|  |
| --- |
| Proyecto final - Desarrollo de aplicaciones web |
|  |
|  |
| [Aplicación de consulta y gestión de puntos de interés en diferentes localizaciones] bg.jpg |
| **logo1.pngRubén Cancelo Rodríguez** |
|  |

|  |
| --- |
| Centro: CIFP. PONFERRADA  Tutor: Fidel Apellidos  ROUTES OF INFINITY |
|  |

Índice

[Índice de ilustraciones 4](#_Toc452411826)

[1. Estudio del problema y análisis del sistema 5](#_Toc452411827)

[1.1. Introducción 5](#_Toc452411828)

[1.2. Funciones y rendimientos deseados 5](#_Toc452411829)

[1.3. Objetivos 5](#_Toc452411830)

[2. Planteamiento y evaluación de diversas soluciones 6](#_Toc452411831)

[3. Justificación de la solución elegida 8](#_Toc452411832)

[3.1. Estructura de la web 8](#_Toc452411833)

[3.2. Estructura de la base de datos 14](#_Toc452411834)

[3.3. Estructura y programación 16](#_Toc452411835)

[3.4. Funcionamiento de la aplicación 17](#_Toc452411836)

[4. Modelado de la solución 24](#_Toc452411837)

[4.1. Recursos humanos 24](#_Toc452411838)

[4.2. Recursos hardware 25](#_Toc452411839)

[4.3. Recursos software 25](#_Toc452411840)

[5. Planificación temporal 26](#_Toc452411841)

[6. Costes del proyecto 27](#_Toc452411842)

[7. Conclusiones finales 27](#_Toc452411843)

[7.1. Grado de cumplimiento de los objetivos fijados 27](#_Toc452411844)

[7.2. Propuesta de modificaciones o ampliaciones futuras del sistema implementado 28](#_Toc452411845)

[8. Bibliografía 28](#_Toc452411846)

# Índice de ilustraciones

[Index (Boceto inicial) 3‑1 7](#_Toc452409497)

[Discover (Boceto inicial) 3‑2 8](#_Toc452409498)

[Show (Boceto inicial) 3‑3 9](#_Toc452409499)

[Create (Boceto inicial) 3‑4 10](file:///C:\Users\Cancelo\Desktop\Proyecto.docx#_Toc452409500)

[Login (Boceto inicial) 3‑5 11](#_Toc452409501)

[Profile (Boceto inicial) 3‑6 12](#_Toc452409502)

[Diagrama Base de datos 3‑7 14](#_Toc452409503)

[Diagrama de casos de uso 3‑8 14](#_Toc452409504)

[Index (Resultado final) 3‑9 16](#_Toc452409505)

[Discover (Resultado final) 3‑10 16](#_Toc452409506)

[Show (Resultado final) 3‑11 17](#_Toc452409507)

[Login (Resultado final) 3‑12 18](#_Toc452409508)

[Perfil de usuario sin rutas (Resultado final) 3‑13 18](#_Toc452409509)

[Perfil de usuario con rutas creadas (Resultado fina)l 3‑14 19](#_Toc452409510)

[Crear, introducción de ubicaciones (Resultado final) 3‑15 20](#_Toc452409511)

[Crear, ventana de finalizar (Resultado final) 3‑16 20](#_Toc452409512)

# Estudio del problema y análisis del sistema

## Introducción

*Routes of infinity* es una aplicación web que permite la consulta, creación y posterior gestión de la rutas creadas por el usuario.

La aplicación está destinada a personas que les guste viajar y conocer lugares nuevos, entrando en *Routes of infinity* podrán ver las rutas compartidas por otros usuarios, teniendo una referencia a la hora de moverse por la zona en la que se encuentre, así como posibles eventos de bares, museos, etc.

## Funciones y rendimientos deseados

Lo que se busca con esta aplicación web es crear un lugar donde la gente comparta los lugares o puntos de interés que más le han gustado de una ciudad, así como los locales donde mejor se lo han pasado o más le han gustado. Recomendado al resto de usuarios sus rutas favoritas.

*Routes of infinit*y tendrá un apariencia sencilla, inspirada por el estilo *Material Design*. Se buscará la facilidad de uso y la simplicidad a la hora de consultar, crear y gestionar las diferentes rutas.

Un usuario no registrado podrá acceder a la aplicación web y consultar rutas creadas por otros usuarios. Se han establecido tres tipos de rutas por defecto: *Ocio*, *Cultural* y *Otros*.  
Un usuario registrado podrá acceder a la aplicación web y consultar rutas además de crear las suyas propias. Dispondrá de un perfil donde gestionarlas.

## Objetivos

Uno de los objetivos de la realización de esta aplicación web ha sido profundizar más en las APIs de *Google Maps*, además de crear un proyecto real, más allá de lo realizado en clase. Se busca crear una aplicación web robusta y segura a nivel de programación, y que en cuanto a diseño pueda adaptarse a las diferentes resoluciones de los dispositivos que existen hoy en día.

**Objetivos planteados para la aplicación web:**

Un usuario no registrado solamente podrá consultar rutas. No podrá crearlas, no podrá votarlas y no tendrá acceso a su perfil, ya que no existe.

Un usuario registrado tendrá acceso a todo el contenido de la aplicación web, es decir, consultar, crear, gestionar, votar. En su perfil podrá eliminar sus rutas, así como editarlas o eliminarlas. También podrá modificar sus datos personales y añadir una imagen de perfil. Las rutas que haya votado aparecerán en su perfil, de esta forma podrá acceder rápidamente a ellas. Al no ser el autor de ellas no podrá eliminarlas ni editarlas.

Se busca una forma sencilla de crear rutas sobre un mapa de Google Maps.

Las rutas se podrán ordenar según los votos recibidos, fecha y tamaño de la ruta, es decir, cuantas ubicaciones tiene esa ruta. Se podrán aplicar otros filtros como el tipo de ruta, y la ordenación ascendente o descendente.

# Planteamiento y evaluación de diversas soluciones

Para conseguir los objetivos se han planteado las siguientes soluciones, intentando que sean lo más simples y fiables en su implementación.

**Uso de APIs**

Utilización de las APIs de Google Maps. Es una solución efectiva hacer uso de estas APIs en el proyecto, ya que es necesario un mapa respaldado por una gran base de datos, bien testeado y sólido.

Se utilizará *Google Maps Javascript API* para la creación de rutas, pudiendo crear marcadores (*markers*) donde el usuario desee. También se utilizará a la hora de visualizar las rutas guardadas, proporcionándole las coordenadas de los *markers* y la información que el usuario introduzca.

Se utilizará *Google Maps Places API* para buscar en el mapa no sólo ciudades, pueblos, etc., sino también para poder buscar locales y establecimientos, como bares, restaurantes, museos, etc.

Se utilizará *Google Static Maps API* para mostrar imágenes estáticas de los mapas, estas imágenes se utilizarán en una vista previa de la ruta, antes de que el usuario acceda a toda la información de la misma. Crea el mapa según parámetros de URL enviados a través de una solicitud HTTPS estándar. Muestra el mapa como una imagen.

Estas APIs tiene limitaciones de peticiones por día en la versión utilizada en Routes of infinity. Con la posibilidad de aumentar sus capacidades en un futuro.

**Diseño**

Se utilizará un framework CSS, en este caso será Materializecss.

Es un framework relativamente nuevo que proporciona rapidez a la hora de crear el diseño de la aplicación web. Sigue los patrones establecidos por Material Design y proporciona componentes acorde a este estilo.

Aunque se utilice un framework para el estilo de la aplicación web hace falta crear un estilo CSS propio para adaptar los componentes proporcionados al resultado deseado. Nos proporcionará un diseño atractivo y con una implementación relativamente rápida.

La interfaz deberá ser clara e intuitiva, facilitando al usuario el aprendizaje de la aplicación web. Se mostrará en todo momento en que parte de la web se encuentra, para esto se utilizará un árbol de navegación o similar.

**Base de datos**

Se han planteado varias posibilidades a la hora de elegir la base de datos, en primer lugar MySql y en segundo lugar MariaDB. La duda entre estas dos bases de datos es debido a las licencias que está aplicando Oracle a MySQL, viniendo a día de hoy MariaDB en el paquete de instalación de XAMPP.

La elección final ha sido utilizar MySQL por el conocimiento sobre está y por el uso que se ha hecho de ella durante el curso del ciclo formativo de Desarrollo de aplicaciones web.

Se utilizará PDO para acceder a la base de datos en PHP, proporcionando una capa de abstracción de acceso a datos, lo que significa que, independientemente de la base de datos que se esté utilizando, se emplean las mismas funciones para realizar consultas y obtener datos. Esto nos permitirá cambiar de base de datos en un futuro sin modificar o modificando ligeramente el código.

**Lenguajes de programación**

Se utilizará HTML5 y CSS3 para crear la estructura de la web y adaptarla al diseño deseado.

Del lado del cliente utilizaremos JavaScript, para desarrollar ciertas funciones de los formularios, temas de validación y avisos al usuario. Los mapas, así como las APIs de Google también se implementaran con JavaScript. Se hará uso de jQuery para inicializar ciertos componentes del framework Materializecss.

