****

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA**

**INGENIERÍA TÉCNICA INFORMÁTICA DE GESTIÓN**

**PATHGENERATOR**

**GESTIÓN DE RUTAS CATEGORIZADAS EN LA NUBE USANDO GWT, DATASTORE, APPENGINE Y API GOOGLE MAPS GWT**

**Realizado por**

**MIGUEL CENTENO MUÑOZ**

**28633254W**

**Dirigido por**

**SERGIO SEGURA RUEDA**

**Departamento**

**LENGUAJES Y SISTEMAS INFORMÁTICOS**

**Sevilla, Septiembre de 2014**

# Resumen

EL origen de esta idea surgió como una necesidad particular de un familiar quien, planificando un viaje turístico, necesitaba de la experiencia de personas que ya hubieran estado en diferentes lugares de España. Dicha necesidad se convirtió³ en idea de negocio.

Entre las necesidades que busca cubrir esta aplicación se encuentran dos preguntas que frecuentemente abordan a las personas en la planificación de sus actividades favoritas o el sacar partido a su tiempo libre: ¿Qué hacer y qué ofrece un destino concreto? y, sobre todo, ¿Cómo puedo saber si una actividad en concreto que consulto en la red es realmente lo que busco y cubre mis necesidades?

Posteriormente a esta primera interpretación del problema, surge otro factor determinante en la funcionalidad de *PathGenerator*: Ofrecer alternativas, dentro de los gustos de los usuarios, cerca de la zona en la cual residen e, incluso, cerca de la geolocalización en la que se encuentran en un momento concreto.

*PathGenerator* es una aplicación que busca cubrir las necesidades de los usuarios para realizar actividades categorizadas, presentarlas haciendo uso de las posibilidades de *Google Maps*, y, sobre todo, poder compartir opinión sobre dichas actividades así como poder consultar las opiniones de usuarios o amigos que ya las conozcan.

Los usuarios de *PathGenerator* pueden crear sus propias actividades, gestionarlas, compartirlas con grupos de amigos (tanto privadas como públicamente) y, sobre todo, ofrece a los usuarios la posibilidad de organizarse para realizar sus actividades favoritas, así como sacar el máximo provecho a su tiempo libre.

Para el desarrollo de *PathGenerator* se ha tenido muy en cuenta la posibilidad de que la aplicación sea escalable, tenga un posible despliegue multidispositivo en el futuro, y su código sea mantenible en el tiempo.

Entre las tecnologías y herramientas usadas con este fin se encuentran (enumerar: *projetsii*, *GWT* Google Framework usando *API Google Maps*, *Spring*, *Hibernate*, *Maven*, *SVN*, *Sonar*, y las que salgan).

**Tabla de contenidos**

[Resumen 2](#_Toc391493247)

[Parte I.- Introducción y planificación 7](#_Toc391493248)

[1.- Introducción 9](#_Toc391493249)

[1.1.- Motivación 9](#_Toc391493250)

[1.2.- Objetivos del proyecto 9](#_Toc391493251)

[1.2.1.- Objetivos docentes 9](#_Toc391493252)

[1.2.2.- Objetivos técnicos 10](#_Toc391493253)

[1.3.- Desarrollo del proyecto 10](#_Toc391493254)

[1.3.1.- Projetsii 10](#_Toc391493255)

[1.4 Estructura del documento 11](#_Toc391493256)

[2.- Planificación 12](#_Toc391493257)

[2.1.- Planificación temporal 12](#_Toc391493258)

[2.1.1. Estimación de tiempos 12](#_Toc391493259)

[2.1.2.- Desviación respecto estimación de tiempos 13](#_Toc391493260)

[2.2.- Planificación de costes 13](#_Toc391493261)

[2.2.1.- Estimación de costes 13](#_Toc391493262)

[2.2.2- Desviación respecto estimación de costes 13](#_Toc391493263)

[Parte II.- Materias relacionadas 14](#_Toc391493264)

[3.- Materias relacionadas 16](#_Toc391493265)

[3.1.- Google Web Toolkit (GWT) 16](#_Toc391493266)

[3.2.- MVP aplicado a Gwt: 18](#_Toc391493267)

[Parte III.- Sistema desarrollado 20](#_Toc391493268)

[4.- Dominio del problema 22](#_Toc391493269)

[4.1.- Descripción del problema 22](#_Toc391493270)

[4.2.- Descripción del sistema actual 22](#_Toc391493271)

[4.3.- Glosario de términos 22](#_Toc391493272)

[5.- Visión general del sistema desarrollado 23](#_Toc391493273)

[5.1.- Participantes en el proyecto 23](#_Toc391493274)

[5.1.1.- Participantes 23](#_Toc391493275)

[5.1.2.- Organizaciones 23](#_Toc391493276)

[5.2.- Objetivos del sistema 24](#_Toc391493277)

[6.- Elicitación de requisitos 24](#_Toc391493278)

[6.1.- Requisitos de información 25](#_Toc391493279)

[6.2.- Reglas de negocio y restricciones 26](#_Toc391493280)

[6.3.- Requisitos funcionales 27](#_Toc391493281)

[6.3.1 Diagrama de subsistemas del sistema 27](#_Toc391493282)

[6.3.2.- Diagramas de casos de uso 27](#_Toc391493283)

[6.3.2.- Definición de actores 28](#_Toc391493284)

[6.3.3.- Casos de uso del sistema 29](#_Toc391493285)

[6.3.- Requisitos no funcionales 33](#_Toc391493286)

[6.4.- Matriz de rastreabilidad requisitos-objetivos 33](#_Toc391493287)

[6.5.- Matriz de rastreabilidad casos de uso-objetivos 33](#_Toc391493288)

[7.- Análisis de requisitos 33](#_Toc391493289)

[7.1.- Modelo estático del sistema 33](#_Toc391493290)

[7.1.1.- Diagramas de tipos 33](#_Toc391493291)

[7.1.2.- Escenario de prueba 33](#_Toc391493292)

[7.1.3.- Tipos 33](#_Toc391493293)

[7.2.- Modelo dinámico del sistema 33](#_Toc391493294)

[7.2.1.- Subsistema de gestión de usuarios 33](#_Toc391493295)

[7.2.2.- Subsistema de gestión de rutas 33](#_Toc391493296)

[7.3.- Prototipos de interfaz de usuario 33](#_Toc391493297)

[7.3.1.- Pantalla de inicio y búsqueda 34](#_Toc391493298)

[7.3.2.- Pantalla de login 34](#_Toc391493299)

[7.3.3.- Pantalla de registro de usuario 35](#_Toc391493300)

[7.3.4.- Pantalla Mis Rutas 36](#_Toc391493301)

[7.3.5.- Pantalla Añadir Ruta 37](#_Toc391493302)

[7.3.6.- Pantalla Editar Perfil 38](#_Toc391493303)

[7.3.7.- Pantalla Búsqueda (Usuario logado) 39](#_Toc391493304)

[7.3.8.- Pantalla Resultados Búsqueda (Usuario logado) 39](#_Toc391493305)

[7.3.9.- Pantalla Resultados Búsqueda (Usuario NO logado) 40](#_Toc391493306)

# Parte I.- Introducción y planificación

## 1.- Introducción

En la actualidad, las personas cada vez gozan de un tiempo libre más limitado debido principalmente a las situaciones laborales y personales de cada individuo. Desde hace algún tiempo, cada vez es más frecuente que las personas hagan uso de servicios ofrecidos por las diferentes tecnologías y dispositivos que nos rodean para organizarse y planear dicho tiempo libre en base a los gustos y aficiones particulares.

Dicho tiempo libre es muy utilizado para hacer deporte (jogging, mountain bike...), viajar y hacer turismo, conocer la cultura y gastronomía de diferentes puntos del mundo, acudir a espectáculos y eventos y todo tipo de actividades.

### 1.1.- Motivación

En base a aficiones compartidas con conocidos, familiares y amigos, surge la ideas de desarrollar NOMBREAPP con el fin de facilitar la organización y planificación del tiempo libre en base a las aficiones de cada uno.

