar en un 82%, tomando en cuenta que las pantallas son similares a hojas de cálculo con funcionalidades mínimas como Insertar, Eliminar, Guardar.

6. En un trabajo colaborativo, se debe hacer mucho énfasis en el uso de herramientas que permitan ahorrar tiempo en la resolución de problemas. Algunas de ellas son la creación de wikis, blogs, trabajo en línea con el equipo de desarrollo.
7. La interacción que tiene el framework X-PRIME con el servidor de aplicaciones Glassfish es muy versátil en el momento que fuere necesario cambiar el motor de la base de datos, ya que usa un pool de conexiones haciendo que este proceso se reduzca a una simple parametrización en el lado del servidor.
8. Primefaces 3.2 es una librería de componentes que brinda la posibilidad de modificar su código y adaptarlo a las necesidades de quién la use. X-PRIME es una implementación de esta librería que ayudó a los desarrolladores a optimizar su tiempo y agilizar así su trabajo. Esta característica es una de las principales ventajas que tiene el uso de software libre además de reducir costos.
9. Mantener reuniones cada cierto tiempo con el equipo de trabajo durante la elaboración de un sistema, junto con la correcta selección de las herramientas a utilizar permiten llevar un desarrollo ordenado y sin complicaciones. Estas reuniones deben ser estrictamente necesarias y para tratar temas que no se puedan resolver utilizando los medios o herramientas de soporte como Sourceforge ya que por el contrario no sería de ayuda sino una pérdida de tiempo.

RECOMENDACIONES

1. Para lograr un mejor rendimiento con el uso del sistema es posible considerar a JBOSS 7.0 como servidor de aplicaciones ya que su rendimiento es mucho mejor que Glassfish 3.1.2. En caso de seguir utilizando Glassfish 3.1.2 es recomendable hacer un “Tuning” del servidor. Principalmente aumentar de la memoria de la JVM (máquina virtual de java), implementar balanceo de carga, liberar memoria “basura”, etc. Esto permite que en el futuro el servidor no se sature y deje de funcionar.
2. Para futuros desarrollos se podrían agregar herramientas que faciliten el uso del sistema, como por ejemplo: un buscador en cada pantalla, filtros para las tablas, una opción para exportar a Excel la tabla que se está mostrando, ordenamiento de las columnas, etc. Adicionalmente, el tema de visualización y diseño se lo puede mejorar incluyendo elementos visuales para aumentar el tamaño de la letra, la fuente, etc.
3. Revisar la página oficial de Primefaces www.primefaces.org y trabajar con la última versión disponible para tener más opciones en los componentes utilizados para el desarrollo.
4. La seguridad y auditoría se la puede complementar llevando un registro de cambios en contraseñas de usuarios incluyendo una opción de generación automática de contraseñas y obligando a éste a cambiarla en su primer ingreso al sistema. La implementación de un EJB que controle la seguridad y registre cada acción que se haga en el sistema, sería lo recomendable para futuros desarrollos.
5. Para mejorar un tema fundamental dentro de un sistema que es la comunicación, se puede implementar un chat en el que los usuarios conectados puedan interactuar entre sí. A su vez, una pantalla en la que el usuario reciba notificaciones o alertas también sería una solución para este tema.