Para el sistema de votos se utilizará AJAX, se implementará de una forma muy básica.

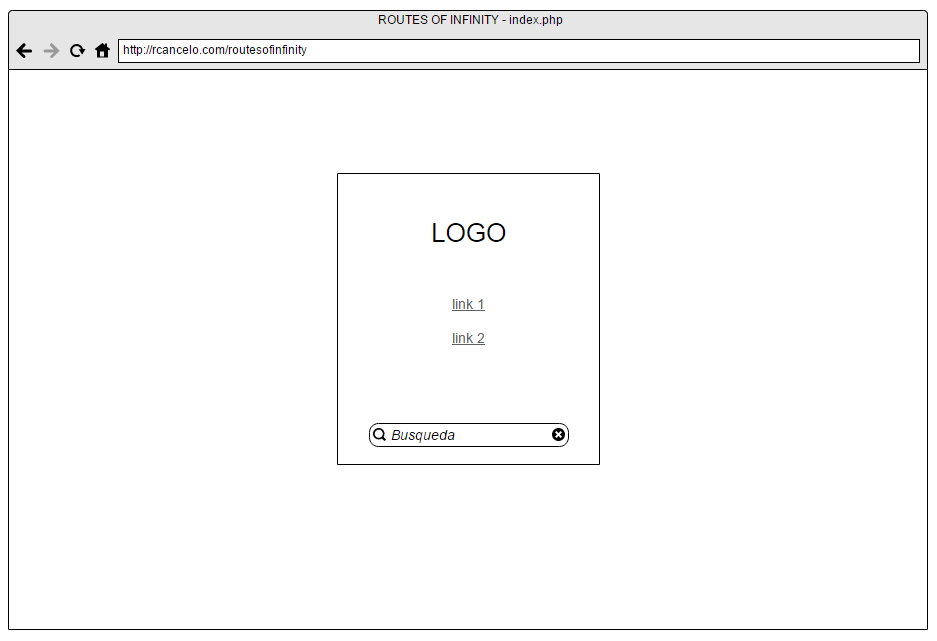
Del lado del servidor se usará PHP, se intentará crear una estructura propia separada por capas y clases, en la que el acceso a la base de datos sea independiente del resto de funciones. En este punto hubiera sido muy interesante el uso de algún framework como CodeIgniter, pero debido a la falta de tiempo no ha sido posible dedicarle el tiempo necesario al aprendizaje de una herramienta de este tipo.

# Justificación de la solución elegida

## Estructura de la web

Se buscará la simplicidad de uso y la claridad. Exceptuando el index, todas las páginas constarán de una barra de navegación, una barra indicando la situación actual en la que se encuentra el usuario y un pie de página con información relevante de la aplicación web.

**Index.php**

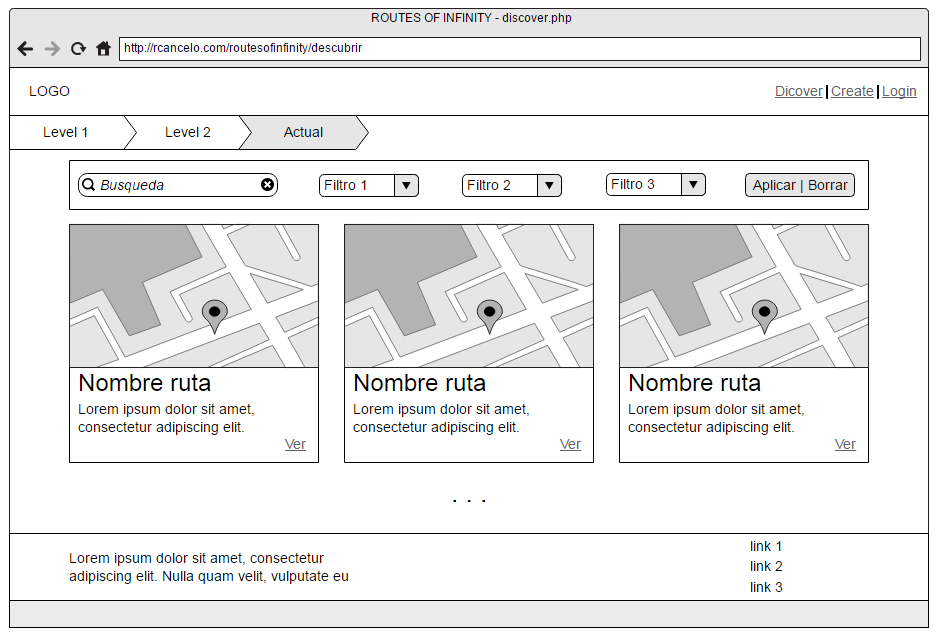


Index (Boceto inicial) 3‑1

Será la página principal de la aplicación web, solo tendrá el logo y un buscador. Se podrá acceder al resto de la aplicación utilizando el buscador para visualizar las rutas que coincidan con la búsqueda o saltándose este paso y visualizar todas las rutas.

Me parece una estructura sencilla, el usuario puede buscar lo que desee o directamente pasar a ver todas las rutas disponibles. En futuras versiones de la aplicación se podría incluir la opción de geolocalizar al usuario y mostrarle las rutas más cercanas.

**Discover.php**



Discover (Boceto inicial) 3‑2

Será la página donde se mostraran todas las rutas de una forma rápida y si entrar en detalle de ellas. A parte de los elementos comunes con otras páginas, tendrá un buscador y una serie de filtros que se podrán aplicar o limpiar con dos botones situados en la misma barra de búsqueda.

La estructura del contenedor de la ruta estará compuesta por una imagen de la ruta, el nombre de la misma y demás información introducida durante la creación. Se añadirá un icono y un color característico en función del tipo que sea. Icono de copa y color azul: *Ocio*, icono de museo y color verde: *Cultural* e icono de mapa y color naranja: *Otros.* De esta forma se podrá saber rápidamente que tipo de ruta es.

Haciendo uso de Materializecss, al hacer click izquierdo en la ruta deseada, el contenedor se transformará y mostrará más información. Esta función se podría desarrollar en JavaScript puro desde cero, pero de esta forma sería más lento el tiempo empleado en finalizar la aplicación web. Gracias al framework utilizado estas funciones son relativamente fáciles de implementar y se ahorra tiempo en la creación del proyecto.

Este contenedor tendrá un enlace, que nos llevará a la página donde podremos ver la ruta con más detalle y explorar las ubicaciones introducidas en el mapa.

Se ha buscado presentar la información más relevante de las rutas almacenadas. Con las posibilidad de aplicar ciertos filtros que podrían expandirse en futuras versiones.

**Show.php**

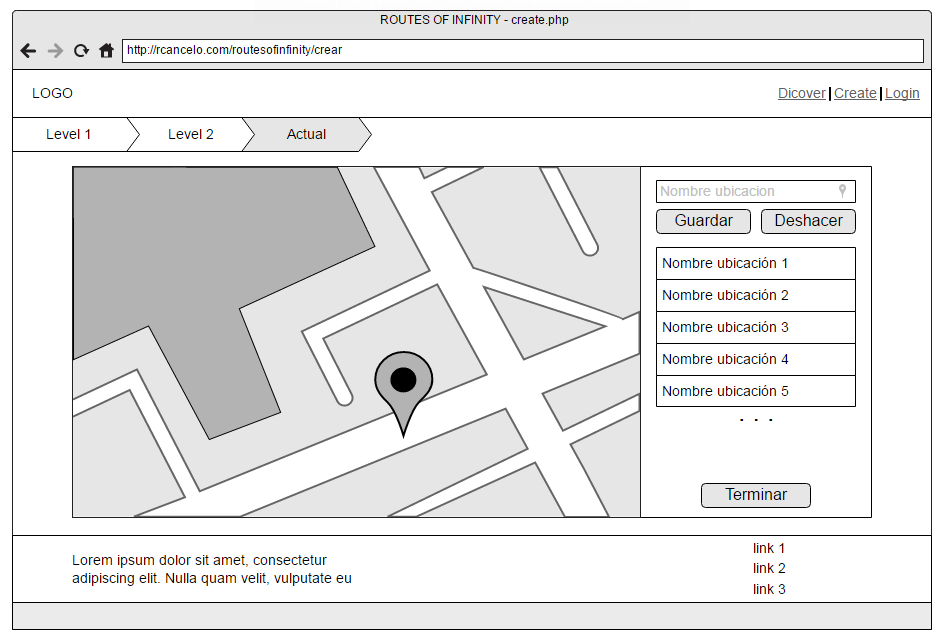


Show (Boceto inicial) 3‑3

Esta página será la que muestre toda la información de la ruta. El contenedor en el que se encuentra estará dividido en dos, ocupando la parte izquierda un 70% y la derecha un 30% del mismo. En la derecha se mostrará el mapa con las ubicaciones guardadas y el usuario podrá interactuar con él, haciendo zoom, cambiando el tipo de mapa o utilizando Google Street View.

En la derecha, se mostrará el título, el autor, la fecha, el tipo y la valoración. En el planteamiento de la aplicación se pretendía un sistema de valoración basado en la puntuación media que le dieran los usuarios, pero finalmente se ha aplicado un sistema basado en votos. Un click = 1 voto.