Tras tener la idea del desarrollo de NOMBREAPP como una mera afición, conversaciones entabladas con los potenciales usuarios de mi entorno (clientes) hacen que sea aún más motivante pues entiendo que cubre las necesidades de un amplio espectro de posibles usuarios de diferentes edades y gustos muy distintos.

En NOMBREAPP se busca, haciendo uso de la geolocalización y de las posibilidades ofrecidas por el plugin de Google para Eclipse (principalmente su librería Google Maps), representar en un mapa rutas y senderos con información detallada, estando englobadas en diferentes categorías.

NOMBREAPP ofrece la posibilidad a un usuario de crear sus rutas personales y que. tanto el propio usuario como sus círculos de amistades dentro de la aplicación, puedan disfrutar de actividades diversas cercanas al lugar donde se encuentran.

Otra motivación mportante es el uso que el framework GWT (Google Web Toolkit) puede aportarme en el desarrollo continuo de mi carrera y mis conocimientos. GWT aporta múltiples funcionalidades orientadas a la integración de servicios en el desarrollo de aplicaciones web, haciendo el uso de AJAX mucho más sencillo, debido al uso de Java que, como ya es conocido, es un lenguaje robusto y continuamente actualizado. Con GWT se puede desarrollar bajo cualquier IDE de java siendo el código compilado a JavaScript y HTML.

### 1.2.- Objetivos del proyecto

#### 1.2.1.- Objetivos docentes

- Uso de metodologías estudiadas para la planificación, diseño y ejecución de un proyecto software

- Practicar el proceso de planificación y organización de un proyecto software, fundamental para un ingeniero informático.

- Uso de tecnologías no estudiadas durante la etapa universitaria que puedan aportarme en mi carrera profesional.

- Afianzar el uso de Java como lenguaje y plataforma de programación para proyectos software.

#### 1.2.2.- Objetivos técnicos

- Utilización de patrones de diseño y análisis de las ventajas del uso de los mismos durante el desarrollo del proyecto.

- Estudiar y profundizar en el uso de tecnologías en las cuales no se profundiza durante la carrera, y hacer uso de ellas para el desarrollo del proyecto.

- Hacer simulación profesionalizada del ciclo de vida del software y funciones de consultoría del software, análisis de costes y de tiempos.

- Ser capaz de adaptar el proceso de desarrollo de un proyecto software a necesidades reales de clientes.

### 1.3.- Desarrollo del proyecto

La idea a desarrollar surge como consecuencia de las primeras reuniones entre el tutor y el alumno. En la decisión tomada sobre el proyecto a desarrollar se unen necesidades reales de conocidos (usuarios-clientes) con el desarrollo de una aplicación web con integración de servicios y la búsqueda de mejorar la experiencia con tecnologías bastante demandadas en la actualidad entre profesionales del sector de las TIC.

Las reuniones entre tutor y alumno se han ido sucediendo de manera muy habitual con la intención de concretar objetivos y generar tareas a realizar entre las mismas. A las tareas se les ha asignado un tiempo de investigación y ejecución razonables, tiempo tenido en cuenta a la hora de realizar la estimaciones temporales en el desarrollo del proyecto. En las reuniones posteriores se decide las modificaciones necesarias a realizar además de las nuevas tareas.

#### 1.3.1.- Projetsii

Esta herramienta ofrecida por la Universidad de Sevilla se ha utilizado para la gestión de tareas. Las tareas se añaden al sistema asignándole un número de horas estimado y, posteriormente, se le asignan las horas reales utilizadas en el desempeño de las mismas. Esto se utiliza posteriormente en el apartado X.X. de la documentación para la estimación temporal del proyecto.

Para el resto de pasos o etapas superadas para el desarrollo del proyecto, se indica el proceso seguido para la realización de cada apartado o sección en su apartado correspondiente en este documento.

### 1.4 Estructura del documento

Para una mejor comprensión de los pasos seguidos en el desarrollo del proyecto, se resume en este apartado las partes de las que la documentación consta:

**PARTE I – Introducción y planificación.** En esta primera parte, se resumen las motivaciones y los orígenes de la idea a desarrollar, así como se realiza una introducción a la metodología a seguir en su desarrollo, estimación de los tiempos y estimación de los costes de llevar a cabo el proyecto.

**PARTE II – Materias relacionadas.** En la segunda parte, se detallan las tecnologías empleadas en el desarrollo del proyecto. (MÁS)

**PARTE III – Sistema desarrollado.** Esta es la parte principal del documento, en ella se incluye: documentación de la aplicación desarrollada: elicitación y análisis de requisitos, diseño, implementación, pruebas y manuales.

**PARTE IV – Comentarios y conclusión.**

**Parte I – Introducción y planificación**

1.- Introducción

2.- Planificación

Memoria

**Parte II – Materias relacionadas**

3.- Tecnologías Web

**Parte III – Sistema desarrollado**

4.- Dominio del problema

5.- Visión general del sistema

6.- Elicitación de requisitos

7.- Análisis de requisitos

8.- Diseño

9.- Implementación

10.- Pruebas

11.- Manual de instalación

12.- Manual del usuario

**Parte II – Materias relacionadas**

13.-Cumplimiento de los objetivos del proyecto

14.- Conclusiones

## 2.- Planificación

### 2.1.- Planificación temporal

La planificación temporal consiste en la distribución de tiempos estimados para la realización de determinados conjuntos de tareas (los llamaremos HITOS), de manera que se consigue una mejora en la estimación temporal para el desarrollo del proyecto en su conjunto.

En todas las funciones del mundo real en las que se trabaja mediante proyectos, la minimización de los costes y, por ende, de los tiempos de desarrollo de los mismos es fundamental, además de ser una cualidad muy valorable para un buen ingeniero de software.

En el caso de este proyecto en cuestión, la tarea es algo más sencilla pues el equipo de desarrollo consta de un único miembro, ahorrándonos la planificación para desarrollo de tareas de forma paralela.

El uso de tecnologías, frameworks y herramientas, que no son conocidas en profundidad por el autor del proyecto, dificulta la tarea pues tiene un trabajo importante previo de documentación y prácticas con las mismas.

La división del trabajo a realizar en hitos queda definido de la siguiente manera:

|  |  |
| --- | --- |
| HITO | DESCRIPCIÓN |
| H1 | Estudio y prácticas realizadas con las tecnologías a utilizar |
| H2 | Documento de elicitación de requisitos |
| H3 | Documento de análisis de requisitos |
| H4 | Documento de diseño |
| H5 | Desarrollo de la aplicación |
| H6 | Instalación e implantación final de la aplicación |
| H7 | Fin de proyecto |

#### 2.1.1. Estimación de tiempos

La fecha en la que se comenzó la realización del proyecto fue en torno al 15 de Marzo de 2014, y la fecha fin del mismo es en el mes de Septiembre, a principios del mismo, pero debido a que el mes de agosto no es lectivo, se estimará la fecha de finalización para documentación, desarrollo, pruebas e implantación en torno al 30 de Julio.

La estimación en horas de trabajo para el único miembro del equipo de trabajo será de **3 horas al día 5 días a la semana**, lo que hace un total de **15 horas semanales**. Estas 15 horas semanales es fruto de una ponderación, pues hay fechas en las que la capacidad de dedicación en horas es inferior a dicha fecha, así como hay épocas en las que es bastante mayor, haciendo uso de horas en fin de semana.

Otro factor a tener en cuenta es la cantidad de horas mínima que el PFC tiene como asignatura de Ingeniería Técnica Informática de Gestión, que son **225 horas**.

En la tabla que a continuación se presenta, realizamos una división del desarrollo del proyecto en tareas y una estimación de horas divididas por cada una de ellas.

|  |  |
| --- | --- |
| TAREA | DÍAS |
| T1.- Estudio del problema y prácticas previas realizadas | **10** |
| T2.- Familiarización con el entorno de desarrollo y herramientas | **71** |
| T3.- Desarrollo de la aplicación  T3.1. Elicitación de requisitos  T3.2. Análisis de requisitos  T3.3. Diseño del sistema  T3.4. Implementación de la aplicación  T3.5. Pruebas Unitarias  T3.6. Implantación del sistema | 15  15  5  30  3  3 |
| T4.- Documentación del proyecto | **14** |
| T5.- Manual de usuario | **5** |
| TOTAL | **100 días** |

Según el reparto de días en función a tareas realizado anteriormente, nos salen 120 días.