LISTA DE REFERENCIAS

- Borja, C., & Cuji, V. (agosto de 2013). *Metodología para la especificación de requerimientos de software basados en el estándar IEEE 830-1998*. Recuperado el 15 de enero de 2014, de <http://dspace.ups.edu.ec/bitstream/123456789/5264/1/UPS-CT002757.pdf>
- Burns, E., & Schalk, C. (2010). *Java Server Faces 2.0 La referencia completa*. Editorial McGraw - Hill.
- Características de PrimeFaces. (2014). *Primefaces*. Recuperado el 8 de abril de 2014, de <http://es.scribd.com/doc/182828665/Caracteristicas-de-PrimeFacfes-docx>
- Fuentes, J. M. (2009). *Manual de Ajax, las entrañas de Ajax*. Recuperado el 10 de marzo de 2013, de Manual de Ajax: <http://www.elrincondeajax.com/wp-content/uploads/Manual.pdf>
- Glassfish. (2012). *Glassfish*. Recuperado el 15 de mayo de 2014, de <http://es.scribd.com/doc/207074993/Glassfish-1-docx>
- Guamantica, L., & Tupiza, S. (noviembre de 2013). *Análisis, diseño y construcción del módulo de seguridad y edición del mapa para el geoportal de la comunidad salesiana*. Recuperado el 8 de abril de 2014, de <http://dspace.ups.edu.ec/bitstream/123456789/5342/1/UPS-ST001025.pdf>
- InterGraphic Design Agency. (2014). *InterGraphic Design*. Recuperado el 5 de abril de 2014, de <http://www.intergraphicdesigns.com/tools/test-usabilidad-web/ExpertReviewCheckpoints-Spanish.xls>
- Loor, J. M. (2013). *Líder en desarrollo de aplicaciones java*. Recuperado el 6 de abril de 2014, de http://www.ndeveloper.com/ndeveloperDocuments/documents/nDeveloper_JavaServerFaces.pdf
- NetBeans. (2013). *NetBeans*. Recuperado el 15 de mayo de 2014, de https://netbeans.org/index_es.html
- PostgreSQL. (2012). *PostgreSQL*. Recuperado el 20 de mayo de 2014, de PostgreSQL: <http://microbuffer.wordpress.com/2011/05/04/que-es-postgresql/>
- Presman, R. (2002). *Ingeniería de Software*. MC. Graw-Hill.

- Primefaces. (s.f.). *Documentation*. Recuperado el 5 de febrero de 2013, de User Guide: <http://www.primefaces.org/documentation.html>
- Primefaces. (s.f.). *Primefaces Showcase*. Recuperado el 5 de febrero de 2013, de Primefaces Showcase: <http://www.primefaces.org/showcase/>
- Proyecto GlassFish. (2012). *Proyecto GlassFish*. Recuperado el 15 de mayo de 2014, de Proyecto GlassFish: <https://glassfish.java.net/es/downloads/3.0.1-final.html>
- Roldon Martínez, D. (s.f.). *Aplicaciones web un enfoque práctico*.
- Sánchez, J. M. (2010). *Introducción a Primefaces*. Autentia.
- Smart, J. F. (2007). *JSF Jumpstart*. Editorial Wakaleo Consulting Ltd.
- SQL Power Architect. (2014). *Modelado de datos y herramientas de generación de perfiles*. Recuperado el 15 de mayo de 2014, de <http://www.sqlpower.ca/page/architect>
- Stinson, B. (2009). *PostgreSQL essential reference*. New Riders.
- Tufiño, R. E. (2014). *Sistema UPS Schedule para la Universidad Politécnica Salesiana Sede Quito*. Quito, Pichincha, Ecuador.
- Universidad Pplitécnica Salesiana. (2009). *Sistema Nacional Académico*. Quito, Pichicha, Ecuador.

ANEXO

Anexo 1: Documento excel con formato para la evaluación de pruebas de usabilidad

Instrucciones de utilización

[Última actualización: 6 de Julio del 2009. Revisar la última versión](#)

Paso 1: Haga clic en la Hoja "Página de Inicio"

Paso 2: Para cada item del checklist, digite un -1 (no satisface la directriz), +1 (sí satisface la directriz) ó 0 (no aplica). Si una directriz no es relevante, puede dejar el espacio en blanco. Puede agregar algún comentario a la derecha si lo desea

on users' key tasks ("testutils" has been avoided)	1
visible on the homepage	-1

Paso 3: Haga clic en las hojas de cálculo restantes para calificar las otras características del sitio.