**Create.php**



Create (Boceto inicial) 3‑4

En esta sección de la aplicación web los usuarios que se hayan registrado podrán crear las rutas. Constará de los elementos comunes y un contenedor principal. Se repetirá la estructura de la página show.php, un mapa de Google Maps con el que el usuario podrá interactuar y un panel derecho para introducir los nombre de las ubicaciones seleccionadas en el mapa. Habrá tres botones:

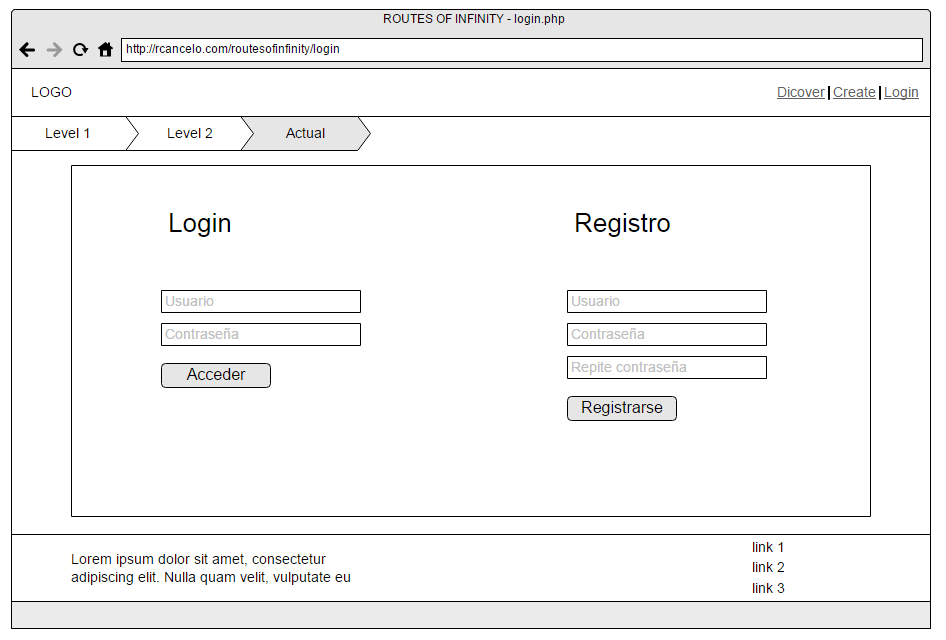
Guardar, almacenará la ubicación marcada en el mapa con su título.

Deshacer, eliminará la última ubicación guardada.

Continuar, abrirá una ventana donde el usuario podrá introducir el resto de información asociado a la ruta. Nombre de la ruta, ciudad o localización, seleccionar el tipo de ruta y una descripción de la misma.

En la ventana de finalizar habrá un botón para almacenar la ruta. Se ha optado por abrir una ventana por encima del contenido para no ocupar mucho espacio que habría que quitar al mapa. De esta forma el mapa queda con un tamaño bueno para interactuar con el de forma cómoda.

**Login.php**



Login (Boceto inicial) 3‑5

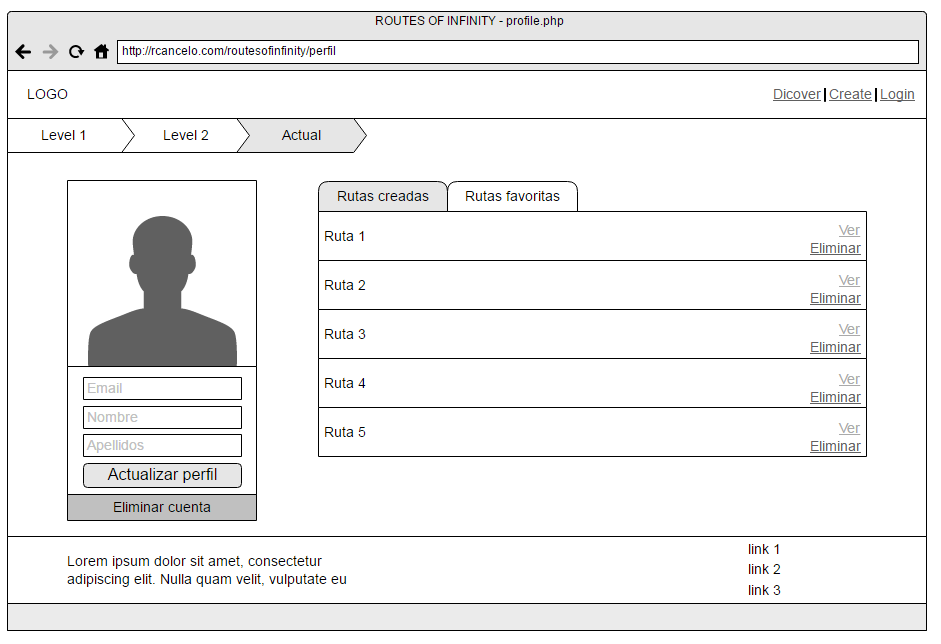
Será la sección donde el usuario se registrará y podrá iniciar sesión. Constará de un contenedor principal y con los elementos comunes, como son el header, el footer y la barra que indica la situación actual dentro de la aplicación web.

En la izquierda aparecerá el formulario de login y en la derecha un formulario de registro.

Es una sección simple y sencilla, cumple su función que es el registro y acceso a la aplicación web.

En un planteamiento inicial está página no existiría como tal, sino que sería un popup o un modal que aparecería al hacer click izquierdo sobre login. Al iniciar sesión se crearan las variablse de SESSION y se hará el el seguimiento del usuario, podrá acceder a u su perfil.

**Profile.php**



Profile (Boceto inicial) 3‑6

Una vez el usuario haya iniciado sesión podrá acceder a su perfil. En esta página podrá gestionar sus rutas y modificar su cuenta.

Constará de los elementos comunes y un contenedor principal. Este contenedor tendrá a la izquierda una imagen de perfil de usuario que en futuras versiones podrá cambiar. Podrá modificar sus datos, también en futuras versiones. Y podrá eliminar su cuenta.

Una vez el usuario haya iniciado sesión podrá acceder a su perfil. En esta página podrá gestionar sus rutas y modificar su cuenta.

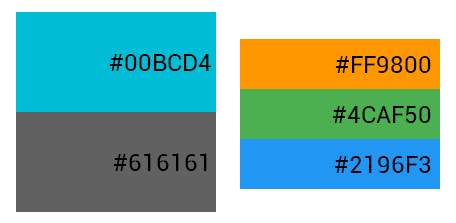
Es una estructura sencilla que muestra la mayor información posible de un solo vistazo.

**Colores**

Los colores utilizados en la aplicación web serán los siguientes:

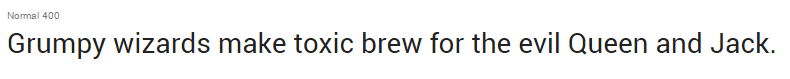
El color principal será el cyan, con código hexadecimal: #00BCD4

Se utilizarán como colores secundarios varios tipos de grises.

Habrá algunos elementos que lleven un color específico como los iconos de los tipos de rutas, los iconos de ver ruta y eliminar ruta desde el perfil de suario. 

**Tipografía**

La tipografía utilizada en toda la aplicación web será Roboto en su versión Normal, por su facilidad de lectura y su simplicidad. Dándole a la aplicación web un acabado más serio.



## Estructura de la base de datos

El fundamento principal de la base de datos de la aplicación web es la automatización y la gestión de las rutas de los usuarios, controlando los usuarios registrados, las rutas que crean y como las visualizan los demás usuarios.

Para ello se ha creado únicamente tres tablas:

La tabla usuario, controlará los datos de registro de los usuarios en la aplicación. Estará formada por la siguientes columnas: *id, nombre, password, fecha\_registro, vip y bloqueado.*

Los campos *vip* y *bloqueado* no se dan uso en esta versión de la aplicación pero la idea es tener diferentes tipos de usuarios. Si un usuario es vip sus rutas aparecerán promocionadas respecto a las otras, es útil para bares o restaurantes que quieran promocionar una ruta gastronómica o rutas de vinos, etc. Deberán pagar una cuota.

El campo bloqueado poda tener los valores 0, no bloqueado y 1, bloqueado. Se utilizará para usuarios molesto que utilicen la aplicación para fines que no coinciden con la temática de la web. Estos usuarios bloqueados no tendrán acceso a la aplicación.

La tabla ruta, contendrá información de la ruta creada, como el tamaño de ubicaciones guardadas, la fecha de creación, ciudad, votos. Será útil para aplicar diferentes filtros para buscar con más facilidad. Las columnas dentro de esta tabla serán: *id, nombre, ciudad, descripción, tipo, ubicaciones, id\_usuario, fecha\_creacion, votos, tamaño*.

Y por último la tabla control\_votos, solo tendrá dos columnas: *id\_uusario y id\_ruta*. Se utilizará para que un usuario registrado no pueda volver a votar la misma ruta.