Como ya hemos comentado, asignamos 3 horas/día, quedándonos una dedicación total en horas: **100 días x 3 horas/día = 300 horas.**

Diagrama de gantt: TOMSPLANNER

#### 2.1.2.- Desviación respecto estimación de tiempos

Pendiente

### 2.2.- Planificación de costes

Pendiente

#### 2.2.1.- Estimación de costes

Pendiente

#### 2.2.2- Desviación respecto estimación de costes

Pendiente

# Parte II.- Materias relacionadas

## 3.- Materias relacionadas

En el capítulo tercero, se repasan las tecnologías y herramientas de las que se ha hecho uso durante el desarrollo de *PathGenerator*. En ellas presentaremos brevemente sus características y las ventajas que ofrece su uso.

### 3.1.- Google Web Toolkit (GWT)

Google Web Toolkit (GWT) es un framework, un conjunto de APIs y widgets que permiten desarrollar aplicaciones AJAX en Java para después compilar el código fuente a Javascript de manera muy optimizada. Este código JavaScript generado corre perfectamente en todo tipo de navegadores web incluyendo navegadores móviles de Android o iOs.

Una ventaja que tiene GWT es que, a pesar de estar basado en una serie de Widgets en su capa más alta de presentación, permite la inserción de código Javascript estándar por parte del usuario. Digamos que no limita su uso a sus puntos fuertes, dejando hueco a la creatividad del desarrollador.

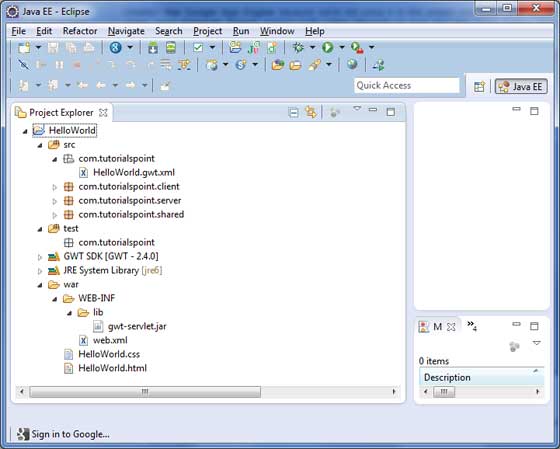
Otra de las principales ventajas que ofrece GWT es la posibilidad de hacer depuración en código Javascript, debido a que se implementa en Java, además de que puede ser usado mediante un IDE (Eclipse en nuestro caso) con las ventajas y comodidades que ofrece.

En el momento del despliegue de la aplicación es cuando el código desarrollado en Java es compilado mediante el compilador GWT generando un fichero Javascript y ficheros HTML.

La aplicaciones desarrolladas en GWT pueden ser, a su vez, ejecutadas de dos modos diferentes: *Dev-Mode* o en modo desarrollo, ejecutándose como código bytecode de Java dentro de la Máquina Virtual de Java (JVM) o en modo Web, ejecutándose la aplicación como código JavaScript y HTML puro.

Probablemente, la ventaja de mayor peso de todas las que ofrece este framework es la siguiente: GWT resuelve, de manera sencilla, la parte del cliente y de comunicación remota usando un contenedor de Servlets en el lado del cliente. Además, GWT es sencillo de combinar con tecnologías maduras de Java, como pueden ser JPA, EJB, Spring Framewrok...

Google ofrece desde hace bastante tiempo, un plugin oficial para eclipse llamado Googl Plugin for Eclipse. Con él en sencillos pasos podemos dar comienzo al desarrollo de un proyecto GWT con Eclipse.

Imagen 1: Estructura de un proyecto GWT en Eclipse.

En el paquete contenedor del código Java (src) categorizamos el código en 3 paquetes básicos: client, shared, server.

Client contiene el código del lado del cliente, shared contendrá clases útiles en ambas direcciones, y server contendrá clases con métodos que se ejecutarán del lado del servidor, así como el mapeo de la Base de Datos.

Otros componentes a destacar en la estructura de un proyecto base GWT:

Módulo XML (cuyo nombre será: nombreapp.gwt.xml). En él se definen el módulo GWT, los recursos del proyecto o shared package y también define cuál es la clase Entrypoint.

Página host (nombreapp.html). Página que ejecuta la aplicación. Referencia las hojas de estilos CSS. También referencia el código JavaScript que será un mezcla de elementos estáticos y dinámicos.

Css de la aplicación.

En GWT la comunicación entre el código que se ejecutará en e lado del cliente, y el que lo hará en el lado del servidor, se realiza mediante llamadas RPC (Remote Procedure Calls).

### 3.2.- MVP aplicado a Gwt:

A la hora de desarrollar una aplicación en GWT tenemos la posibilidad de usar el patrón MVP (modelo vista presentador). ¿Qué ventajas tiene su uso? 1º desacopla el desarrollo en una dirección permitiendo trabajar a varios desarrolladores a la vez.

Y 2º minimiza el uso de GWTTestCases (se realizan en navegador, siendo más costosas) y permite la realización de pruebas de software basadas en JRE (véase Junit) lo cuál es más rápido y eficiente, desechando el uso de un navegador para testear el software.

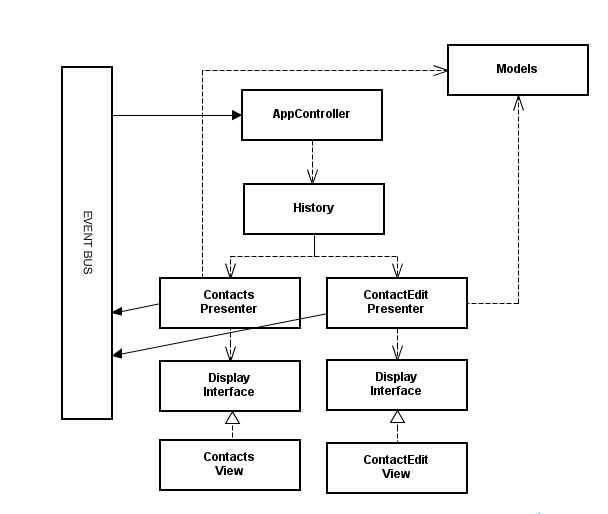
A la hora de usar el patrón MVP, existen dos conceptos fundamentales:

**a) Se particiona la aplicación en componentes:**

Model

View

Presenter

AppController

**b) Interactuación de los componentes:**

Enlazar Presenter – Views

Eventos y Bus de Eventos

Histórico y transición entre las vistas.

Pruebas (testing)

Imagen 2: Diagrama patrón MVP en Gwt en App de prueba de gestión de contactos.

# Parte III.- Sistema desarrollado

## 4.- Dominio del problema

### 4.1.- Descripción del problema

En la actualidad, se pueden encontrar muchas aplicaciones orientadas a cubrir las necesidades que tienen las personas de cubrir su, en muchos casos, escaso tiempo libre. Dichas aplicaciones suelen ser específicas de un tipo de actividad, como puede ser la gastronomía, el deporte o el turismo de una localidad, con lo cual buscan solucionar necesidades puntuales para una actividad o un ámbito de necesidades concreto.

Son herramientas pensadas para llegar al mayor número de usuarios posibles y, sobre todo, son orientadas con desde el punto de vista del comercio o local, más que en la creación de comunidad de usuarios. Bien es verdad que la mayoría de las mismas permiten el uso de herramientas sociales pero no el crear comunidad como tal.

También cada vez es más frecuente tener conocido por las personas las actividades que hacer en su lugar habitual de residencia, pero actualmente muchos trabajamos o vivimos temporalmente en sitios que no nos son familiares.

