



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Diseño e implementación del servidor web "Ruta de la Tapa"

Trabajo Fin de Grado
Grado en Ingeniería Informática

Autor: David Valero García

Tutor: Sergio Sáez Barona

2014/2015



Resumen

En este Trabajo Final de Grado se va a desarrollar una aplicación web para gestionar los concursos de Rutas de la Tapa, en la que distintos establecimientos gastronómicos se unen para competir entre ellos.

Esta web permite a los usuarios registrarse como Organizadores o Participantes, crear rutas con estilos personalizados, inscribirse en rutas existentes con imágenes de las tapas, y descargar documentos PDF con los códigos QR que se les proporcionarán a los clientes para que los repartan entre los Consumidores para poder votar.

Al final del concurso, el sistema hace un recuento de los votos y se anuncian los ganadores.

Palabras clave: web, ruta, tapa, PHP, QR, API

Resum

En aquest Treball de Final de Grau es va a desenvolupar una aplicació web per a gestionar els concursos de “Rutes de la Tapa”, en la que distints establiments gastronòmics s’uneixen per a competir entre ells.

Aquesta web permet als usuaris registrar-se com Organitzadors o Participants, crear rutes amb estils personalitzats, inscriure’s en rutes existents amb imatges de les tapes, i descarregar documents PDF amb els codis QR que es proporcionaran als clients perquè els repartisquen entre els Consumidors per a poder votar.

Al final del concurs, el sistema fa un recompte dels vots i s’anuncien els guanyadors.

Paraules clau : web, ruta, tapa, PHP, QR, API



Abstract

In this Final Degree Project, we are going to develop a web application to manage "Tapa Routes" contests, where several gastronomic establishments join themselves in order to compete against them.

This website allows users to register as organizers or participants, creating routes with custom styles, joining existing routes with tapas images and downloading QR codes that will be provided to customers for distributing among consumers to vote.

At the end of the contests, system makes a recount of the votes and the winners are announced.

Keywords : web, route, tapa, PHP, QR, API

Tabla de contenidos

1	Introducción	9
2	Motivación y objetivos	10
2.1	Introducción	10
2.2	Motivación	10
3	Estado del arte	12
3.1	Lenguajes de programación para servidores	12
3.2	Aplicaciones relacionadas	13
4	Especificación de requisitos	17
4.1	Introducción	17
4.1.1	Propósito y alcance.....	17
4.1.2	Objetivos.....	17
4.1.3	Definiciones y acrónimos	18
4.2	Descripción general	18
4.2.1	Perspectiva del producto.....	18
4.2.2	Funcionalidad del producto	19
4.2.3	Características de usuario	19
4.2.4	Restricciones del sistema	21
4.2.5	Dependencias del sistema	21
4.3	Requisitos específicos.....	21
4.3.1	Requisitos de interfaz externos.....	21
4.3.2	Requisitos funcionales	21
4.3.3	Requisitos de rendimiento	23
4.3.4	Requisitos lógicos de base de datos	23
5	Análisis	24
5.1	Diagrama de clases / entidades.....	24
5.2	Casos de uso.....	25
6	Diseño	31
6.1	Introducción	31
6.2	Capa de presentación.....	31
6.3	Capa de persistencia	34
6.4	Capa de lógica	37



7	Detalles de implementación	42
7.1	Tecnologías utilizadas.....	42
7.1.1	PHP.....	42
7.1.2	Marcos de trabajo para PHP	42
7.1.3	Librerías auxiliares PHP	43
7.2	Plataformas de desarrollo.....	44
7.2.1	Control de versiones.....	47
7.3	Estructura de los ficheros y directorios.....	48
7.3.1	Carpetas de Codeigniter	48
7.3.2	Carpetas propias.....	48
7.4	Estructura de la Base de Datos.....	49
7.4.1	Ion Auth.....	49
7.4.2	Propias de la aplicación web	50
7.5	APIs REST.....	50
7.5.1	Obtención de rutas disponibles	51
7.5.2	Obtención de detalles de ruta	51
7.5.3	Obtención de resultados del concurso.....	52
7.5.4	Inserción de nueva valoración de usuario	52
8	Despliegue.....	53
8.1	Instalación de servicios	53
8.2	Desplegado de código	53
8.3	Configuración de BD	53
8.4	Configuración router	56
9	Pruebas	57
9.1	Resultado de diseño final	57
9.2	Pruebas en diferentes navegadores.....	64
9.3	Pruebas en diferentes resoluciones.....	66
10	Conclusiones	71
11	Bibliografía	73



Lista de ilustraciones

Figura 1. Portal de rutadeltardeovalencia.es	13
Figura 2. Detalles de una tapa en rutadeltardeovalencia.es	14
Figura 3. Portada de estamosdetapas.com	15
Figura 4. Diagrama de clases	24
Figura 5. Casos de uso para usuarios	26
Figura 6. Casos de uso para organizadores y participantes	30
Figura 7. Diseño original de la portada	32
Figura 8. Diseño inicial de una de las rutas	33
Figura 9. Diseño inicial del panel de control del organizador	33
Figura 10. Esquema de BD para usuarios	34
Figura 11. Esquema de BD de las rutas	35
Figura 12. Esquema completo de la BD	37
Figura 13. Diagrama de alcanzabilidad	41
Figura 14. Captura de pantalla de PhpStorm	44
Figura 15. Captura de pantalla de Sequel Pro	45
Figura 16. Captura de pantalla de MAMP Pro	45
Figura 17. Captura de pantalla de SourceTree	46
Figura 18. Captura de commits en proyecto GIT	47
Figura 19. Captura de SequelPro. Ventana de conexión a BD	54
Figura 20. Captura de SequelPro. Creación de nueva BD	54
Figura 21. Fichero de configuración de BD	55
Figura 22. Configuración de URL base	55
Figura 23. Ejemplo de configuración del router	56
Figura 24. Diseño final de la portada de la ruta de la tapa	58
Figura 25. Diseño final de la página de características	59
Figura 26. Diseño final de la portada de una ruta	60
Figura 27. Panel superior del panel de control de los organizadores	61
Figura 28. Diseño final de la pantalla de creación de rutas	62
Figura 29. Diseño final de las solicitudes en el panel de control de los organizadores	63
Figura 30. Diseño final del panel de control de los establecimientos	64
Figura 31. Composición de la portada en diferentes navegadores	65
Figura 32. Composición de la ventana de creación de ruta en diferentes navegadores	66
Figura 33. Captura de la aplicación en un dispositivo grande	67
Figura 34. Captura de la aplicación en un dispositivo mediano	68
Figura 53. Captura de la aplicación en un dispositivo pequeño	69

1 Introducción

Es indudable que vivimos en la era de Internet. La posibilidad de tener una gran cantidad de información centralizada hace que los usuarios, las personas, suelen acudir en primera instancia a la red para buscar cualquier tipo de servicio.

Por ello la presencia y el buen posicionamiento en este medio cobra cada día más importancia.

La industria gastronómica, a pesar de ser un negocio tradicional, está claramente influenciado por el terreno tecnológico. Desde permitir consultar la carta online, hasta webs dedicadas a recolectar opiniones acerca de ellos.

La comida, ha pasado de ser una necesidad humana básica a convertirse en un “espectáculo”. Prueba de ello son la multitud de fotos de comida que inundan la red.

Pero esta manera de disfrutar la comida no se ciñe exclusivamente a Internet. Son muchos los eventos gastronómicos, degustaciones, concursos de TV que hacen que incluso ver a otros cocinar, sea un placer.

Una seña de identidad de España, son las Tapas. Esencialmente es un aperitivo que , opcionalmente, se puede acompañar con una bebida. Es un alimento ligero, más rápido de preparar y permiten que el cliente pueda degustar una gran variedad de platos con una sola visita.

Por ello es normal que se celebren ferias en las que varios establecimientos puedan darse a conocer haciendo que los clientes los visiten de una manera divertida, fácil y económica.

Las rutas de las tapas siempre se han movido en un ámbito local con publicidad tradicional y no cuentan con las facilidades de gestión que un sistema informatizado les puede brindar por lo que el propósito de este trabajo, es el de crear una herramienta que permita: que los clientes puedan encontrar fácilmente establecimientos cerca de su zona de una manera atractiva; que organizadores puedan crear y gestionar los detalles de las rutas en un sólo sitio muy fácilmente: por último, que los distintos establecimientos puedan consultar las rutas más cercanas e inscribirse en ellas.



2 Motivación y objetivos

Vamos a tratar de describir las razones por las que se eligió este proyecto frente a otros. Haremos una breve introducción a las Rutas de la Tapa y los motivos por los que se eligió.

2.1 Introducción

La gastronomía es uno de los grandes atractivos para los turistas que vienen a España así como para los propios españoles.

Las tapas son una exquisita, variada y selecta muestra de la gastronomía popular que permite compartir platos y disfrutar de una generosa oferta de productos. Todo el sabor de España en una sola comida y a precios generalmente muy módicos.

En todos los pueblos y ciudades de España hay vares y tabernas, incluso barrios enteros, especializados en tapas.

Las presentaciones de las mismas son muy diversas, así como sus ingredientes. Pinchos, banderillas, montaditos, cazuelitas son denominaciones diferentes que se aplican a las distintas formas de servir las tapas, desde la mas sencilla, hasta las más sofisticadas dignas de cocina de vanguardia.

Por todo ello no es de extrañar que se celebren las denominadas "Rutas de la tapa", en la que distintos establecimientos se organizan para celebrar un concurso con el motivo de fomentar el turismo y el consumo.

Los establecimientos presentan una tapa al concurso y son los propios consumidores los que se encargan de valorarlas.

Las rutas suelen estar organizadas por barrios en las grandes ciudades o en municipios enteros. Así los consumidores, pueden desplazarse andando entre un establecimiento y otro.

2.2 Motivación

Así pues, junto con el deseo de los establecimientos promocionarse, haciendo uso de las Rutas de la Tapa, surge la necesidad de organizarse.

Existen multitud de soluciones, normalmente gestionadas por los ayuntamientos de las localidades o empresas privadas, por lo que la idea de un sistema centralizado y totalmente independiente no resulta una idea descabellada.

Al usuario que desea participar en una de estas rutas se le ofrece la posibilidad de buscar por localidad y encontrar una sin tener que visitar una por una las páginas de los ayuntamientos o utilizar un buscador que puede no ofrecerle lo que necesite. Con una aplicación web dedicada a estos eventos, es seguro, que lo que encuentre, será una ruta en la que podrá participar.



A los establecimientos y organizadores se les proporciona un entorno en el que poder auto gestionarse sin depender de entidades externas que puedan manipular los resultados o promover sus productos. Además, al ser un sistema automatizado, los resultados son obtenidos de las votaciones de los clientes por lo que se asegura la imparcialidad.

Como ventaja adicional, es una manera de promocionarse gratuitamente, y se pueden alcanzar unos grados de visibilidad mucho mas altos que con páginas independientes o por medios locales.



3 Estado del arte

En este capítulo hablaremos acerca de cómo se encuentra el ecosistema de aplicaciones web en el contexto en que se mueve nuestro proyecto.

Empezaremos haciendo un repaso de los lenguajes de programación del lado del servidor y tras esto, veremos que aplicaciones similares se encuentran disponibles hoy en día.

3.1 Lenguajes de programación para servidores

Existe una multitud de lenguajes concebidos o no para Internet. Cada uno de ellos explota más a fondo ciertas características que lo hacen más o menos útiles para desarrollar distintas aplicaciones web.

La elección de un lenguaje u otro dependerá de la aplicación que se desee desarrollar pero sobretodo, de la experiencia previa del programador. Hoy en día, es difícil encontrar lenguajes en los que hayan funciones que no se puedan desarrollar en otros.

Entre los lenguajes para servidores más utilizados para el desarrollo de páginas dinámicas se encuentran Ruby, PHP y JavaScript entre otros. A continuación haremos una breve introducción a algunas de ellas.

Ruby: Es uno de los lenguajes de programación que mayor crecimiento ha experimentado en los últimos años. Creado por Yukihiro “Matz” Matsumoto, Ruby es un lenguaje de programación de código abierto orientado a objetos y que permite la programación tanto en web como en escritorio. Para la programación de webs o webapps, se utiliza su conocido framework “Rails”.

ASP.NET: Es un lenguaje creado por Microsoft y que requiere un servidor Windows para su utilización. Se utiliza principalmente para desarrollar sitios web dinámicos, aplicaciones web o servicios web XML.

PHP: Se trata de un lenguaje de servidor de código abierto muy extendido, flexible y potente, muy adecuado para el desarrollo web. Ha sido utilizado para la creación de millones de webs en el mundo, incluidas algunas tan conocidas como Wikipedia o Facebook. Esta popularidad hace que existan varios frameworks que se pueden utilizar a la hora de abordar un proyecto web o una WebApp como Codeigniter, Symfony, CakePHP, Laravel o Zend.

Python: También de código abierto, orientado a objetos que permite la creación de sitios web dinámicos y complejos pero de actualización fácil y con un corto tiempo de carga. Actualmente, se utiliza en webs tan populares como Pinterest, Instagram o Dropbox. A pesar de que hay múltiples frameworks, el más utilizado y conocido es Django, su framework para web.

Javascript: Es uno de los lenguajes, si no “el lenguaje”, más utilizado y popular para ejecutar del lado del cliente. Gracias a su multitud de frameworks, se pueden realizar desarrollos realmente impresionantes. Los más conocidos son jQuery, Backbone.js,

Ember.js, Knockout.js o Angular.js. Para ejecutar del lado del servidor, se utiliza su framework Node.js.

Estos son sólo un pequeño grupo de lenguajes posibles para el desarrollo en la parte del servidor. Los lenguajes evolucionan y surgen otros nuevos por lo que al final, la elección de uno u otro se basará mas en la facilidad de aprendizaje, la velocidad de desarrollo y por supuesto, la comodidad del programador serán las que determine cual usar.

3.2 Aplicaciones relacionadas

Como ya se ha comentado anteriormente, ya existen distintas soluciones para la gestión de Rutas de la Tapa. El inconveniente de estas plataformas es que ofrecen una solución a una zona en particular.

A continuación se muestran algunos ejemplos.

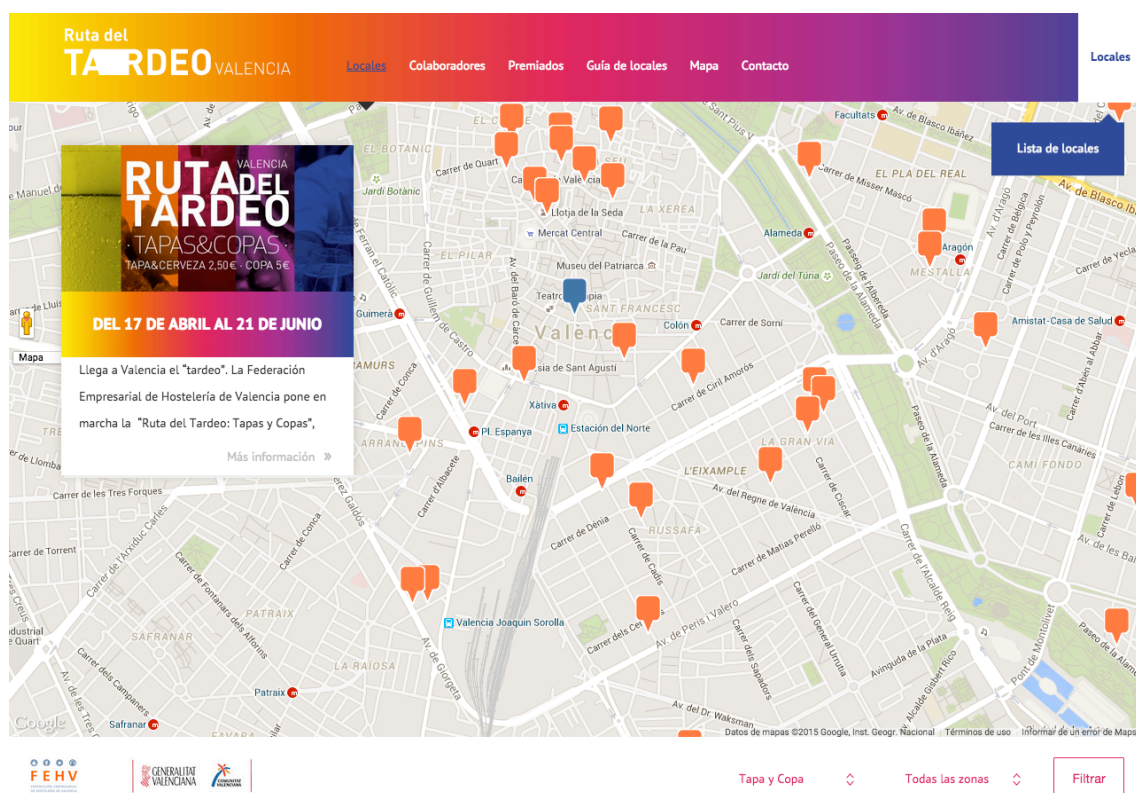


Figura 1. Portal de rutadeltardeovalencia.es

En esta captura se puede muestran todos los establecimientos participantes geolocalizados en el mapa de valencia. Al intentar obtener más información de alguna tapa en concreto se nos redirige a la página en cuestión.



Diseño e implementación del servidor web "Ruta de la Tapa"

The screenshot shows the website interface for 'Ruta del TARDEO VALENCIA'. At the top, there is a navigation menu with links: Locales, Colaboradores, Premiados, Guía de locales, Mapa, and Contacto. Below the menu is a map of Valencia with a red location marker. A banner below the map reads '37. LLOC DE TAPES'. The main content area features a large, high-quality photograph of a burrito on a wooden board, garnished with fresh herbs and a side of sauce. To the right of the photo is a blue box containing the following information:

TAPA
Burrito Lloc

Ingredientes
Burrito de Habitas y Longanizas con all i oli de naranja y soja.

Dirección
C. Marqués de Zenete, 7
46007 - Valencia

Teléfono
963 827 162

Teléfono
www.llocdetapes.com

Horario
De 12:00 a 13:30 y de 19:30 a 21:00 h.
Cerrado: domingo noche.