Estarán relacionadas por la id de cada una. Tabla ruta tendrá una clave foránea relacionada con la clave primaria de la tabla usuario. Y la tabla de control de votos tendrá la id de la tabla usuario y la id de la tabal ruta.

De esta forma cada ruta está asociada a un usuario y cada voto de un usuario estará relacionado con una ruta.

Estructura de la base de datos:

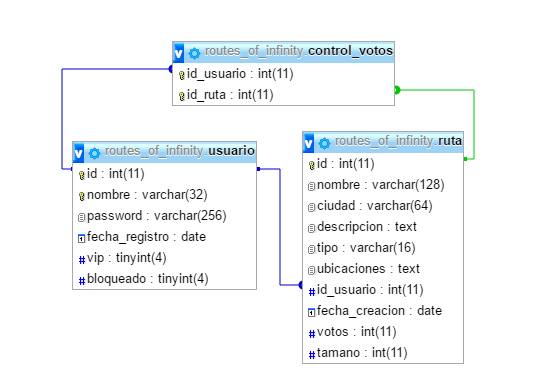


Diagrama Base de datos 3‑7

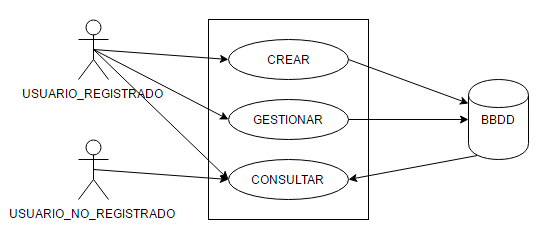
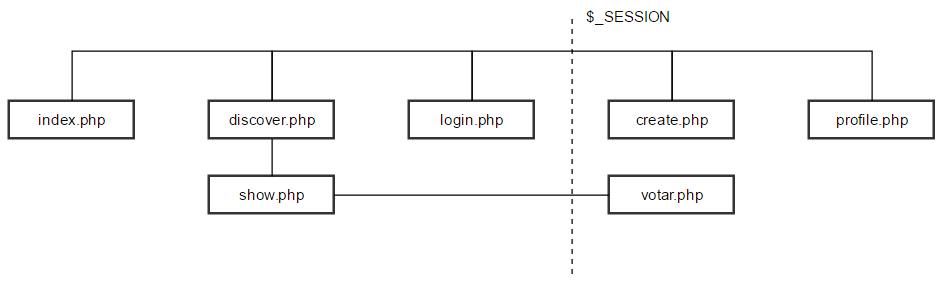
Diagrama de casos de uso:

Diagrama de casos de uso 3‑8

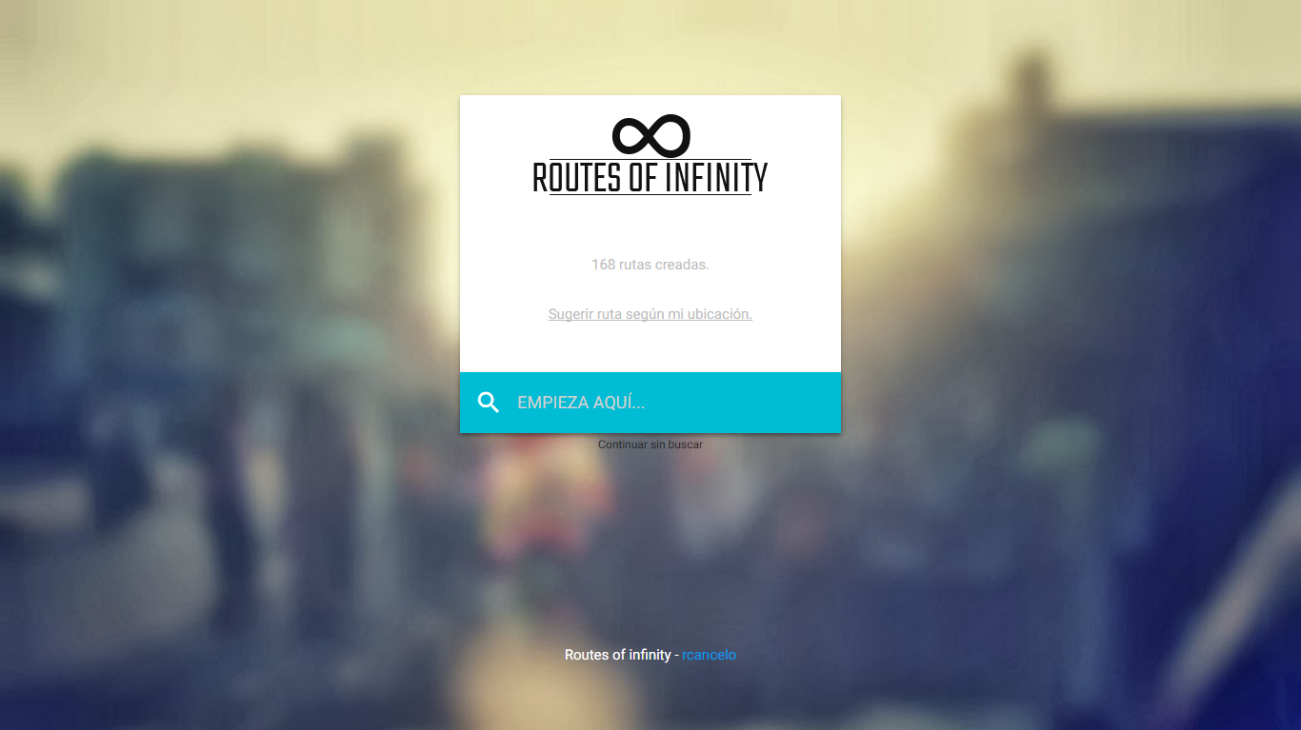
## Estructura y programación



## Funcionamiento de la aplicación

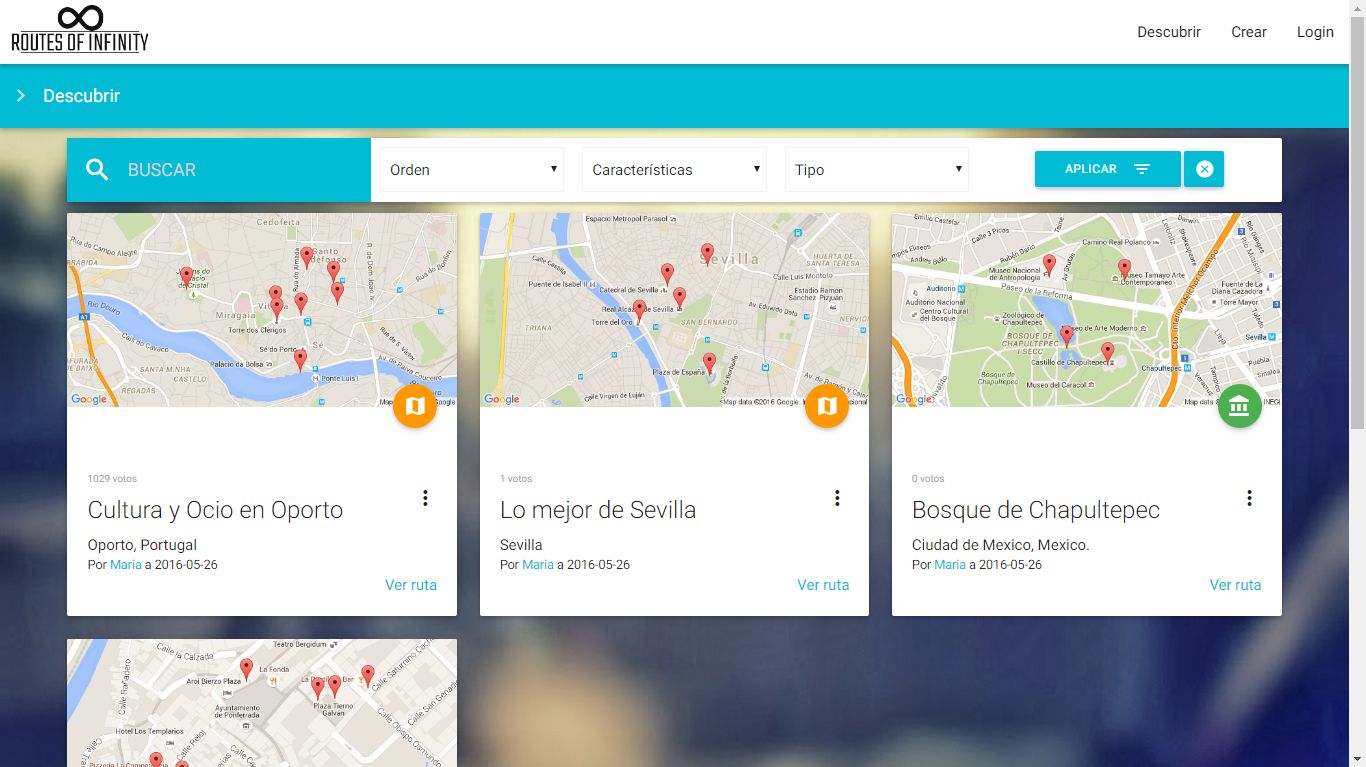
A continuación se mostrará el resultado actual de la aplicación, puede presentar alguna modificación con la versión final. También se mostrará el funcionamiento de la misma:

En la página principal se podrá buscar un lugar o una descripción para acceder rápidamente a la lista de todas las rutas con un filtro de búsqueda ya aplicado. Este paso se podrá saltar y acceder directamente.