Figura 1. Pestaña de instrucciones de utilización del archivo

Elaborado por: Marco Montaguano y Eduardo Pacheco

Revisión de Experto

Resumen de resultados

	Calificación	# Preguntas	# Respuesta	Calificación
Página de Inicio	0	20	0	
Orientación a Tareas y Funcionalidad	0	44	0	
Navegabilidad y Arq. De la Informac	0	29	0	
Formularios y entrada de datos	0	23	0	
Confianza y Credibilidad	0	13	0	
Calidad del Contenido y Escritura	0	23	0	
Diagramación y Diseño Gráfico	0	38	0	
Búsquedas	0	20	0	
Ayuda, Retroalimentación & Recuper	0	37	0	
Calificación Final		247	0	

Figura 2. Pestaña con los resultados obtenidos

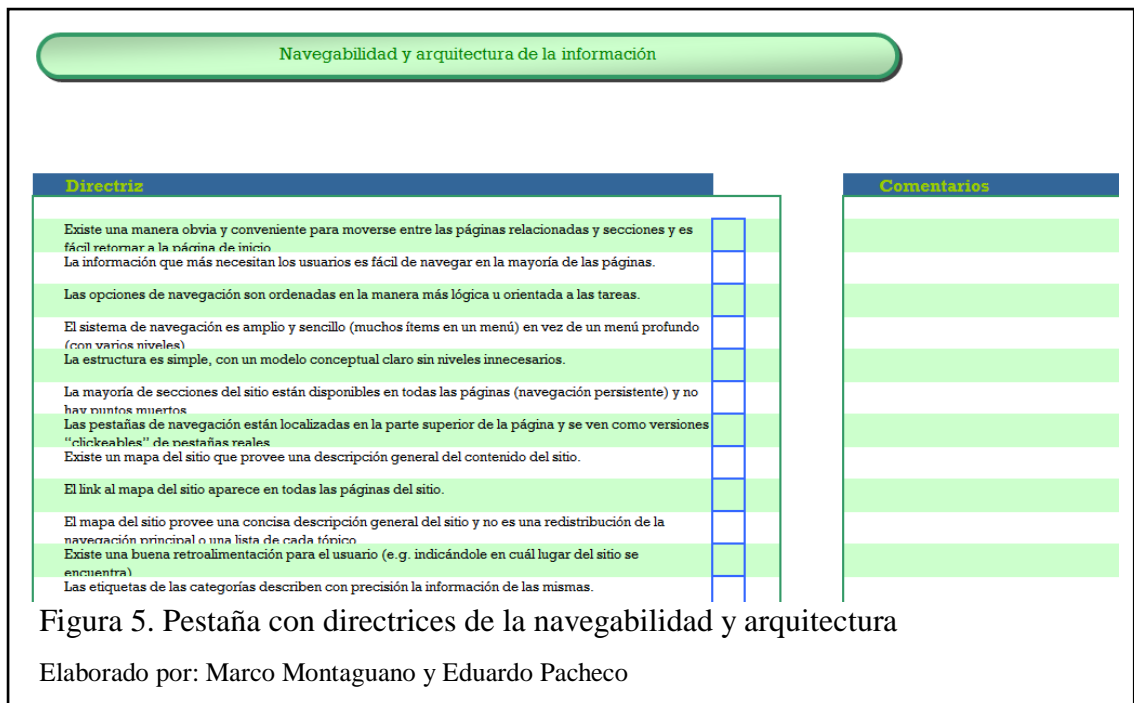
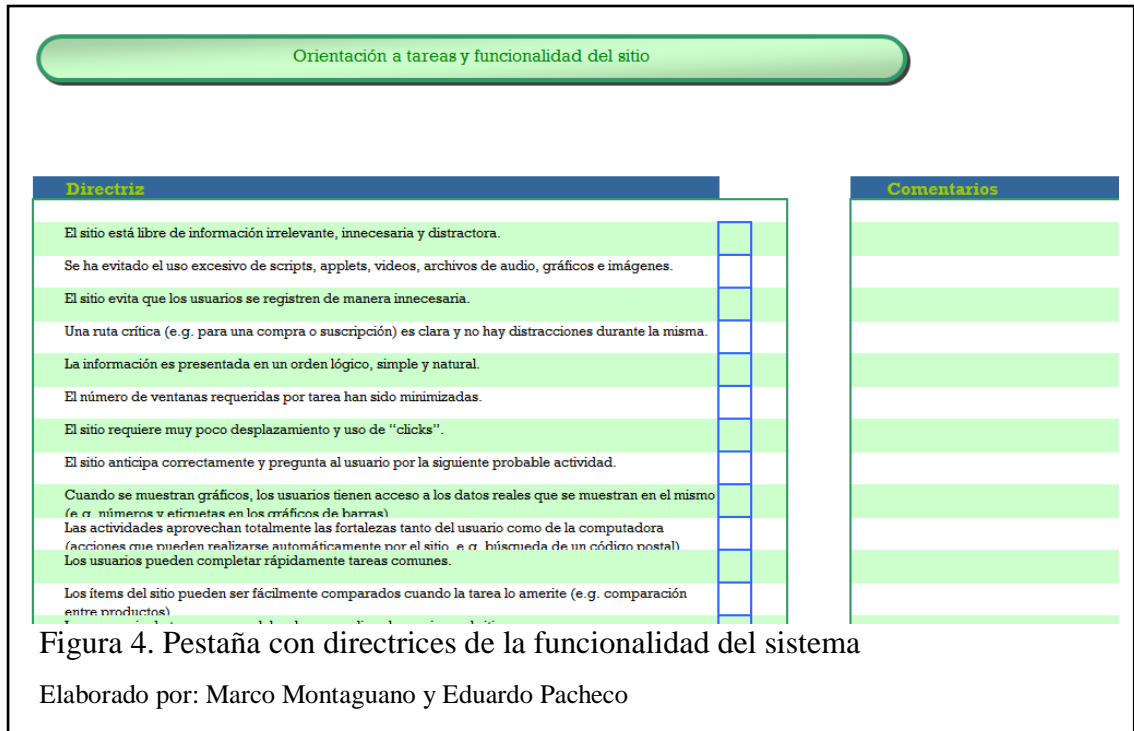
Elaborado por: Marco Montaguano y Eduardo Pacheco

