

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

14-12-2017

# WEB BUSCAR DESTINO

Proyecto de Fin de Curso

Several thin, curved lines in dark blue and light grey originate from the bottom left and sweep upwards and to the right.

PATRICIA MARTÍNEZ LUCENA  
2º DAW

## Contenido

Contenido .....	2
IMPLEMENTAR UNA WEB-BUSCA-DESTINO .....	3
OBJETIVOS .....	3
CATÁLOGO DE REQUISITOS.....	3
INFORMACIÓN RELATIVA A CADA PÁGINA.....	4
INICIO/BÚSQUEDA:.....	4
REGISTRO:.....	7
LOGIN: .....	7
PERFIL: .....	8
EDITAR USUARIO:.....	8
BORRAR USUARIO: .....	8
BÚSQUEDAS/HISTORIAL:.....	9
BORRAR BÚSQUEDA:.....	9
TECNOLOGÍAS Y HERRAMIENTAS .....	9
MODELOS Y DIAGRAMAS .....	10
MODELO DE CASOS DE USO .....	10
DIAGRAMA DE ACTIVIDADES.....	10
ÁRBOL DE NAVEGACIÓN .....	11
ÁRBOL DE NAVEGACIÓN DE FICHEROS .....	11
DIAGRAMA DE COLABORACIONES.....	12
DIAGRAMA DE SECUENCIA .....	13
MODELO ENTIDAD RELACIÓN.....	13
DIAGRAMA DE CLASES .....	14
VISTA DESPLEGADA DE LA ARQUITECTURA .....	15
ALMACENAMIENTO .....	15
ARQUITECTURA.....	16
DESPLIEGUE DE LA APLICACIÓN.....	16
SERVICIOS REST UTILIZADOS.....	17
TIEMPO EMPLEADO EN HACER LA APLICACIÓN: .....	17
ESTIMACIÓN DE COSTES EN BASE AL TIEMPO EMPLEADO: .....	17
POSIBLES MEJORAS / COSAS QUE AÑADIR A LA APLICACIÓN .....	17

# IMPLEMENTAR UNA WEB-BUSCA-DESTINO

ALUMNO: PATRICIA MARTÍNEZ LUCENA

CURSO: 2º DAW

EMAIL: patrimartinez31@hotmail.com

## OBJETIVOS

Una aplicación en la cual un usuario puede realizar búsquedas de trayectos en coche entre dos lugares de España, ya sea registrándose para almacenar sus búsquedas y poder verlas en otro momento, o sin registrarse.

La aplicación ofrece datos como la distancia que existe entre los dos lugares, la duración del trayecto en vehículo y datos meteorológicos del lugar de destino (temperatura, nubes, precipitaciones, humedad, viento y descripción). También mostrará un mapa en el cual se dibuja el trayecto que se realizaría.

En la aplicación se podrá editar la información del usuario o incluso eliminarlo.

Las búsquedas también podrán ser eliminadas por el usuario en el historial de búsqueda.

## CATÁLOGO DE REQUISITOS

### Usuario:

La aplicación guardará la información de los usuarios que se registren.

Un usuario registrado podrá:

- Editar su perfil.
- Eliminar su cuenta.

Usaremos el código del usuario (usuario en la aplicación) y la contraseña para controlar el acceso a la aplicación en el apartado login.

Datos del Usuario:

- Código del Usuario (codUsuario)
- Nombre del Usuario (nomUsuario)
- Email (email)
- Contraseña (password)

### Búsqueda:

La aplicación guarda las búsquedas que realiza un usuario registrado automáticamente.

Un usuario registrado podrá:

- Almacenar las búsquedas que realice en la aplicación (proceso automático).
- Visualizar el historial de búsquedas que realice en la aplicación.

- Eliminar las búsquedas una a una.
- Eliminar todo el historial de una vez.

Datos de la Búsqueda:

- |  |                 |
|--|-----------------|
| • Código de la Búsqueda                                | (codBusqueda)   |
| • Código del Usuario que realizó la búsqueda           | (codUsuario)    |
| • Fecha de realización de la búsqueda                  | (fecha)         |
| • Lugar de salida                                      | (salida)        |
| • Lugar de llegada                                     | (llegada)       |
| • Distancia entre los dos lugares                      | (distancia)     |
| • Duración del trayecto en coche                       | (duracion)      |
| • Temperatura en el lugar de llegada                   | (temperatura)   |
| • Probabilidad de nubes en el lugar de llegada         | (nubes)         |
| • Probabilidad de precipitación en el lugar de llegada | (precipitacion) |
| • Humedad relativa en el lugar de llegada              | (humedad)       |
| • Velocidad del viento en el lugar de llegada          | (viento)        |
| • Descripción del tiempo que hará                      | (frase)         |

## INFORMACIÓN RELATIVA A CADA PÁGINA

### INICIO/BÚSQUEDA:

Página principal en la que se puede realizar la búsqueda sin importar si se ha iniciado sesión o no.

Si no se ha iniciado sesión, únicamente realiza la búsqueda y muestra una tabla con los datos correspondientes a la búsqueda y un mapa en el que se dibuja la ruta del trayecto. Si por el contrario se ha iniciado sesión, dichos datos de la búsqueda serán almacenados en la base de datos del usuario correspondiente.

La aplicación sólo permite realizar búsquedas en España y el trayecto se realiza en coche siempre.

Elementos de la página:

- Un formulario de dos campos: lugar de salida y lugar de llegada. Ambos son de tipo cadena o string.
- Un botón de enviar que recoge los valores escritos en los campos anteriores y redirige a la página del controlador de la página de búsqueda.
- Un botón de iniciar sesión, por si el usuario quiere iniciar sesión para guardar sus búsquedas. El botón redirige a la página de login.
- Un botón de registro, por si el usuario quiere iniciar sesión para guardar sus búsquedas y aún no tiene una cuenta. El botón redirige a la página de registro.

Trozo de código para extraer información JSON de una URL:

```
//URL DEL SERVICIO REST DE LA API DE GOOGLE (DISTANCIA Y DURACIÓN).
$gmUrl="https://maps.googleapis.com/maps/api/distancematrix/json?units=imperial&origins="
.urlencode($campos['salida']).",spain".&destinations=".urlencode($campos['llegada'])
.",spain".&key=AIzaSyBOTmJGCrDskI_CD6DePfvMP-SCsKLRRt0";
//ARCHIVO JSON QUE RECIBE LA URL
$gmJson = @file_get_contents($gmUrl);
if ($gmJson === false) {
    print("<br/><span class='errorphp'>NO SE PUEDE ACCEDER A LA URL GMAPS</span> <br/>");
}
//ARRAY QUE DEVUELVE EL JSON DE LA API.
$gmArray=json_decode($gmJson,true);
if($gmArray['rows'][0]['elements'][0]['status'] == "NOT_FOUND" ||
$gmArray['rows'][0]['elements'][0]['status'] == "ZERO_RESULTS"){
    print " <br/>NO SE HA ENCONTRADO LA BÚSQUEDA";
}
```

Existe validación de entrada para los dos campos, tanto en el lado del cliente como en el lado del servidor.

