29-11-2018

Documentación

ArtisanBeer - Proyecto final DAW

Miguel Ángel García y Julián Rodenas

2º DAW

INDICE

**DOCUMENTACION PROYECTO FINAL**

**Miguel Ángel García Martínez**

**Julián Díaz Rodenas**

**Fecha 2018**

**Ciclo de Grado Superior y Grupo Informática**

**IES LEONARDO DA VINCI**

**DESARROLLO DE APLICACIONES WEB**

# Información acerca del proyecto

## Software utilizado

Para el desarrollo de este proyecto hemos utilizado:

* Lenguaje principal Java enfocado a desarrollo web
* Otros lenguajes utilizados: Bootstrap (HTML5 y CSS3), JavaScript, AJAX
* Entorno de desarrollo NetBeans en su versión 8.2
* Base de datos: MYSQL
* Gestor de base de datos: HeidiSQL
* Servidor de Aplicaciones: Glassfish Server 4.1.1
* Hibernate 4.1
* GitHub
* GitHub Desktop

## Descripción del proyecto

Aplicación web enfocada a la compraventa de cerveza artesanal. La idea surge de las dificultades que existen a la hora de distribuir este tipo de producto, ya que suele estar elaborado por pequeños productores con dificultades de publicitarse.

La aplicación permite ver un amplio catálogo de productos subidos por los propios usuarios vendedores. La aplicación permite registrarse tanto como usuario consumidor o como usuario vendedor, de forma que un usuario vendedor pueda acceder a sus pedidos y realizarlos, y un usuario vendedor puede acceder a un panel de control para poder poner a la venta productos y categorizarlos a su gusto.

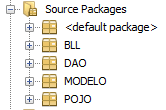
La aplicación está implementada con Java como lenguaje de programación principal utilizando el framework Hibernate para gestionar el acceso a la base de datos.

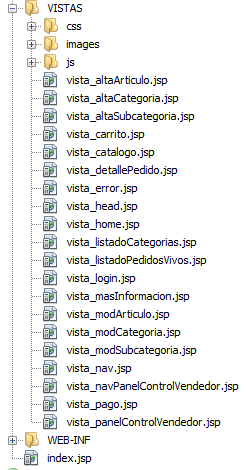
Las vistas han sido realizadas con HTML5 y hemos utilizado las librerías de Bootstrap 4 junto con CSS3 para realizar el diseño de la web.

El proyecto al realizarlo de forma conjunta ha tenido que ser actualizado utilizando un repositorio común, en nuestro caso utilizamos Git-Hub y usamos su aplicación de escritorio Git-Hub Desktop para sincronizar nuestro trabajo.

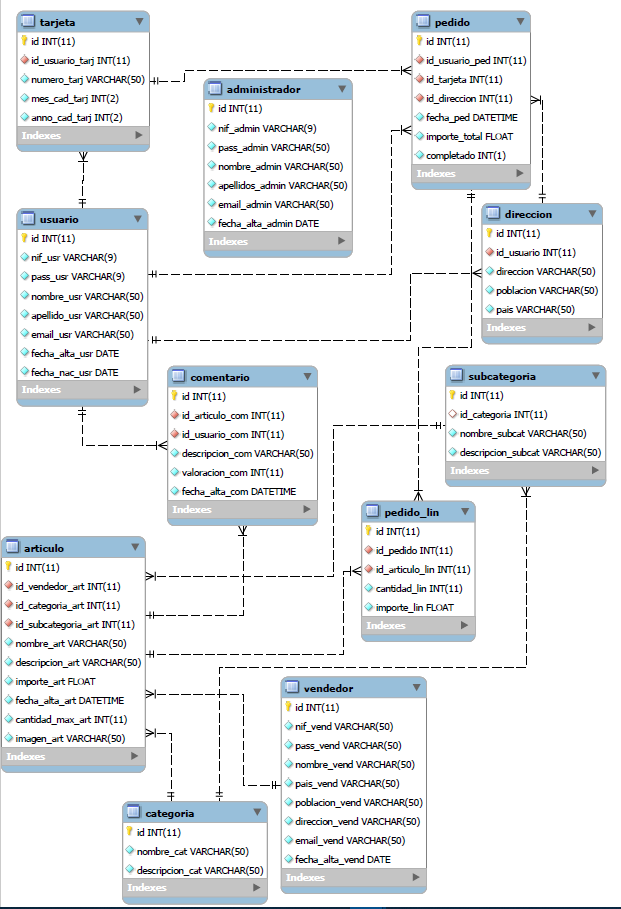
## Modelo de organización

Se ha utilizado un modelo de organización modelo-vista-controlador (MVC):





# Diagrama de tablas



# API de la aplicación

Se adjunta junto a la documentación del proyecto.

# Casos de uso

# Variables de session