Con PathGenerator se busca solucionar la necesidad de las personas de realizar sus actividades preferidas --o sencillamente ofrecer algo que hacer en su tiempo libre—allá donde se encuentre. Actividades que son compartidas por los mismos usuarios de la aplicación y, de todos es sabido, que las recomendaciones y sugerencias de las personas que han realizado ya alguna actividad, han comido en algunos restaurantes o simplemente han paseado por una ciudad, es la mejor fuente de conocimiento y garantía de confianza.

### 4.2.- Descripción del sistema actual

Ya se comentó al inicio de este documento que la idea surge de un familiar, muy aficionado tanto al jogging como a la mountain bike, cuando se encontraba planificando sus vacaciones y descubrió que existía un vacío a la hora de poder disfrutar del turismo, conocer una ciudad, su gastronomía y, a la vez, poder hacer deporte en lugares que fueran realmente interesantes, de modo que pudiera hacer su tiempo más provechoso.

El principal objetivo es no tener que visitar un elevado alto de sitios web, donde en dos webs diferentes puedes encontrar opiniones o sugerencias contrapuestas para un mismo lugar, una misma actividad.

Nace así la idea de desarrollar una herramienta web en la cual exista una variedad de categorías de actividades a realizar y, además, puedan consultarse en el mapa en forma de ruta, para poder hacer más accesible, y en definitiva más útil, la información a los usuarios.

### 4.3.- Glosario de términos

pendiente

## 5.- Visión general del sistema desarrollado

### 5.1.- Participantes en el proyecto

En este apartado presentamos brevemente a las diferentes personas y organizaciones involucradas en el proyecto.

#### 5.1.1.- Participantes

|  |  |
| --- | --- |
| Participante | Miguel Centeno Muñoz |
| Organización | Alumnado |
| Rol | Desarrollador |
| Es desarrollador | Sí |
| Es cliente | No |
| Es usuario | Sí |
| Comentarios | miguelcent@gmail.com |

|  |  |
| --- | --- |
| Participante | Sergio Segura Rueda |
| Organización | Dpto. Lenguajes y Sistemas Informáticos |
| Rol | Tutor |
| Es desarrollador | No |
| Es cliente | No |
| Es usuario | Sí |
| Comentarios | sergiosegura@us.es |

|  |  |
| --- | --- |
| Participante | Sara Álvarez Romero |
| Organización | Alumnado |
| Rol | Cliente |
| Es desarrollador | No |
| Es cliente | Sí |
| Es usuario | Sí |
| Comentarios | Ninguno |

#### 5.1.2.- Organizaciones

|  |  |
| --- | --- |
| Organización | Dpto. Lenguajes y Sistemas Informáticos |
| Dirección | Avenida Reina Mercedes S/N 41012 Sevilla |
| Teléfono | 954 557 139 |
| Fax | 954 557 139 |
| Comentarios | E-mail: lsi@lsi.us.es  Web: www.lsi.us.es |

|  |  |
| --- | --- |
| Organización | Alumnado |
| Dirección | Luis Montoto, 135 - 2º C |
| Teléfono | 617673924 |
| Fax |  |
| Comentarios | Organización a la que pertenece el alumno desarrollador y el cliente. |

### 5.2.- Objetivos del sistema

|  |  |
| --- | --- |
| OBJ-0001 | Gestión Usuarios Aplicación |
| Versión | 1.0 ( 22/06/2014 ) |
| Autores | * Miguel Centeno Muñoz |
| Fuentes | * Sara Álvarez Romero |
| Descripción | El sistema deberá gestionar toda la información referente a los usuarios de la aplicación y sus perfiles, incluyendo la capacidad de Alta de usuario, así como la modificación del perfil de un usuario. |
| Subobjetivos | * **[OBJ-0002] Alta de usuario**: El sistema deberá gestionar el alta de un nuevo usuario en el sistema * **[OBJ-0003] Modificación perfil de usuario**: El sistema deberá gestionar la modificación del perfil de los usuarios. |
| Importancia | vital |
| Urgencia | inmediatamente |
| Estado | en construcción |
| Estabilidad | alta |
| Comentarios | Ninguno |

|  |  |
| --- | --- |
| OBJ-0004 | Gestión de Rutas |
| Versión | 1.0 ( 23/06/2014 ) |
| Autores | * Miguel Centeno Muñoz |
| Fuentes | * Sara Álvarez Romero |
| Descripción | El sistema deberá gestionar toda la información referida a las rutas, indluyendo la creación, modificación de las mismas. |
| Subobjetivos | * **[OBJ-0005] Alta de una Ruta**: El sistema deberá gestionar el alta de una nueva ruta creada por un usuario en el sistema. * **[OBJ-0006] Modificación de una Ruta**: El sistema deberá gestionar la modificación de los atributos propios de una ruta. * **[OBJ-0007] Eliminación de una Ruta**: El sistema deberá gestionar la capacidad de un usuario para eliminar una ruta creada por el del sistema. |
| Importancia | vital |
| Urgencia | inmediatamente |
| Estado | en construcción |
| Estabilidad | alta |
| Comentarios | Ninguno |

## 6.- Elicitación de requisitos

Se conoce como Elicitación de requisitos a todos los procesos que se realizan anteriormente a la creación del diseño y la implementación de un proyecto software. Un requisito es cada una de las funciones que debe realizar el sistema que se quiere realizar. Normalmente se presenta en forma de lista de requisitos.

Para la recopilación de estos requisitos, existen técnicas variadas, desde las más básicas, aplicables a todo tipo de proyectos, como pueden ser las entrevistas, los talleres, las reuniones en grupo… hasta otras herramientas más orientadas a aspectos tecnológicos, aspectos por los que, además pueden ser categorizados los requisitos. Por ejemplo, habrá requisitos de interfaz, requisitos funcionales, o requisitos de información a almacenar.

Los requisitos y los procesos por los cuáles sean obtenido quedan recogidos habitualmente en un documento de Elicitación de requisitos. Si partimos que el objetivo final del proyecto es, a grandes rasgos, el desarrollo de un aplicación web que presenta rutas en mapas para el desarrollo de actividades diversas, veremos que a partir de este concepto muy generalista nos surgirán requisitos más concretos, pudiéndolos seccionar.

### 6.1.- Requisitos de información

Las tablas que se presentan a continuación, hacen referencia a requisitos referentes a la información que el sistema debe almacenar.

|  |  |
| --- | --- |
| IRQ-0001 | Información sobre el perfil del usuario |
| Versión | 1.0 ( 23/06/2014 ) |
| Autores | * Miguel Centeno Muñoz |
| Fuentes | * Sara Álvarez Romero |
| Dependencias | Ninguno |
| Descripción | El sistema deberá almacenar la información correspondiente a datos identificativos de cada usuario. En concreto: |
| Datos específicos | * Nombre de usuario * Password * Email * Nombre * Apellidos * País * Foto de perfil |
| Importancia | vital |
| Urgencia | inmediatamente |
| Estado | en construcción |
| Estabilidad | alta |
| Comentarios | Ninguno |

|  |  |
| --- | --- |
| IRQ-0002 | Información sobre las Rutas |
| Versión | 1.0 ( 23/06/2014 ) |
| Autores | * Miguel Centeno Muñoz |
| Fuentes | * Sara Álvarez Romero |
| Dependencias | Ninguno |
| Descripción | El sistema deberá almacenar la información correspondiente a datos identificactivos e informativos de cada una de las rutas del sistema.. En concreto: |
| Datos específicos | * Título de la Ruta ( El usuario tiene la capacidad de nombrar cada ruta.) * Categoría de la Ruta: Mountain Bike, Turística, Gastronómica, Senderismo o Jogging. * Descripción de la ruta. ( El usuario puede escribir una descripción explicativa sobre cada una de sus rutas) * Fichero de la Ruta ( la ruta específica al fichero físico, así como el fichero en si.) |
| Importancia | vital |
| Urgencia | inmediatamente |
| Estado | en construcción |
| Estabilidad | alta |
| Comentarios | Ninguno |

### 6.2.- Reglas de negocio y restricciones

|  |  |
| --- | --- |
| CRQ-0001 | Unicidad de los nombres de usuario |
| Versión | 1.0 ( 23/06/2014 ) |
| Autores | * Miguel Centeno Muñoz |
| Fuentes | * Sara Álvarez Romero |
| Dependencias | Ninguno |
| Descripción | La información almacenada por el sistema deberá satisfacer la siguiente restricción: La información que el sistema almacene como nombr de usuario de un usuario debe ser única. |
| Importancia | vital |
| Urgencia | hay presión |
| Estado | en construcción |
| Estabilidad | alta |
| Comentarios | Ninguno |

### 6.3.- Requisitos funcionales

A continuación se representan los requisitos funcionales que deberá satisfacer la aplicación a desarrollar. Se categorizan en Subsistemas diferenciados por la información con la que tratan.