At the bottom of the page, there are logos for FEHV, Generalitat Valenciana, and the website's design and development by Conca&Marzal.

Figura 2. Detalles de una tapa en rutadeltardeovalencia.es

En esta pantalla nos encontramos con la información de la tapa así como los datos del establecimiento. Quizás la información que se muestra es demasiado pobre y se hace demasiado énfasis en la foto, la cual ocupa el 40% de la página.

III RUTA DE LA TAPA DE VALENCIA
Del 4 al 7 de Junio de 2015



[LISTADO DE TAPAS](#) [MAPA DE LA RUTA](#) [DESCARGAR RUTERO](#)



HOMENAJE TABERNA GOURMET
PEQUEÑO HOMENAJE

Ganador de:
III RUTA DE LA TAPA DE VALENCIA

[Ver la tapa](#)



Listado de tapas

Ordenar por: **Valoración** Establecimiento Tapa

[Ver mapa de la ruta](#)



CERVECERÍA RESTAURANTE CÁNOVAS
MEDITERRANEO



RTE. AGUA NA BOCA
AGUA NA BOCA



TASCA MI NIÑA
CROQUETA DE CREMA DE JALAPEÑOS



MARABÚ
SUPREMA DE BERENJENA



SAN REMO
MONTADITO DE ANCHOA DEL CANTÁBRICO



RTE. LOS VIÑEDOS
AJO-ARRIERO UTIELANO



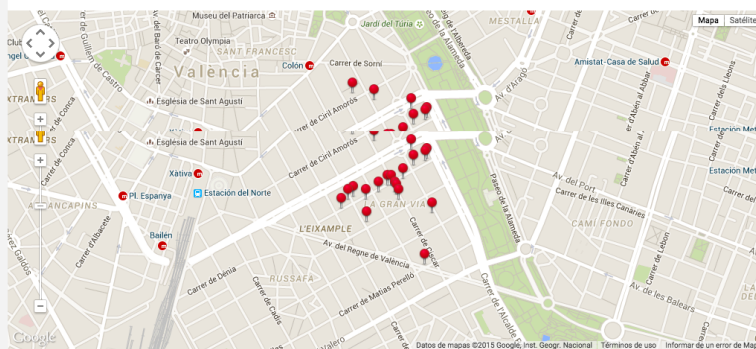
TASCA LA FLAMENCA
MARRAQUÉS



PANTALÁN 5
PANTALÁN

Mapa de la ruta

[Ver el listado de tapas](#)



Establecimientos participantes:

CERVECERÍA RESTAURANTE CÁNOVAS
BODEGA PASCUAL
HOMENAJE TABERNA GOURMET
RTE. AGUA NA BOCA
TASCA MI NIÑA
MARABÚ
RTE. LUMAYA

TABERNA EL SITIO
RTE. CISCAR 28
SAN REMO
CERVECERÍA DOCE
RTE. LA TOFONA
RTE. ALFABEGA
RTE. LOS VIÑEDOS

TASCA EL PUNTO Y LA I
TASCA LA FLAMENCA
CAFÉ BALLI
RTE. MI
PANTALÁN 5
EL ARENAL DE CÁNOVAS

III RUTA DE LA TAPA DE VALENCIA
Del 4 al 7 de Junio de 2015
Valencia



Figura 3. Portada de estamosdetapas.com



Diseño e implementación del servidor web "Ruta de la Tapa"

En la web www.estamosdetapas.com nos encontramos un diseño mas cuidado, pero como ya se ha comentado en casos anteriores, vemos que esta patrocinado por la marca de cerveza Mahou por lo que nos encontramos su logo por toda la web.

En estos casos, corremos el peligro de que los patrocinadores influyan demasiado en el desarrollo de la ruta, bien puedan denegar la participación de participantes por no ofrecer sus productos, o favoreciendo a los que si lo hagan.



4 Especificación de requisitos

En este capítulo abordaremos de una manera más detallada la visión que se tiene de este proyecto. Primero haremos una descripción general en la que se obtiene un primer contacto acerca del producto, funcionalidad y los distintos tipos de usuario que formarán parte de la aplicación.

Tras esto, haremos una recopilación de definiciones y acrónimos que nos serán muy útiles a lo largo del proyecto para comprender algunos de los términos utilizados.

A continuación, analizaremos los requisitos propios de la aplicación para detectar cuales serán las necesidades básicas que deberemos ofrecer a nuestros usuarios.

4.1 Introducción

4.1.1 Propósito y alcance

El propósito de este proyecto es el de crear una plataforma en el que se ofrezcan tapas y rutas creadas por los mismos usuarios.

En la página web ofrecerá (previo registro) la posibilidad de crear rutas nuevas geolocalizadas así como la posibilidad para otros usuarios de inscribirse a dichas rutas creadas. En todos los casos, la creación de rutas o tapas vendrá acompañada de una imagen que proporcionarán también los mismos usuarios.

A parte de los servicios a los usuarios, se ofrecerá una API para el consumo de la aplicación móvil desarrollada por la alumna Marta Gaona Jiménez, en la que se le ofrecerá la información de las rutas y tapas, así como métodos para registrar los votos de los usuarios.

El alcance, por lo tanto, debería cubrir los siguientes aspectos:

- Creación de rutas personalizadas por los usuarios.
- Creación de tapas con imágenes.
- Visualización de todos los detalles de una ruta.
- API rest para obtención de resultados y votaciones en JSON.

4.1.2 Objetivos

Así pues, una vez visto el propósito y el alcance, procedemos a describir el objetivo del proyecto.

El objetivo principal del proyecto es el de proporcionar a los usuarios un sistema mediante el cual poder auto-gestionar un concurso de tapas local.

Como hemos visto anteriormente, están muy de moda los concursos gastronómicos pero muchos pecan de estar patrocinados, o que la gestión recaiga sobre alguna entidad, como un ayuntamiento. Por lo que la meta principal del proyecto es permitir a



los usuarios tomar el control de sus rutas, no tener que depender de ningún agente externo, y ofrecerles todo este control de una manera sencilla y atractiva.

Así pues, como objetivo secundario, tendríamos la capacidad de poder agrupar todas las rutas de España en un mismo portal. Esto reduciría considerablemente los tiempos de búsqueda de los usuarios que deseen participar en una ruta como consumidores.

4.1.3 Definiciones y acrónimos

En este apartado se muestran las definiciones y acrónimos de ciertos términos quizás más técnicos que ayudarán a entender los requisitos en su totalidad

Definiciones

- **Geolocalización:** Se trata de una técnica de posicionamiento espacial de una entidad en una localización geográfica única y bien definida en un sistema de coordenadas.

Acrónimos

- **API:** Application Programming Interface. En español: Interfaz de programación de aplicaciones. Se trata de un conjunto de subrutinas, funciones y procedimientos que ofrece una biblioteca para ser utilizado por otro software como una capa de abstracción.
- **BD:** Base de Datos. Es la capa software encargada de la persistencia de los datos.
- **JSON:** Javascript Object Notation. Es un formato ligero para el intercambio de datos en forma de mensaje.

4.2 Descripción general

En este apartado hablaremos del producto de una manera abstracta sin entrar en detalles técnicos. Esto nos permitirá tener una perspectiva de lo que se va a realizar y las funciones que se implementarán.

4.2.1 Perspectiva del producto

Se pretende desarrollar una aplicación web donde se ofrezca al usuario la posibilidad de buscar rutas existentes, todos sus datos, como localización, noticias y los establecimientos inscritos.

También se le ofrecerá la posibilidad de crear sus propias rutas o tapas, permitiéndole incluir fotos suyas, y en el caso de las rutas, personalizar ciertos detalles con colores a su elección.

Dado que el sistema no permite valorar las tapas, ya que este aspecto se ha delegado en la aplicación móvil, se ofrecerá una API para su consumo.

4.2.2 Funcionalidad del producto

Las funciones que forman la aplicación son:

Gestión de usuarios: Se debe permitir a los usuarios registrarse como organizadores o participantes. Dependiendo del tipo de cuenta, se le ofrecerán unos servicios u otros.

Gestión de rutas: Un usuario registrado como organizador, puede crear nuevas rutas. Se verificará que no haya creado ya otra ruta en el mismo rango de fechas.

Gestión de tapas: Un usuario registrado como establecimiento, puede crear nueva tapa con foto e inscribirse en alguna de las rutas ya existentes. Se verificará que no esté inscrito en otra ruta en las mismas fechas.

Gestión de las solicitudes: Un usuario registrado como organizador, puede aceptar o rechazar las solicitudes que los establecimientos le envíen.

Gestión de las votaciones: El sistema se encargará de calcular periódicamente el estado de las rutas, y para las finalizadas, hará un recuento de los votos y se publicarán los resultados.

Gestión de las noticias: Un usuario registrado podrá crear noticias nuevas, las cuales se etiquetarán con la hora de creación y se mostrarán en la portada de la ruta en cuestión.

4.2.3 Características de usuario

En la aplicación se distinguen 3 tipos de usuarios: Organizadores, Establecimientos y Consumidores.

Los Organizadores son los que generan las rutas y aceptar o rechazan las solicitudes de participación; los Establecimientos los que generan las tapas y se adhieren a las rutas y por último, los Consumidores son los clientes de los Establecimientos que buscan información.

Dependiendo del tipo de usuario que sean, se les deberá ofrecer y limitar los servicios que se les ofrecen. Ya que, por ejemplo, un Organizador no puede apuntarse con un bar a una ruta creada por él mismo, así como un Establecimiento no puede gestionar una ruta. Los motivos son claros, que la competición sea lo más justa posible.

Así pues, atendiendo a esta clasificación, los servicios disponibles a cada uno de los usuarios serían:

Gestión de usuarios

Proceso	Organizador	Participante	Consumidor	Sistema
Crear cuenta	X	X	X	
Iniciar sesión	X	X		



Gestión de rutas

Proceso	Organizador	Participante	Consumidor	Sistema
Crear ruta	X			
Ver rutas	X	X	X	

Gestión de tapas

Proceso	Organizador	Participante	Consumidor	Sistema
Crear tapa		X		
Ver tapas	X	X	X	
Inscribirse con una tapa		X		

Gestión de las solicitudes

Proceso	Organizador	Participante	Consumidor	Sistema
Ver solicitudes		X		
Aceptar / Rechazar solicitud	X	X	X	
Inscribirse con una tapa		X		

Gestión de las votaciones

Proceso	Organizador	Participante	Consumidor	Sistema
Votar tapa			X	
Calcular resultados				X

Gestión de las noticias

Proceso	Organizador	Participante	Consumidor	Sistema
Ver noticias	X	X	X	
Insertar noticia	X			

4.2.4 Restricciones del sistema

Obviamente, al tratarse de una aplicación web, el usuario necesitará disponer de un dispositivo capaz de usar internet (Smartphone, Tablet u ordenador de sobremesa).

La aplicación ha sido diseñada para ser completamente responsive por lo que no hay limitaciones a la hora de elegir dispositivo.

4.2.5 Dependencias del sistema

Al tratarse de una aplicación web, se necesita un servidor que pueda alojar los archivos necesarios.

Por otra parte, algunas funciones depende de la API de Google Maps por lo que se necesitará que acepte peticiones. Al tratarse de un servicio de Google, es bastante improbable que no se encuentre disponible.

4.3 Requisitos específicos

En este apartado, definiremos que aspectos de la aplicación se deben cumplir para que se considere la aplicación como plenamente funcional.

4.3.1 Requisitos de interfaz externos

Cuando hablamos de requisitos de interfaz externos, hacemos referencia a los aspectos visuales que deberá cumplir la aplicación.

- **IR-Ro1.** Idioma: El idioma principal debe ser el castellano.
- **IR-Ro2.** Debe ser compatible con la mayoría de navegadores web (Chrome, Firefox, Opera, Safari)
- **IR-Ro3.** Debe adaptarse la interfaz al tamaño de pantalla del navegador.

4.3.2 Requisitos funcionales

Dependiendo del tipo de usuario, se le ofrecerán unos determinados servicios. A continuación se detallan las funciones que tendrán disponibles divididos en los tipos de usuario: consumidores (CR), organizadores (OR) y participantes (PR). Además se describen los requisitos que deberá tener la API que se ofrece para el consumo de la aplicación móvil (AR).



4.3.2.1 *Servicios a los consumidores*

- **CR-Ro1. Búsqueda de rutas.** La materia prima de la aplicación web son las rutas. Por lo que se debe ofrecer un sistema de búsqueda para poder filtrarlas por Localidad o Nombre.
- **CR-Ro2. Ver todos los detalles de una ruta en concreto:** Una ruta se compone de algo más que una localización. Una vez localizada una ruta en concreto, se deben mostrar los establecimientos participantes y sus localizaciones, así como los premios que se repartirán una vez acabada la misma.
- **CR-Ro3. Registrarse o iniciar sesión:** Los usuarios no identificados tendrán siempre disponible la opción de registrarse como organizadores o participantes. Y en el caso de que ya tuvieran cuenta, poder iniciar sesión.

4.3.2.2 *Servicios a los organizadores*

- **OR-Ro1. Poder crear nuevas rutas geo localizadas.** El servicio está mantenido por los mismos usuarios con roles de Organizador. Éstos serán los que se encarguen de dar contenido a la página.
Deben poder crear nuevas rutas identificándose con un nombre, una localización y definir unos premios para los participantes.
- **OR-Ro2. Aceptar y rechazar solicitudes de participación.** Dado que las rutas son propiedad de los Organizadores, la aceptación o Rechazo de nuevos participantes corre a cargo de ellos. Se les debe proporcionar mecanismos para poder decidir acerca de la participación de nuevos establecimientos a la misma.
- **OR-Ro3. Insertar noticias en rutas.** Se debe proporcionar un sistema para poder informar a los usuarios interesados de las distintas novedades que puedan haber.

4.3.2.3 *Servicios a los participantes*

- **PR-Ro1. Creación de nuevas tapas con foto, nombre y descripción.** Al igual que con las rutas, las tapas son generadas por los mismos usuarios, en este caso, con roles de Participantes. Se les debe permitir crear nuevas tapas con imagen, nombre y descripción.
- **PR-Ro2. Inscribirse en rutas existentes con una tapa.** Los participantes son los que deciden unirse a una ruta ya creada por lo que deben ser capaces de solicitar permiso para participar en una ruta.
- **PR-Ro3. Permitir a los consumidores votar.** Los consumidores que se presentan físicamente al establecimiento son los encargados de juzgar las tapas, por lo que se deberán tener métodos para recoger sus valoraciones.

4.3.2.4 Servicios de API

- **AR-Ro1. Ofrecer rutas disponibles geolocalizadas.** Se debe devolver un JSON con las rutas disponibles respecto de una coordenada dada.
- **AR-Ro2. Ofrecer datos de una ruta en concreto.** Se devolverán los datos de una ruta (nombre, descripción, participantes, fechas....).
- **AR-Ro3. Ofrecer datos del estado de la votación.** Se informará de los resultados de una ruta. Los ganadores y el premio recibido.
- **AR-Ro4. Guardar votaciones.** Se almacenará la puntuación que un cliente da a una tapa dentro de concurso.

4.3.3 Requisitos de rendimiento

No existen requisitos de rendimiento como tal. Simplemente, el sistema debe responder a las peticiones del usuario en un tiempo aceptable (hasta 500ms).

4.3.4 Requisitos lógicos de base de datos

La base de datos deberá ser una de tipo relacional. El motivo es que la mayoría de los datos guardan una relación con otro y eligiendo una BD de este tipo, nos aseguramos que los datos que hay almacenados sean coherentes.

Por ejemplo, si eliminamos una ruta, todas las participaciones de la misma se deberán eliminar también, ya que no deben quedar participaciones que no estén asociadas a una ruta inexistente.



5 Análisis

En este apartado, se analizará el sistema para hacernos una idea de cómo será el producto final. Empezaremos con un diagrama de clases para ver las distintas clases junto con sus métodos y atributos.

A partir de estas clases, definiremos los escenarios y de ahí obtendremos los casos de uso.

5.1 Diagrama de clases / entidades

El diagrama de clases general sería el mostrado en la figura siguiente.

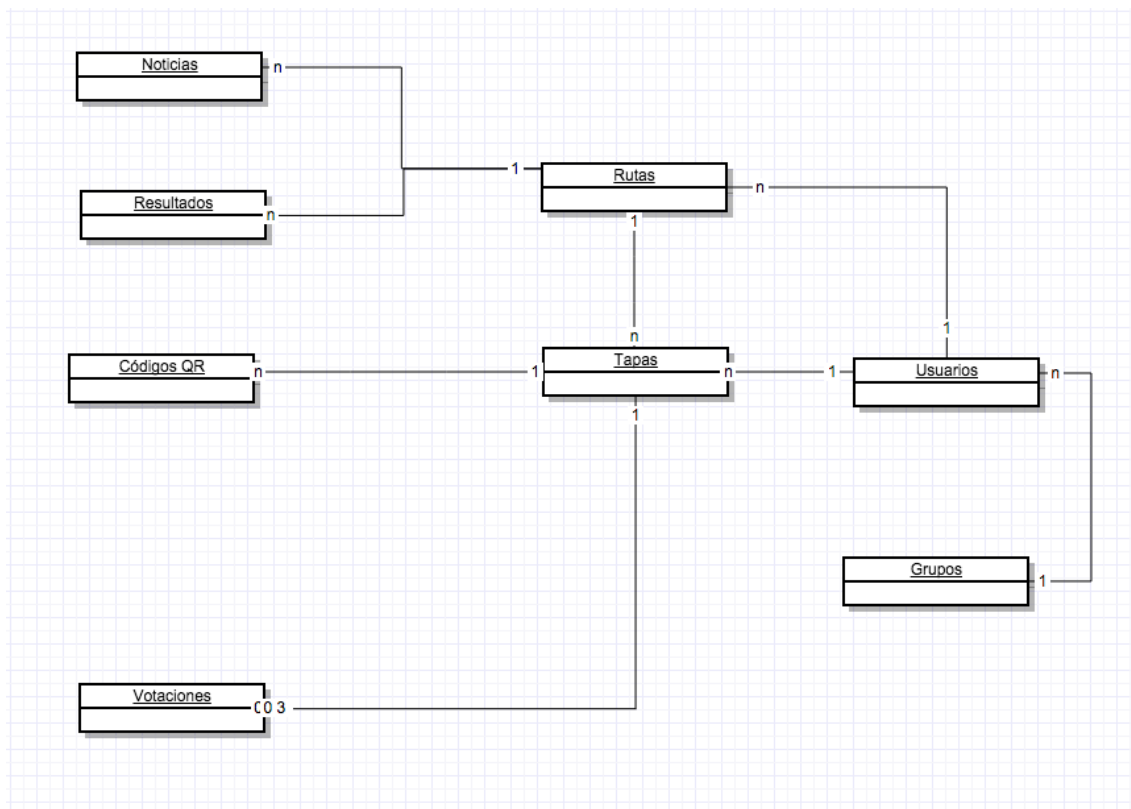


Figura 4. Diagrama de clases

Partiendo de los usuarios, vemos que todos tienen un único grupo asignado. Puede (y debe) haber varios usuarios en el mismo grupo. Los grupos serían organizadores y participantes. Los consumidores o simples visitantes no tienen cuenta de usuario asociada.