Index (Resultado final) 3‑9

Una vez entrado en la siguiente página, aparecerá un listado con todas las rutas, bien con el filtradas por la búsqueda del usuario o bien ordenadas por fecha. En esta sección se podrá aplicar filtros para encontrar de forma más precisa la ruta buscada. Las rutas mostraran la máxima información posible.

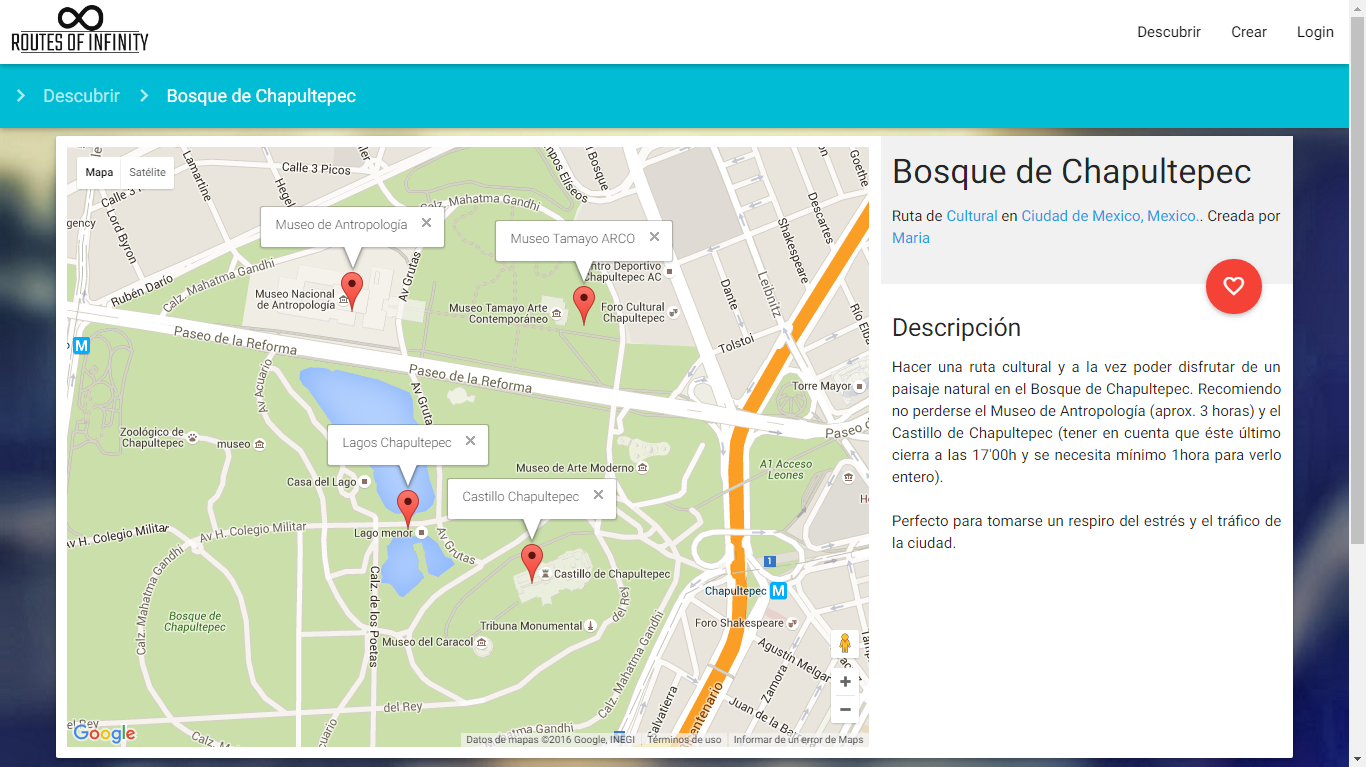


Discover (Resultado final) 3‑10

Si hacemos click izquierdo la ruta el contenedor hará un efecto de volteo y mostrará la descripción de la ruta. Si hacemos click izquierdo en ver ruta accederemos a la siguiente página, en donde se verá toda la información de la ruta, así como las ubicaciones sobre el mapa que es lo principal de la aplicación.

En esta página veremos el título, autor, lugar, descripción, tipo y un botón de un corazón que servirá para darle un voto a la ruta. Solo se podrá votar si el usuario está registrado. Además se mostrará un tooltip con el número de votos totales.

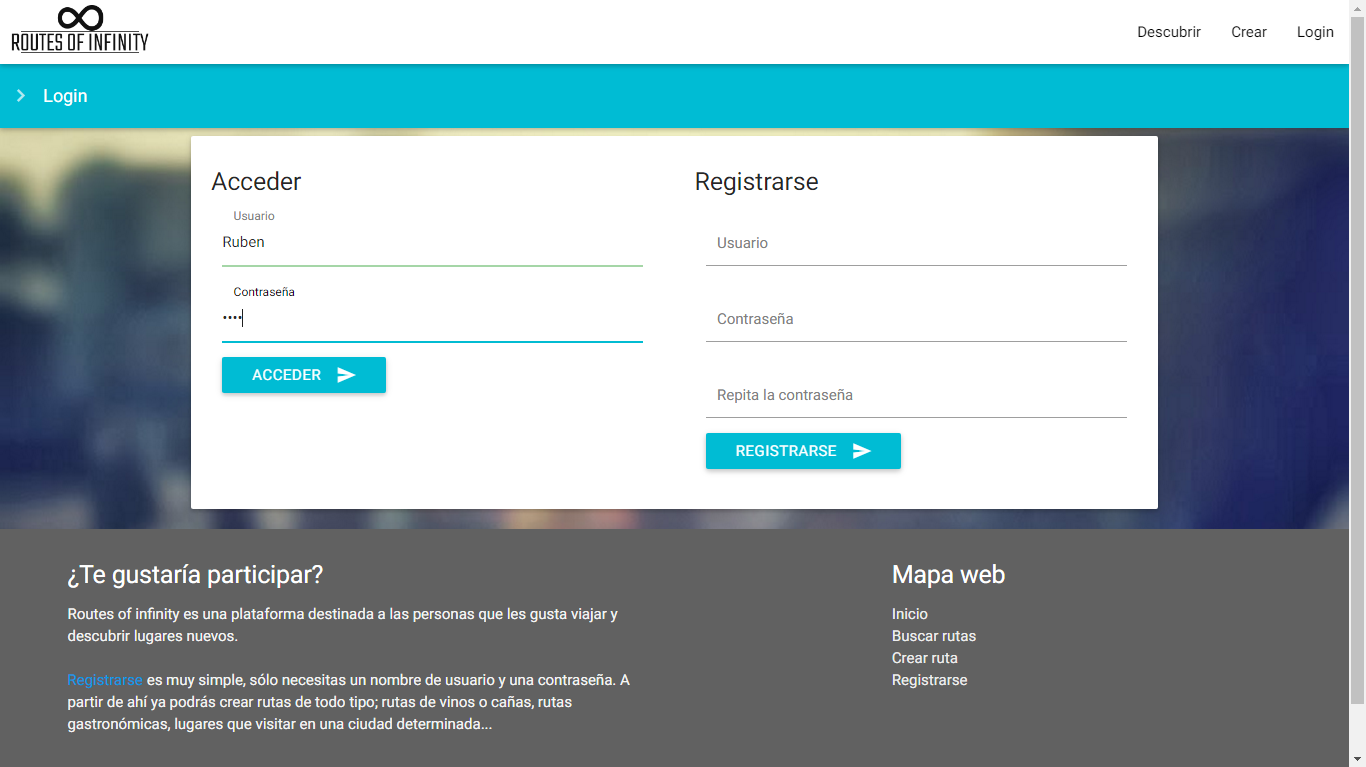
Se podrá interactuar con el mapa de manera normal, como se usa un mapa de Google maps. Si el usuario lo permite y comparte su posición la aplicación mostrará en el mapa donde se encuentra. El sistema de geolocalización no funciona en todos los navegadores.