En el lado del cliente he utilizado Javascript, mientras que en el lado del servidor he utilizado PHP para validar el formulario.

Código Javascript:

clnicio.php

```
var patron=/^[a-zA-ZàèìòùÀÈÌÒÙáéíóúÁÉÍÓÚâëîôûÂÊÎÔÛãäöÿÃäÿÿçç\`s]+$/;
function validacionsalida(){
    var camposalida = document.getElementById("desde");
    //SI EL CAMPO ESTÁ VACÍO
    if(camposalida.value.length < 1){
        sessionStorage.setItem("errorsalida","Campo vacío");
        return false;
    }
    //SI NO COINCIDE CON EL PATRÓN
    else if(!camposalida.value.match(patron)){
        sessionStorage.setItem("errorsalida","Introduce solo letras");
        return false;
    }
    //SI COINCIDE CON EL PATRÓN
    else{
        sessionStorage.setItem("errorsalida","");
        return true;
    }
}
function validacionllegada(){
    var campollegada = document.getElementById("address");
    //SI EL CAMPO ESTÁ VACÍO
    if(campollegada.value.length < 1){
        sessionStorage.setItem("errorllegada","Campo vacío");
        return false;
    }
    //SI NO COINCIDE CON EL PATRÓN
    else if(!campollegada.value.match(patron)){
        sessionStorage.setItem("errorllegada","Introduce solo letras");
        return false;
    }
    //SI COINCIDE CON EL PATRÓN
    else{
        sessionStorage.setItem("errorllegada","");
        return true;
    }
}
```

vInicio.php

```
function mostrarerrores() {
    if(errorsalida.length>0) {
        document.getElementById("errorsalida").innerHTML=errorsalida;
        sessionStorage.setItem("errorsalida","");
    }if(errorllegada.length>0){
        document.getElementById("errorllegada").innerHTML=errorllegada;
        sessionStorage.setItem("errorllegada","");
    }
    if(errorsalida.length<1 && errorllegada.length<1){
        calculaRuta();
    }
}
```

Código PHP:

clnicio.php

```
//ACCIÓN SI SE PULSA EL BOTÓN ACEPTAR
if(isset($_POST['aceptar'])){
    $entradaOK=true;
    //RECOGER EN UN ARRAY LOS VALORES DE LOS CAMPOS DEL FORMULARIO.
    $campos['salida']=$_POST['salida'];
    $campos['llegada']=$_POST['llegada'];
    //INCLUIR LA LIBRERÍA DE FUNCIONES
    include "core/funciones.php";
    if(validartexto($campos['salida'])){
        $entradaOK=false;
        //ALMACENAR EN EL ARRAY EL ERROR RECIBIDO.
        $errores['salida']=validartexto($campos['salida']);
        $campos['salida']="";
    }
    if(validartexto($campos['llegada'])){
        $entradaOK=false;
        //ALMACENAR EN EL ARRAY EL ERROR RECIBIDO.
        $errores['llegada']=validartexto($campos['llegada']);
        $campos['llegada']="";
    }
}
```

funciones.php

```
function validartexto($texto) {
    $patrontexto="/^[a-zA-ZàèìòùÀÈÌÒÙáéíóúýÁÉÍÓÚÝâëîôûÂÊÎÔÛãñõÃÑÕäëïöÿÄËÏÖŸçÇ]+$/";
    $error="";
    if(empty(trim($texto))){ //Se genera error si el campo está vacío
        $error = "Campo vacío";
    }else{
        if(!preg_match($patrontexto, $texto)){
            $error="Introduce solo letras";
        }
    }
    return $error;
}
```

## REGISTRO:

Página donde se puede dar de alta a un usuario.

Elementos de la página:

- Un formulario donde se insertan los datos del usuario (código de usuario, nombre, email y password).
- Un botón de aceptar, que añade el usuario a la base de datos si no ha ocurrido ningún error y redirige a la página de inicio con la sesión iniciada automáticamente.

Si el usuario ya existe, la aplicación mostrará un mensaje conforme el usuario ya existe.

- Un botón de cancelar, que redirige otra vez a la página de inicio.

## LOGIN:

Página de inicio de sesión de un usuario (control de acceso).

Elementos de la página:

- Un formulario donde se insertan los datos del usuario (código y password). La aplicación realiza la validación del usuario accediendo a la base de datos del servidor.
- Un botón de aceptar, que inicia la sesión del usuario si se ha validado correctamente y redirige a la página de inicio con la sesión iniciada. Si no se ha encontrado el usuario o no coincide la contraseña, la aplicación mostrará por pantalla un mensaje informativo.
- Un enlace a la página de registro, que redirige a la página de registro por si no el usuario no se había dado de alta.
- Un botón de cancelar, que redirige otra vez a la página de inicio.

En esta página sólo utilizo validación en el lado del servidor. Compruebo si el usuario coincide con el usuario guardado en la base de datos del servidor.

```
//COMPROBAR SI EXISTE EL USUARIO LLAMANDO A LA FUNCIÓN DE LA CLASE USUARIO Y ALMACENARLO EN LA VARIABLE
$usuario=Usuario::validarUsuario($_POST['usuario'],hash('sha256', $_POST['password']));
//ACCIÓN SI NO SE HA ENCONTRADO EL USUARIO
if($usuario==null){
    //LA ENTRADA SERÁ FALSA
    $entradaOK=false;
    //INTRODUCIR VALORES EN LAS VARIABLES DE ERROR
    $error = "Debes introducir un nombre de usuario y una contraseña válidos";
    $span="<span class='error' name='error'>".$error."</span>";
}
```

## PERFIL:

Página con información del usuario que inició sesión donde se puede editar o eliminar su cuenta. Se muestran los datos del usuario en un formulario visual.