Estos usuarios, crearán tapas o rutas (dependiendo de que tipo de usuario sean), y quedarán enlazados a este único autor. Así mismo, las tapas pertenecerán a una sólo ruta, mientras que las rutas llevarán asociadas varias tapas.

Como clases adicionales de las rutas, tendríamos Noticias, y Resultados. Estas clases pertenecerán a una sola ruta. Obviamente, una ruta, tendrá varias noticias, y de 0 (no hay participantes) a 3 votaciones.

Así mismo, para una tapa, habrá varios códigos QR asociados o varias votaciones. Pero sólo pertenecerán a una tapa.

5.2 Casos de uso

A continuación vamos a enumerar las distintas actividades que los distintos usuarios podrán hacer en la plataforma.

Se especificará para cada uno de ellos, que tipo de usuario puede realizar dicha acción, las condiciones previas necesarias para realizarlas así como el estado en el que quedará el sistema tras la acción.

Agruparemos los casos de uso en torno a un elemento en común, y a continuación mostraremos el diagrama de caso de uso.

Caso de uso 1: Buscar rutas	
Actores	Todos los usuarios.
Objetivo	Obtener un listado de las rutas disponibles.
Resumen	El sistema ofrecerá a los usuarios un sistema de búsqueda en la que filtrar las rutas disponibles filtradas por nombre o ciudad.
Precondición	El usuario debe estar en la página web.
Postcondición	El sistema devolverá al usuario una lista de rutas disponibles con su enlace para poder dirigirse a ellas.
Requisitos asociados	CR-R01. Búsqueda de rutas, CR-R02. Ver detalles de una ruta

Caso de uso 2: Registrarse como organizador	
Actores	Organizadores
Objetivo	Obtener una cuenta de usuario con permisos de creación de rutas.
Resumen	El usuario introducirá su correo electrónico y una contraseña definida por él para poder crear rutas.
Precondición	No debe existir una cuenta con el mismo correo electrónico y no debe tener una sesión iniciada.



Postcondición	Se guardan los credenciales del usuario en Base de Datos y se le redirige al panel de Administración de los Organizadores.
Requisitos asociados	CR-R03. Registrarse o iniciar sesión

Caso de uso 3: Registrarse como participante	
Actores	Participantes
Objetivo	Obtener una cuenta de usuario con permisos de inscripción en rutas ya existentes y creación de tapas.
Resumen	El usuario introducirá su correo electrónico y una contraseña definida por él para poder inscribirse en rutas y crear tapas.
Precondición	No debe existir una cuenta con el mismo correo electrónico y no debe tener una sesión iniciada.
Postcondición	Se guardan los credenciales del usuario en Base de Datos y se le redirige a la pantalla de creación de tapas.
Requisitos asociados	CR-R03. Registrarse o iniciar sesión

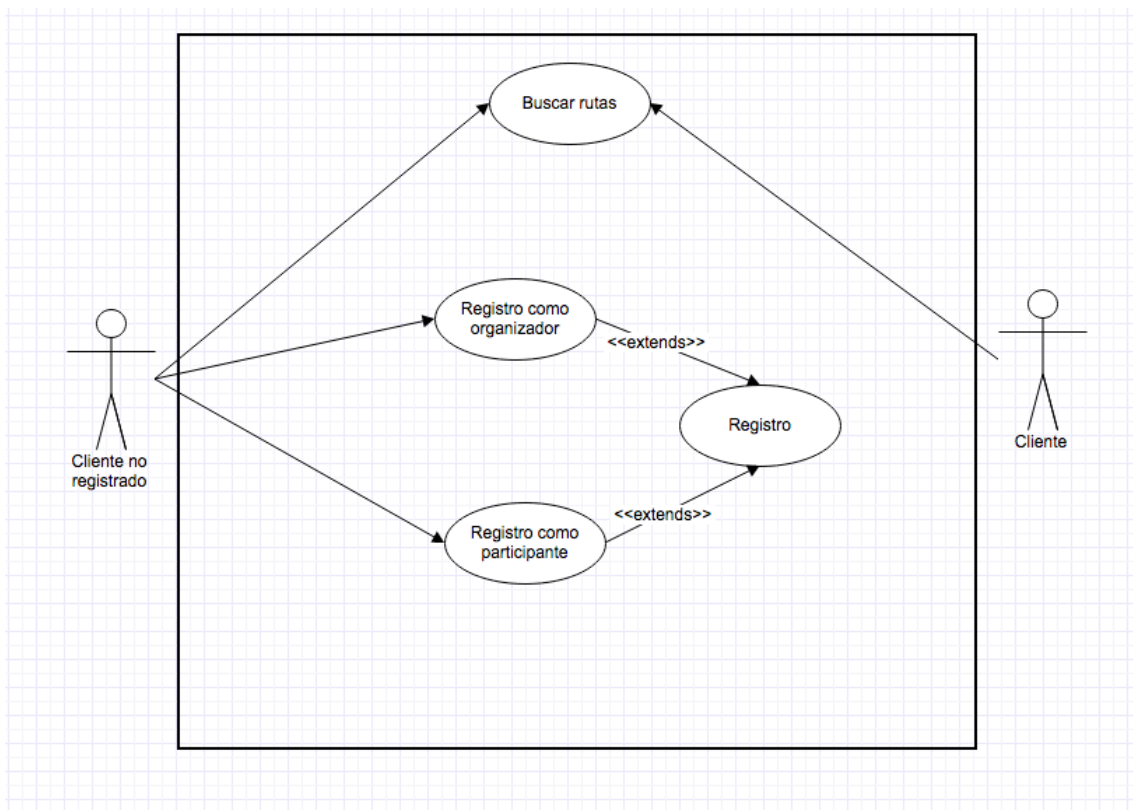


Figura 5. Casos de uso para usuarios

Caso de uso 4: Crear una ruta	
Actores	Organizadores.
Objetivo	Crear una nueva ruta con los datos que decida el usuario y almacenarla en base de datos para que esté disponible para su consulta.
Resumen	El usuario introducirá todos los datos de la ruta (nombre, lugar, fechas, premios...) y una vez validados se almacenarán en Base de Datos.
Precondición	No haber creado una ruta en el mismo intervalo de fechas o con fechas coincidentes.
Postcondición	La ruta queda registrada en Base de Datos y queda a la espera de aceptar solicitudes por parte de los Establecimientos.
Requisitos asociados	OR-R01. Poder crear nuevas rutas geo localizadas

Caso de uso 5: Responder a una solicitud de participación	
Actores	Organizadores.
Objetivo	Tras recibir una solicitud de participación. El Organizador debe decidir si acepta o no la solicitud. En caso de rechazarla, deberá especificar un motivo para que el Establecimiento pueda corregir su solicitud.
Resumen	El Organizador navega hasta el panel de Solicitudes y selecciona si acepta o no la solicitud del Establecimiento. En caso de rechazarla, describe los motivos del rechazo.
Precondición	Un Establecimiento debe haber rellenado previamente la solicitud para dicha ruta.
Postcondición	Si acepta, el Establecimiento aparece como participante en la ruta y se le ofrece la posibilidad de descargar códigos para repartir a sus clientes.
Requisitos asociados	OR-R02. Aceptar y rechazar solicitudes de participación



Caso de uso 6: Insertar noticia en una ruta	
Actores	Organizadores
Objetivo	Añadir alguna noticia en el tablón de la ruta para consulta de Establecimientos y Usuarios.
Resumen	El Organizador, desde la ficha detallada de la ruta, redacta el texto de la noticia y la envía para que se almacene en Base de Datos.
Precondición	Debe tener alguna ruta previamente creada.
Postcondición	La noticia se almacena y queda reflejada en la portada de la ruta etiquetada por la fecha en la que se redactó.
Requisitos asociados	OR-RO3. Insertar noticias en rutas

Caso de uso 7: Crear una nueva tapa	
Actores	Establecimientos.
Objetivo	Crear una nueva tapa para poder participar en alguna de las rutas existentes.
Resumen	El Establecimiento, rellena los datos necesarios para la creación de la tapa (imagen, nombre y descripción) y, tras validarse en el servidor, se almacena en Base de Datos.
Precondición	Estar registrado y logueado en la plataforma como participante.
Postcondición	Los datos de la tapa quedan almacenados para su uso en alguna inscripción posterior.
Requisitos asociados	PR-RO1. Creación de nuevas tapas con foto, nombre y descripción.

Caso de uso 8: Inscribirse en una ruta con una tapa existente	
Actores	Establecimientos.
Objetivo	Inscribirse en una ruta existente para poder optar a los premios de la misma.
Resumen	El usuario accede al área de registro de una ruta en concreto y selecciona una de sus tapas previamente creadas para poder inscribirse.
Precondición	Debe haber creado alguna tapa anteriormente.
Postcondición	La solicitud queda a la espera de validación por el Organizador de la ruta y se le envía una solicitud al mismo.
Requisitos asociados	PR-R02. Inscribirse en rutas existentes con una tapa

Caso de uso 9: Descargar códigos QR	
Actores	Establecimientos.
Objetivo	Descargar un PDF con códigos para ofrecérselos a los Consumidores.
Resumen	El establecimiento accede a su área de administración y descarga un PDF con códigos QR únicos para dicha tapa y ruta.
Precondición	Haberse inscrito en alguna ruta y que el Organizador acepte la solicitud.
Postcondición	El Establecimiento obtendrá un fichero PDF que deberá imprimir para ofrecer a los Consumidores.
Requisitos asociados	PR-R03. Permitir a los consumidores votar.

Diseño e implementación del servidor web "Ruta de la Tapa"

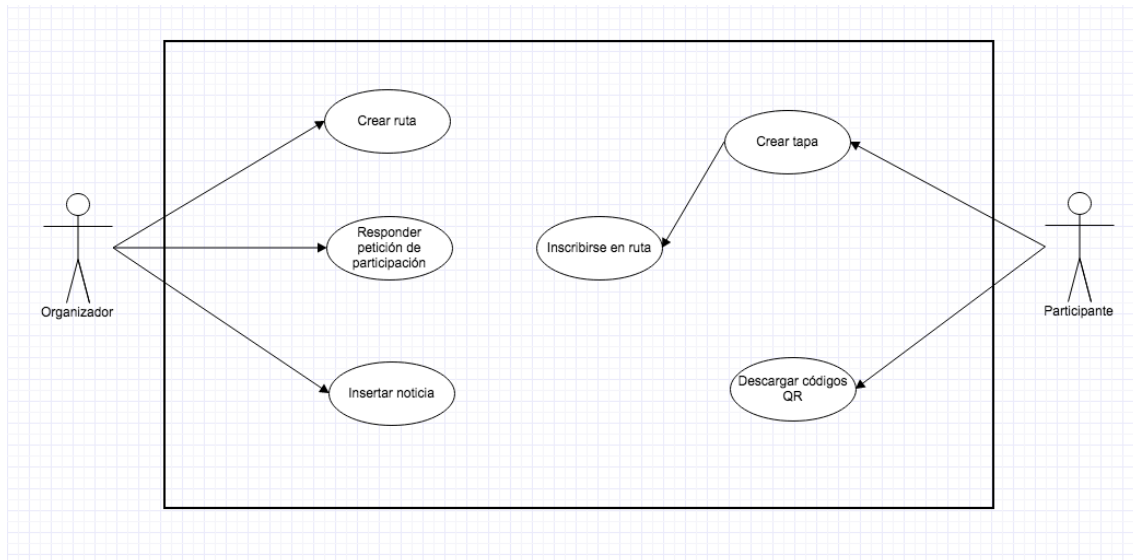


Figura 6. Casos de uso para organizadores y participantes

Caso de uso 10: Puntuar una tapa	
Actores	Consumidores.
Objetivo	Valorar una tapa de un Establecimiento.
Resumen	A los consumidores, una vez han consumido en algún establecimiento participante, se les entrega un código QR para escanear y proceder a puntuar la tapa.
Precondición	Haber consumido y tener la aplicación móvil.
Postcondición	Si el código QR es correcto, se almacena su valoración para el recuento una vez acabada la ruta.
Requisitos asociados	PR-R03. Permitir a los consumidores votar.

6 Diseño

En este apartado veremos detalles más cercanos al producto final que se desarrollará. Definiremos un esquema general para el proyecto en su totalidad definiendo cada uno de las capas que conformarán la aplicación.

Tras una introducción de todo el ecosistema que será la aplicación web, analizaremos las tres capas de la misma: presentación, persistencia y lógica.

6.1 Introducción

Todo nuestro proyecto, estará centralizado en nuestro servidor. La única dependencia externa que tendremos, como ya se ha comentado anteriormente, es con de Google Maps.

La aplicación tiene un fuerte componente de geolocalización por lo que debemos asegurarnos que los usuarios (que a fin de cuentas, son los que van a introducir los datos), sepan exactamente dónde están creando las rutas, tapas...

El uso de esta API, nos permite mostrar de forma visual, la posición en la que podrán ser localizados. Mediante una consulta en forma de texto (la dirección física), Google Maps nos devuelve tanto las coordenadas, como la dirección correctamente formateada, así como una imagen con un marcador en la que el usuario enseguida puede reconocer si los datos son correctos o no, y proceder a corregirlos.

Por conocimientos anteriores, se ha optado por usar el ecosistema AMP (Apache, MySQL y PHP).

El uso del lenguaje PHP nos permite un desarrollo ágil y estable, el cual acompañado de algún framework, nos permite centrarnos en el diseño de nuestra aplicación sin preocuparnos de detalles como la gestión de usuarios o el saneamiento de los formularios.

La forma de organizar todo el conjunto, sigue la metodología MVC, la cual se desarrollará en PHP. Así pues, separaremos las 3 capas que conformarán el sistema entero.

Nuestra aplicación tiene unas necesidades en las cuales los elementos están relacionados entre sí, por lo que se nos hace necesario una BD de tipo relacional. Así pues, se ha optado por usar MySQL como base de datos.

A continuación, veremos cada una de las capas, sus responsabilidades y resultados.

6.2 Capa de presentación

Para la capa de presentación se programaron unos bocetos gracias a la herramienta Bootstrap. Esta herramienta nos ofrece bloques de código ya preparados por lo que se puede hacer un primer desarrollo muy rápidamente.





Figura 7. Diseño original de la portada

En la figura 7, se ve un primer diseño de la portada de la web. En rediseños posteriores se modificaron los tamaños ya que nada más abrir la web sólo se veía una imagen y resultaba poco atractivo para el usuario.



Figura 8. Diseño inicial de una de las rutas

Este fue un primer diseño de la portada de una ruta donde se mostraba la información básica. Participantes, mapa, lugar y noticias.

En un principio se le iba a dar a la ficha de los participantes una apariencia de foto tipo Polaroid, pero después se descartó por no resultar tan atractivo como se pensaba al principio, o no haber sabido integrarlo correctamente.



Figura 9. Diseño inicial del panel de control del organizador

Aquí vemos una primera versión del panel de control del organizador.

En esta primera versión se eligió como menú de navegación una barra lateral tipo dashboard. Más tarde se descartó por ocupar demasiado espacio y se agruparon las categorías en la barra superior.

6.3 Capa de persistencia

Como ya se ha comentado anteriormente, se hará uso de MySQL para la capa persistencia.

En ella, deberemos almacenar toda la información relativa a rutas, tapas, usuarios, inscripciones, noticias, valoraciones y resultados.

Para los usuarios, se ha preferido optar por la librería Ion Auth para que gestione los usuarios. La decisión se ha tomado en base a que el propósito del proyecto no era la de crear un sistema de gestión de usuarios por lo que se ha preferido delegar dicha responsabilidad.

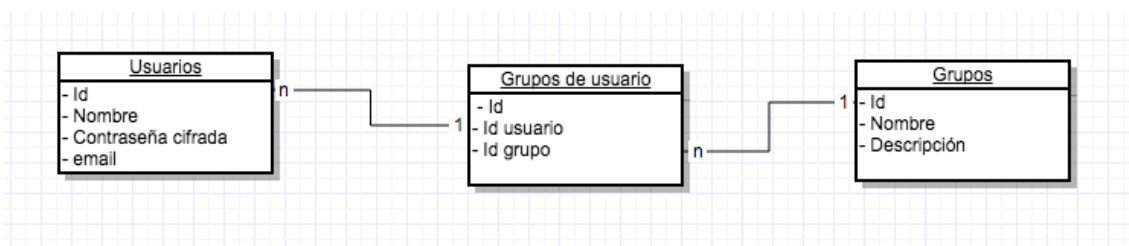


Figura 10. Esquema de BD para usuarios

Para la identificación de los usuarios harán falta los siguientes datos:

- **ID:** Identificador único.
- **Nombre:** Nombre del usuario.
- **Contraseña:** Contraseña cifrada del usuario.
- **Email:** Email del usuario

Los grupos estarán formados por los siguientes atributos:

- **ID:** Identificador del grupo
- **Nombre:** Nombre del grupo (organizador o participante)
- **Descripción:** Una breve descripción del grupo

Por último, se usará una tabla intermedia para relacionar los grupos con los usuarios.