Página de Inicio

Directriz	Comentarios
Todos los elementos de la página de inicio están claramente enfocados en las tareas claves de los usuarios (La "featuritis" - proliferación de características - ha sido evitada). La página de inicio contiene un campo de texto para búsquedas.	
Se presentan las categorías de productos y están claramente visibles en la página de inicio.	
Existe contenido de utilidad en la página de inicio o a un clic de distancia de la página inicial.	
La página de inicio contiene buenos ejemplos sobre el contenido en el resto del sitio web.	
Los links de la página de inicio empiezan con la palabra clave más importante (ej. en inglés "Sun holidays" en vez de "Holidays in the sun"). Existe en la página de inicio una lista breve de elementos recientes, con un link hacia el contenido completo.	
Las áreas de navegación en la página de inicio no sufren de un abuso de formato/diseño y los usuarios no las podrían confundir con publicidad.	
La proposición de valor está claramente establecida en la página de inicio (e.g. con un lema o fraseo de bienvenida).	
La página de inicio contiene información gráfica con sentido, sin clip art por ejemplo.	
Las opciones en la navegación están ordenadas en el orden más lógico o de alguna forma orientada a las tareas más importantes (con la información corporativa menos importante al final).	
El título de la página de inicio va a proveer buena visibilidad a máquinas de búsqueda como Google.	
Toda la información corporativa está bien agrupada en una sola área por aparte (ej. "Acerca de").	
Los usuarios van a entender la proposición de valor.	
Con solo un vistazo a la página de inicio el usuario que ingresa por primera vez puede entender por dónde comenzar.	

Figura 3. Pestaña con directrices de la pantalla de inicio

Elaborado por: Marco Montaguano y Eduardo Pacheco



Formularios y entrada de datos	
Directriz	Comentarios
Los campos de entrada contienen valores predeterminados cuando así se requiera y muestran la estructura de los datos y la longitud del campo.	
Cuando una tarea involucra documentos fuente (como un formulario de papel), la interfaz es compatible con las características del documento fuente.	
El sitio automáticamente introduce el formateo de los datos (e.g. símbolos de moneda, comas para los miles, espacios al principio o al final). Los usuarios no necesitan introducir caracteres como f. or %.	
Las etiquetas para los campos explican claramente cuáles campos son requeridos.	
Las cajas de texto en los formularios tienen el tamaño adecuado para el dato que se debe introducir.	
Existe una clara distinción entre campos "requeridos" y "opcionales" en los formularios.	
El mismo formulario es utilizado tanto para conectarse o registrarse (e.g. como Amazon).	
Los formularios deben advertir al usuario si información externa es necesitada para su completitud (e.g. número de pasaporte).	
Las preguntas en los formularios están agrupadas de manera lógica y cada grupo tiene un título descriptivo.	
Los campos en los formularios contienen ayudas, ejemplos o modelos de respuestas para demostrar el dato que se debe introducir.	
Cuando hay campos en el formulario que toman la forma de preguntas, dichas preguntas deben ser claras y en lenguaje simple.	
Las listas de opciones, botones de radio y casillas son preferibles a las cajas de texto en los formularios (e.g. no se debe abusar del uso de las cajas de texto).	

