

Optinet

Software de Gestión Web para centros Ópticos

José Ángel Parada Jiménez

Ingeniería Técnica en Informática de Gestión
Universidad de Cádiz

28 de junio de 2013



Índice

- 1 Introducción
- 2 Desarrollo del proyecto
- 3 Conclusiones
- 4 Posibles mejoras

1 Introducción

2 Desarrollo del proyecto

3 Conclusiones

4 Posibles mejoras

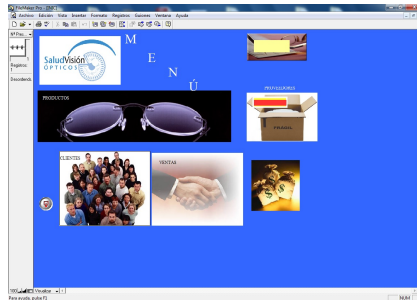
Salud visión

El proyecto surge por:

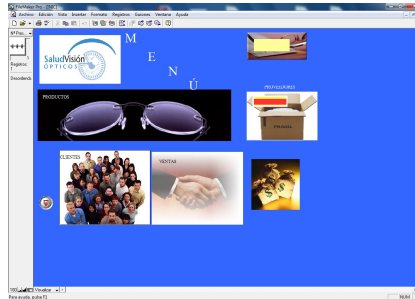
- Petición del gerente de Salud Visión construir un sistema de gestión.
- Interés personal por el desarrollo de un software de gestión.
- Interés personal por programación web.



Software Actual



Software Actual



Problemas

- Aspecto visual.
- Problema de usabilidad.
- Carencias de funcionalidades.

Objetivos del proyecto



Objetivos principales

- Construir aplicación web para gestionar un centro óptico.

Objetivos del proyecto



Objetivos principales

- Construir aplicación web para gestionar un centro óptico.
- Gestión de productos, proveedores, pedidos, ventas...

Objetivos del proyecto



Objetivos principales

- Construir aplicación web para gestionar un centro óptico.
- Gestión de productos, proveedores, pedidos, ventas...
- Generar informes.

Objetivos del proyecto



Objetivos principales

- Construir aplicación web para gestionar un centro óptico.
- Gestión de productos, proveedores, pedidos, ventas...
- Generar informes.
- Control de las acciones que realizan los usuarios.

Objetivos del proyecto



Objetivos principales

- Construir aplicación web para gestionar un centro óptico.
- Gestión de productos, proveedores, pedidos, ventas...
- Generar informes.
- Control de las acciones que realizan los usuarios.
- Multi-idioma y multi-plataforma.

Objetivos del proyecto



Objetivos principales

- Construir aplicación web para gestionar un centro óptico.
- Gestión de productos, proveedores, pedidos, ventas...
- Generar informes.
- Control de las acciones que realizan los usuarios.
- Multi-idioma y multi-plataforma.
- Segura, fiable y tener un rendimiento adecuado.

Desarrollo del proyecto

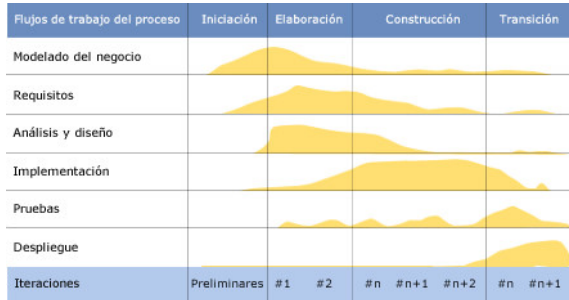
1 Introducción

2 Desarrollo del proyecto

3 Conclusiones

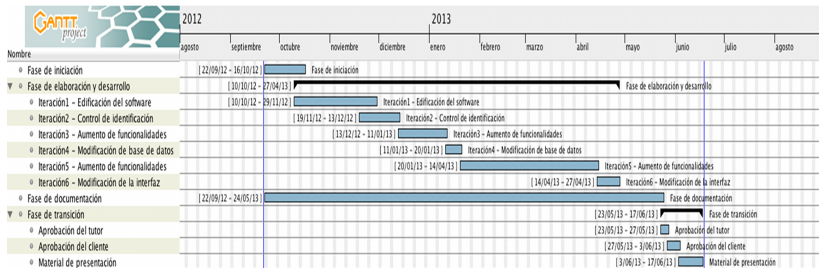
4 Posibles mejoras

Metodología



- Se ha utilizado la metodología RUP al ser la más utilizada para la construcción de sistemas orientados a objetos.
- En cada fase participan todas las disciplinas, pero dependiendo de la fase el esfuerzo dedicado a una disciplina varía.

