

Tienda Virtual Andaser

DAW

MEMORIA FINAL DE PROYECTO

Tienda online Andaser
Comercial Andaser

COMERCIAL
ANDASER
ALMAGRO

CICLO FORMATIVO DE GRADO SUPERIOR

Desarrollo de Aplicaciones Web

AUTORES

Jose Luis Gómez Álvarez

TUTOR

Jose Luis Gallego García

COORDINADOR

Jose Luis Gallego García

INDICE

1 INTRODUCCIÓN

2 ALCANCE DEL PROYECTO

3 ESTUDIO DE VIABILIDAD

- Estado actual del sistema

- Requisitos del cliente

- Posibles soluciones

- Solución elegida

- Planificación de los recursos a utilizar

4 ANÁLISIS

- Modelo de datos

- Requisitos funcionales

- Requisitos no funcionales

5 DISEÑO

- Estructura de la aplicación

- Componentes del sistema

- Arquitectura de la red

- Herramientas

6 IMPLEMENTACIÓN

- Tablas creadas

- Carga de datos

7 PRUEBAS

- Casos de pruebas

8 EXPLOTACIÓN

- Planificación

- Implantación

9 DEFINICIÓN DE PROCEDIMIENTOS DE CONTROL Y EVALUACIÓN

10 CONCLUSIONES

11 BIBLIOGRAFÍA

12 ANEXOS

1. INTRODUCCIÓN

Este documento recoge el trabajo realizado para el **módulo de Proyecto** del CFGS en **Desarrollo de Aplicaciones Web**.

Este módulo profesional complementa la formación establecida para el resto de los módulos profesionales que integran el título en las funciones de **análisis** del contexto, **diseño** o **desarrollo** del proyecto y organización de la **ejecución**.

2. ALCANCE DEL PROYECTO Y ANÁLISIS PREVIO

El propósito de este proyecto consiste en la realización de una tienda online que mi cliente, la empresa Comercial Andaser me ha encargado realizar, en la cual se podrán visualizar los productos y los usuarios registrados podrán realizar pedidos.

El desarrollo de éste proyecto se ha llevado a cabo en varias fases: análisis, diseño, implementación y pruebas, y ejecución. A continuación se detallan los procedimientos de cada una de estas fases.

3. ESTUDIO DE VIABILIDAD

En esta fase se considera si el proyecto se puede realizar teniendo en cuenta las circunstancias internas y externas de la empresa, las diferentes soluciones posibles y los recursos de los cuales se dispone.

Para ello se hace una valoración del estado actual del sistema y de los requisitos del cliente, se presentará un estudio de soluciones alternativas y la solución elegida por el cliente.

Estado actual del sistema

El sistema administra la gestión de productos que se ofertan y la gestión de pedidos que los usuarios realizan.

Resumen de requisitos del cliente

Mi cliente desea una tienda online que sirva tanto de expositor como para realizar pedidos online, los usuarios registrados deben ser aceptados por el administrador del sistema.

Posibles soluciones

Realizar la aplicación con la herramienta PrestaShop.

Realizar la aplicación desde 0, utilizando algún lenguaje de programación, gestor de base de datos y maquetador web (HTML)

Solución elegida

He elegido realizar la aplicación desde 0, utilizando el lenguaje de programación JAVA(J2EE) apoyándome en el framework struts2 y la herramienta AngularJS para implementar vistas dinámicas dentro de la aplicación.

Planificación temporal de las tareas del proyecto Tienda Online Andaser

- Análisis y diseño de la base de datos; Tiempo estimado: 8 horas.
- Maquetación y diseño general de la aplicación web; Tiempo estimado: 4 horas.
- Implementación de frameworks y herramientas; Tiempo estimado: 2 horas.
- Sistema de registro de usuarios y login; Tiempo estimado 8 horas.
- Sistema de mantenimiento de categorías, subcategorías, productos y usuarios; Tiempo estimado: 20 horas.
- Visualización de productos; Tiempo estimado 4 horas.
- Realización de pedidos; Tiempo estimado 5 horas.
- Administración de pedidos; Tiempo estimado 5 horas.