Elementos de la página:

- Un formulario visual con los datos del usuario (código de usuario, nombre y email). En esta sección no se podrán editar los campos del formulario.
- Un botón de Editar usuario, que redirige la información de la sesión del usuario a la página de editar usuario.
- Un botón de Borrar usuario, que redirige la información de la sesión del usuario a la página de borrar usuario.
- Un botón de volver, que redirige a la página de inicio.

## EDITAR USUARIO:

Página de edición del usuario que inició sesión.

Elementos de la página:

- Un formulario donde muestra los datos del usuario en cada campo correspondiente. El código del usuario (Usuario) no se podrá editar. El resto de los campos sí se pueden editar (nombre de usuario, email y contraseña). Si se cambia algún campo, se deberá introducir la contraseña actual del usuario o no se podrá editar la información. Los campos de password (password antiguo y nuevo password) estarán vacíos para introducir la contraseña, bien sea sólo la antigua o las dos si se quiere cambiar.
- Un botón de aceptar, que actualiza la información del usuario en la base de datos y en la sesión si no existe ningún error al insertar los datos y redirige a la página de inicio. Si existe algún error, simplemente muestra un mensaje por pantalla advirtiendo de que no se ha podido editar el usuario.
- Un botón de cancelar, que redirige otra vez a la página de inicio.

## BORRAR USUARIO:

Página que pregunta al usuario si realmente quiere borrar su perfil.

Elementos de la página:

- Un texto informativo que advierte de que se está intentando eliminar una cuenta.
- Un formulario donde se visualizan los datos del usuario (código, nombre, email y password). No se podrán editar los campos.
- Un botón de aceptar, que elimina la cuenta de usuario y la sesión correspondiente y redirige a la página de inicio, si ha podido eliminar el usuario. Si no ha podido eliminar el usuario, redirige otra vez a la página de perfil del usuario.
- Un botón de cancelar, que redirige otra vez a la página de perfil del usuario.



## BÚSQUEDAS/HISTORIAL:

Página que lista todas las búsquedas que ha realizado ese usuario, con opción a poder borrar una a una o todo el historial de una vez.

Elementos de la página:

- Una tabla con los datos de la búsqueda del usuario (Usuario que realizó la búsqueda y datos de la búsqueda).
- En la última celda del encabezado de la tabla existe un botón de limpiar historial, el cual elimina todas las búsquedas realizadas por el usuario que inició sesión.
- En la última celda de cada fila existe un botón de borrar, el cual redirige los datos de la búsqueda a la página de borrar búsqueda.
- Un botón de volver, que redirige a la página de inicio.

## BORRAR BÚSQUEDA:

Página que pregunta al usuario si realmente quiere borrar la búsqueda seleccionada en el historial de búsquedas.

Elementos de la página:

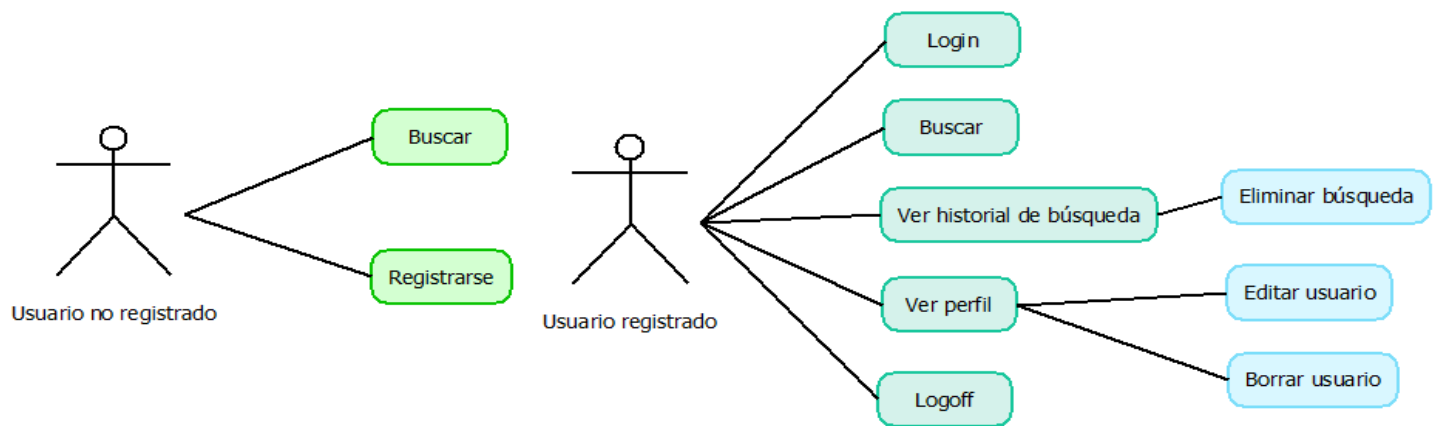
- Un formulario donde se visualizan algunos datos de la búsqueda (código del usuario, fecha, lugar de salida y lugar de llegada). Estos datos no se podrán editar.
- Un texto informativo que advierte de que se está intentando eliminar una búsqueda.
- Un botón de aceptar, que elimina la búsqueda y redirige a la página del historial si no ha ocurrido ningún error. Si no se ha podido eliminar la búsqueda muestra por pantalla un mensaje informativo y redirige de nuevo a la página del historial.
- Un botón de cancelar, que redirige otra vez a la página del historial.