Calendario



Estimación de tiempos

Fase	Tiempo estimado	Tiempo real
Fase de iniciación	60 horas	75 horas
Fase de elaboración y construcción	400 horas	550 horas
Fase de documentación	30 horas	50 horas
Fase de transición	50 horas	70 horas
Total	540 horas	745 horas

Se realizó la planificación de los tiempos de las tareas pero no se cumplieron debido a problemas no esperados o dificultad añadida no prevista.

Requisitos Software

- Control de acceso y roles de los usuarios.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.
- Control de los proveedores del sistema.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.
- Control de los proveedores del sistema.
- Control de las citas que se generen.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.
- Control de los proveedores del sistema.
- Control de las citas que se generen.
- Control de los pedidos, ventas, devoluciones, reservas y apartados que generen los usuarios.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.
- Control de los proveedores del sistema.
- Control de las citas que se generen.
- Control de los pedidos, ventas, devoluciones, reservas y apartados que generen los usuarios.
- Control de los cambios realizados en el sistema.

Requisitos Software

- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.
- Control de los proveedores del sistema.
- Control de las citas que se generen.
- Control de los pedidos, ventas, devoluciones, reservas y apartados que generen los usuarios.
- Control de los cambios realizados en el sistema.
- Control de los arqueos realizados en el sistema.

Requisitos Software

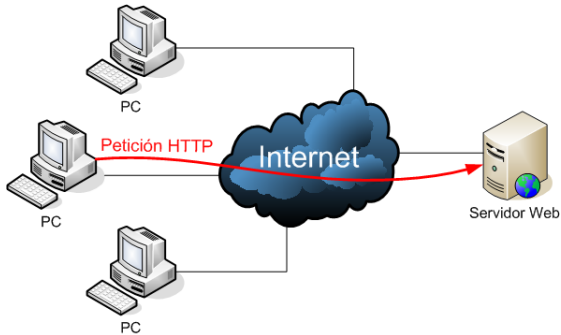
- Control de acceso y roles de los usuarios.
- Organización de los productos agrupados por familias.
- Creación de documentos e informes con el logotipo.
- Control de los clientes.
- Control de los proveedores del sistema.
- Control de las citas que se generen.
- Control de los pedidos, ventas, devoluciones, reservas y apartados que generen los usuarios.
- Control de los cambios realizados en el sistema.
- Control de los arqueos realizados en el sistema.
- Control de los permisos disfrutados por los usuarios.

Actores del sistema

Se definieron las distintas responsabilidades de los usuarios del sistema quedando de la siguiente manera:

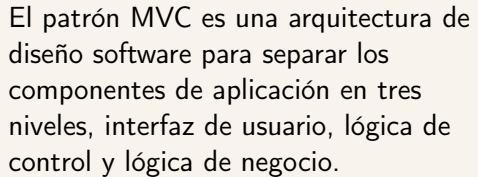
	Administrador	Empleado	Médico
Gestión operaciones	X	X	X
Ver informes	X	X	X
Crear informes	-	-	X
Administración	X	-	-

Arquitectura física



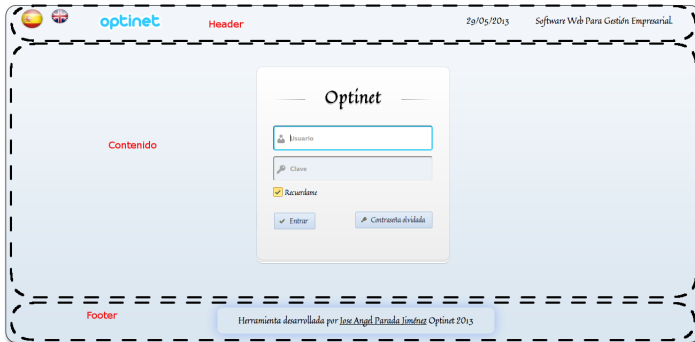
Varios usuarios trabajando sobre la aplicación simultáneamente.