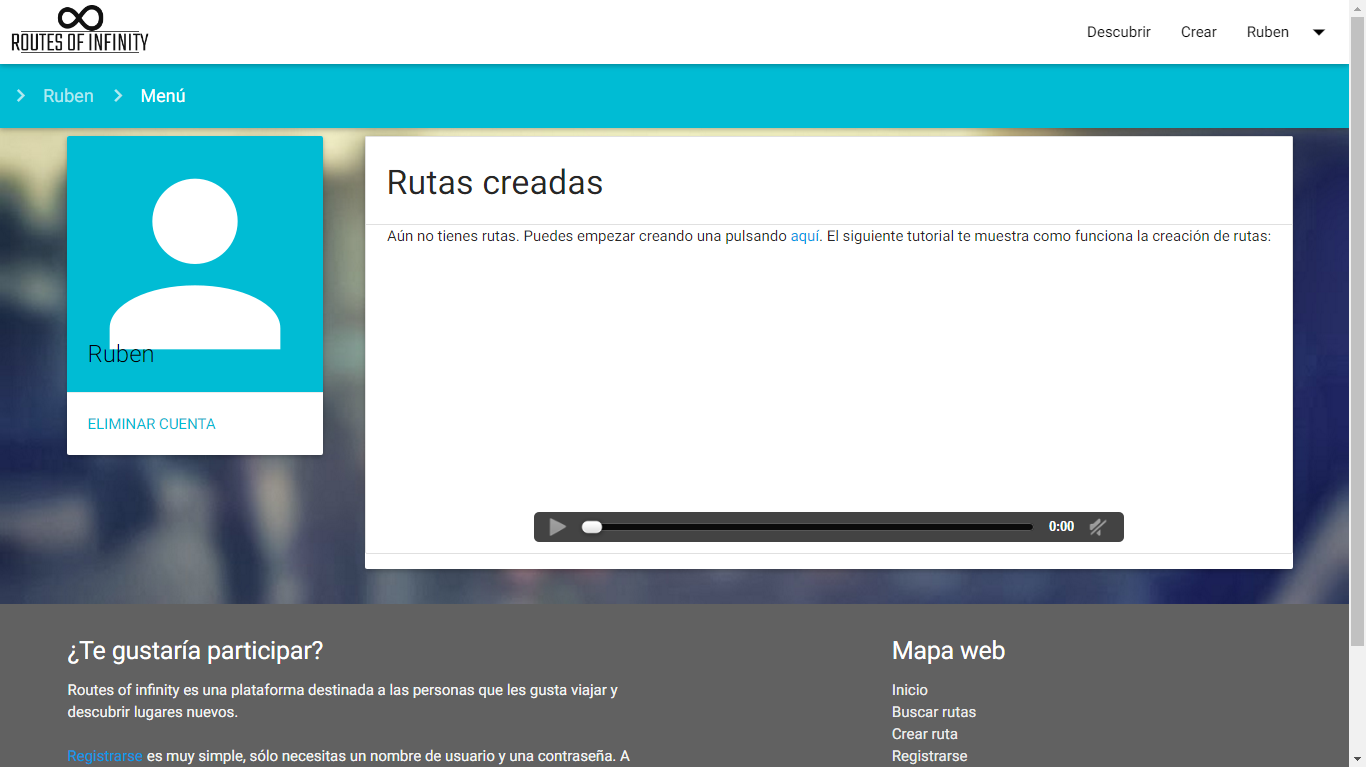
Show (Resultado final) 3‑11

El acceso a la aplicación y registro será desde la página de login. Dos formularios compondrán esta página.

A esta página se redireccionará al usuario si quisiera acceder a la sección de crear o otra página que requiera que el usuario se autentifique.

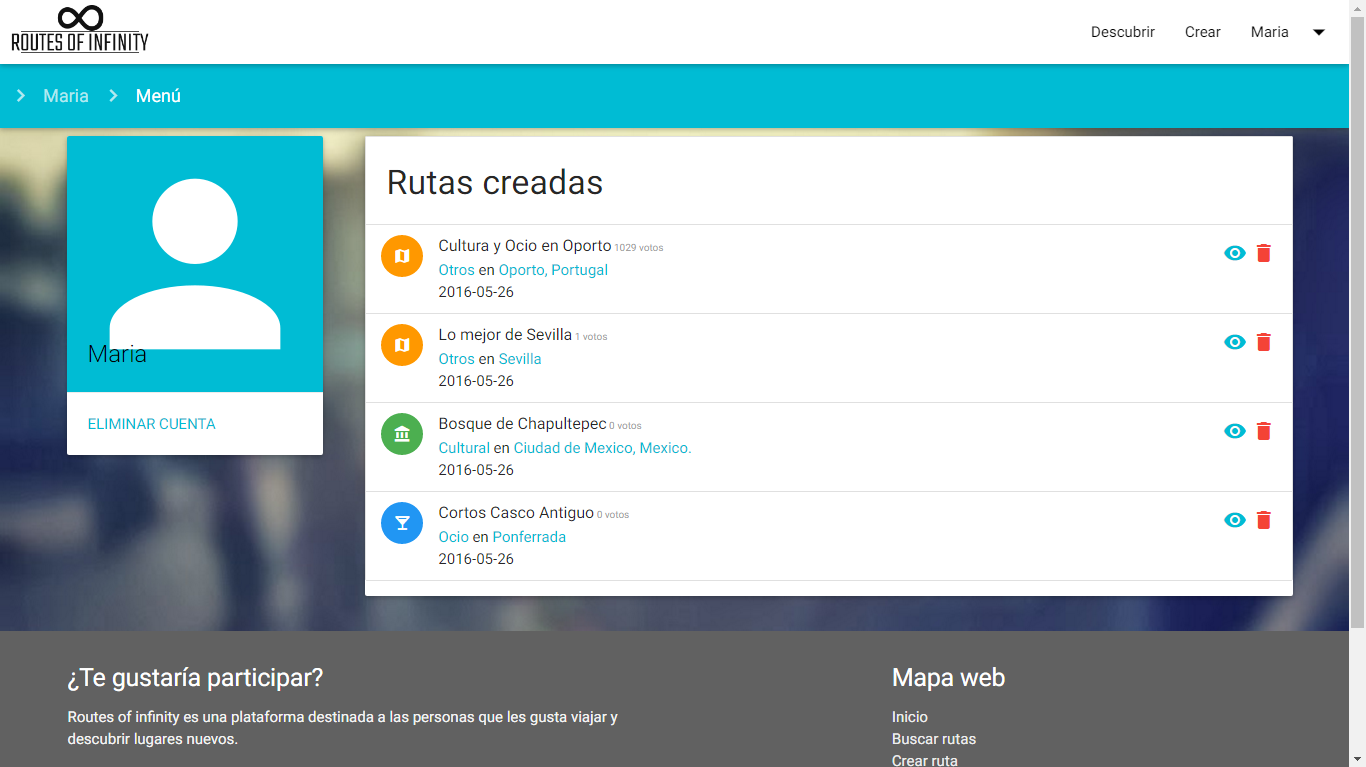


Login (Resultado final) 3‑12

Una vez registrado podrá crear rutas y acceder a su perfil. A continuación se muestra el perfil de usuario o menú de usuario. La primera imagen es de un usuario recién registrado al que se le lleva a su perfil automáticamente. Hasta que no cree alguna ruta le aparecerá un mensaje informativo indicándole que ya puede crear rutas y un vídeo tutorial de cómo sería el proceso de creación. Esto debido al trabajo que supone no está implementado en esta versión.

Perfil de usuario sin rutas (Resultado final) 3‑13

La siguiente imagen es de un usuario que ya ha creado varias rutas. El mensaje informativo no aparece y se le muestran todas sus rutas creadas hasta la fecha. Podrá ver los votos que tiene cada ruta, así como eliminaras, icono de la papelera, o acceder rápidamente a ellas desde el icono del ojo.

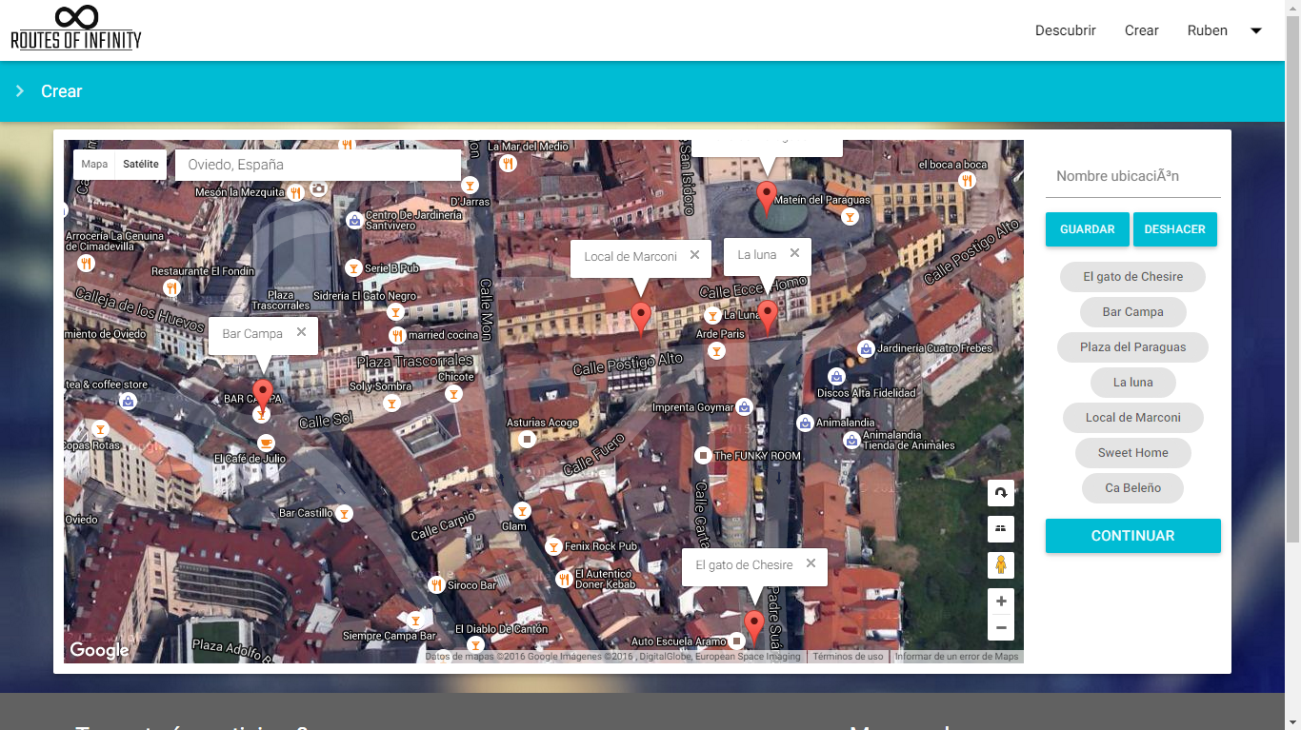


Perfil de usuario con rutas creadas (Resultado fina)l 3‑14

La última imagen corresponde a la página de creación, aquí se podría buscar manualmente en el mapa las ubicaciones deseadas o bien utilizar el buscador situado en la esquina superior izquierda del mapa.