En el diagrama se han obviado los campos no relevantes. Como se puede apreciar, se guardarán todos los datos del usuario y se le asignará un rol (grupo) el cual decidirá que tipo de usuario es y que funciones tendrá disponibles.

Para esta aplicación web, se deberán añadir los siguientes datos opcionales a la tabla de usuarios:

- Dirección del establecimiento.
- Coordenadas del establecimiento.
- Numero de peticiones del organizador.

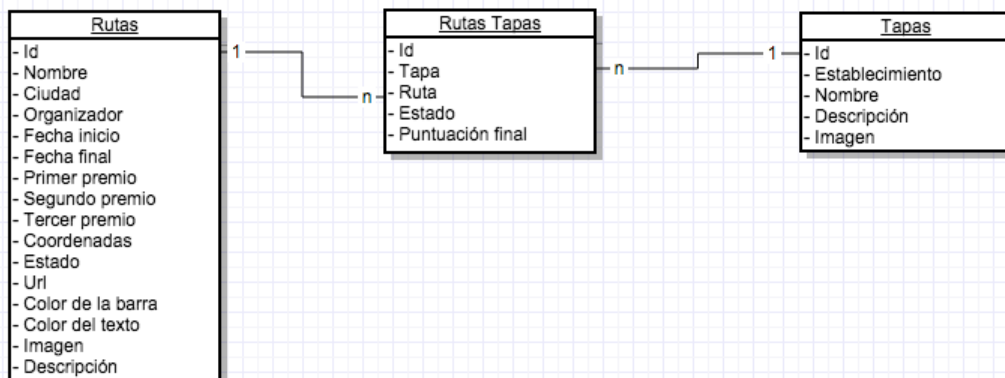


Figura 11. Esquema de BD de las rutas

Como se puede ver en la figura anterior, una tapa puede participar en más de una ruta, así como en una ruta pueden (y deben) haber múltiples participantes.

Para las rutas los campos necesarios serán:

- **Id:** Identificador único de la ruta.
- **Nombre:** Nombre de la ruta.
- **Ciudad:** Ciudad en la que se ubicará.
- **Organizador:** Identificador del organizador.
- **Fecha inicio:** Fecha de inicio de la ruta.
- **Fecha final:** Fecha de finalización de la ruta.
- **Primer premio:** Primer premio que se otorgará.
- **Segundo premio:** Segundo premio que se otorgará.
- **Tercer premio:** Tercer premio que se otorgará.
- **Coordenadas:** Coordenadas donde se ubicará la ruta.
- **Estado:** Estado de la ruta (activa, finalizada...)
- **Url:** Url única de la ruta.
- **Color de la barra:** Color de la barra superior de la portada de la ruta.
- **Color del texto:** Color del texto de la barra superior de la portada de la ruta.
- **Imagen:** Banner superior de la portada de la ruta.
- **Descripción:** Descripción de la ruta.

Diseño e implementación del servidor web "Ruta de la Tapa"

Para las rutas, tanto el color de la barra como las imágenes serán opcionales.

Para las tapas deberemos almacenar los siguientes datos:

- **Id:** Identificador único para la tapa.
- **Establecimiento:** Identificador único del establecimiento que la creó
- **Nombre:** Nombre de la tapa.
- **Descripción:** Descripción de la tapa.
- **Imagen:** Imagen de la tapa.

Así pues, se necesitará una tabla para relacionar las rutas con las tapas y así saber que tapas están participando en que rutas:

- **Id:** Identificador único.
- **Tapa:** Identificador de la tapa.
- **Ruta:** Identificador de la ruta.
- **Estado:** Estado de la solicitud de participación.
- **Puntuación final:** Total de puntos conseguidos en las votaciones.

Para las noticias, usaremos una tabla separada cuyos atributos serán:

- **Ruta:** ID de la ruta.
- **Fecha:** Fecha en la que se creó la noticia.
- **Texto:** Texto de la noticia

Para el sistema de votaciones necesitaremos de varias tablas; una para los códigos QR, otra para las votaciones, y otra para los resultados.

Respecto a los códigos QR que se generen para los clientes, así como el estado del mismo crearemos una tabla que deberá llevar los siguientes atributos:

- **Ruta:** Identificador de la ruta
- **Establecimiento:** Identificador del establecimiento.
- **Ruta-Tapa:** Identificador único de la participación del establecimiento.
- **Usado:** Indica si se ha usado ya o no.
- **Token:** Cadena aleatoria

Para almacenar los resultados, la tabla será la siguiente:

- **Ruta-tapa:** Identificador único de la participación del establecimiento.
- **Ruta:** Identificador de la ruta
- **Posición:** Posición en la que ha quedado tras el recuento.
- **Valor del premio:** Valor en metálico del premio.

Por último, deberemos recoger los datos obtenidos mediante las llamadas a nuestra API hechas desde la aplicación Android.

Para almacenar las valoraciones de los usuarios haremos uso de una tabla con los siguientes atributos:

- **Ruta-tapa:** Identificador único de la participación del establecimiento.
- **Ruta:** Identificador de la ruta.
- **Valoración:** Puntuación que ha dado el usuario.



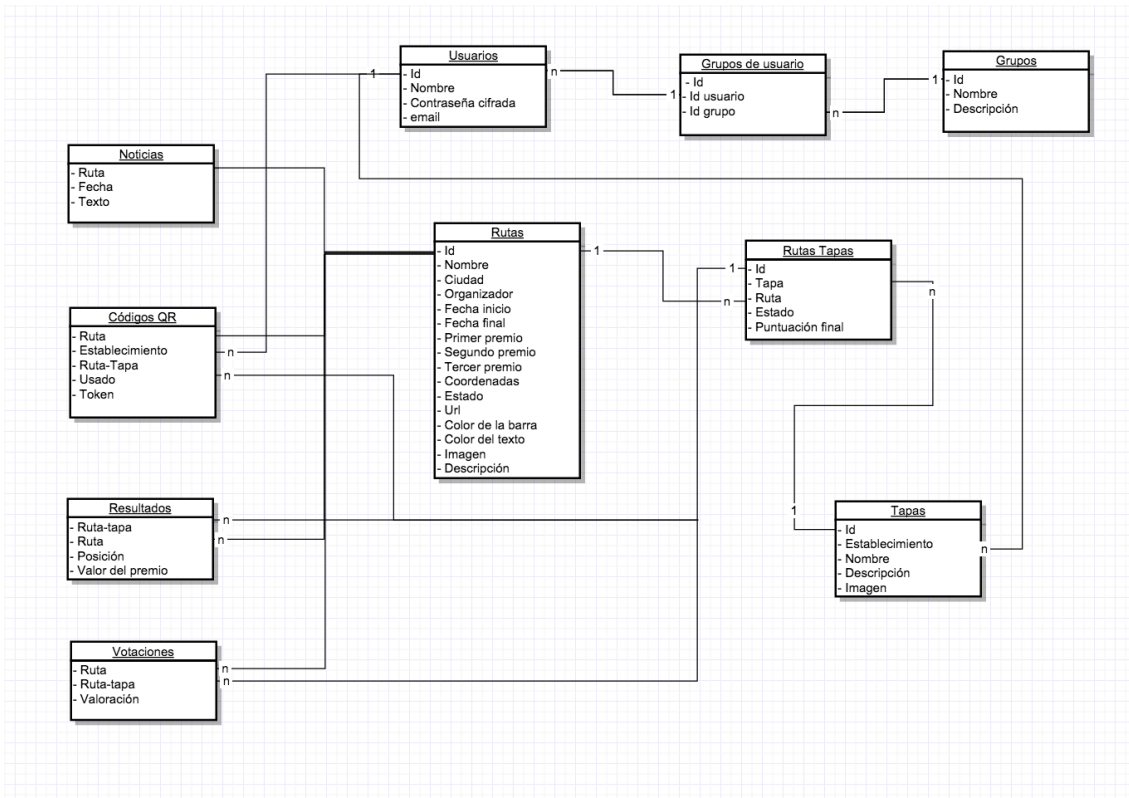


Figura 12. Esquema completo de la BD

Así pues, en la figura 15 se muestra una aproximación del sistema completo de persistencia y sus relaciones.

Se puede observar cierta duplicidad de los atributos de las tablas, como por ejemplo, que la tabla de Códigos QR tenga tanto el identificador de la Ruta como el de la Ruta-Tapa. El código de la ruta se podría obtener a partir del de Ruta-Tapa, pero se ha optado por guardar ambos valores para una búsqueda más rápida.

6.4 Capa de lógica

En esta capa se define toda la funcionalidad de la aplicación. Que acciones hay que ejecutar y cómo. Los casos de uso son los que definirán toda la funcionalidad que se debe implementar. A continuación, se explican los pasos que se realizan para cada una de las acciones.

CU1 - Buscar rutas: el usuario accederá a la portada de la aplicación y se le mostrará un cuadro de texto, en el que podrá introducir sus palabras clave.

En el servidor, se realizará una búsqueda en BD tanto por los campos nombre como población. Se devolverá un JSON para que la parte cliente lo interprete y muestre los resultados como un desplegable bajo el campo de texto.

Al seleccionar uno de los resultados, se le redirigirá a la portada de la ruta en cuestión.



CU2 - Registrarse como organizador: El formulario de registro consistirá en 4 campos; nombre, email, contraseña y confirmación de contraseña. Mediante la petición POST del formulario se enviarán los datos para procesar en el servidor. Éste, se encargará de que todos los campos estén rellenos y tengan el formato adecuado. El campo email tiene que tener el formato adecuado, así como la contraseña debe de ser de una longitud aceptable.

Si los datos son correctos, se crea la cuenta, y se le redirige al panel de control del Organizador.

Si los datos son incorrectos, se le vuelve a redirigir al formulario mostrándole los datos incorrectos.

CU3 - Registrarse como participante: Normalmente el registro como participante se hace desde la portada de una ruta. Al usuario se le pedirá que introduzca los datos necesarios, los cuales son nombre del establecimiento, dirección postal, correo electrónico, contraseña y confirmación de contraseña. Al introducir su dirección, se hace una búsqueda mediante la API de Google Maps para recuperar automáticamente las coordenadas y la dirección correctamente formateada. Estos datos también se enviarán al servidor.

Una vez comprobados los datos y validados, se creará la cuenta y se le redirigirá a la creación de una nueva tapa.

Si los datos fueran incorrectos, se le volvería a mostrar el formulario indicándole los datos incorrectos.

CU4 - Crear una ruta: Una vez el organizador en el panel de control, en la sección de creación de rutas, se le presentará un formulario. En él tendrá que rellenar los datos de la ruta. En este caso, los datos serían: nombre de la ruta, localidad y una breve descripción, las fechas de inicio y final, los premios en metálico que se otorgarán al finalizar y opcionalmente, colores de la barra superior y una imagen en forma de banner.

Al igual que en el CU3, se hará una llamada a la API de Google Maps para recuperar las coordenadas de la localidad en segundo plano.

Cuando estos datos se envíen al servidor, se deberán hacer varias comprobaciones.

En primer lugar, todos los datos deben estar completos y con el formato adecuado.

Si los datos son correctos, se comprobará que el organizador no tuviera alguna ruta creada en las mismas fechas. Por motivos de coherencia de datos, también se comprobará que las fechas sean lógicas; que la fecha de finalización no sea anterior a la de inicio y que no coincidan entre si.

Si todos los datos son coherentes, se almacenarán los datos en BD y se le redirigirá a la pantalla de detalles de la ruta recién creada. Adicionalmente, se creará una noticia automáticamente con un mensaje que indique cuando fue creada.

CU5 - Responder a una solicitud de participación: El organizador, desde su panel de control se le presentarán los datos de la solicitud y las opciones de aceptar o rechazar. En caso de rechazar, deberá especificar los motivos.

En la petición al servidor se le enviará el ID de la ruta, de la tapa, si ha aceptado o no, y opcionalmente los motivos del rechazo.

El servidor deberá comprobar que el organizador es el creador de la ruta y que el ID de la tapa de la solicitud existe. Si los datos son correctos, marcará la solicitud como aceptada o rechazada.

CU6 - Insertar noticia en una ruta: Al organizador, mediante el panel de control, se le presentará un cuadro de entrada de texto en el que podrá introducir la noticia en cuestión.

En la petición al servidor, irá incluido el ID de la ruta. El servidor verificará que la ruta fue creada por dicho usuario y en caso afirmativo guardará la noticia en BD.

La noticia irá acompañada con la marca de tiempo en la que fue creada.

CU7 - Crear una nueva tapa: Para inscribirse en una ruta, los establecimientos deberán crear previamente una tapa con la que participar. Los datos necesarios para su creación serán: nombre para la tapa y una breve descripción.

En este caso, deberá ir acompañado obligatoriamente de una imagen.

Cuando el servidor reciba estos datos mediante una petición POST, verificará que están todos los datos, y almacenará la imagen con un nombre generado aleatoriamente para evitar el escaneo automático de las imágenes.

En BD se almacenará el nombre generado para la imagen y el resto de los datos.

CU8 - Inscribirse en una ruta con una tapa existente: Cuando un establecimiento participante desea inscribirse en una ruta y ya tenía alguna tapa creada, puede seleccionarla sin necesidad de volver a introducir los datos.

En este caso, el servidor recibiría mediante POST el ID de la tapa y de la ruta. El servidor deberá comprobar que el establecimiento no esté ya inscrito en otra ruta en las mismas fechas de la ruta en que se desee inscribir.

En caso de que hubiera conflicto con otra ruta, en vez de mostrar el formulario se le informaría de que ya está inscrito en otra ruta.

Si no es el caso, se genera una nueva solicitud en espera de que el organizador la acepte o rechace.

CU9 - Descargar códigos QR: Los establecimientos con participaciones en rutas aceptadas, necesitarán disponer de códigos QR para repartir a los consumidores y que puedan valorar su tapa.



Diseño e implementación del servidor web "Ruta de la Tapa"

Mediante el panel de control del establecimiento, se envía una petición al servidor con el ID de la participación y empieza el proceso de generación del PDF.

En primer lugar, se comprueba que el ID de la participación pertenece al usuario. Si es así, se generan las URL de las votaciones, que incluirán este ID, la de la tapa, y una cadena generada aleatoriamente para cada uno de los códigos.

Se almacena en BD este código aleatorio junto con el ID de la tapa y de la ruta.

Una vez generados los enlaces, se generarán las imágenes QR con los enlaces y se almacenarán. Tras este paso, se crea el PDF con la información de la ruta y las imágenes y se envían al cliente.

CU10 - Puntuar una tapa: Mediante la aplicación móvil, se escanea un QR proporcionado por el establecimiento y se realiza una petición GET al servidor.

En esta petición, se incluirá el código de la participación, así como de la ruta, un token y la valoración del usuario.

Cuando la petición llega al servidor, se comprobará que el token existe, y todos los datos concuerdan. Para evitar que los usuarios voten más de una vez con el mismo código, en BD se encuentra almacenado el estado del código; es decir, si ya se había usado anteriormente.

En caso de que sea una petición válida, se marca el código como usado y se almacena la valoración del usuario.

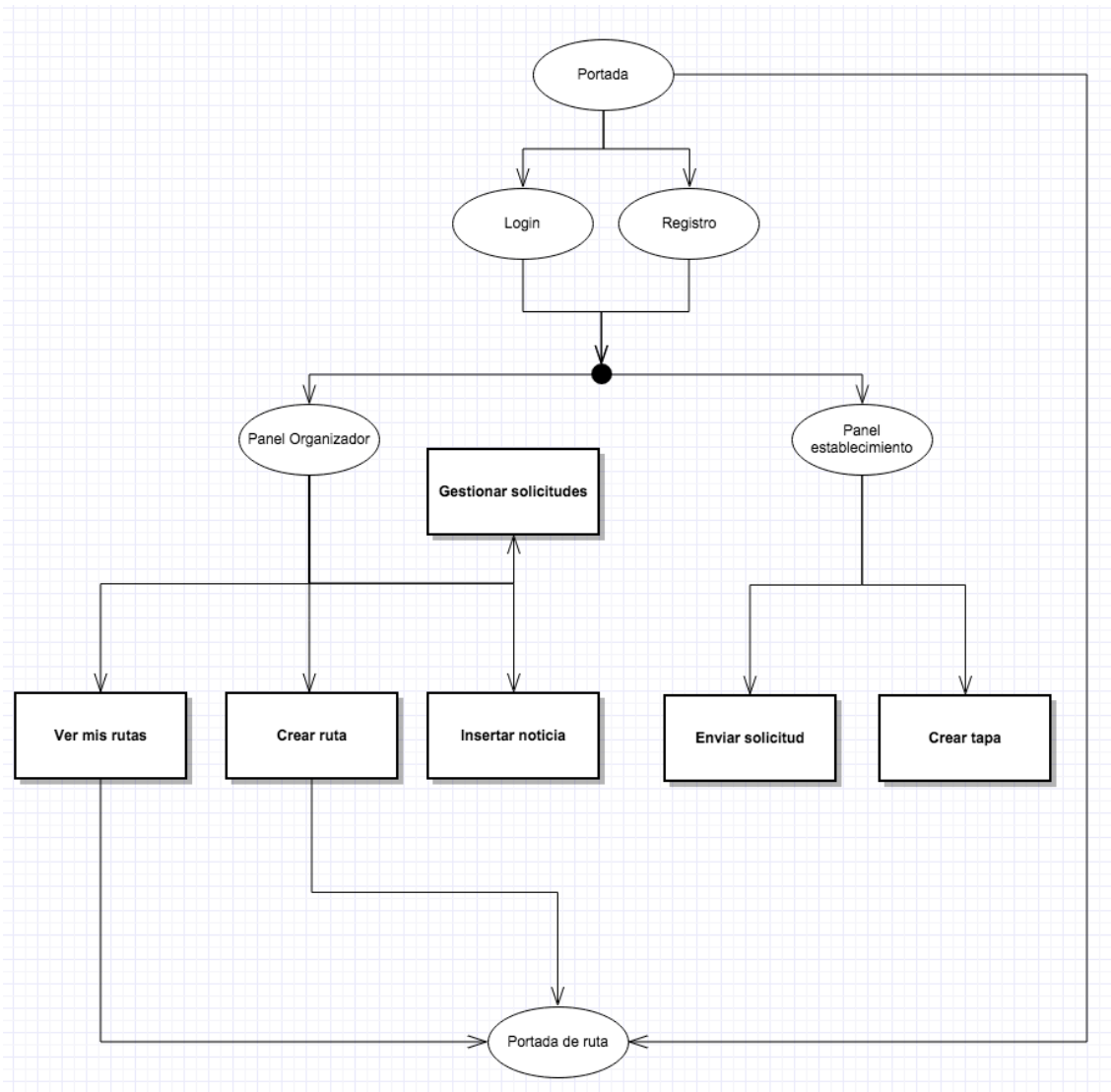


Figura 13. Diagrama de alcanzabilidad

En el diagrama se puede observar los flujos normales de un usuario. Desde la portada, si ya tiene cuenta de organizador o establecimiento, o se registra, accederá a su respectivo panel de control. Ahí hará las acciones que desee (insertar noticia, enviar solicitud de participación...). Tendrá acceso directo a las rutas en las que participe o gestione.