#### 6.3.1 Diagrama de subsistemas del sistema

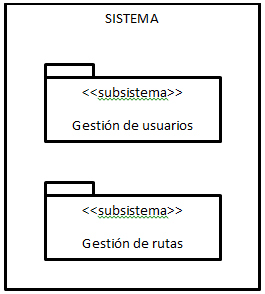


Imagen X: Diagrama subsistemas de la aplicación

#### 6.3.2.- Diagramas de casos de uso

##### 6.3.2.1.- Subsistema de Gestión de Usuarios

En el Subsistema de Gestión de Usuarios se gestiona toda la información referente al perfil de los usuarios de la aplicación, incluyendo modificación de la contraseña y los datos identificativos.

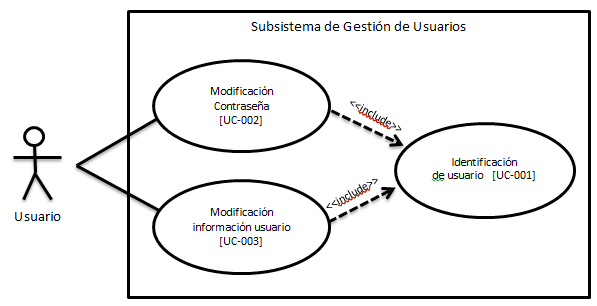


Imagen X: Subsistema de gestión de usuarios

##### 6.3.2.2.- Subsistema de Gestión de Rutas

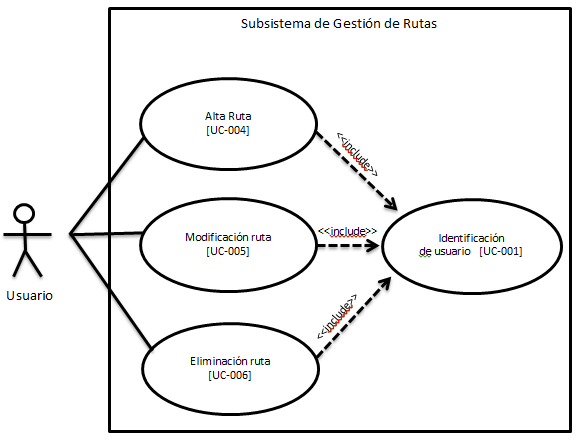
En el Subsistema de Gestión de Rutas se gestiona toda la información referente a las rutas de la aplicación, incluyendo alta, modificación y eliminación de las mismas.

Imagen X: Subsistema de gestión de rutas

#### 6.3.2.- Definición de actores

|  |  |
| --- | --- |
| ACT-0001 | Usuario |
| Versión | 1.0 ( 21/06/2014 ) |
| Autores | * Miguel Centeno Muñoz |
| Fuentes | * Sara Álvarez Romero |
| Descripción | Este actor representa el usuario registrado en el sistema |
| Comentarios | Ninguno |

#### 6.3.3.- Casos de uso del sistema

##### 6.3.3.1.- Subsistema de gestión de usuarios

|  |  |  |
| --- | --- | --- |
| UC-0001 | Identificación de usuario | |
| Versión | 1.0 ( 21/06/2014 ) | |
| Autores | * Miguel Centeno Muñoz | |
| Fuentes | * Sergio Segura Rueda | |
| Dependencias |  [OBJ-0001] Gestión Usuarios Aplicación   [IRQ-0001] Información sobre Usuario | |
| Descripción | El sistema deberá comportarse tal como se describe en el siguiente caso de uso abstracto durante la realización de los siguientes casos de uso: [UC-0002] Modificación contraseña, [UC-0003] Modificación informaciñón del usuario, [UC-0004] Alta ruta, [UC-0005] Modificación ruta, [UC-0006] Eliminación ruta | |
| Precondición | Ninguna | |
| Secuencia normal | **Paso** | **Acción** |
| 1 | El sistema solicita los datos de identificación del usuario |
| 2 | El actor Usuario (ACT-0001) introduce los datos de identificación |
| 3 | El sistema valida los datos introducidos por el usuario mostrando un mensaje de confirmación |
| Postcondición | Usuario reconocido por la aplicación, asignándosele sus respectivos permisos. | |
| Excepciones | **Paso** | **Acción** |
| 3 | Si alguno de los datos aportados por el usuario no es correcto, el sistema muestra un mensaje informando de la no-correcta introducción de los datos de usuario.El sistema permite modificar los datos introducidos erroneamente., a continuación este caso de uso continúa |
| Importancia | vital | |
| Urgencia | inmediatamente | |
| Estado | en construcción | |
| Estabilidad | alta | |

|  |  |  |
| --- | --- | --- |
| UC-0002 | Modificación contraseña | |
| Versión | 1.0 ( 21/06/2014 ) | |
| Autores | * Miguel Centeno Muñoz | |
| Fuentes | * Sergio Segura Rueda | |
| Dependencias |  [OBJ-0001] Gestión Usuarios Aplicación   [IRQ-0001] Información sobre Usuario | |
| Descripción | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando cuando un usuario solicite cambiar su contraseña. | |
| Precondición | Ninguna | |
| Secuencia normal | **Paso** | **Acción** |
| 1 | Se realiza el caso de uso Identificación de usuario (UC-0001) |
| 2 | El sistema solicita al usuario una nueva contraseña (por duplicado para evitar confusiones) |
| 3 | El actor Usuario (ACT-0001) introduce la nueva contraseña |
| 4 | El sistema Registra y asigna nueva contraseña al usuario y muestra confirmación por pantalla |
| Postcondición | El sistema asigna una nueva contraseña al usuario | |
| Importancia | vital | |
| Urgencia | inmediatamente | |
| Estado | en construcción | |
| Estabilidad | alta | |

|  |  |  |
| --- | --- | --- |
| UC-0003 | Modificación informaciñón del usuario | |
| Versión | 1.0 ( 25/06/2014 ) | |
| Autores | * Miguel Centeno Muñoz | |
| Fuentes | * Sergio Segura Rueda | |
| Dependencias |  [IRQ-0001] Información sobre Usuario   [OBJ-0003] Modificación perfil de usuario | |
| Descripción | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario solicita la modificación/actualización de los datos de su perfil | |
| Precondición | Ninguno | |
| Secuencia normal | **Paso** | **Acción** |
| 1 | Se realiza el caso de uso Identificación de usuario (UC-0001) |
| 2 | El sistema solicita al usuario los datos del perfil que desea actualizar |
| 3 | El actor Usuario (ACT-0001) rellena los datos del perfil que desea actualizar |
| 4 | Si dato identificativo a modificar es foto de perfil, el actor Usuario (ACT-0001) provee el fichero con la foto de perfil, buscándolo en su equipo mediante explorador de archivos |
| 5 | El sistema registra/actualiza los datos identificativos facilitados mostrando confirmación del proceso por pantalla |
| Postcondición | Ninguna | |
| Importancia | vital | |
| Urgencia | inmediatamente | |
| Estado | en construcción | |
| Estabilidad | alta | |