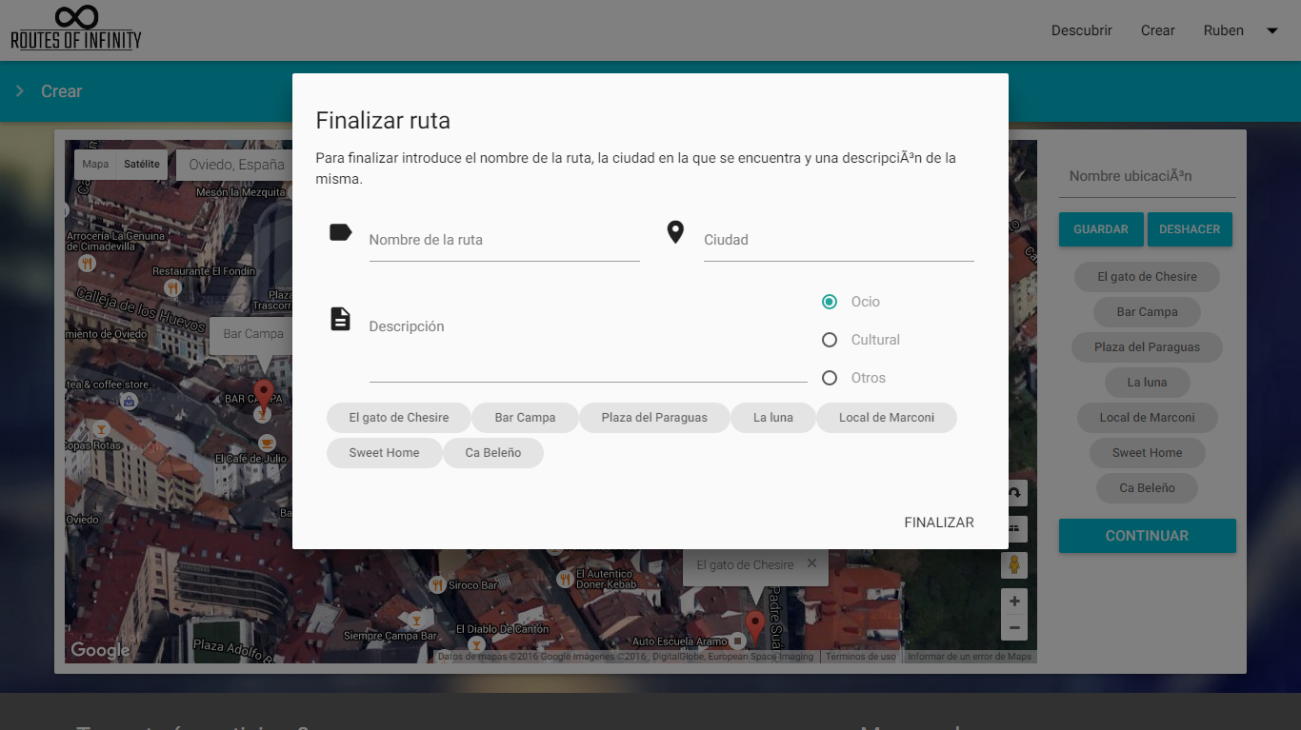
En el momento que se seleccione una ubicación se habilitará el cuadro de texto para introducir un nombre o título. Se pulsará en guardar y esa ubicación quedará fija en el mapa y almacenada. Si pulsa deshacer se eliminará la última ubicación almacenada. Se intentará guiar al usuario con una serie de animaciones tanto de los markers en el mapa, como en el panel derecho de que la ubicación ha sido almacenada.

Una vez introducidas todas las ubicaciones y sus respectivos nombres, el usuario deberá hacer click izquierdo en continuar para pasar al siguiente paso de la creación de rutas.



Crear, introducción de ubicaciones (Resultado final) 3‑15

Se le abrirá un panel flotante para introducir el resto de información de la ruta. Deberá rellenar los campos de este panel y pulsar finalizar.

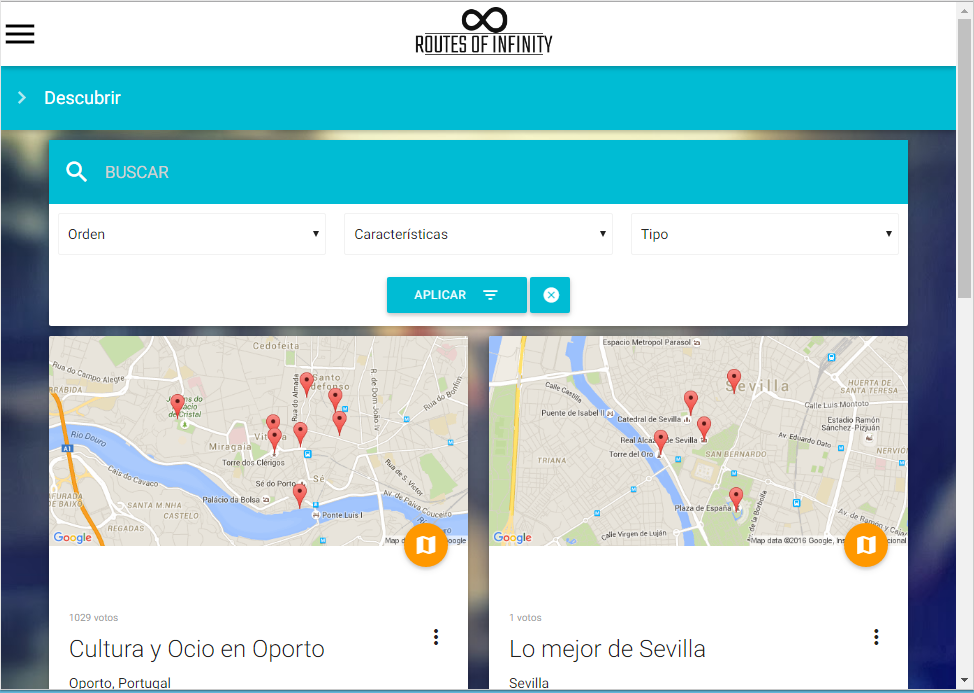


Crear, ventana de finalizar (Resultado final) 3‑16

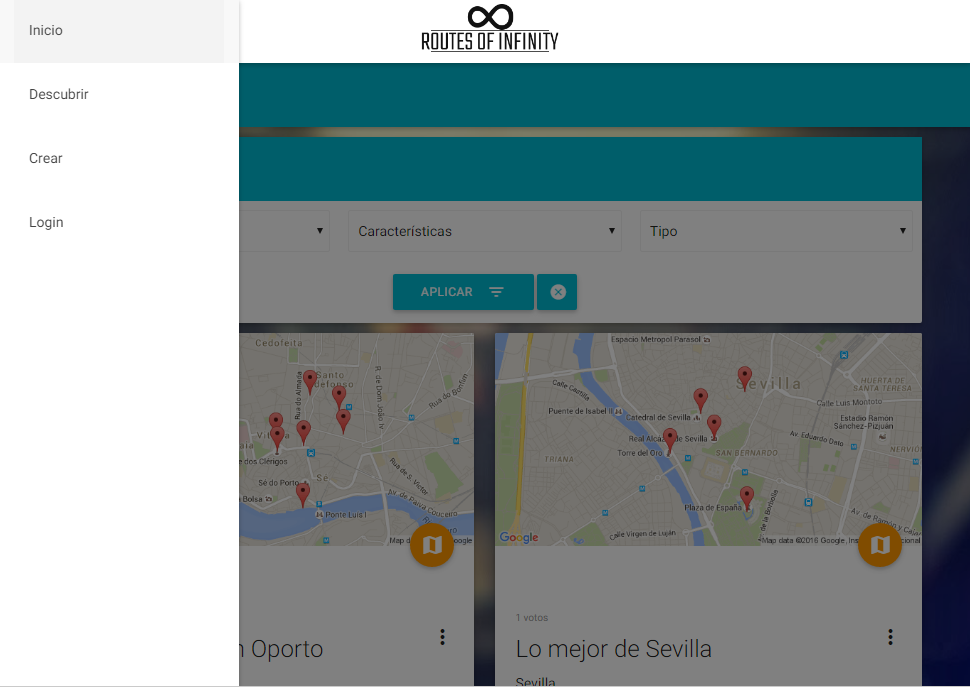
Se ha intentado que fuera lo más intuitivo posible, para ello se modifica el árbol DOM y se crean nuevos elementos en función de que el usuario guarde nuevas ubicaciones o deshaga su última acción.