Figura 6. Pestaña con directrices de formularios y entrada de datos

Elaborado por: Marco Montaguano y Eduardo Pacheco

Confianza y Credibilidad	
Directriz	Comentarios
El contenido está actualizado, autorizado y fidedigno.	
El sitio contiene soporte de terceros (e.g. citas, testimonios) para verificar la precisión de la información.	
Es claro que hay una organización real detrás del sitio (e.g. existe una dirección física o una foto de la oficina).	
La compañía está compuesta por expertos reconocidos (buscar credenciales).	
El sitio evita los anuncios, especialmente los de tipo "pop-up".	
Los costos de entrega están destacados al inicio del "checkout".	
El sitio evita la palabrería de mercadeo.	
Cada página debe contener el logo de la marca de la compañía para que el usuario sepa que se mantiene en el mismo sitio.	
Es fácil contactar a alguien para asistencia y una respuesta es recibida rápidamente.	
El contenido es fresco: es actualizado con frecuencia y el sitio tiene contenido reciente.	
El sitio está libre de errores tipográficos y de errores ortográficos.	
El diseño visual complementa a la marca de la compañía y a los mensajes de mercadeo.	

Figura 7. Pestaña con directrices de confianza y credibilidad

Elaborado por: Marco Montaguano y Eduardo Pacheco

Calidad del Contenido y Escritura		
Directriz		Comentarios
El sitio tiene contenido único y atractivo.	<input type="checkbox"/>	
El texto es conciso, sin instrucciones innecesarias ni mensajes de bienvenida.	<input type="checkbox"/>	
El contenido de cada página comienza con conclusiones o implicaciones y el texto es escrito en estilo de pirámide invertida	<input type="checkbox"/>	
Las páginas usan viñetas y listas numeradas en vez del texto narrativo.	<input type="checkbox"/>	
Las listas son precedidas de una introducción concisa (e.g. una palabra o frase) ayudando a los usuarios a apreciar cómo los ítems se relacionan unos con otros	<input type="checkbox"/>	
Los ítems más importantes de una lista están situados al inicio de la misma.	<input type="checkbox"/>	
La información está organizada jerárquicamente, de lo general a lo específico, y la organización es clara y lógica	<input type="checkbox"/>	
El contenido ha sido específicamente creado para la web (las páginas web no contienen material de brochuras por ejemplo)	<input type="checkbox"/>	
Las páginas de productos contienen el detalle necesario para hacer una compra y los usuarios pueden ver las imágenes del producto con acercamiento	<input type="checkbox"/>	
El hipertexto ha sido usado apropiadamente en la estructura del contenido.	<input type="checkbox"/>	
Las oraciones están escritas en voz activa.	<input type="checkbox"/>	
Las páginas son rápidas de examinar, con títulos grandes, subtítulos y párrafos cortos.	<input type="checkbox"/>	