## TECNOLOGÍAS Y HERRAMIENTAS

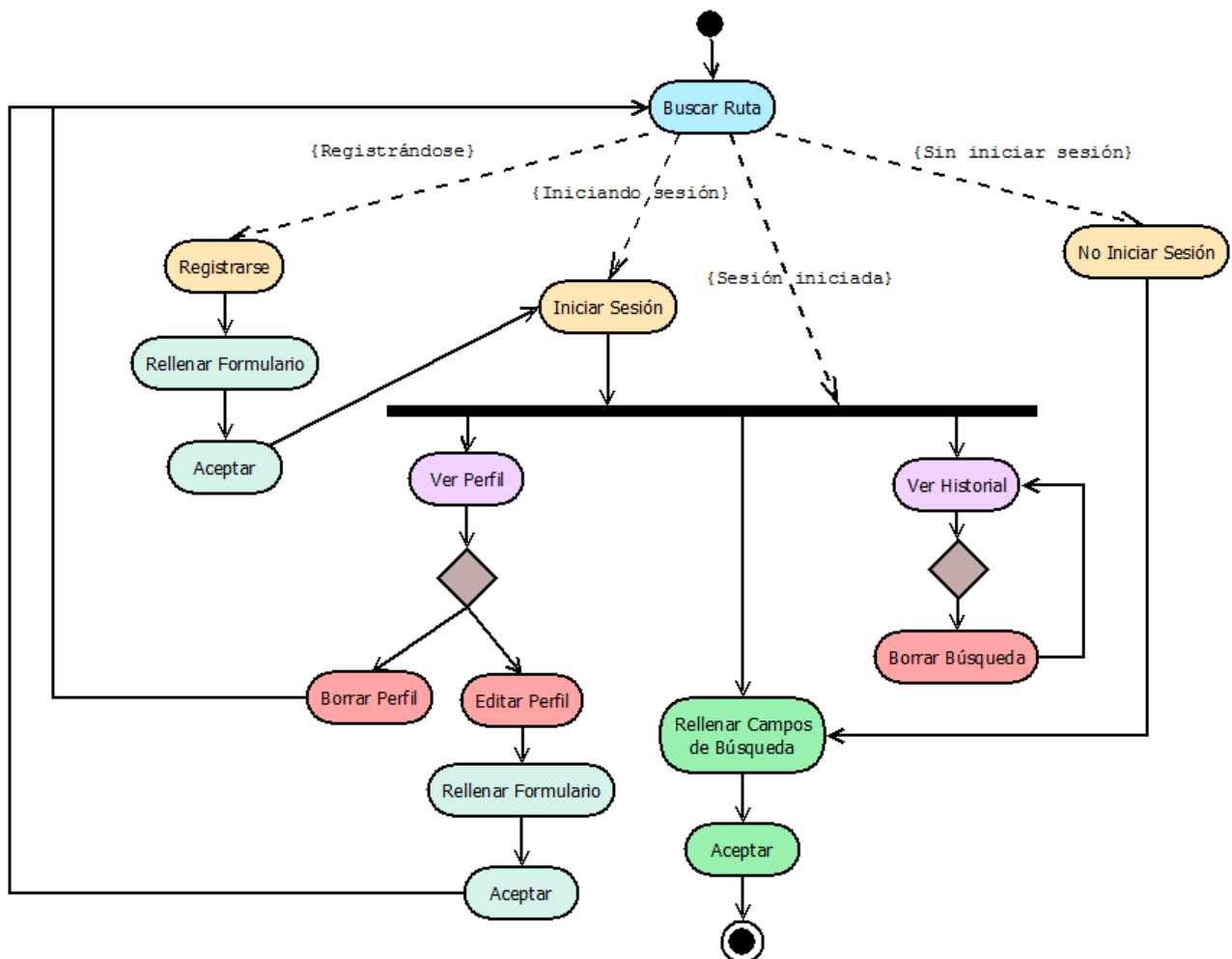
<b>Entorno de desarrollo</b>	Windows 7 Enterprise x64
<b>Entorno de explotación</b>	Ubuntu Server x64
<b>Servidor HTTP</b>	Apache 2
<b>Sistema Gestor de Base de Datos</b>	MySQL-Server
<b>Lenguajes en el lado del servidor</b>	PHP
<b>Lenguajes en el lado del cliente</b>	JavaScript, JQuery
<b>Lenguajes de marcas</b>	HTML y CSS
<b>Editor de texto</b>	Sublime Text 3