##### 6.3.3.2.- Subsistema de gestión de rutas

|  |  |  |
| --- | --- | --- |
| UC-0004 | Alta ruta | |
| Versión | 1.0 ( 25/06/2014 ) | |
| Autores | * Miguel Centeno Muñoz | |
| Fuentes | * Sergio Segura Rueda | |
| Dependencias |  [OBJ-0004] Gestión de Rutas   [OBJ-0005] Alta de una Ruta   [IRQ-0002] Información sobre Rutas | |
| Descripción | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario solicita que desea registrar una nueva ruta | |
| Precondición | Ninguna | |
| Secuencia normal | **Paso** | **Acción** |
| 1 | Se realiza el caso de uso Identificación de usuario (UC-0001) |
| 2 | El sistema solicita al usuario información referente a la ruta que se desea añadir |
| 3 | El actor Usuario (ACT-0001) provee la información necesaria para registrar una nueva ruta |
| 4 | El sistema registra el alta de la nueva ruta así como muestra un mensaje por pantalla de confirmación |
| Postcondición | Nueva ruta queda registrada en el sistema | |
| Importancia | vital | |
| Urgencia | inmediatamente | |
| Estado | en construcción | |
| Estabilidad | alta | |

|  |  |  |
| --- | --- | --- |
| UC-0005 | Modificación ruta | |
| Versión | 1.0 ( 25/06/2014 ) | |
| Autores | * Miguel Centeno Muñoz | |
| Fuentes | * Sergio Segura Rueda | |
| Dependencias |  [OBJ-0006] Modificación de una Ruta   [OBJ-0004] Gestión de Rutas | |
| Descripción | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario solicita modificar la información referente a una ruta | |
| Precondición | Ninguna | |
| Secuencia normal | **Paso** | **Acción** |
| 1 | Se realiza el caso de uso Identificación de usuario (UC-0001) |
| 2 | El sistema solicita los atributos de la ruta que el usuario desea modificar |
| 3 | Si el atributo de la ruta que desea modificar es fichero ruta, el actor Usuario (ACT-0001) busca el nuevo fichero a registrar en su equipo mediante el explorador de archivos |
| 4 | El sistema el sistema registra los cambios realizados por el usuario a la ruta mostrando un mensaje de confirmación |
| Postcondición | Ninguna | |
| Importancia | vital | |
| Urgencia | inmediatamente | |
| Estado | en construcción | |
| Estabilidad | alta | |

|  |  |  |
| --- | --- | --- |
| UC-0006 | Eliminación ruta | |
| Versión | 1.0 ( 25/06/2014 ) | |
| Autores | * Miguel Centeno Muñoz | |
| Fuentes | * Sergio Segura Rueda | |
| Dependencias |  [IRQ-0002] Información sobre Rutas   [OBJ-0007] Eliminación de una Ruta   [OBJ-0004] Gestión de Rutas | |
| Descripción | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario solicita que desea eliminar una ruta | |
| Precondición | Ninguna | |
| Secuencia normal | **Paso** | **Acción** |
| 1 | Se realiza el caso de uso Identificación de usuario (UC-0001) |
| 2 | El actor Usuario (ACT-0001) selecciona la ruta que desea eliminar haciendo click en el botón designado para eliminar ruta en la pantalla de gestión de las rutas del usuario |
| 3 | El sistema solicita confirmación de que se quiere eliminar la ruta seleccionada |
| 4 | Si cliente quiere eliminar la ruta , el actor Usuario (ACT-0001) confirma la eliminación de la ruta en cuestión |
| 5 | Si cliente ha confirmado la eliminación de la ruta, el sistema elimina la ruta en cuestión mostrando un mensaje de confirmación |
| 6 | Si cliente no quiere eliminar la ruta en cuestión, el actor Usuario (ACT-0001) cancela el proceso de eliminación haciendo uso del botón desiegnado a tal función |
| 7 | Si cliente cancela eliminación de la ruta, el sistema sistema muestra un mensaje confirmando la no-eliminación de la ruta |
| Postcondición | La ruta queda eliminada del sistema | |
| Importancia | vital | |
| Urgencia | inmediatamente | |
| Estado | en construcción | |
| Estabilidad | alta | |

### 6.3.- Requisitos no funcionales

### 6.4.- Matriz de rastreabilidad requisitos-objetivos

### 6.5.- Matriz de rastreabilidad casos de uso-objetivos

## 7.- Análisis de requisitos

### 7.1.- Modelo estático del sistema

#### 7.1.1.- Diagramas de tipos

##### 7.1.1.1.- Subsistema de gestión de usuarios

##### 7.1.1.2.- Subsistema de gestión de rutas

#### 7.1.2.- Escenario de prueba

#### 7.1.3.- Tipos

##### 7.1.3.1.- Subsistema de gestión de usuarios

##### 7.1.3.2.- Subsistema de gestión de rutas

### 7.2.- Modelo dinámico del sistema

#### 7.2.1.- Subsistema de gestión de usuarios

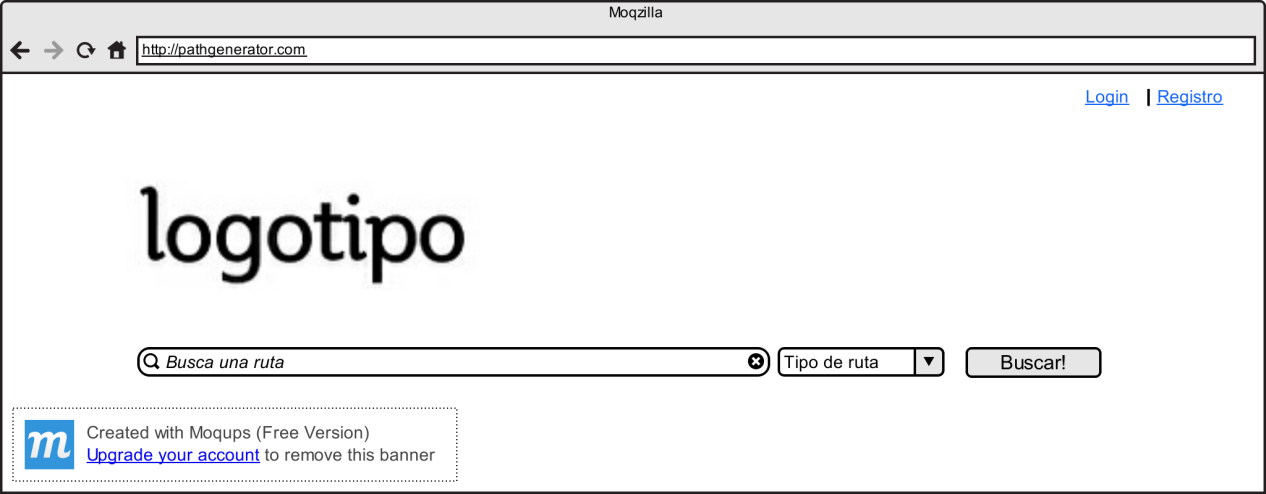
#### 7.2.2.- Subsistema de gestión de rutas

### 7.3.- Prototipos de interfaz de usuario

A continuación mostraremos las vistas más importantes que conformarán nuestra aplicación. Para la realización de estos diseños hemos usado el software *Moqups* en su versión de prueba. Estos prototipos han sido obtenidos tras varias reuniones con el cliente y a partir de los requisitos obtenidos. Serán utilizados como medio de comunicación con el cliente, como una forma sencilla para él de comprobar que los requisitos que nos está pidiendo aparecen todos en las interfaces mostradas. Por nuestra parte, serán usados para cerciorarnos de que lo que nos está pidiendo se adecua a lo que hemos entendido a partir de los requisitos, y también para comprobar si existen inconsistencias en los requisitos obtenidos o si es necesario añadir o eliminar alguno.