Extras

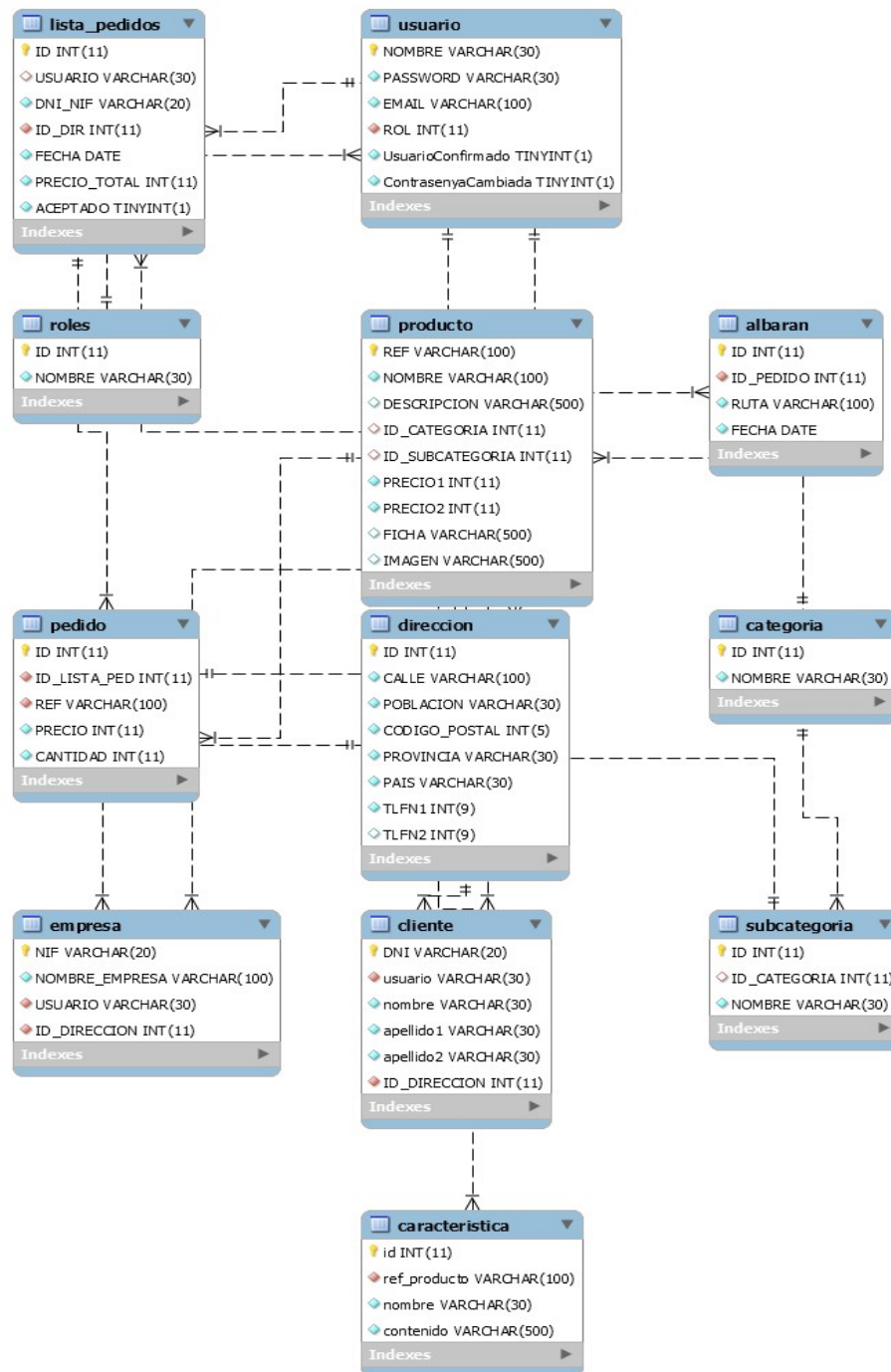
- Exportar pedidos a PDF
- Envío de email al registrarse y al aceptar pedido.
- Validación de formularios con AngularJS

Planificación de los recursos a utilizar

Para solventar los problemas que plantea el proyecto **Tienda Virtual Andaser** es necesario descargar las librerías de Struts2, JDBC, AngularJS... Disponer de un servidor web TomCat y un sistema de gestión de BBDD MySQL (XAMPP).

4. ANÁLISIS

Modelo de datos



Requisitos funcionales

- Visualización y administración de productos
- Gestión de usuarios
- Realización y gestión de pedidos

Requisitos no funcionales

- Diseño atractivo y dinamismo
- La aplicación se quedará en fase prototipo a espera de futuras implementaciones que el cliente decidirá cuando se realizarán. El plazo máximo es marzo de 2017.

5. DISEÑO

Estructura de la aplicación

Es una aplicación web con servicios de tienda online.

Componentes del sistema / arquitectura de red

La aplicación cuenta con una base de datos relacional y con un servidor de aplicaciones TomCat. No es necesaria seguridad a nivel de servidor ya que en el futuro se subirá la aplicación a un host ajeno.

Herramientas y tecnologías utilizadas.

En la aplicación se utiliza el lenguaje de programación JAVA(J2EE), herramientas javaScript como AngularJS, un servidor web Apache Tomcat, MySQL, utilizando el IDE NetBeans y para el diseño WorkBench.

6. Implementación del modelo de datos

Tablas:

- Albarán (fichero pdf de un pedido)
- Característica (característica adicional de un producto, futura implementación)
- Categoría (tipo de producto)
- Cliente (Tipo de usuario)
- Dirección (Dirección de un usuario)
- Empresa (Tipo de usuario)
- Lista_Pedidos (Tabla que recoge los productos de un pedido)
- Pedido (Tabla que contiene un producto de un pedido)
- Producto
- Roles (roles de usuario)
- Subcategoría (Subtipo de producto)
- Usuario

Carga de datos

En la carga de datos inicial se recogerán todas las categorías y subcategorías para mostrarse dichos datos en el menú de usuario.