Arquitectura lógica



En el modelo que se tratará en este libro, la lógica de la vista recibe la información provista por el controlador. Trae el layout (diseño) y lo procesa con la información recibida, antes de entregarlo al usuario. Otros modelos, pueden realizar esto, íntegramente a través del controlador, pero aquí no lo trataremos por considerar que resta mantenibilidad al sistema.

Diseño



Antes de llegar al diseño definitivo de la imagen se crearon bocetos con la herramienta Pencil App.

Implementación

- Se realizó un estudio del entorno de trabajo para la construcción de la aplicación.
- Se barajaron diferentes lenguajes de programación PHP, JSP, ASP
- Se eligió PHP (rapidez, documentación, variedad de módulos, similar a C, especializado para web...)
- Una vez elegido el lenguaje, se estudiaron los diferentes frameworks de PHP: Codeigniter, Symfony, Yii, CakePHP.

Lenguajes utilizados



Framework Symfony2



Symfony2 es un framework de PHP rápido, flexible y fácil de aprender.

Características Symfony2

- Versátil.
- Seguridad.
- Flexible.
- Rendimiento.
- Soporte.
- Documentación.
- Comunidad.
- Popular.

Framework jQuery

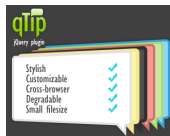
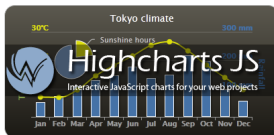


Es un framework de JavaScript que proporciona un conjunto de funciones heredadas de JavaScript listas para utilizarlas de una manera muy simplificada.

Características jQuery

- Eventos.
- Manipulación CSS.
- Efectos y animaciones.
- Selección y modificaciones del DOM.
- AJAX.
- Soporta extensiones.
- Gran compatibilidad con navegadores.

Bibliotecas utilizadas



Entorno de trabajo



Modelo - Vista - Controlador

Vistas: Son las encargadas de mostrar información del usuario.

- Se hace uso de las plantillas twig para poder escribir plantillas concisas y fáciles de leer.
- Permite el uso de herencia de plantillas: una plantilla base de la que heredan todas las demás.

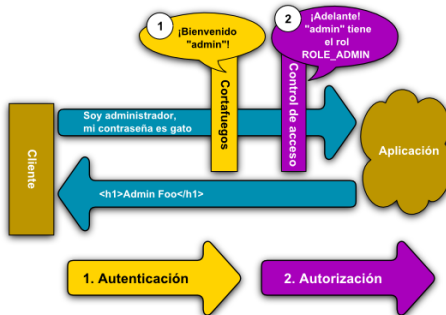


Modelo - Vista - Controlador

Controlador: Es el que se encarga de manejar las peticiones.

- Capa petición es manejada por un controlador frontal (app.php , app_dev.php).
- El enrutador lee la información de la petición y se llama al controlador asociado.
- El controlador es ejecutado creando y devolviendo un objeto respuesta.

Acceso a la aplicación



Se hace uso del algoritmo de encriptación md5 + salt.

Tipos de Pruebas

Pruebas en el desarrollo

- Pruebas unitarias.
- Pruebas de integración.

Tipos de Pruebas

Pruebas en el desarrollo

- Pruebas unitarias.
- Pruebas de integración.

Pruebas finalizado el desarrollo

- Pruebas de sistema.
 - Pruebas funcionales.
 - Pruebas no funcionales.
- Pruebas de aceptación.

Pruebas de sistema

Pruebas funcionales.

- Se probaron todos los escenarios principales y alternativos de los casos de uso.