Por el contrario, si no tiene cuenta, normalmente irá a las diferentes rutas para obtener información.



7 Detalles de implementación

En este apartado, trataremos el proyecto de una manera mas técnica y real respecto a la implementación final. Indicaremos las tecnologías que se han utilizado en cada una de las capas. También repasaremos las librerías auxiliares que han hecho falta para el completo desarrollo del proyecto. Por último, se mostrará la organización de los ficheros y directorios usada.

7.1 Tecnologías utilizadas

Tal y como hemos mencionado anteriormente, el abanico de posibilidades a la hora de desarrollar del lado del servidor es inmenso. Para este proyecto, se ha elegido PHP como lenguaje principal para gestionar la lógica por experiencia previa del autor.

7.1.1 PHP

Se trata de un lenguaje de programación interpretado para el desarrollo de páginas web de forma rápida y dinámica.

Un fichero PHP contiene código HTML con fragmentos de PHP rodeados de su propia etiqueta (`<?php ... ?>`).

Al realizarse una petición por parte de un cliente, el documento es procesado y los fragmentos de código PHP se ejecutan y se muestran los resultados. Éste código nunca se le muestra al cliente.

7.1.2 Marcos de trabajo para PHP

Un marco de trabajo (Framework en inglés), es una plataforma sobre la que se pueden desarrollar aplicaciones.

Su objetivo es el de facilitar el desarrollo permitiendo a los programadores utilizar su tiempo en realizar los requisitos de la aplicación en vez de tener que pelear con las inconveniencias a bajo nivel del lenguaje, consiguiendo por tanto reducir el tiempo requerido para el desarrollo de la aplicación.

La utilización de un marco de trabajo suele implicar que el tamaño de la aplicación será mayor, ya que el marco de trabajo añade una serie de utilidades y estructuras que pueden no ser necesarias para la solución a implementar. También restringe parte de la libertad del programador, que debe regirse por la estructura que propone el marco de trabajo, por lo que hasta que se consigue cierta familiarización con el mismo, hay que invertir cierto tiempo en este aprendizaje.

Sin embargo, una vez se tiene soltura con un marco de trabajo, la rapidez en la que se puede desarrollar la solución y la facilidad de producir código sostenible y ampliable se ven enormemente incrementadas.

Como ejemplos de Frameworks tendríamos Zend, Yii, Laravel, Symfony, Codeigniter



Para este proyecto se ha elegido CodeIgniter. Ya se tenía experiencia previa, y permite un despliegado inicial muy rápido.

Hace poco, se liberó la 3 versión con múltiples mejoras, por lo que, al estar todavía siendo actualizado, nos da cierta tranquilidad saber que si hubiera fallos críticos, se corregirían en un tiempo razonable.

7.1.3 Librerías auxiliares PHP

IonAuth

Es una librería que nos permite delegar la gestión de usuarios. Gestiona el autenticado, registro, login y grupos de usuario mediante una sencilla API. Está específicamente diseñado para integrarse a la perfección con CodeIgniter.

Un ejemplo de su API es:

```
$this->ion_auth->login($identity, $password, true);
```

Este comando de PHP iniciaría sesión a un usuario y la librería se encargaría de generar y almacenar todas las variables de sesión necesarias.

```
$this->ion_auth->logout();
```

Así de sencillo sería cerrar la sesión del usuario.

PhpQRCode

Esta librería permite la generación dinámica de códigos QR en forma de imágenes. Está basado en la librería en C libqrencode. Proporciona una sencilla API para su generación sin ninguna dependencia externa.

```
QRcode::png($qrURL, $qrFileName, "L", 5, 3);
```

Mediante esta sencilla orden, la librería genera una imagen con el QR generado a partir de la URL que se le pasa por parámetro. Lo almacena en el directorio definido por \$qrFileName. El resto de argumentos definen el tamaño y calidad del mismo.

DomPDF

Es una librería que permite la conversión de código HTML a PDF. Maneja estilos CSS2.1 y algunas propiedades CSS3 por lo que no podremos usar las propiedades más nuevas de este lenguaje. No tiene dependencias externas y nos proporciona una API para generar los documentos.

```
$oDompdf = new DOMPDF();  
$oDompdf->load_html($htmlData);  
$oDompdf->render();
```

Con estas 3 líneas, se nos creará un documento PDF al que podremos acceder con:

```
$oDompdf->output();
```



Diseño e implementación del servidor web "Ruta de la Tapa"

El contenido del documento viene dado por el código de la variable \$htmlData.

7.2 Plataformas de desarrollo

Para este proyecto se ha usado principalmente PhpStorm en su versión 8.0.3 para el desarrollo del código.

Este programa cuenta con auto-completado de código, inspección de código y marcado de errores, resaltado de sintaxis con códigos de colores.

Maneja PHP, HTML, CSS, JS entre otros por lo que se adapta perfectamente a las necesidades de este proyecto.

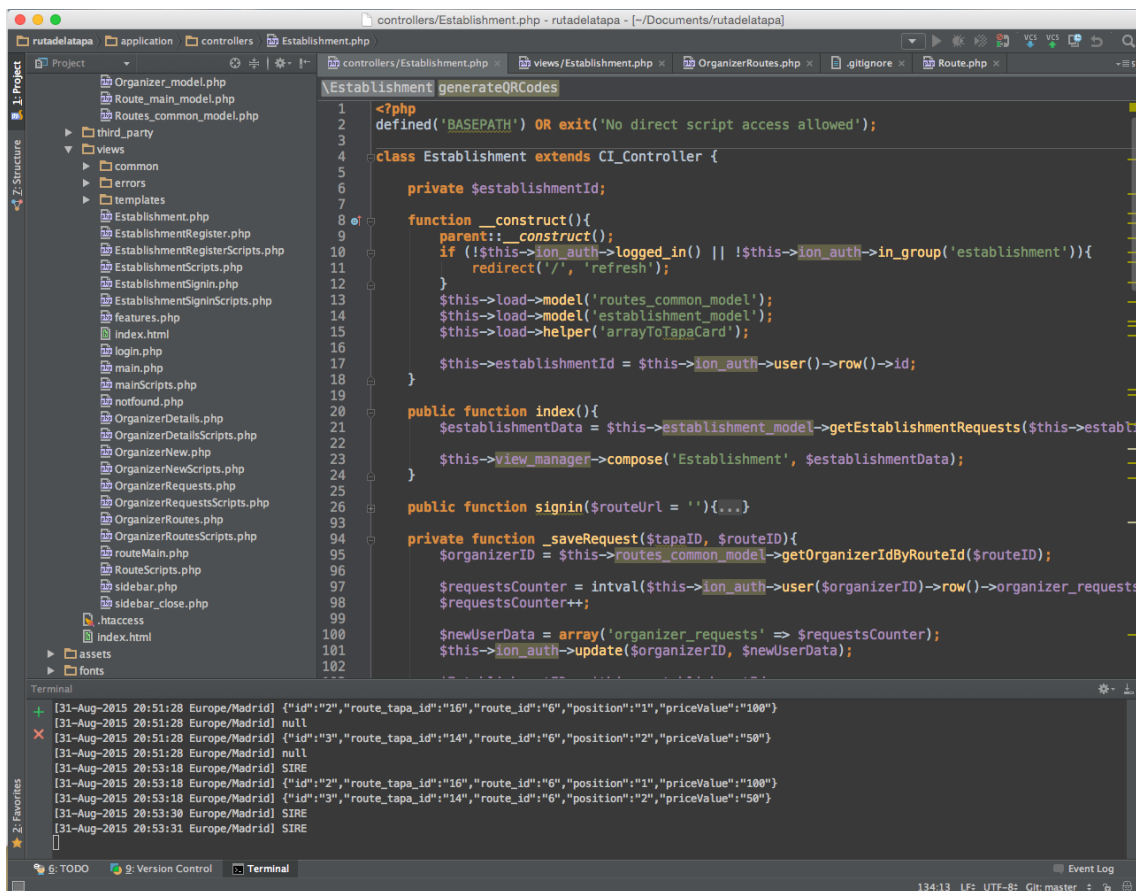


Figura 14. Captura de pantalla de PhpStorm

Para la gestión de BD se ha usado SequelPro ya que permite trabajar con la BD de una manera visual muy sencilla.



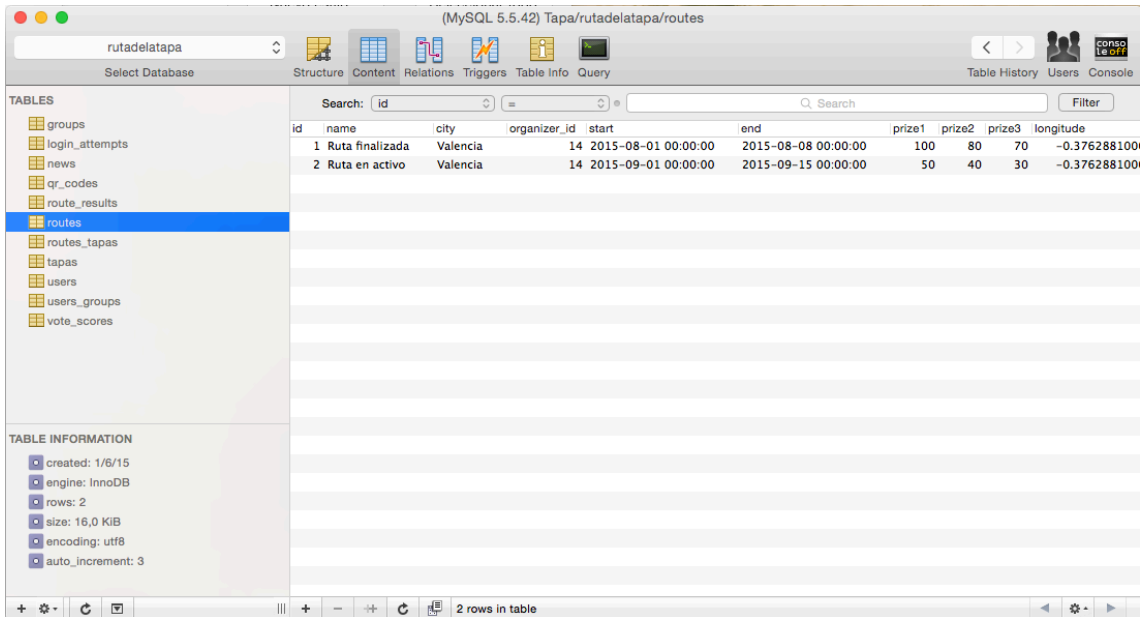


Figura 15. Captura de pantalla de Sequel Pro

Como servidor de pruebas local se ha usado MAMP ya que permite el despliegado de Apache, MySQL y PHP con un click.

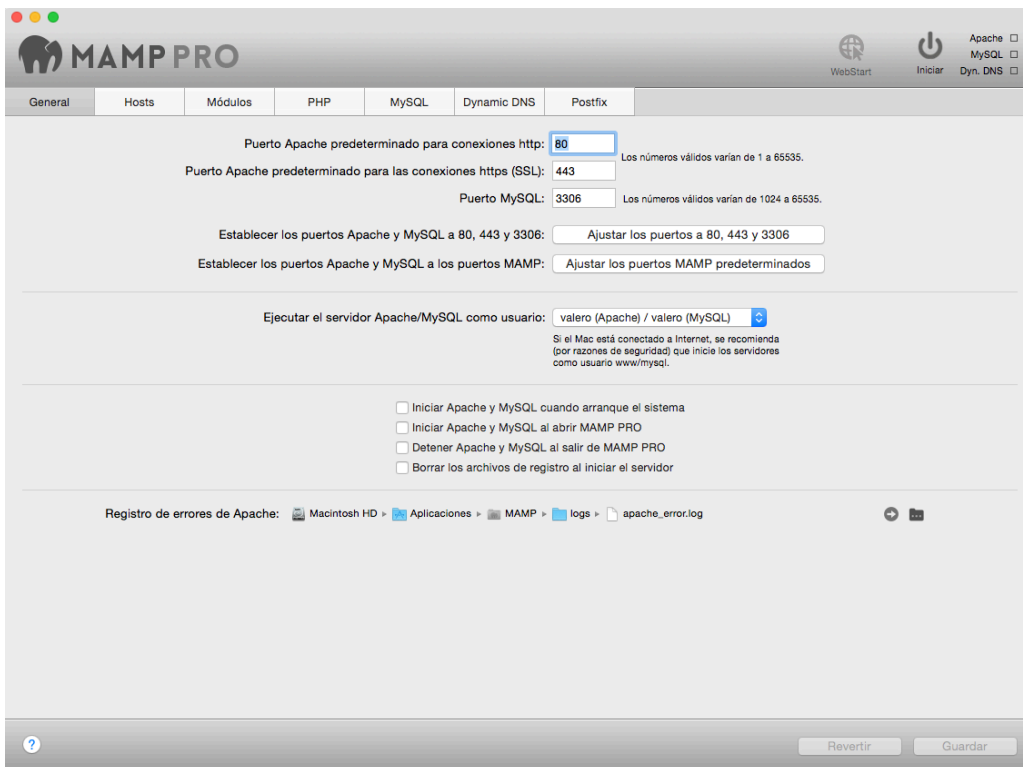


Figura 16. Captura de pantalla de MAMP Pro



Diseño e implementación del servidor web "Ruta de la Tapa"

Para la gestión de git se ha utilizado SourceTree ya que permite toda la gestión mediante una interfaz visual atractiva y sobretodo sencilla.

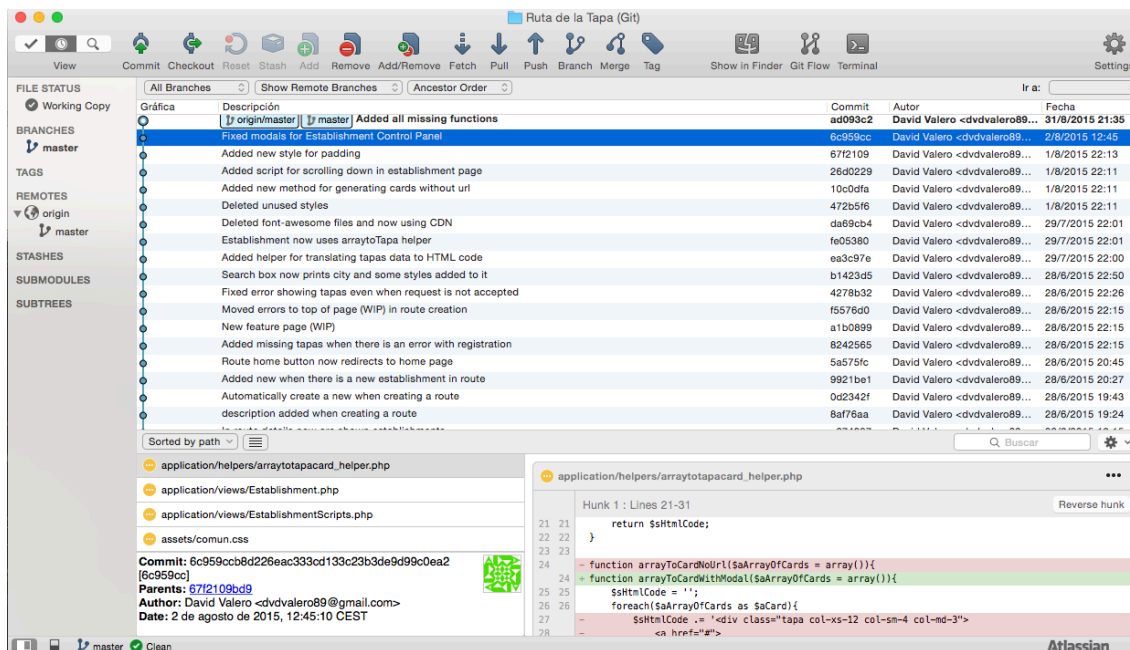


Figura 17. Captura de pantalla de SourceTree

Todo el código ha sido desarrollado y testado con Google Chrome y Firefox en sus últimas versiones.