7. PRUEBAS

Casos de pruebas

1.

- Fecha 11/12/2016
- Propósito: Realizar pedido
- Se procede a realizar un pedido el cual ya tiene elementos agregados.
- Descripción: Comprobar la correcta realización del pedido y que aparezca en la zona de administración para aceptarse.
- Condiciones de ejecución: que se inserte en la base de datos correctamente.
- Para que el pedido se inserte correctamente no debe estar ningún campo vacío y el usuario debe ser un particular o una empresa.
- Entrada: Lista_Pedido.
- Datos necesarios para poder ejecutar la prueba: producto, id_Dirección, usuario, fecha.
- Resultado esperado: Se muestra en la zona de pedidos pendientes.
- Resultado obtenido: Se inserta en la tabla de pedidos, pero no aparece en la zona de pedidos pendientes.

- Se comprueba en la base de datos que el valor que comprueba que esté confirmado el pedido es correcto
- Posible causa de error.

El valor introducido en la función de añadir pedido es erróneo.

- Posible corrección: corregir dicho valor.
- Áreas afectadas: campo de la tabla pedidos.

2.

- Fecha 26/12/2016
- Propósito: Borrar categoría
- Se procede a borrar una categoría la cual tiene dependencias de subcategorías y productos
- Descripción: Comprobar que se realice correctamente la acción de borrado.
- Condiciones de ejecución: que se inserte borre en la base de datos la categoría y no influya en las subcategorías y productos de la cual dependen.
- Para que se borre correctamente, las relaciones en las tablas de la base de datos tienen que estar correctas.
- Entrada: ID_Categoría.
- Datos necesarios para poder ejecutar la prueba: id_categoria.
- Resultado esperado: Se borra dicha categoría.
- Resultado obtenido: Se produce un error de sql.

- Se comprueban las relaciones de la base de datos
- Posible causa de error: La relación de borrado está en 'No Action'
- Posible corrección: Cambiar relación para que al borrar, en las tablas que dependen se quede dicha relación a null.
- Áreas afectadas: tabla categoría.

3.

- Fecha 29/12/2016
- Propósito: Solicitud de registro.
- Se realiza una solicitud de registro de un particular.
- Descripción: Comprobar que se realice correctamente dicha solicitud y realice correctamente las validaciones en los campos del formulario.
- Condiciones de ejecución: que se inserte en la base de datos correctamente dicha solicitud.
- Para que la solicitud se inserte correctamente no debe estar ningún campo vacío y los campos tienen que tener el formato correspondiente.
- Entrada: dni, cp, tfn1, tfn2, usuario, email, calle, población, país.
- Datos necesarios para poder ejecutar la prueba: cp, tfn1, tfn2, usuario, email, calle, población, país.
- Resultado esperado: Se inserta el usuario en la tabla de usuarios y el campo USUARIOCONFIRMADO de la tabla tiene el valor 0, que corresponde a pendiente de confirmación.
- Resultado obtenido: error sql al haber campos vacíos.

- Se comprueban las validaciones de AngularJS.

- Posible causa de error.

Struts2 y AngularJS aún no tienen compatibilidad y la validación no se realiza.

- Posible corrección: cambiar la forma de validar, para hacerlo desde el servidor.

- Áreas afectadas: formulario de registro.

8. EXPLOTACIÓN

La implantación es la fase más crítica del proyecto ya que el sistema entra en producción, es decir opera en un entorno real, con usuarios reales.

Planificación

Para la explotación de la aplicación se procederá a contratar un host para alojar la aplicación y aplicar la seguridad del servidor y la aplicación.

Dicha migración al host se realizará antes de la fecha límite, que es 6 de marzo de 2017.

Manual de usuario y administrador

Dichos están en los anexos anexo1 y anexo2.

9. DEFINICIÓN DE PROCEDIMIENTOS DE CONTROL Y EVALUACIÓN

Se ha utilizado el programa SourceTree para administrar un repositorio GitHub en el cual está alojada la aplicación y su información.

10. CONCLUSIONES

Está completada la funcionalidad de la aplicación gracias a la implementación de envío de correo.

En un futuro se realizarán cambios en los usuarios para que solo haya un tipo y ponerle un campo descuento, que será distinto para cada usuario, aún no está claro si se realizará, por lo que se hará más adelante.

La versión de Struts2 que compatibiliza con AngularJS aún está en fase beta, por lo que en cuanto esté estable dicha versión, se actualizará la aplicación y se implementará la validación con AngularJS.

11. Bibliografía

www.stackoverflow.com

<http://struts.apache.org/docs/>

<https://docs.angularjs.org/api>

Ubicación en GitHub: <https://github.com/joseluisGA/Andaser>

12. ANEXOS

Anexo1.manual de usuario

Anexo2.manual de administrador

Anteproyecto