# MODELOS Y DIAGRAMAS

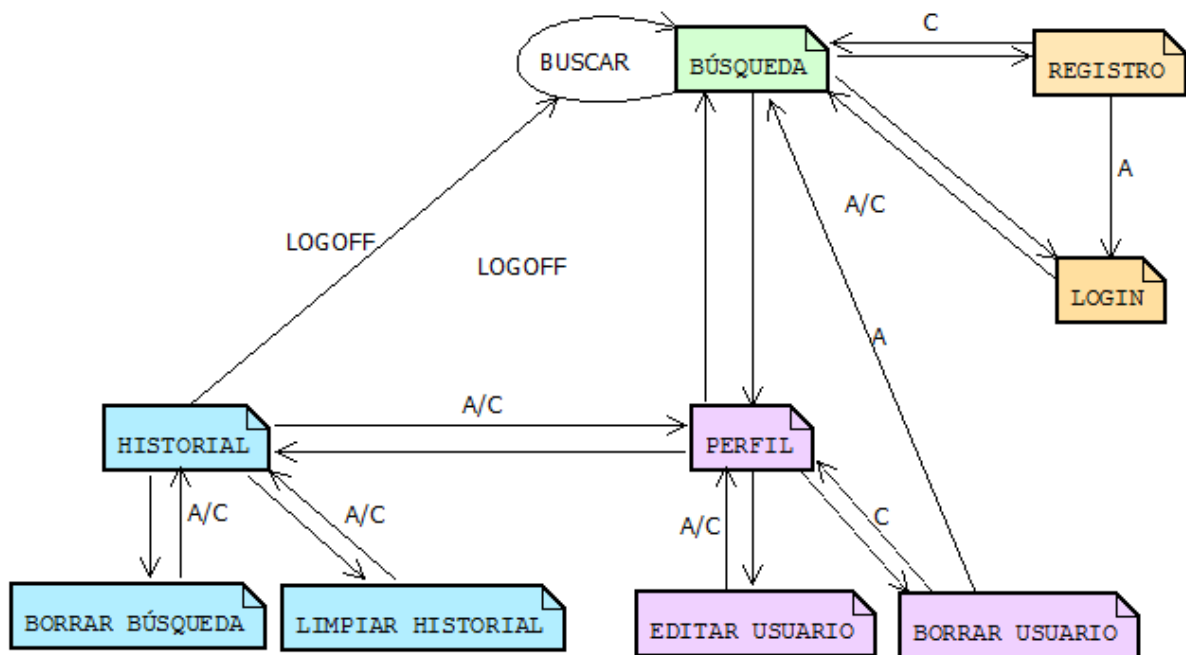
## MODELO DE CASOS DE USO



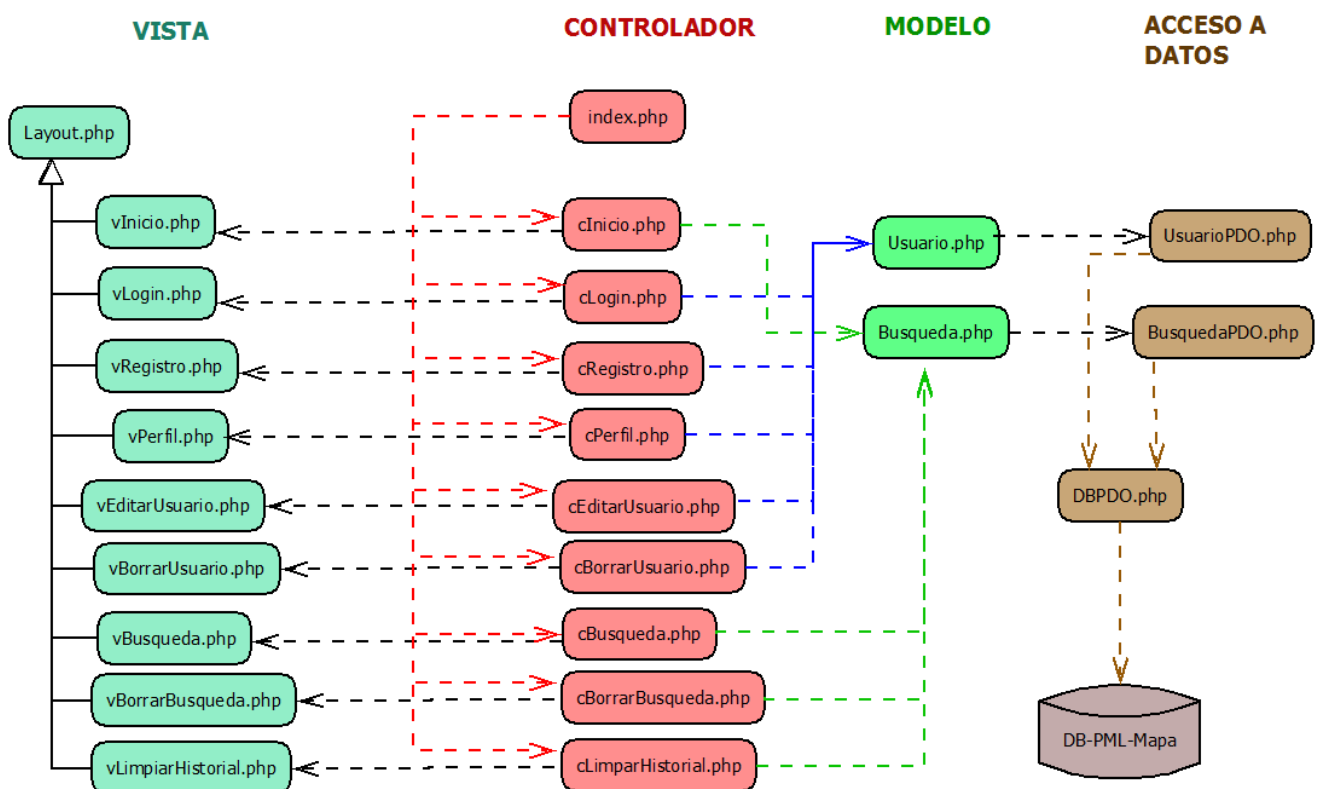
## DIAGRAMA DE ACTIVIDADES



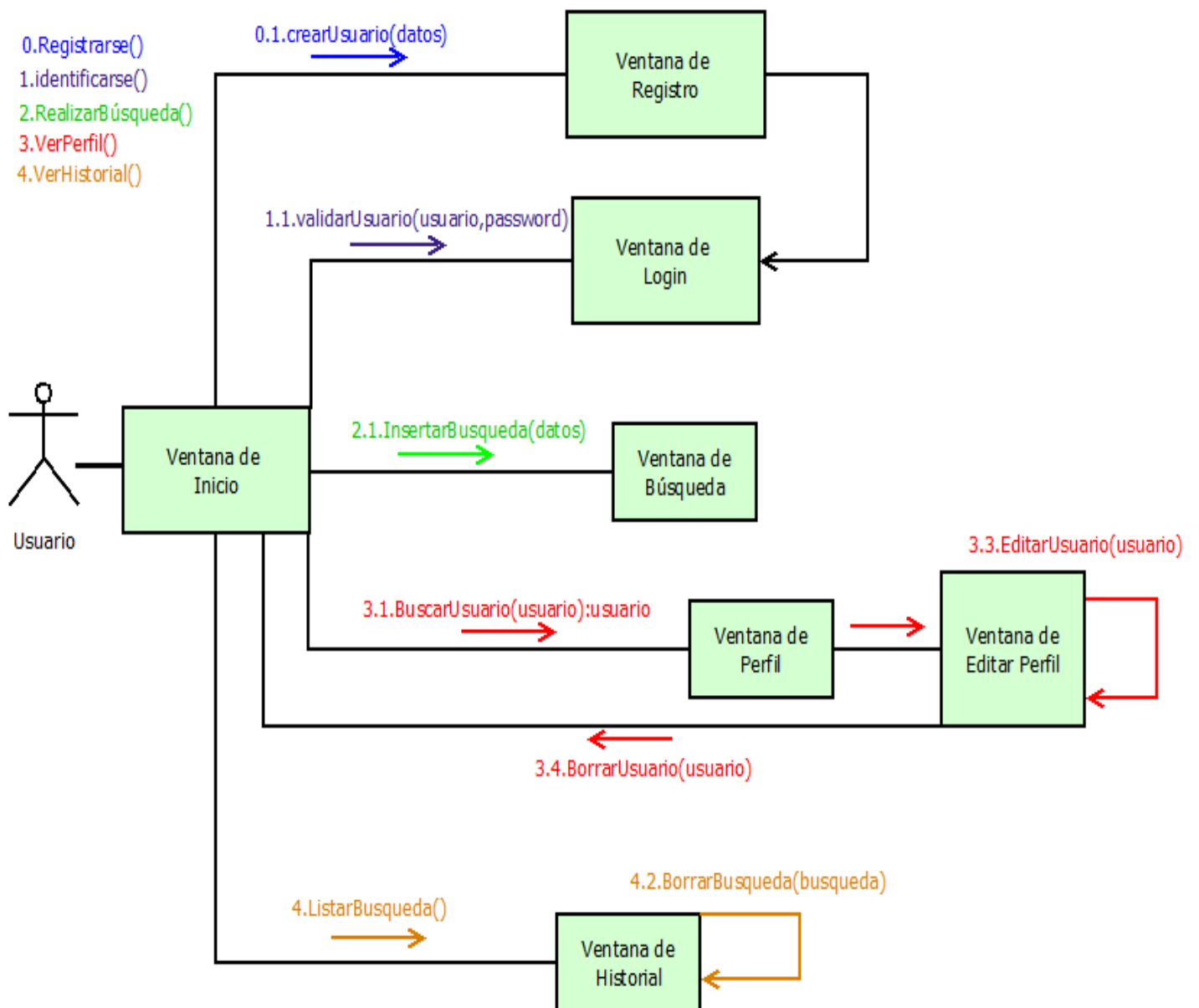
## ÁRBOL DE NAVEGACIÓN



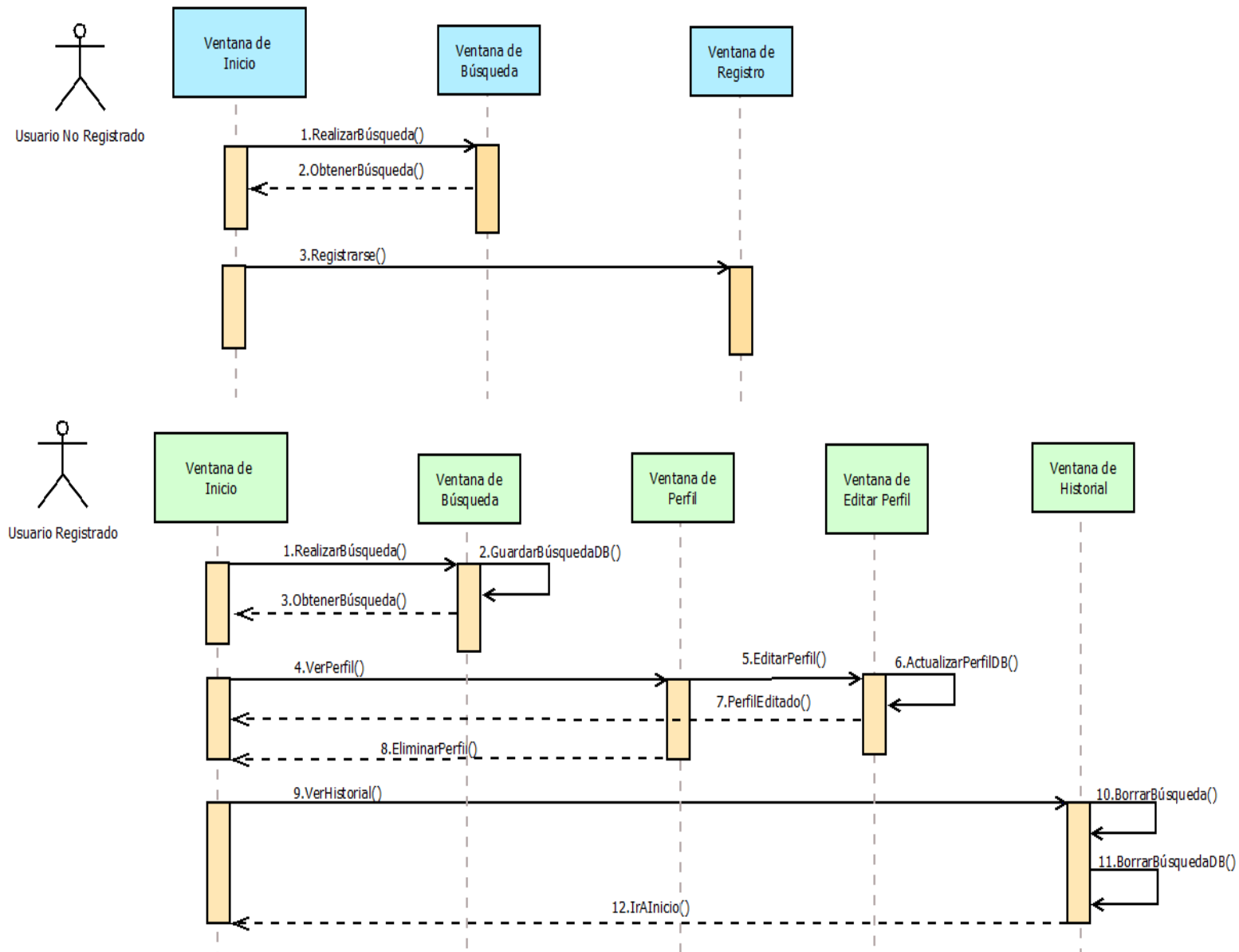
## ÁRBOL DE NAVEGACIÓN DE FICHEROS



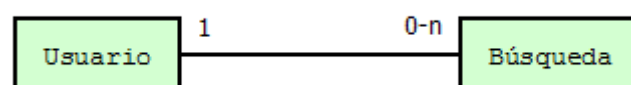
## DIAGRAMA DE COLABORACIONES



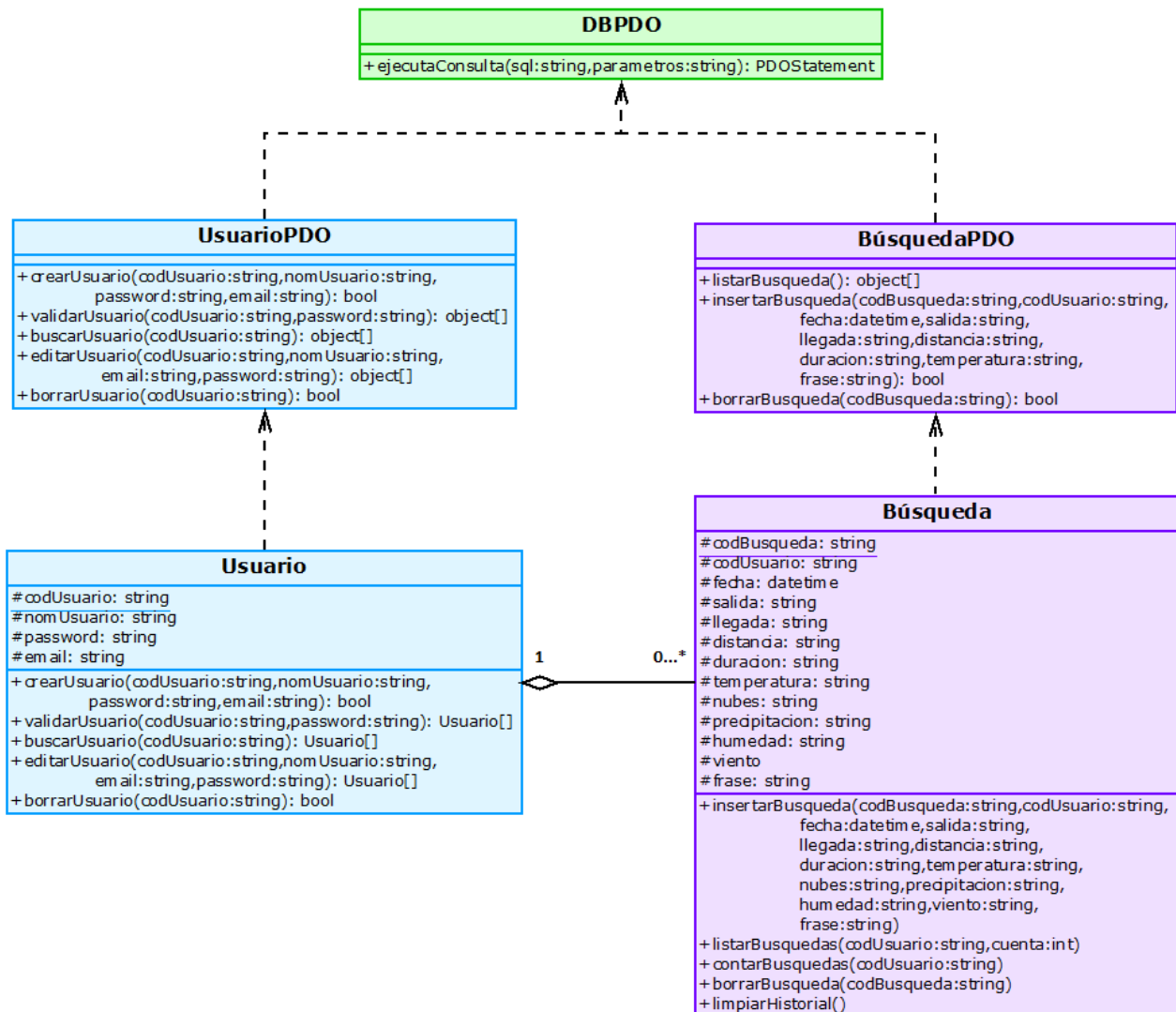