Pruebas de sistema

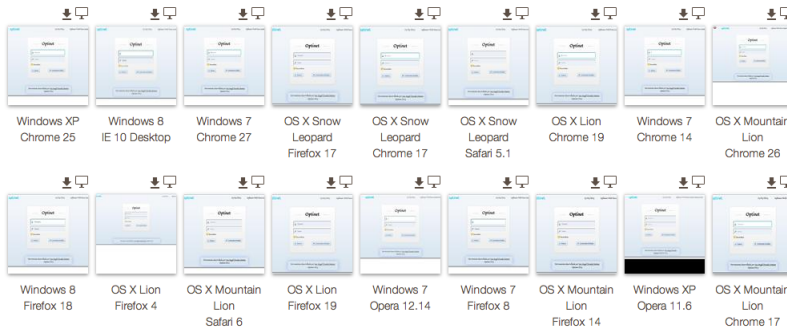
Pruebas funcionales.

- Se probaron todos los escenarios principales y alternativos de los casos de uso.

Pruebas no funcionales.

- Portabilidad - Herramienta online BrowseStack.
- Mantenibilidad - Estructura de directorios Symfony2.
- Seguridad - Fundación OWASP.
- Fiabilidad - Plan de pruebas exhaustivo.
- Rendimiento - Herramienta online GTmetrix y PigDom(Usa de YuiCompressor, CDN, gzip).

Pruebas de sistema - Portabilidad



Pruebas de sistema - Rendimiento

- GTmetrix:

Latest Performance Report for: <http://parada85.sytes.net/optinet/web/login>

 [Download PDF](#)

Report generated: Fri, May 17, 2013, 2:09 AM -0700

Test Server Region: Vancouver, Canada

Using: Firefox (Desktop) 14.0.1, Page Speed 1.12.16, YSlow 3.1.6

Summary

Page Speed Grade:

(94%)_↑

A

YSlow Grade:

(94%)_↑

A

Page load time: 2.70s

Total page size: 314KB

Total number of requests: 22

- PigDom



parada85.sytes.net/optinet/web/...

Tested from New York City, New York, USA on May 17 at 12:48:45

Perf. grade	Requests	Load time	Page size
96 ₁₁₀₀	21	2.15 _s	265.9 _{kB}

Your website is faster than 69% of all tested websites

[DOWNLOAD HAR](#) [Tweet](#) [Email](#)

Pruebas de aceptación

Se realizaron diferenciando dos tipos de personas:

- Personas con nivel alto de conocimientos informáticos.
Compañeros de la universidad con acceso al código de la aplicación.
 - Fallos: Detectaron algunos fallos de seguridad.



Conclusiones

- 1 Introducción
- 2 Desarrollo del proyecto
- 3 Conclusiones**
- 4 Posibles mejoras

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓
- La aplicación gestiona correctamente los productos, proveedores, ventas...✓

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓
- La aplicación gestiona correctamente los productos, proveedores, ventas...✓
- La aplicación genera informes.✓

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓
- La aplicación gestiona correctamente los productos, proveedores, ventas...✓
- La aplicación genera informes.✓
- La aplicación tiene control de las acciones que hacen los usuarios.✓

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓
- La aplicación gestiona correctamente los productos, proveedores, ventas...✓
- La aplicación genera informes.✓
- La aplicación tiene control de las acciones que hacen los usuarios.✓
- La aplicación es multi-idioma y multi-plataforma.✓

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓
- La aplicación gestiona correctamente los productos, proveedores, ventas...✓
- La aplicación genera informes.✓
- La aplicación tiene control de las acciones que hacen los usuarios.✓
- La aplicación es multi-idioma y multi-plataforma.✓
- La aplicación es segura, fiable y tiene un rendimiento adecuado.✓

Objetivos

Objetivos cumplidos

- Se ha construido una aplicación web para gestionar un centro óptico.✓
- La aplicación gestiona correctamente los productos, proveedores, ventas...✓
- La aplicación genera informes.✓
- La aplicación tiene control de las acciones que hacen los usuarios.✓
- La aplicación es multi-idioma y multi-plataforma.✓
- La aplicación es segura, fiable y tiene un rendimiento adecuado.✓
- Plena satisfacción del cliente.✓

Lecciones aprendidas

Lecciones aprendidas

- Lenguajes de programación: HTML, CSS, JavaScript, PHP.
- Utilización de un framework PHP: Symfony2.
- Utilización de un framework Javascript: JQuery.
- Repaso y ampliación de conocimientos de ingeniería del software

Posibles mejoras

Posibles mejoras

- Registro de citas por internet.
- Realización de una aplicación móvil.

Gracias por su atención