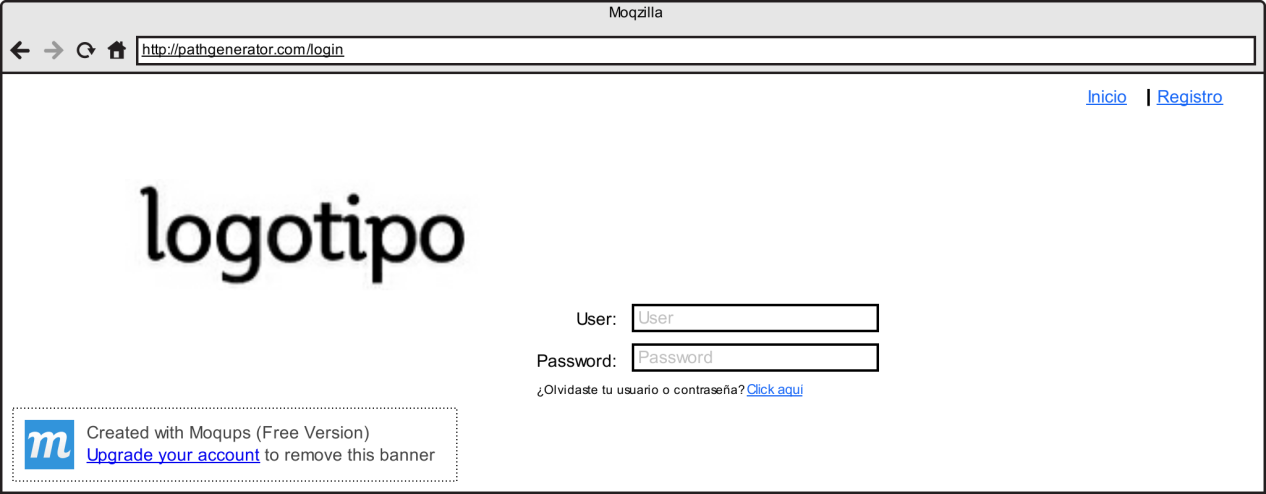
#### 7.3.1.- Pantalla de inicio y búsqueda

texto



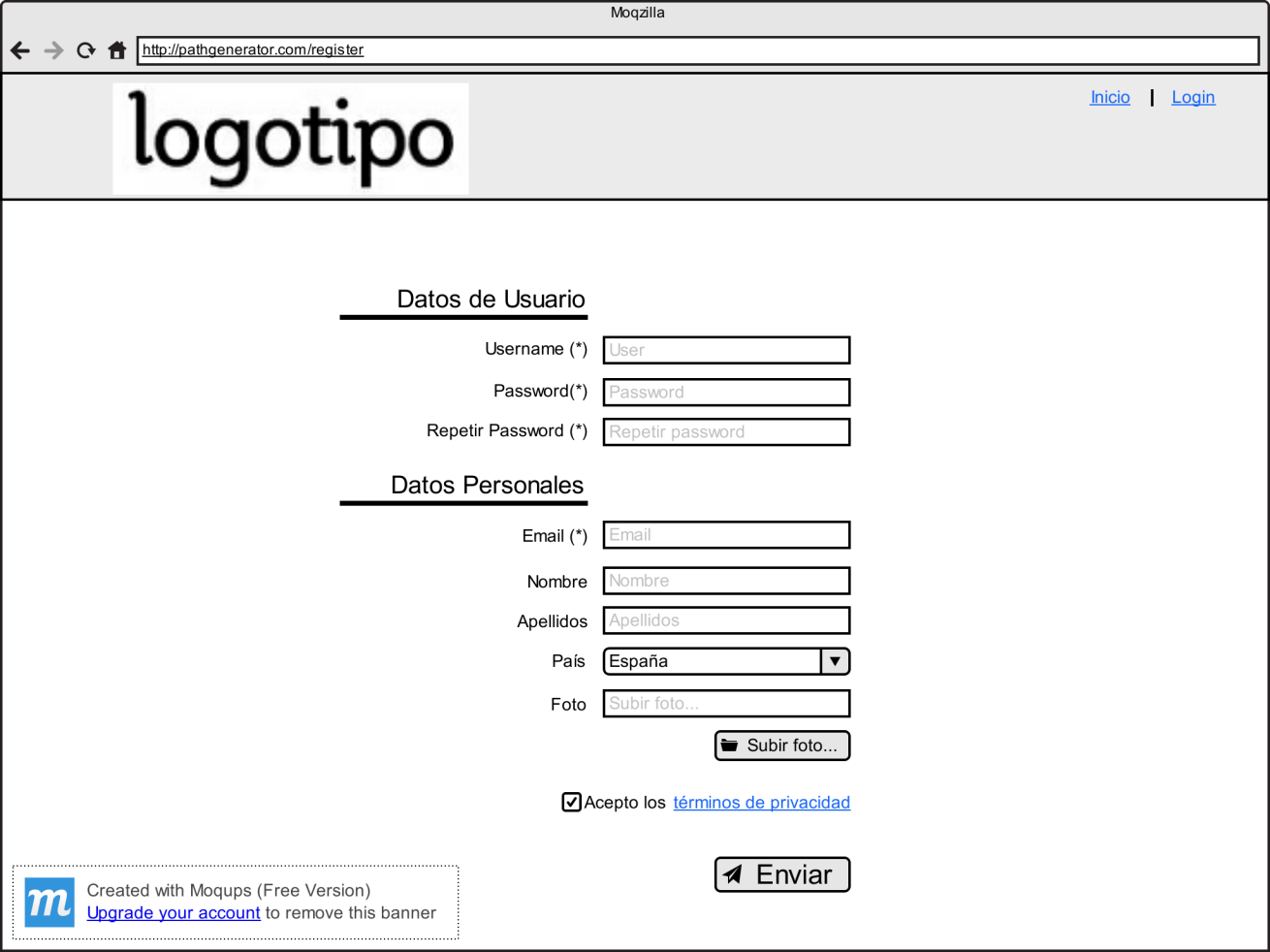
#### 7.3.2.- Pantalla de login

Texto



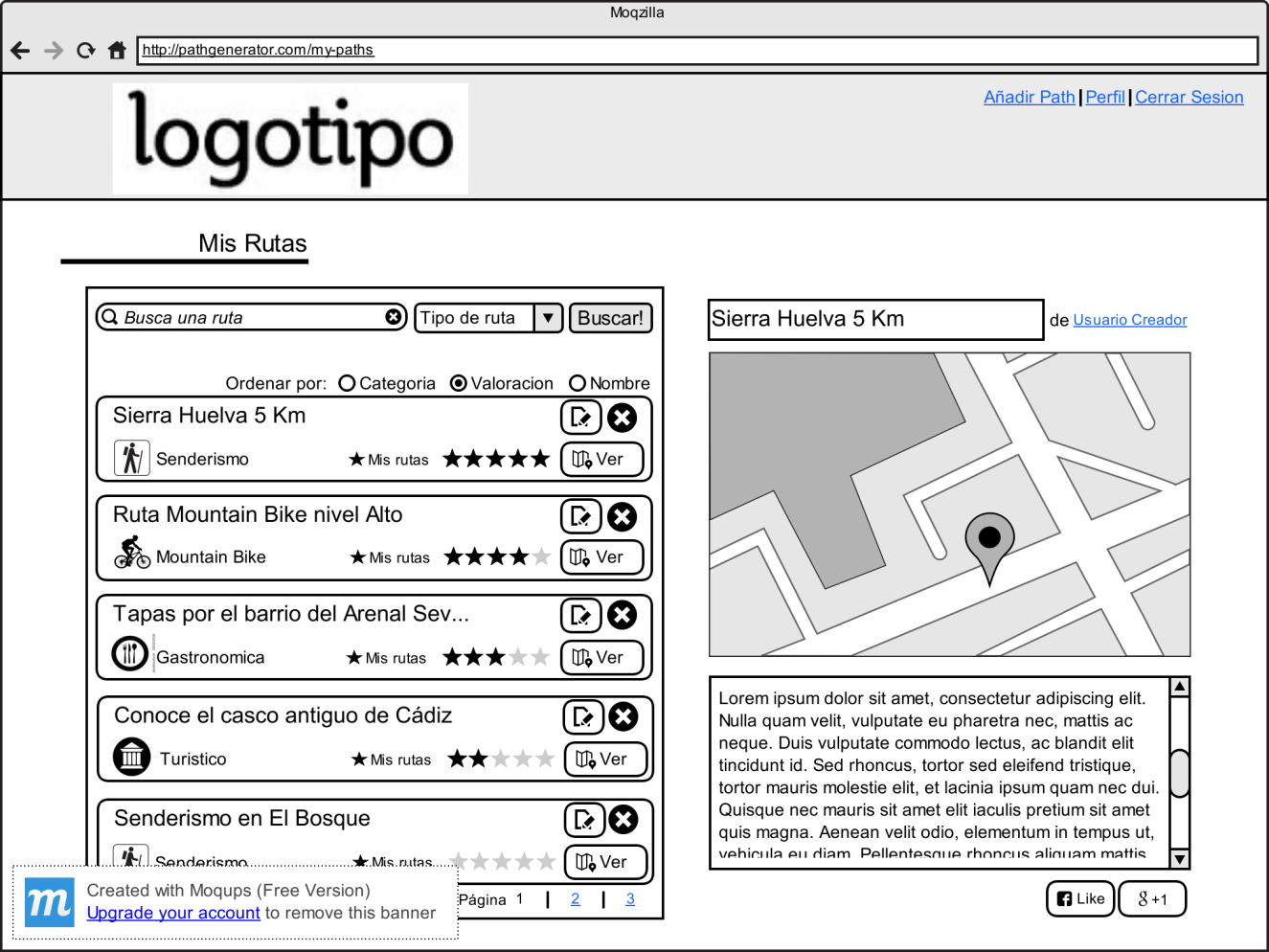
#### 7.3.3.- Pantalla de registro de usuario