Una vez pulsado finalizar y si los campos están cubiertos, la ruta se almacenará en la base de datos y será visible para todos los usuarios.

La aplicación tendrá un diseño responsive, adaptándose y modificando su estructura en función de la resolución del dispositivo.



El menú horizontal desaparece en resoluciones menores y aparece un icono que al pulsarlo muestra las diferentes secciones de la aplicación web. Al igual que la barra de navegación este cambiará si es usuario ha iniciado sesión.



# Modelado de la solución

## Recursos humanos

Para la realización de este proyecto se necesitarán:

Un diseñador gráfico encargado de realizar el diseño de la aplicación web, eligiendo la paleta de colores, la tipografía a utilizar y la posición de los diferentes elementos de la web.

Un programador especializado en diseño front-end, con conocimientos de HTML5, CSS3, JavaScript y diseño adaptativo. Sería e encargado de toda la programación en el lado cliente.

Un programador back-end, centrado en lenguajes del lado servidor como PHP, servidores y conocimiento de bases de datos relacionales.

Se podría incluir una persona que “mueva” la web por redes sociales, dándole publicidad y haciendo que se conozca de forma poco intrusiva. Con conocimientos sobre marketing y publicidad.

## Recursos hardware

Serán necesarios los siguientes elementos para el alojamiento y puesta en marcha de la aplicación web:

Un hosting que tenga soporte para PHP 5.6 como mínimo, con una base de datos MySql y con bastante ancho de banda para poder dar soporte a todas las peticiones.

También se necesitarán ordenadores para los desarrolladores, con un sistema operativo relativamente actual y con potencia suficiente para soportar el software utilizado con fluidez.

## Recursos software

Para el desarrollo de la aplicación web se utilizará el siguiente software:

* **Sublime Text 2**: [editor de texto](https://es.wikipedia.org/wiki/Editor_de_texto) y [editor de código fuente](https://es.wikipedia.org/wiki/Editor_de_c%C3%B3digo_fuente) está escrito en [C++](https://es.wikipedia.org/wiki/C%2B%2B) y [Python](https://es.wikipedia.org/wiki/Python) para los plugins. Desarrollado originalmente como una extensión de [Vim](https://es.wikipedia.org/wiki/Vim" \o "Vim), con el tiempo fue creando una identidad propia, por esto aún conserva un modo de edición tipo [vi](https://es.wikipedia.org/wiki/Vi) llamado *Vintage mode*.   
    
  Se puede descargar y evaluar de forma gratuita. Sin embargo no es [software libre](https://es.wikipedia.org/wiki/Software_libre) o de [código abierto](https://es.wikipedia.org/wiki/C%C3%B3digo_abierto) y se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad.

Ideal para HTML5 y CSS3, así como JavaScript. Similar a Notepad++ pero más potente gracias a los plugins, desde mi opinión personal.

* **NetBeans**: Entorno de Desarrollo Integrado con muchas funcionalidades y facilidades a la hora de programar. Se usará para el desarrollo de los archivos PHP de la aplicación. Permite ver posibles errores y tener un repositorio de versiones.
* **Visual Estudio 2015**: Entorno de Desarrollo Integrado con facilidades para trabajr con clases. Facilita la creación de la estructura PHP al trabajar con clases y objetos. Relaciona y sugiere código facilitando y evitando así escribir los métodos de forma errónea. Además de la capacidad para detectar errores.
* **Photoshop CS6**: es un [editor de gráficos rasterizados](https://es.wikipedia.org/wiki/Editor_de_gr%C3%A1ficos_rasterizados) desarrollado por [Adobe Systems Incorporated](https://es.wikipedia.org/wiki/Adobe_Systems_Incorporated). Usado principalmente para el retoque de fotografías y gráficos.

Utilizado para la creación del logo de la aplicación web, la creación de ciertos iconos y la creación de bocetos de la aplicación web. También se ha utilizado para la edición de ciertas imágenes de esta documentación.

* **XAMPP**: Servidor que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor Apache y los interpretes para lenguajes de script, como PHP.

# Planificación temporal

La aplicación web se ha desarrollado durante tres meses, en el periodo comprendido entre el 14 de marzo y el 31 de mayo, durante el periodo de FCT. Al coincidir las prácticas del ciclo de Desarrollo de Aplicaciones Web con el desarrollo del proyecto, el tiempo dedicado a sido menos del deseado, no pudiendo cumplir todos los objetivos planteados en la planificación inicial.

Se ha dedicado 40 horas semanales a las prácticas en la empresa, por lo que el tiempo restante dedicado durante semana ha variado. Los días fijos dedicados al proyecto han sido los sábados y los domingos, desarrollando el proyecto durante 4 horas aproximadamente cada día.

Si el tiempo hubiera sido mayor se hubieran añadido más funcionalidades al proyecto, cumpliendo muchos más objetivos y mejorando ciertos aspectos de la aplicación web.

Las primeras semanas después del planteamiento de la aplicación se han utilizado para el aprendizaje de las APIs y del framework CSS utilizado. Se debe tener en cuenta todo el esfuerzo realizado y el resultado obtenido en tan poco tiempo. Así como las soluciones que se han tomado para superar los objetivos planteados.

En resumen:

* Horas aproximadas de planteamiento y diseño de la aplicación web: 10 horas
* Horas aproximadas de aprendizaje: 15 horas.
* Horas aproximadas de desarrollo front-end: 40 horas
* Horas aproximadas de desarrollo back-end: 35 horas
* Horas aproximadas de la documentación: 10 horas

En total las horas aproximadas para el desarrollo del proyecto han sido 110 horas.

# Costes del proyecto

Los costes del proyecto serían los siguientes:

|  |  |  |  |
| --- | --- | --- | --- |
| **PRECIO** | **Bajo** | **Medio** | **Profesional** |
| Dominio | 12 Euros | 12 Euros | 12 Euros |
| Hosting | 0 Euros | 12 Euros/mes | 23 Euros/mes |
| Creación de la aplicación web | 1100 Euros | 1100 Euros | 1100 Euros |

**Hosting**: Necesario para alojar la aplicación web, con soporte para PHP y MySQL.

**Dominio**: Habría que comprobar cuales están disponibles, buscando *routesofinfinity.com*. La dirección actual será un subdominio de *rcancelo.com* - ***routesofinfinity.rcancelo.com***

**Creación de la página web**: Los desarrolladores del proyecto deberían tener un sueldo mensual de 10€ la hora como mínimo.

# Conclusiones finales

## Grado de cumplimiento de los objetivos fijados

Se han cumplido la mayoría de objetivos planteados en un comienzo. Hace falta pulir ciertos aspectos de la aplicación web e intentar evolucionar a una arquitectura de MVC.

Se ha conseguido una aplicación sencilla y elegante, con un acabado profesional, aunque mejorable.

Todo lo planteado en el anteproyecto es funcional de una manera o de otra en esta versión.

Posibilidad de ampliar para futuras versiones la aplicación web.

## Propuesta de modificaciones o ampliaciones futuras del sistema implementado

El proyecto desarrollado en este periodo de tiempo podría mejorarse y ampliarse.

* Implementar un sistema de seguimiento de las rutas favoritas o poder seguir a usuarios concretos.
* Poder sugerir rutas según la ubicación del usuario.
* Añadir más filtros de búsqueda.
* Evolución a una arquitectura MVC, uso de CodeIgniter o similar.
* Uso de SASS o LESS para CSS, pero por falta de tiempo no he podido aprender nada de esto.
* Usuarios VIP y rutas promocionadas, como forma de generar dinero.
* Poder modificar las rutas, no solo eliminarlas.
* Habilitar enlace para eliminar cuenta.
* Modificar perfil de usuario y subir foto de perfil.
* Crear e incluir vídeo tutorial de cómo funciona la creación de rutas.

# Bibliografía

* Stackoverflow
  + - <http://stackoverflow.com/>
* W3C
  + <http://www.w3schools.com/ajax/default.asp>
  + <http://www.w3schools.com/css/default.asp>
* Documentación de PHP
  + <http://php.net/docs.php>
* Documentación de Materializecss
  + <http://materializecss.com/>
* Documentación de APIs de Google
  + <https://developers.google.com/maps/documentation/javascript/tutorial?hl=es>
  + <https://developers.google.com/places/?hl=es>
  + <https://developers.google.com/maps/documentation/static-maps/intro?hl=es>
* PDO
  + <http://php.net/manual/es/ref.pdo-mysql.php>