7.2.1 Control de versiones

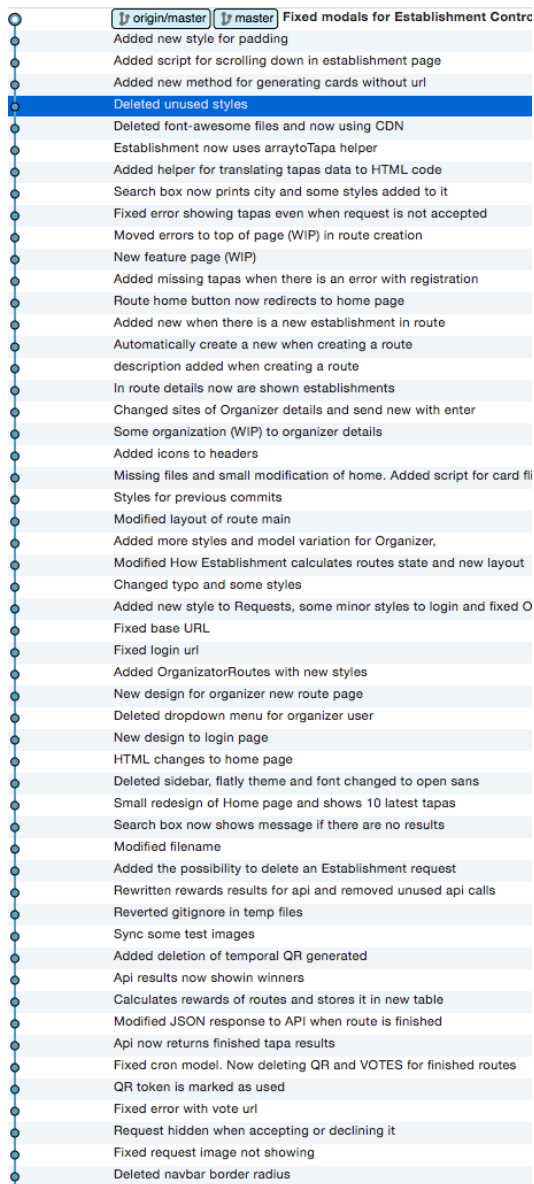


Figura 18. Captura de commits en proyecto GIT

Para gestionar los cambios se ha usado Git usando los servicios de Bitbucket ya que permite el almacenamiento gratuito de repositorios privados.

Esta herramienta nos permite ir haciendo guardados individuales y la posibilidad de deshacer cambios individualmente.

Como ventaja añadida, permite tener una copia de seguridad en la nube del código, así como la posibilidad de desplegar el sistema en otro ordenador para continuar el desarrollo.

Normalmente se hace uso de ramas pero al desarrollar el proyecto una sola persona y por simplificar el proceso se ha hecho el desarrollo íntegramente en la rama Master.



7.3 Estructura de los ficheros y directorios

El funcionamiento de nuestra aplicación web será el siguiente: Los usuarios realizaran peticiones a través del navegador. Estas peticiones llegarán a nuestra aplicación web hospedada en el servidor Apache, y la aplicación realizara los accesos a la base de datos y al sistema de ficheros para obtener la información necesaria para componer la página de respuesta para el usuario.

Al usar Codeigniter, se está bastante restringido a la hora de organizar los ficheros, siendo conveniente seguir sus convenciones.

7.3.1 Carpetas de Codeigniter

Application

En esta carpeta se alojarán todos los ficheros de la aplicación en sí. Estarán divididos en carpetas de entre las que se destacan:

Config

Almacena los ficheros de configuración generales. Conexión a BD, rutas de las URL, librerías que se deben cargar automáticamente entre otras.

Controllers

Aquí se encuentran los ficheros de la lógica de la aplicación. Desde la versión 3, los nombres de los ficheros deben empezar en mayúsculas. Y todas las clases extender CI_Controller.

Libraries

En esta carpeta almacenaremos las librerías externas que queramos incorporar al proyecto. En nuestro caso, hemos añadido la librería de Autenticación Ion Auth, Dompdf para convertir código HTML en PDF y Phpqrcode para crear imágenes QR.

Models

Aquí se almacenan los ficheros que gestionan la persistencia. Al igual que en la carpeta Controllers, deben empezar por mayúscula y extender la clase CI_Model.

Views

Dentro de la carpeta estarán las vistas de la aplicación. Básicamente son ficheros HTML con el código PHP incrustado necesario.

System

En esta carpeta encontramos las librerías propias de CI. Esta carpeta está separada del resto ya que, excepto actualizaciones mayores, para actualizar de una versión a otra del framework sólo tendríamos que sustituir dicha carpeta.

7.3.2 Carpetas propias

Se han creado diversas carpetas auxiliares que se necesitaban para el desarrollo de esta aplicación en concreto.



assets

Aquí se guardan todos los ficheros JS y CSS externos. Como por ejemplo, jQuery, Bootstrap, Font Awesome o MomentJs.

QR_temp

En esta carpeta se almacenan temporalmente las imágenes de los códigos QR que se enviarán a los establecimientos para su distribución.

Una vez generado el PDF y enviado al usuario, se eliminan de esta carpeta.

route_banners

Esta carpeta almacena los banners que los organizadores pueden incluir en las portadas de las rutas creadas por ellos. Antes de almacenarlas, se cambia el nombre de los archivos a uno generado aleatoriamente.

tapa_images

Aquí se almacenan las imágenes de las tapas que suben los establecimientos de sus tapas. Al igual que con los banners, el nombre de los archivos se cambia por una cadena aleatoria.

7.4 Estructura de la Base de Datos

Hay dos tipos de tablas; las necesarias para Ion Auth y las propias de la aplicación.

7.4.1 Ion Auth

groups

Almacena los distintos grupos a los que puede pertenecer un usuario. Administrador (no usado en este desarrollo), Establecimientos y Organizadores de rutas.

login_attempts

Almacena los intentos de login que se hacen sobre una cuenta para bloquearla y evitar ataques de fuerza bruta

users

Aquí se guardan los credenciales de los usuarios. Identificador único, nombre de usuario, contraseña codificada, email, fecha de creación, datos propios de establecimientos (dirección y coordenadas del local) y datos propios de organizadores (número de solicitudes de participación).

users_groups

Esta tabla almacena la relación de los usuarios con el grupo al que pertenecen.



7.4.2 Propias de la aplicación web

news

En esta tabla se almacenan las noticias de cada una de las rutas con su identificador único y la fecha en la que fueron creadas.

qr_codes

Esta tabla guarda los "tokens" para cada uno de los códigos que los establecimientos ofrecen a sus usuarios para votar. También se almacena la ruta a la que pertenecen, el establecimiento y el identificador único de la tapa con la que se presentan y si está ya utilizado o no.

route_results

Aquí se guardan los resultados una vez que una ruta finaliza. Se guarda el identificador de la tapa, la ruta en la que participaba, la posición en la que quedaron y el premio que se llevaron.

routes

Aquí se guardan todas las rutas que han sido creadas por los organizadores y sus datos. Más concretamente el nombre, ciudad, identificador del organizador, fecha de inicio y final, los premios que se ofrecen, las coordenadas en las que se ubica, los códigos hexadecimales de los colores de la barra superior y texto, la descripción de la ruta y opcionalmente, el nombre del banner superior que adjuntó el organizador.

routes_tapas

Aquí se almacena la relación entre la tapa, la ruta, el organizador y el establecimiento por cada participación en una ruta. Así como el estado de la solicitud, el motivo del rechazo (si hubiera sido rechazado) y la puntuación final una vez acabada la ruta.

tapas

En esta tabla encontraremos todas las tapas que crearon los establecimientos. Como datos tenemos: el nombre, la descripción y el nombre del fichero de imagen de la tapa.

vote_scores

En esta tabla se almacenan las puntuaciones que han ido dando los clientes a las distintas tapas participantes. Una vez acabada una ruta, se cuentan los votos y se eliminan de la tabla.

7.5 APIs REST

Uno de los requisitos de la aplicación era la de proporcionar una API para el consumo de la aplicación móvil desarrollada por la alumna Marta Gaona. Todas las peticiones se hacen mediante GET y se devuelve en formato JSON.

A continuación se describen las diferentes rutas que conforman la API.

7.5.1 Obtención de rutas disponibles

URL: /api/routes

Parámetros de entrada GET: longitude, latitud (ambos opcionales)

Devuelve un listado de las rutas disponibles. Si se proporciona la longitud y la latitud, se devolverán las más cercanas a dicha coordenada.

Formato respuesta:

```
[
  {
    "id" : id ruta,
    "name" : nombre de la ruta,
    "latitud" : latitud,
    "longitude" : longitud
  }
]
```

7.5.2 Obtención de detalles de ruta

URL: /api/routes/{ID ruta}

Parámetros de entrada en URL: id de la ruta

En este caso, se devuelve toda la información relativa a una ruta definida en la URL. Devuelve el nombre, coordenadas, dirección y una lista de los establecimientos participantes. De éstos, se devuelve el nombre, la descripción, el nombre de la tapa, la descripción de la tapa y la URL de la imagen de esta.

Formato respuesta:

```
{
  "id" : id establecimiento,
  "name" : nombre del bar,
  "latitud" : latitud del establecimiento,
  "longitude" : longitud del establecimiento,
  "address" : dirección del bar,
  "tapaName" : nombre de la tapa,
  "tapaDescription" : descripción de la tapa,
  "tapaImage" : ruta de la imagen de la tapa
}
```



7.5.3 Obtención de resultados del concurso

URL: /api/results/{ID ruta}

Parámetros de entrada en URL: id de la ruta

Si la ruta ha finalizado, devuelve los resultados del concurso. Más concretamente el ID de las rutas ganadoras, su posición y el premio en metálico que recibirán.

Si no ha finalizado aún, se le devuelve un json indicándole que todavía no ha finalizado.

Formato respuesta:

```
{
  "finished" : true/false,
  "rewards" : [
    {
      "tapaId" : id tapa,
      "position" : posición final,
      "price": premio en metálico
    }
  ]
}
```

7.5.4 Inserción de nueva valoración de usuario

URL: /api/vote/{ID ruta}/{token}/{puntuación}

Parámetros de entrada en URL: id de la ruta, token, valoración del usuario

Esta petición se realiza cada vez que un usuario quiere valorar una tapa tras haber recibido uno de los códigos QR.

El sistema comprueba que el token es válido; es decir, existe en la BD para dicha ruta y todavía no ha sido usado.

En caso de que este todo correcto, se almacena la valoración y se devuelve un código indicando que la valoración se ha realizado correctamente.

Formato respuesta:

```
{
  "status" : "OK"/"FAILED"
}
```


8 Despliegue

Como servidor de pruebas se contaba con una Raspberry Pi 2 Model B con la distribución Raspbian recién instalada.

Al tratarse de una aplicación web sólo ha sido necesario instalarle los servicios asociados necesarios y desplegar el código.

8.1 Instalación de servicios

En este caso mediante la terminal instalamos los paquetes de PHP 5.6, Apache y MySQL.

```
$ apt-get install php56 php56-mcrypt apache2 libapache2-mod-php5 mysql-server mysql-client php5-mysql.
```

8.2 Despliegado de código

Tras instalar todos los paquetes, procedemos a desplegar el código. Como teníamos el código en git, usaremos este método para obtener el código.

Instalamos git y creamos el proyecto en /var/www (carpeta por defecto de Apache).

```
$ cd /var/www
```

```
$ apt-get install git
```

```
$ git clone https://valero89@bitbucket.org/valero89/ruta-de-la-tapa.git rutadelatapa
```

Esto nos creará una carpeta llamada rutadelatapa y copiará todo el código en ella.

En este momento ya podríamos ver en la dirección local <http://localhost/rutadelatapa> el proyecto. Pero nos mostrará un error al no haber configurado correctamente la conexión con MySQL.

8.3 Configuración de BD

En el proceso de instalación de MySQL por terminal nos habrá solicitado una contraseña. Es la que utilizaremos para conectarnos. En este caso, hemos usado SequelPro para conectarnos a la BD por su simplicidad.



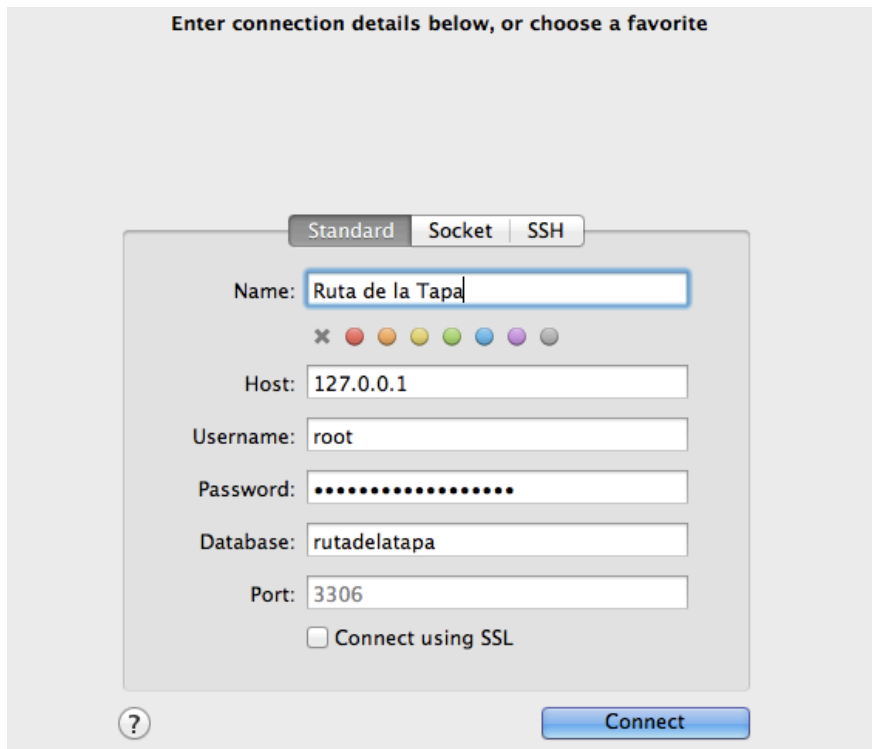


Figura 19. Captura de SequelPro. Ventana de conexión a BD

Una vez conectados, crearemos una nueva base de datos e importaremos el esquema que habremos exportado previamente.

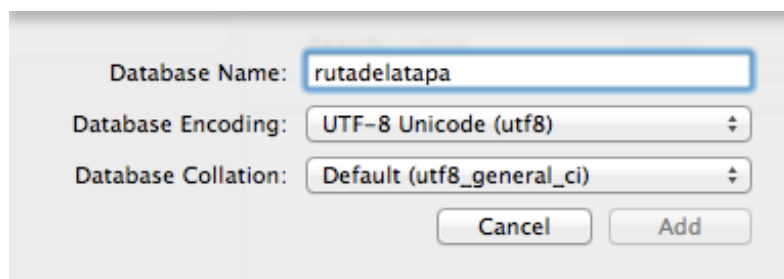


Figura 20. Captura de SequelPro. Creación de nueva BD

Ahora sólo queda configurar la aplicación para que se conecte a nuestra BD recién creada. Abrimos el fichero `application/config/database.php` y editamos los campos correspondientes

```

$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => '*****',
    'database' => 'rutadelatapa',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => TRUE,
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'autoinit' => TRUE,
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```

Figura 21. Fichero de configuración de BD

Una vez configurado el fichero, ya podremos conectarnos. En el caso de estar alojados los archivos en una subcarpeta de la raíz de apache también deberemos especificarlo en el fichero application/config/config.php ya que todos los enlaces y archivos están definidos respecto a la url base.

