FACULTAD DE VALLADOLID

Escuela de Ingeniería Informática

Servicios y Sistemas Web

Prototipo 2

Sergio Esteban Pellejero - Pablo Renero Balgañón - Álvaro Berruezo Franco - Alejandro Martínez Andrés

Índice

1. Estructura de la base de datos

El gestor de base de datos que vamos a utilizar en nuestra página web es MySQL. El diagrama de cómo se relacionan (entidad-relación) las diferentes tablas de nuestra base de datos, y los diferentes campos que almacenan es el siguiente:

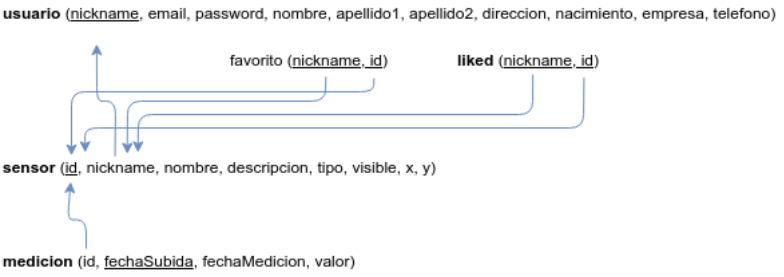

Figura 1: Diagrama de relaciones BB.DD

Los atributos que están subrayados indican las claves primarias de las tablas, y las diferentes flechas indican la relación entre las calves foráneas del resto de tablas y sus correspondientes claves primarias.

2. Tecnologías utilizadas

Nuestro grupo no ha optado por utilizar las tecnologías vistas en clase sino que hemos aprendido a usar unas nuevas. Las diferentes partes que han acabado componiendo nuestro sistema son las siguientes:

- **Flask**: utilizamos la versión 0.12.2 de flask, que es un framework escrito en python, basado en werkzeug y jinja2. Werkzeug es una biblioteca de funciones (python) para aplicaciones WSGI (Web Server Gateway Interface). Jinja2 es un lenguaje de plantillas (templates languajes) que en la práctica es el que nos permite introducir código python en páginas HTML de una manera sencilla.
- **Peewee**: es un ORM (object request manager) para que la interacción entre flask y la bbdd se haga de una manera mucho más cómoda. Peewee nos permite tratar las tablas de la bbdd como si fueran objetos y hacer las consultas en base a sus atributos de manera eficaz y en python, por lo que lo tenemos todo unificado.
- **Virtualenv**: nos permite trabajar con entornos virtuales de python, es decir, para proyectos grandes podemos necesitar versiones diferentes de python, queremos asegurar una mínima compatibilidad hacia atrás... esto se puede conseguir con virtualenv. Nos permite tener instaladas varias versiones de python en diferentes puntos del sistema como si se tratase de un contenedor.

3. Puesta en marcha del proyecto

Como sabemos que no utilizamos la tecnología estándar tenemos que ofrecer una pequeña explicación de como funciona nuestro sistema a grandes rasgos.

3.1. Requisitos

Tenemos que tener instalados los siguientes programas para que nuestro proyecto se ejecute correctamente:

- **Python3**: en distribuciones linux se puede instalar de la siguiente manera:

```
$ sudo apt-get install python3
```

- **MySQL-client**: en distribucioes linux se puede instalar de la siguiente manera:

```
$ sudo apt-get install mysql-client
```

- **Virtualenv**: en distribuciones linux se puede instalar de la siguiente manera:

```
$ sudo apt-get install virtualenv
```

3.2. Instalación y primer arranque

Una vez que hemos descargado los anteriores programas tenemos que ejecutar el fichero install.sh de nuestra práctica, que contiene el siguiente código:

```
#! /bin/bash

# Create virtual environment if it not exist
if [ ! -d menv ]
then
    virtualenv -p python3 menv
fi

# Source virtual environment
source menv/bin/activate

# Install python requirements from requirements
pip install -r requirements.txt

# Deactivate virtual environment
deactivate
```

En el caso de que no exista nos crea un entorno virtual en la carpeta en la que estamos ejecutando el fichero install.sh, después inicializa dicho entorno virtual e instala en él todo lo necesario para que nuestra práctica funcione. Las dependencias y programas se encuentran en un fichero llamado requirments.txt que contiene lo siguiente:

```
Flask
Flask-ini
peewee
configparser
pymysql
```

El instalador de python para el entorno virtual (pip) nos instala las tecnologías dichas en los apartados anteriores, como flask, peewee y sus correspondientes dependencias.

Antes de arrancar por primera vez nuestro proyecto tenemos que configurar el archivo que nos permitirá el acceso a la base de datos. El fichero tiene que ser rellenado por el profesor para atender a las necesidades de su base de datos. Es decir:

```
[DataBase]
# name --> tiene que poner el nombre de la bbdd a la que se conectara la pagina
name =
# user --> usuario de mysql que tengas para las bbdd
user =
# password --> contrasena para el acceso a las bbdd
password =
# host --> una conexion a una bbdd remota, aqui iria la ip, sino va localhost
host = localhost
# port --> puerto aleatorio mayor que 1024
port =
```

Por ejemplo un archivo de configuración relleno para mi máquina, sería:

```
[DataBase]
name = ssw
user = root
password = Sergio23121997
host = localhost
port = 3306
```

Para arrancar el sistema simplemente tenemos que ejecutar el fichero run.sh, que nos ejecutará el flask, nos montará el servidor web y se encargará de devolvernos una url y un puerto local en la que mirar nuestra página web.

4. Objetivos del prototipo 2

En esta entrega había que introducir la interacción con la base de datos y cubrir las funcionalidades básicas para comprobar que sabíamos realizar las interacciones entre los formularios HTML y nuestra base de datos, y presentar al usuario la información personalizada dependiendo de quien sea. Los casos de uso básicos que hemos adoptado implementar, de momento, son:

- **Registrarse**
- **iniciar Sesión**
- **Modificar Perfil**
- **Registrar Medición**

5. Reparto de trabajo

La carga de trabajo de cada uno de los integrantes ha sido:

- **Sergio Esteban**: 25 %.
- **Pablo Renero**: 25 %.
- **Alejandro Martínez**: 25 %.
- **Álvaro Berruezo**: 25 %.