Figura 8. Pestaña con directrices de calidad del contenido y escritura

Elaborado por: Marco Montaguano y Eduardo Pacheco

Diagramación y diseño gráfico		
Directriz		Comentarios
La densidad de la pantalla es apropiada para los usuarios finales y sus tareas.	<input type="checkbox"/>	
El diseño ayuda a poner atención en cuál paso es el siguiente.	<input type="checkbox"/>	
En todas las páginas, la información más importante (como los tópicos, características y funciones más frecuentemente usados) es presentada en la primera pantalla de información ("por encima de la línea")	<input type="checkbox"/>	
El sitio puede ser usado sin desplazamiento horizontal.	<input type="checkbox"/>	
Todo lo que es "clickeable" (como los botones) son efectivamente "presionables".	<input type="checkbox"/>	
Los ítems que no son "clickeables" tienen características que sugieren que efectivamente no son "clickeables"	<input type="checkbox"/>	
La funcionalidad de los botones y controles es obvia a partir de sus etiquetas o de su diseño.	<input type="checkbox"/>	
Las imágenes "clickeables" incluyen textos redundantes (no existe una navegación "misteriosa").	<input type="checkbox"/>	
Los links son fáciles de identificar sin necesidad de ir pasando el mouse para verificar si son links (e.g. están subrayados)	<input type="checkbox"/>	
Las fuentes son usadas consistentemente.	<input type="checkbox"/>	
La relación entre los controles y sus acciones es obvia.	<input type="checkbox"/>	
Los iconos y gráficos son estándar y/o intuitivos (concretos y familiares).	<input type="checkbox"/>	

Figura 9. Pestaña con directrices de diagramación y diseño gráfico

Elaborado por: Marco Montaguano y Eduardo Pacheco

Búsquedas	
Directriz	Comentarios
La búsqueda predeterminada es intuitiva (no hay operadores booleanos).	
La página de resultados de una búsqueda le muestra al usuario lo que se buscó y es fácil editar los criterios de búsqueda y reenviarlos para una nueva búsqueda.	
Los resultados de una búsqueda son claros, útiles y clasificados por relevancia.	
La página de resultados de una búsqueda indica claramente cuántos resultados tuvo la búsqueda y el número de resultados que se muestran por página es configurable por el usuario.	
Si no hubo resultados después de una búsqueda, el sistema ofrece ideas u opciones para mejorar la búsqueda basadas en problemas identificables con la entrada del usuario.	
El motor de búsqueda maneja correctamente las búsquedas vacías (el usuario no introduce nada).	
Las consultas más comunes (reflejado en los registros del sitio) producen resultados útiles.	
El motor de búsqueda incluye plantillas, ejemplos o consejos para usarlo eficazmente.	
El sitio incluye una interfaz más poderosa de búsqueda para ayudar a los usuarios a redefinir sus búsquedas (preferiblemente llamada "redefine su búsqueda" pero no "búsqueda avanzada").	
La página de resultados de una búsqueda no muestra resultados duplicados (ni duplicados reales ni duplicados muy parecidos).	
La caja de búsqueda es suficientemente grande para manejar la longitud de las consultas más comunes.	
Las búsquedas cubren todo el sitio, no una porción de él.	

Figura 10. Pestaña con directrices de búsqueda

Elaborado por: Marco Montaguano y Eduardo Pacheco

Ayuda, retroalimentación y tolerancia a errores	
Directriz	Comentarios
Las preguntas frecuentes (FAQ) o la ayuda en línea provee instrucciones paso a paso que ayudan a los usuarios a llevar a cabo las tareas más importantes.	
Es fácil obtener ayuda en la forma y momento oportuno.	
Los avisos al usuario son breves y no ambiguos.	
El usuario no necesita consultar manuales de usuario u otra información externa para usar el sitio.	
El sitio usa una página 404 personalizada que incluye consejos en cómo encontrar una página perdida y links a la página de inicio y a la búsqueda.	
El sitio provee retroalimentación (e.g. indicador de progreso o mensajes) cuando se necesitan (e.g. durante un "checkout").	
Se le brinda ayuda a los usuarios cuando escogen productos.	
La confirmación del usuario es requerida antes de llevar a cabo acciones potencialmente peligrosas (e.g. al borrar algo).	
Las páginas de confirmación son claras.	
Los mensajes de error contienen instrucciones claras en cuál es el paso siguiente.	
Inmediatamente antes de completar una compra, el sitio muestra al usuario un claro resumen y no debe ser confundido con la página de confirmación de compra.	
Cuando el usuario necesite escoger entre diferentes opciones (como en una ventana de diálogo) las opciones deben ser obvias.	

Figura 11. Pestaña con directrices de retroalimentación y tolerancia a errores

Elaborado por: Marco Montaguano y Eduardo Pacheco