```

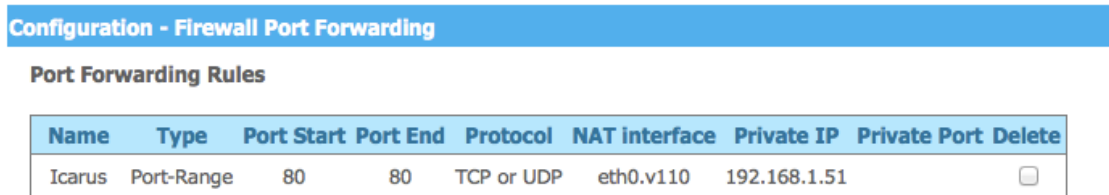
/*
-----
| Base Site URL
|-----
|
| URL to your CodeIgniter root. Typically this will be your base URL,
| WITH a trailing slash:
|
|   http://example.com/
|
| If this is not set then CodeIgniter will try guess the protocol, domain
| and path to your installation. However, you should always configure this
| explicitly and never rely on auto-guessing, especially in production
| environments.
|
|*/
$config['base_url'] = '/rutadelatapa/';

```

Figura 22. Configuración de URL base

8.4 Configuración router

Dado que en el entorno de pruebas en el que se encuentra el servidor, estamos tras un router, para permitir el acceso desde fuera de la intranet deberemos redireccionar los puertos entrantes. Lo haremos configurando el router por lo que este paso dependerá del modelo de router o la ubicación del servidor respecto de este.



The screenshot shows the Mikrotik configuration interface for Firewall Port Forwarding. The title bar reads "Configuration - Firewall Port Forwarding". Below it, the section is titled "Port Forwarding Rules". A table lists the configuration for a rule named "Icarus".

Name	Type	Port Start	Port End	Protocol	NAT interface	Private IP	Private Port	Delete
Icarus	Port-Range	80	80	TCP or UDP	eth0.v110	192.168.1.51		<input type="checkbox"/>

Figura 23. Ejemplo de configuración del router

En nuestro caso, hemos abierto el puerto 80 entrante y se lo hemos asignado a la dirección 192.168.1.51 ya que es la IP interna que hemos definido en el servidor.

9 Pruebas

Para este proyecto, no se han realizado pruebas unitarias programadas.

Se ha optado por hacer revisiones continuas con amigos y familiares. Una vez estuvo la primera versión operativa en el servidor, se invitó a los conocidos a realizar pruebas y que detectaran errores.

Durante todo el desarrollo, se ha modificado varias veces el diseño gráfico y se han arreglado errores de coherencia de los datos.

Los perfiles de los usuarios que realizaron las pruebas, son de gente de 20 a 35 años, con destrezas tecnológicas que van desde usuario medio (la mayoría), véase consumidores de información, hasta usuarios expertos.

En total, unas 25 personas han dado su feedback sobre el producto y se ha rediseñado 3 veces el aspecto visual en base a sus comentarios.

9.1 Resultado de diseño final

A continuación se muestran capturas de pantalla del resultado final de la aplicación.



Diseño e implementación del servidor web "Ruta de la Tapa"



Figura 24. Diseño final de la portada de la ruta de la tapa

En la figura 24 se muestra el diseño final de la portada. El banner con la imagen superior se hizo mas pequeño y se eliminó el color azul oscuro de la barra superior ya que llamaba demasiado la atención.

Bajo la portada, se muestran las últimas tapas que participan en alguna ruta. Estas imágenes enlazan directamente con la portada de las rutas.



Figura 25. Diseño final de la página de características

Quando hacemos click en alguna de las secciones inferiores de la portada, se nos lleva a una página en la que se nos explican con mas detalles que podemos encontrar dentro de la aplicación y los motivos por lo que deberían usarla.

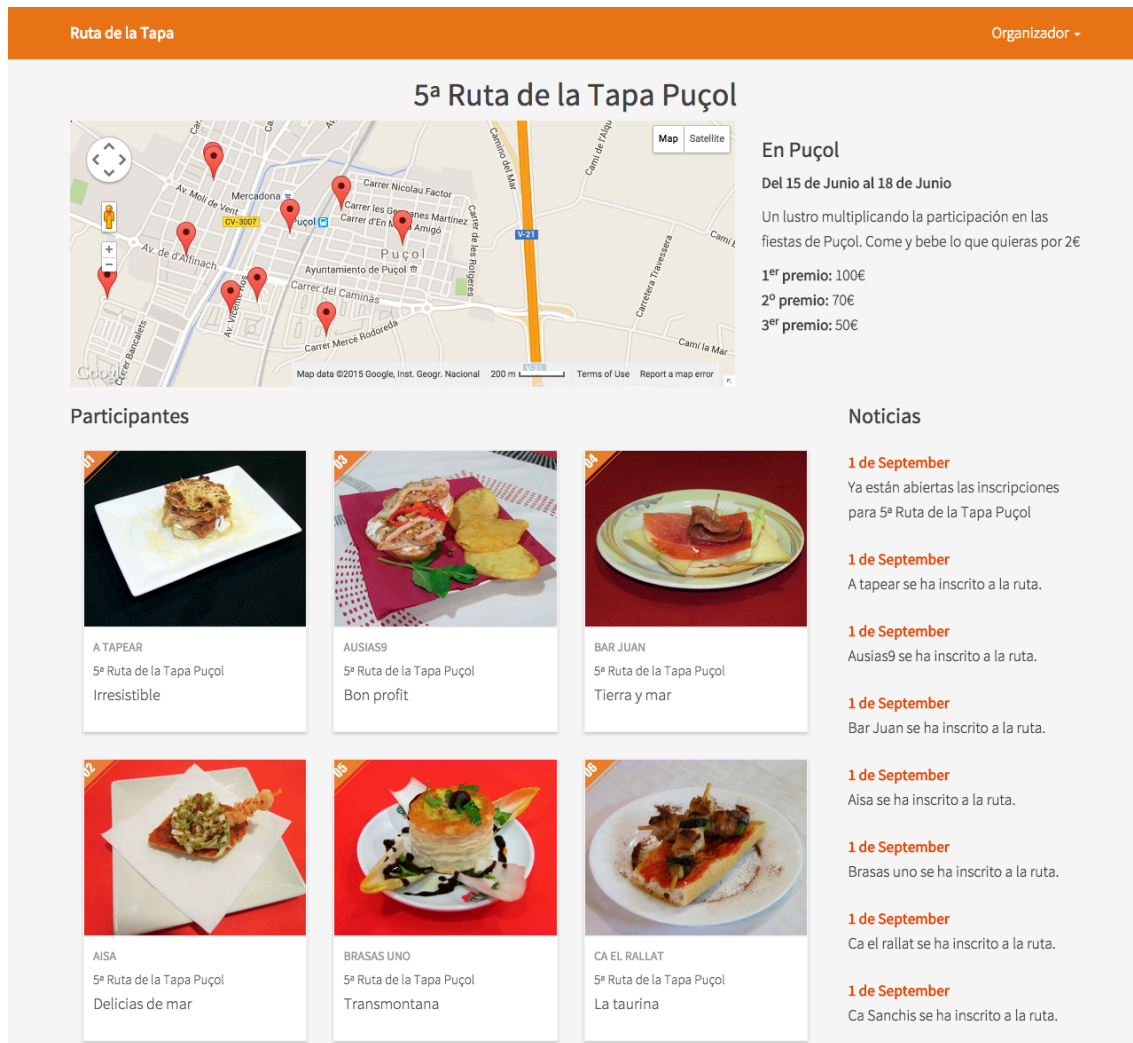


Figura 26. Diseño final de la portada de una ruta

La figura 26 representa la portada de una de las rutas. El color de la barra y del texto viene definido por el organizador de la ruta.

Lo primero que se nos muestra es un mapa con la localización de los establecimientos, así las fechas y la descripción de la ruta.

Abajo se muestran todas las tapas participantes y al lado las noticias.

En esta pantalla, a los usuarios se les pidió que consiguieran descubrir la dirección de una de las tapas. Al ver que no lo solían encontrar, se añadieron características adicionales. Al hacer click en una de las tapas, el mapa de google se centra en el establecimiento.

Además, en versiones anteriores, se desactivaron los controles del mapa, por lo que no se podía hacer zoom. Así que se volvieron a incluir.

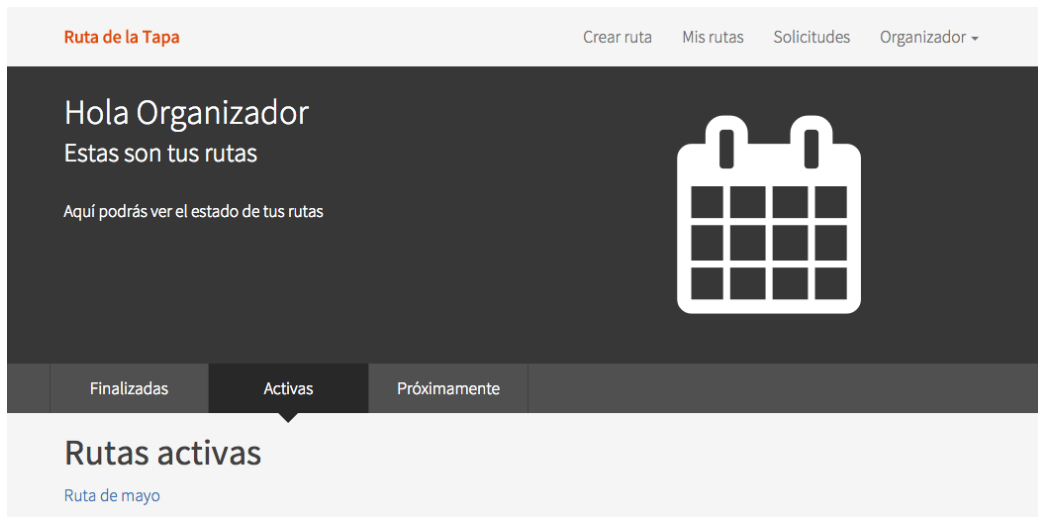


Figura 27. Panel superior del panel de control de los organizadores

En la figura 28 podemos ver el nuevo aspecto del panel de control de los organizadores.

Se ha sustituido la barra lateral por enlaces en la barra superior.

Para darle continuidad a la aplicación, se optó por incrustar un banner en todas las secciones en la parte superior. En este caso, como una imagen distraería, se prefirió usar un icono.

Diseño e implementación del servidor web "Ruta de la Tapa"

The screenshot displays the 'Ruta de la Tapa' web application interface. At the top, there is a navigation bar with the title 'Ruta de la Tapa' and links for 'Crear ruta', 'Mis rutas', 'Solicitudes', and 'Organizador'. Below this, a dark grey header area contains the text 'Hola Organizador' and 'Estás a 5 minutos de crear tu ruta'. To the right of this header is a preview window titled 'Ruta de prueba' which is currently empty. Below the header, there are several input fields: 'Ruta de prueba', 'Valencia', and 'Descripción de la ruta'. The 'Valencia' field has a dropdown arrow and the text 'Población detectada: Valencia' below it. A Google Maps interface is embedded, showing a map of Valencia with a red location pin and a scale bar. Below the map, there are 'Fechas' (start and end dates) set to '31-08-2015', 'Premios' (Primer premio, Segundo premio, Tercer premio) fields, and color selection options for the top bar ('#e14b0c') and text ('#ffffff'). At the bottom, there is a 'Seleccionar archivo' button and a large orange 'Crear' button.

Figura 28. Diseño final de la pantalla de creación de rutas

En la figura 29, tenemos el aspecto final de la creación de una ruta. En la barra superior se encuentra una preview en la que se ve en tiempo real cómo quedará la ruta una vez creada.

El mapa de Google Maps se ha mantenido y una vez se rellena el campo población, el mapa se actualiza para mostrar la localización que se va a guardar. Así el usuario tiene la confirmación de que está introduciendo el campo correcto.

La pantalla de creación de rutas fue una de las más complicadas de realizar. Había que incluir mucha información y no debía resultar muy pesado de rellenar.

Para hacerlo más ameno, se incluyó el mapa de Google a un tamaño mayor, para que el primer golpe de vista no fueran excesivos campos de entrada.

La “preview”, se movió del final de la pantalla al principio, así, en cuanto los usuarios rellenaban el primer campo, veían los cambios reflejados en él y no requirió (en la mayoría de los casos) más información acerca de su uso.

Respecto a los campos de entrada de fechas, se descubrió que haciendo el campo más interactivo, era más fácil de rellenar, y se cometían menos errores. Por lo que se incluyó una librería que solicita los datos en orden (año, mes, día).

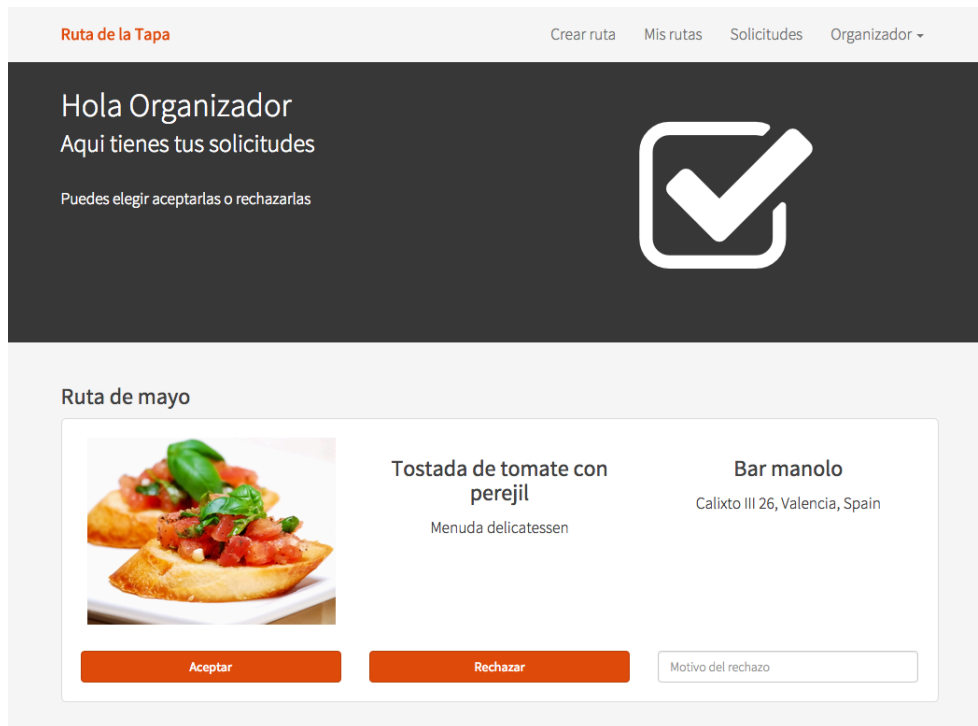


Figura 29. Diseño final de las solicitudes en el panel de control de los organizadores

En la figura 23, vemos la pantalla de solicitudes. Al usuario se le muestra la información introducida por el establecimiento y decide si cumple con los requisitos para entrar. Por ejemplo, que los nombres no contengan palabras malsonantes o que este en la misma población en la que en la que se celebra la ruta.

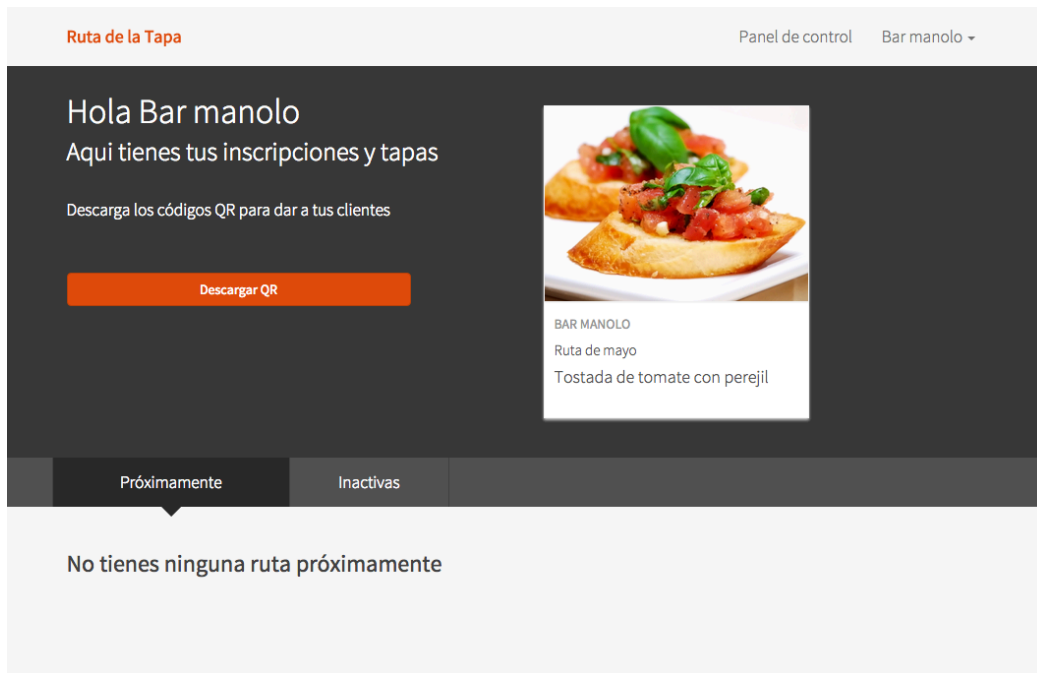


Figura 30. Diseño final del panel de control de los establecimientos

En la figura 24, se muestra el panel de control de los establecimientos. En este caso, el establecimiento tiene una ruta activa, por lo que en la barra superior se le muestra la ficha de la tapa y un botón para descargar los códigos para repartir a sus clientes.

9.2 Pruebas en diferentes navegadores

Uno de los requisitos, era la compatibilidad con la mayoría de navegadores web. Por lo que se realizaron varias pruebas para comprobarlo.