## DIAGRAMA DE SECUENCIA



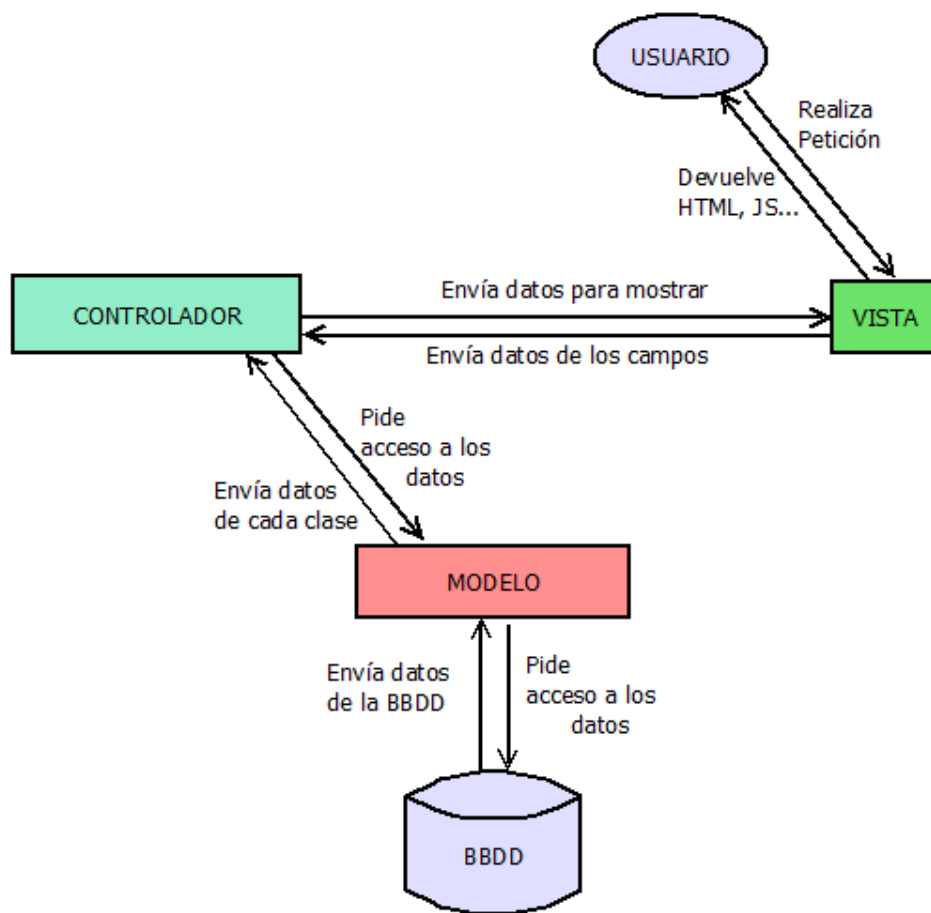
## MODELO ENTIDAD RELACIÓN



## DIAGRAMA DE CLASES



## VISTA DESPLEGADA DE LA ARQUITECTURA

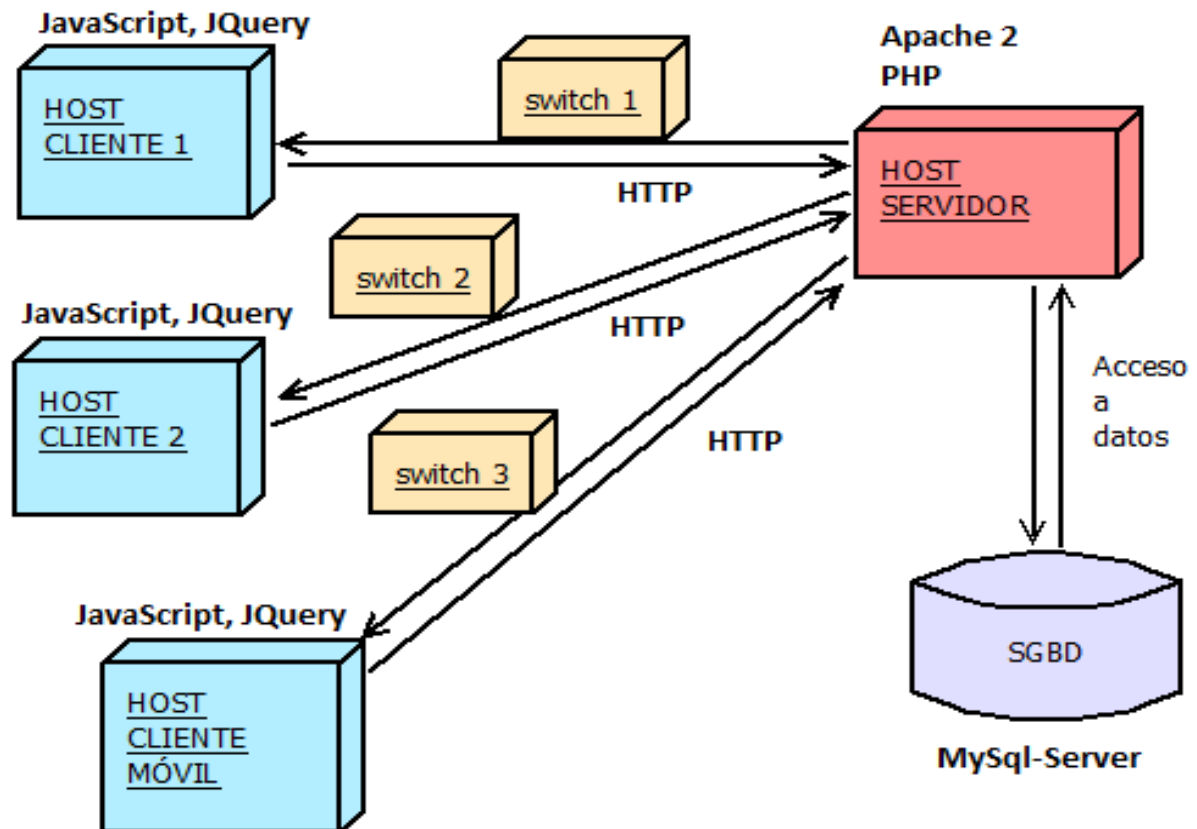


## ALMACENAMIENTO

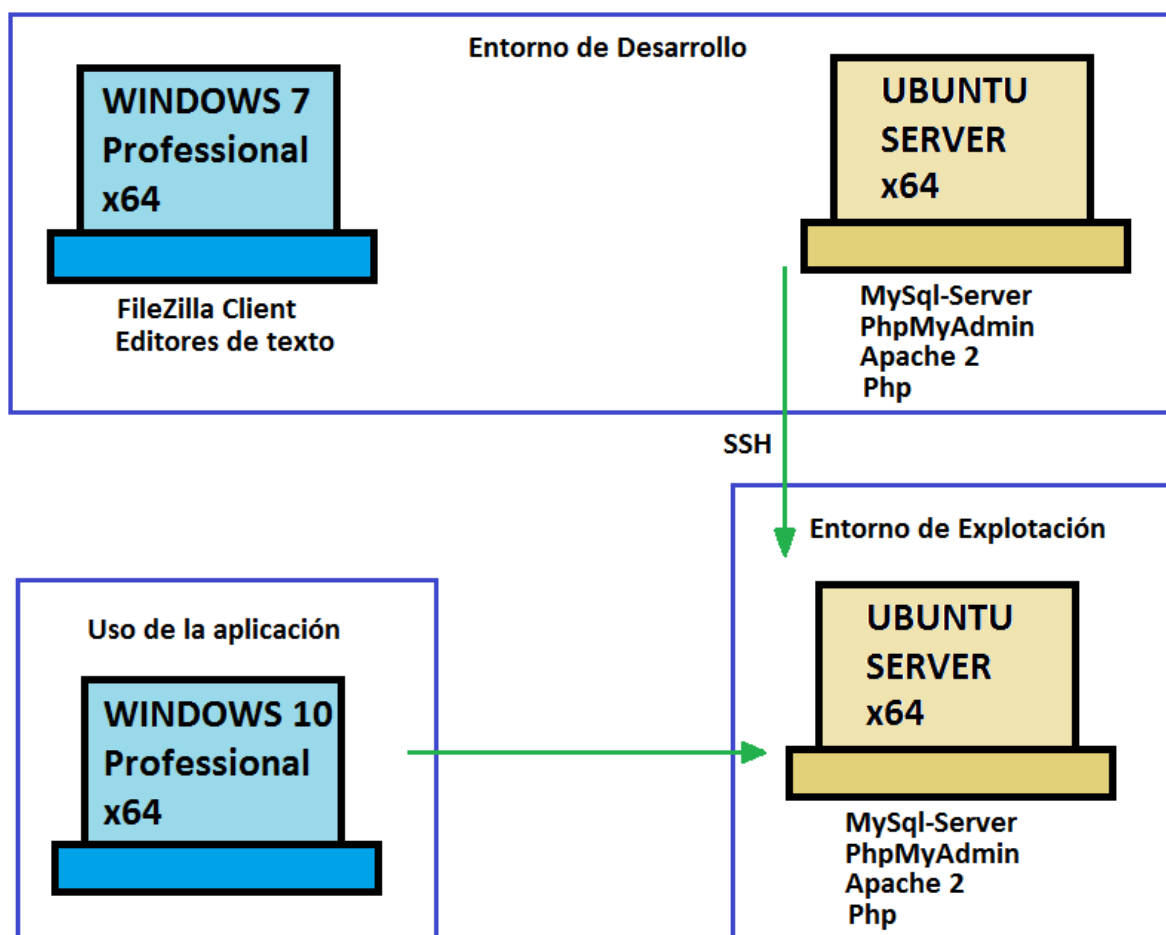
- config
- controller
- core
- doc
- model
- view
- webroot
- index.php

- ▼ Proyecto
  - config
  - controller
  - core
  - ▼ doc
    - Diagramas-Modelos
    - Documentos
    - ScriptDB
  - model
  - view
  - ▼ webroot
    - > css

## ARQUITECTURA



## DESPLIEGUE DE LA APLICACIÓN





## SERVICIOS REST UTILIZADOS

### GOOGLE MAPS API

De esta API saco los datos relacionados a la distancia entre los dos lugares. También calcula la ruta de viaje en coche y la dibuja en el mapa.

URL:

<https://maps.googleapis.com/maps/api/distancematrix/json&key=AlzaSyBOTmJGCrDskICD6DePfvmP-SCsKLRRT0>

### WEATHERBIT.IO

De esta API saco los datos relacionados al tiempo en 48 horas de un lugar en concreto, en este caso del lugar de llegada.

URL Tiempo:

<https://api.weatherbit.io/v2.0/forecast/hourly&lang=es&key=9d1b4156f1cb4c7295d5d203d68969d9>

## TIEMPO EMPLEADO EN HACER LA APLICACIÓN:

## ESTIMACIÓN DE COSTES EN BASE AL TIEMPO EMPLEADO:

## POSIBLES MEJORAS / COSAS QUE AÑADIR A LA APLICACIÓN

### TUTOR:

Baldomero Sanchez

bsanchezpe@educa.jcyl.es