Las pruebas en diferentes navegadores se hicieron sobre OS X en los navegadores a la última versión en el momento de las pruebas; Chrome (Versión 45), Safari (Versión 8.0.7), Opera (Versión 31.0) y Firefox (Versión 38.0.5).

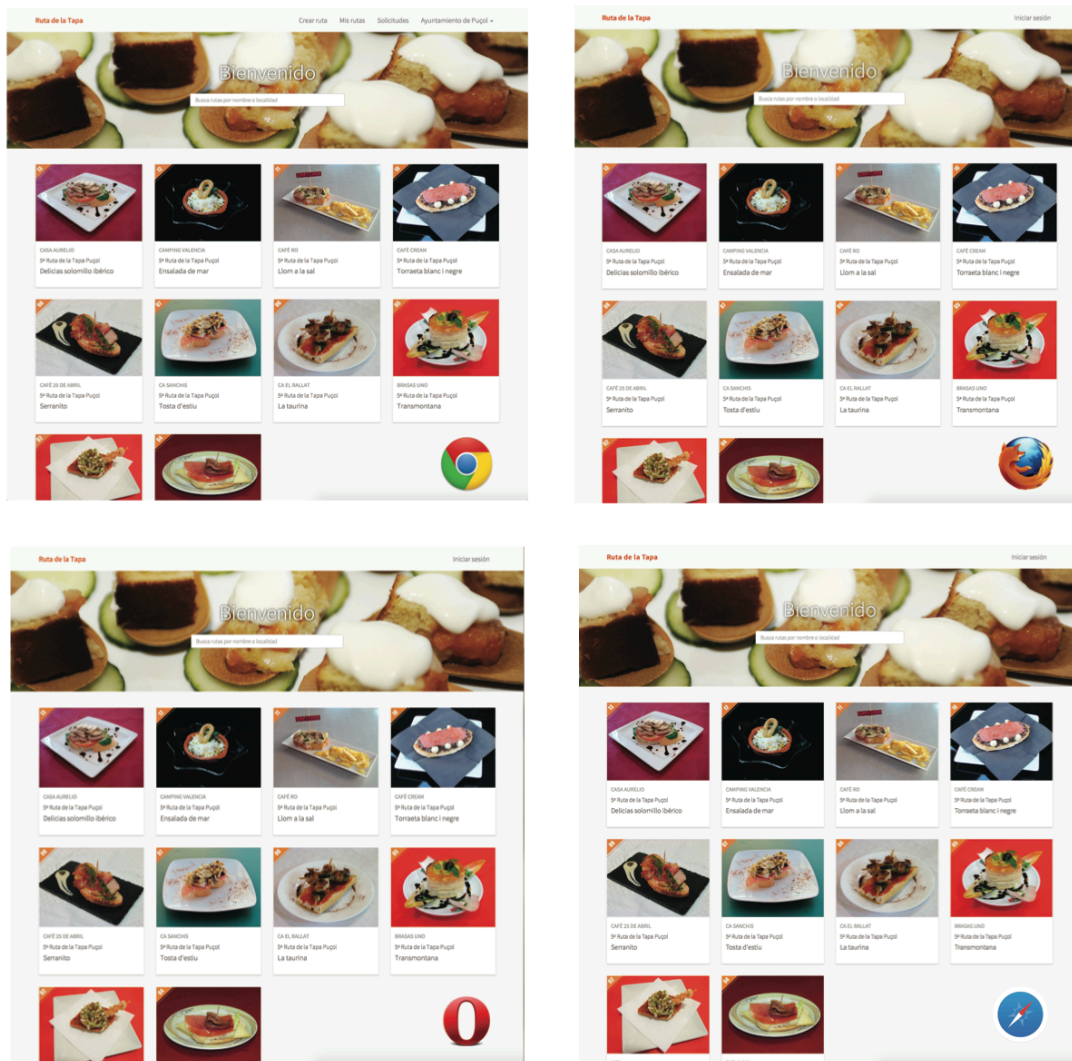


Figura 31. Composición de la portada en diferentes navegadores

La primera prueba que se hizo fue con la portada de la aplicación. Se comprobó que el aspecto era idéntico en todas ellas, tanto a pantalla completa como reduciendo el tamaño de la ventana. Las diferencias de las longitudes de las capturas se deben a que el tamaño de la barra de navegación era distinto en cada uno de los navegadores.

Diseño e implementación del servidor web "Ruta de la Tapa"

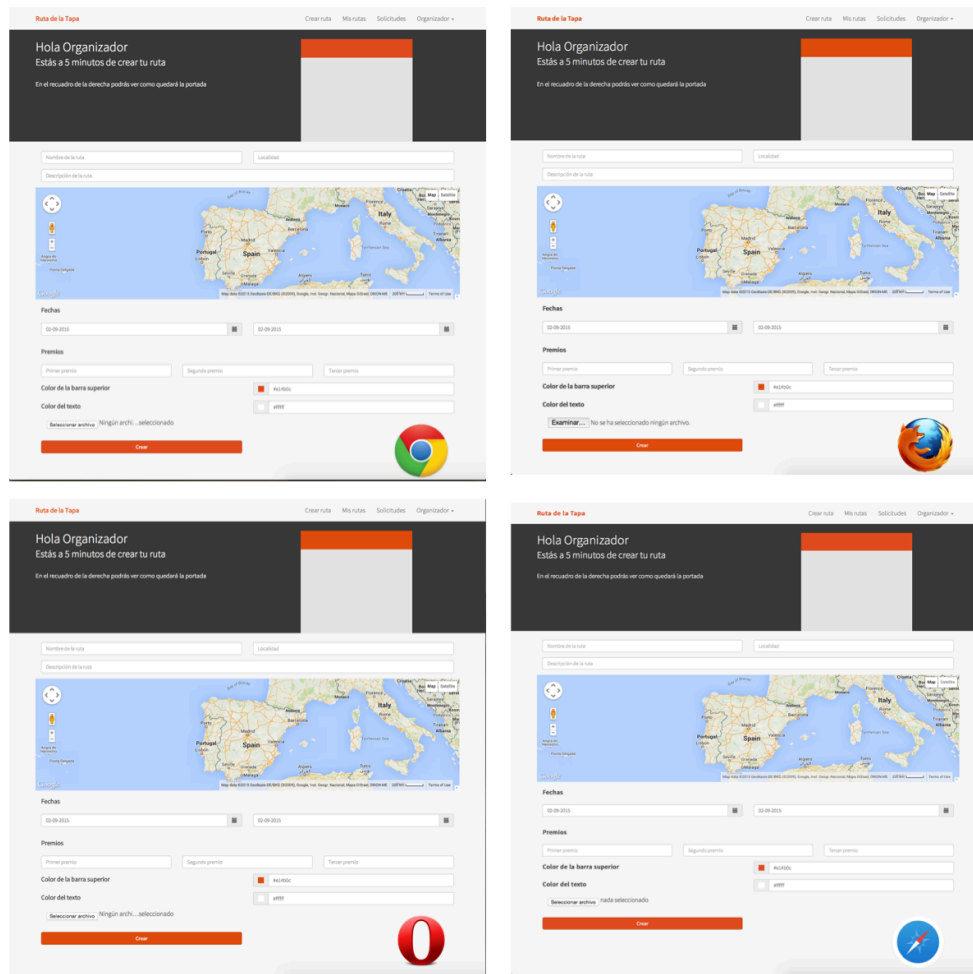


Figura 32. Composición de la ventana de creación de ruta en diferentes navegadores

Tras el éxito de la primera prueba, se repitió el proceso pero en la ventana de creación de ruta. Esta es una de las más complejas al tener más elementos en poco espacio y tener el botón de subir archivo. Este botón, es representado a elección del navegador, y a pesar de que eran diferentes, la distribución de la página no se vio afectada por ello.

9.3 Pruebas en diferentes resoluciones

Otro de los requisitos de la aplicación era que se viera perfectamente en cualquier resolución. Por lo que se comprobó que la web fuera visible en resoluciones grandes (ordenadores), medianas (tablets) y pequeñas (móviles).

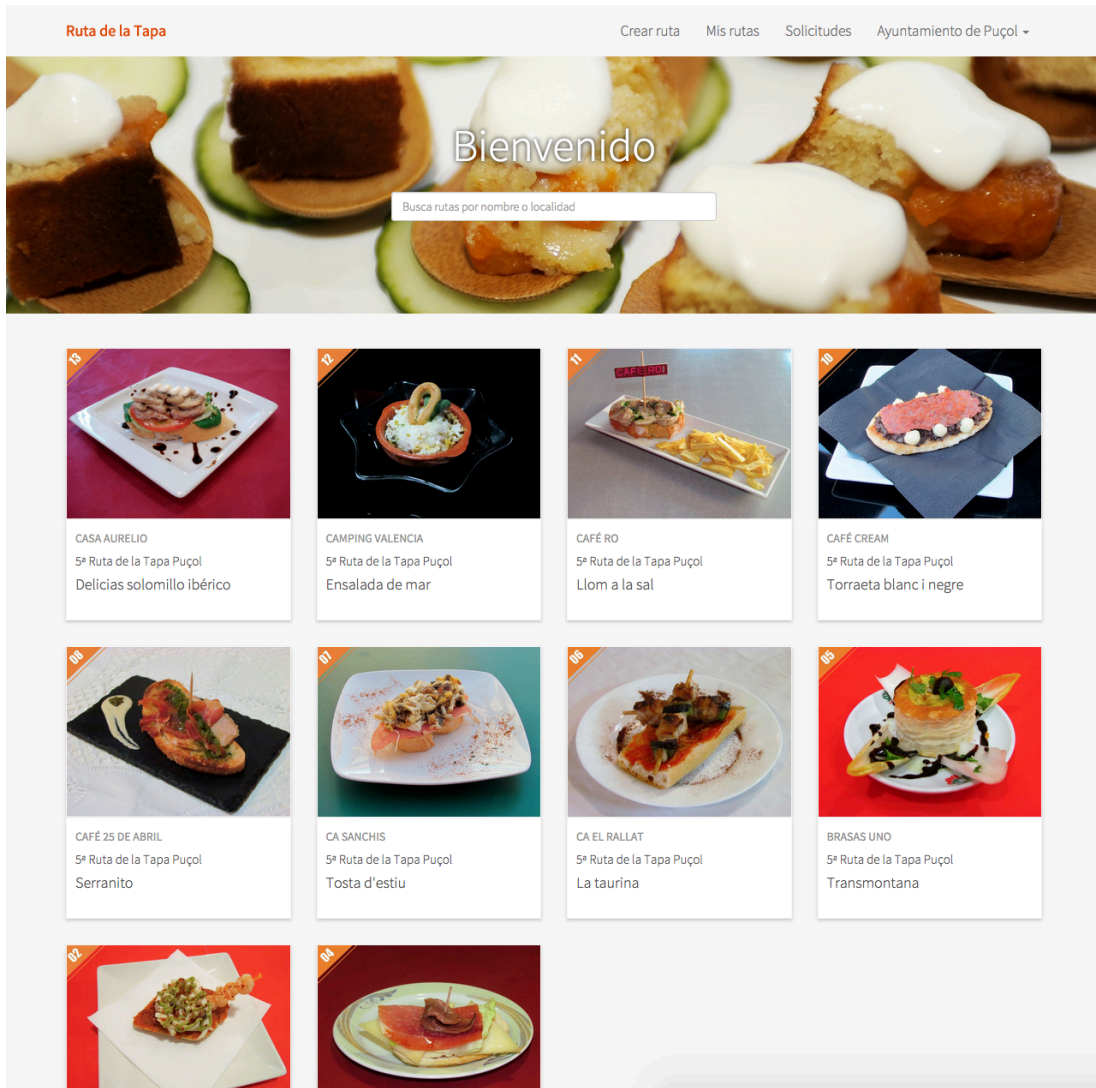


Figura 33. Captura de la aplicación en un dispositivo grande

Diseño e implementación del servidor web "Ruta de la Tapa"

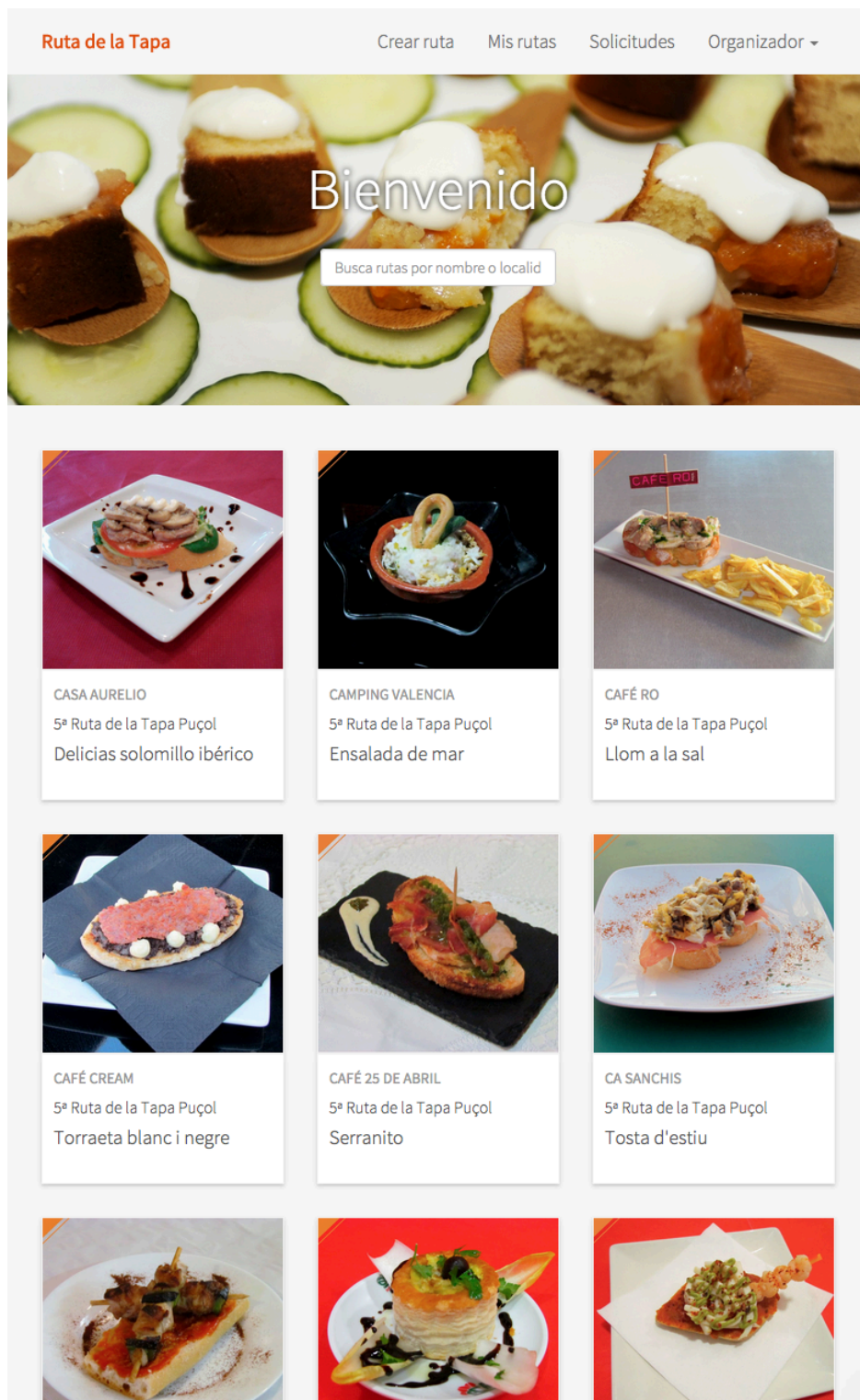


Figura 34. Captura de la aplicación en un dispositivo mediano



Figura 35. Captura de la aplicación en un dispositivo pequeño

Diseño e implementación del servidor web "Ruta de la Tapa"

Como se puede observar, los elementos se adaptan perfectamente. En resoluciones medianas se disminuye el número de elementos visibles, pasando de 4 a 3 columnas.

En cuanto el tamaño se hace demasiado pequeño, todos los elementos pasan a ocupar el ancho de la pantalla, asegurando así que se pueda leer correctamente. Además, para éstas últimas resoluciones, el menú superior se compacta para convertirse en un desplegable.

10 Conclusiones

A lo largo de este proyecto hemos desarrollado la aplicación web La ruta de la Tapa. Hemos visto mediante Google, que es un mercado en auge. Hay muchas rutas de la tapa pero la mayoría se han quedado en ámbitos locales y no han dado el paso a publicitarse en internet. Por lo que se puede decir que la necesidad existe, y de momento, nadie ha conseguido aprovecharla.

Las alternativas más válidas vienen acompañadas de la mano de alguna empresa privada, y todas ellas están restringidas a un ámbito muy local. No hay ninguna solución genérica para todas ellas.

Por eso, se propuso desarrollar una plataforma, donde se puedan encontrar los establecimientos, con los clientes de toda España. De una manera genérica de manera que puedan participar establecimientos de cualquier parte, pero con un cierto grado de personalización para que los usuarios lleguen a sentirla como suya.

Una vez siendo conscientes de la necesidad que existía y se quería cubrir, se procedió a establecer la funcionalidad de la que se iba a dotar la aplicación y los requisitos que se deberían cubrir.

Una vez definido su alcance, se establecieron unos casos de uso, pensando en los usuarios, y que acciones son las que querrían hacer en la plataforma. A partir de ellos, se podrían definir mejor las funciones a desarrollar.

Una vez desarrollados los casos de uso se diseñaron las clases que iban a conformar la aplicación de una manera más real a la implementación final. Con esta información, se hicieron patentes las necesidades tecnológicas que se tenían. Como por ejemplo, para que la información almacenada fuera siempre coherente, se necesitaría una BD de tipo relacional.

Dado que el autor tiene más experiencia desarrollando en la parte servidor, el mayor reto fue diseñar la parte visual de manera que, sabiendo que elementos deberían formar parte de ella, todos fueran accesibles, siguieran un orden lógico y fueran atractivos para el usuario.

Las mejoras visuales han sido constantes durante todo el desarrollo y ha habido grandes modificaciones. Todo gracias al continuo feedback que se ha intentado obtener de amigos y familiares que participaron en el proceso de pruebas.

La experiencia personal ha sido muy positiva. Se podría decir que ha sido el primer proyecto con el que se ha trabajado junto a otro desarrollador, en este caso, la alumna Marta Gaona, la cual era la responsable del desarrollo de la aplicación móvil.

La comunicación entre ambos ha sido constante dado que se quería una experiencia de uso unificada, y aunque la idea inicial era que ambas siguieran la misma guía de diseño, esto no ha sido posible por los ámbitos tan diferentes en las que nos movíamos. En un dispositivo móvil, muchos colores resultan llamativos, mientras que en una web, el exceso puede resultar confuso para el usuario.



Diseño e implementación del servidor web "Ruta de la Tapa"

El resultado de todo el trabajo, ha sido el de una aplicación web completa. Todas las funciones y propósitos que se tenían de ella han sido cumplidos, incluyendo una completa integración con la aplicación móvil. A pesar de ello, se podría complementar con funciones tales como avisos por correo electrónico, permitir la programación de inicio de las rutas, mayor capacidad de personalización de las rutas o incluir funciones sociales (compartir en Facebook, Twitter....).



11 Bibliografía

AJUNTAMENT DE PUÇOL, *Ruta de la Tapa Puçol*, <http://rutadelatapapuzol.blogspot.com.es/>

BEN EDMUNDS, *Ben Edmunds - Ion Auth Docs*, http://benedmunds.com/ion_auth/ [Consulta: Mayo de 2015]

BOOTSTRAP, *Bootstrap · The world's most popular mobile-first and responsive front-end framework*, <http://getbootstrap.com/> [Consulta: Mayo de 2015]

BRIAN SWEENEY, *Home · dompdf/dompdf Wiki · GitHub*, <https://github.com/dompdf/dompdf/wiki> [Consulta: Mayo de 2015]

BRITISH COLUMBIA INSTITUTE OF TECHNOLOGY, *CodeIgniter Web Framework*, <http://www.codeigniter.com/> [Consulta: Mayo de 2015]

DETAPAS, *De Tapas - Todos los bares de tapas de Sevilla*, <http://contenidosabcdesevilla.sevilla.abc.es/detapas/> [Consulta: Mayo de 2015]

DOMINIK DZIENIA, *PHP QR Code - QR code generator, an LGPL PHP library*, <http://phpqrcode.sourceforge.net/> [Consulta: Mayo de 2015]

GENERALITAT VALENCIANA, *Ruta del Tardeo en Valencia | Otro sitio realizado con WordPress*, <http://rutadeltardeovalencia.es/> [Consulta: Mayo de 2015]

GOOGLE, *Google Maps JavaScript API | Google Developers*, <https://developers.google.com/maps/documentation/javascript/> [Consulta: Mayo de 2015]

JEFF STANDEN, *What programming language is the most popular for web development? - Quora*, <http://www.quora.com/What-programming-language-is-the-most-popular-for-web-development> [Consulta: Mayo de 2015]

JOOST VAN VEEN, *CodeIgniter Best Practices - Tuts+ Course*, <http://code.tutsplus.com/courses/codeigniter-best-practices> [Consulta: Mayo de 2015]

JOSE M^a BAQUERO, *¿Cómo elegir el lenguaje de programación más adecuado para cada proyecto?*, <http://www.arsys.info/configuracion-de-programas/como-elegir-el-lenguaje-de-programacion-adecuado-para-cada-proyecto/> [Consulta: Mayo de 2015]

MAHOU SAN MIGUEL, *Estamos de tapas*, <http://www.estamosdetapas.com/> [Consulta: Mayo de 2015]

PHP GROUP, *PHP: Hypertext Preprocessor*, <http://www.php.net/> [Consulta: Mayo de 2015]

TIOBE SOFTWARE, *TIOBE Software: Tiobe Index*, <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html> [Consulta: Mayo de 2015]