texto



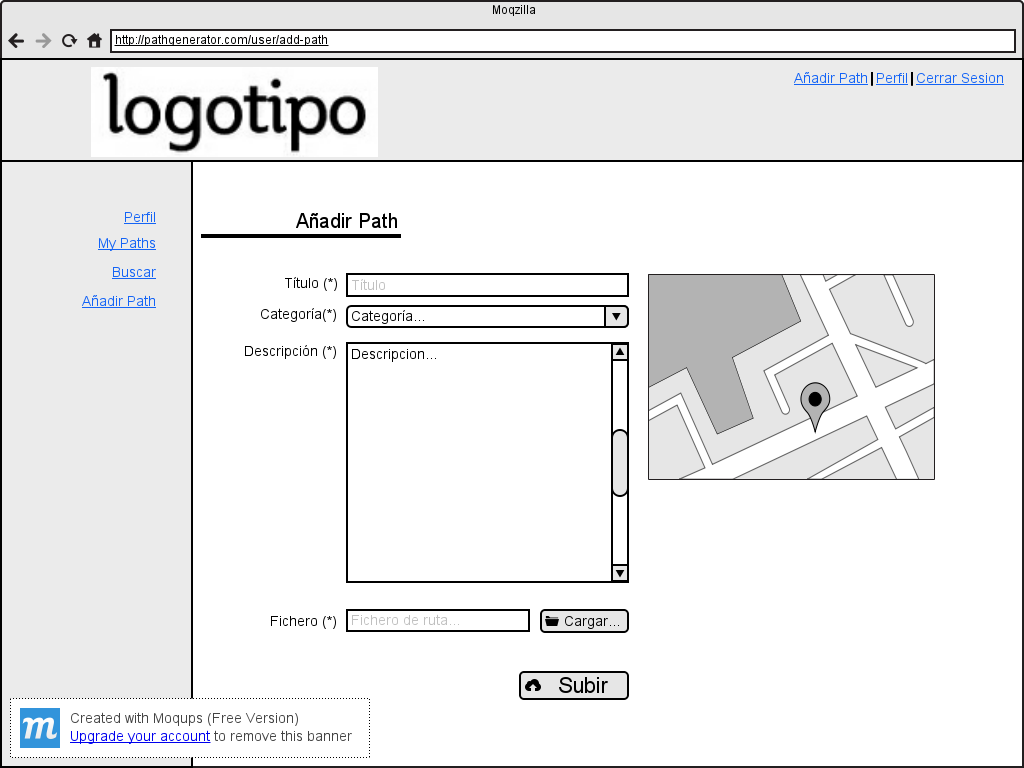
#### 7.3.4.- Pantalla Mis Rutas

texto



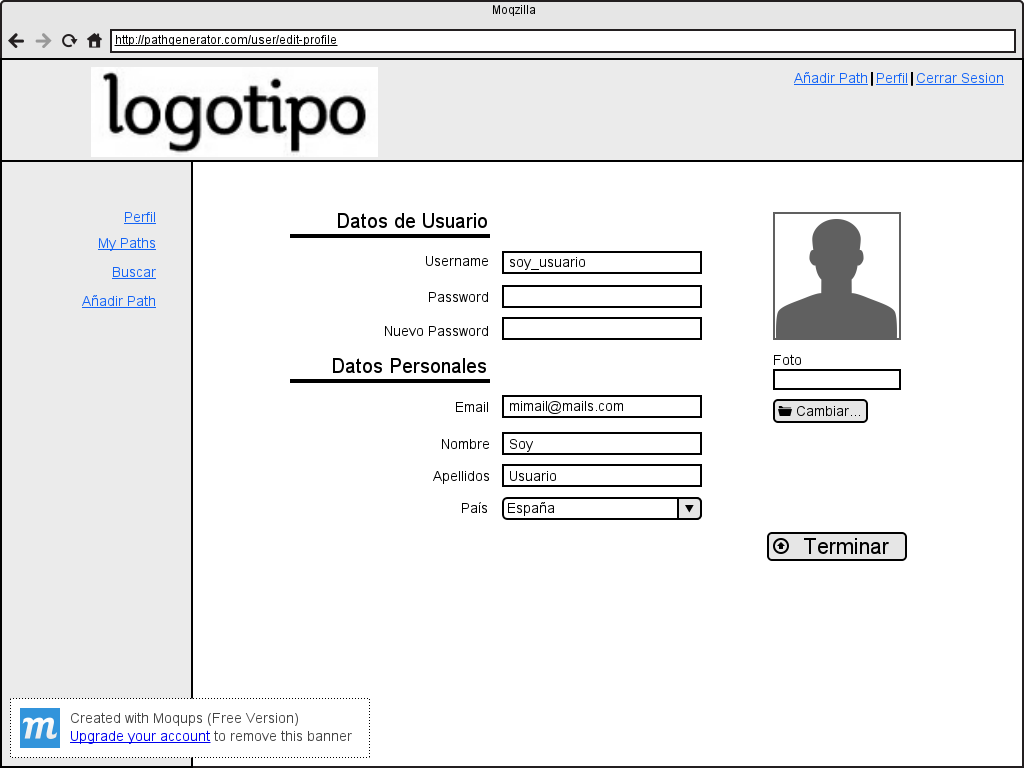
#### 7.3.5.- Pantalla Añadir Ruta

Texto



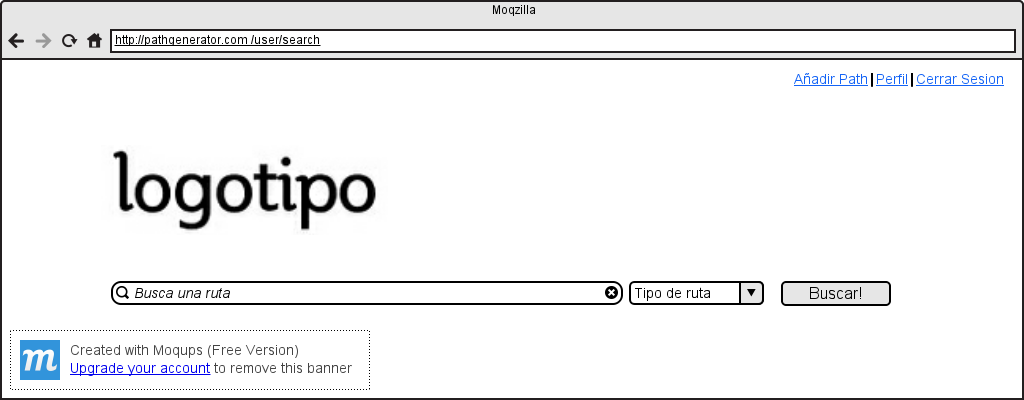
#### 7.3.6.- Pantalla Editar Perfil

Texto



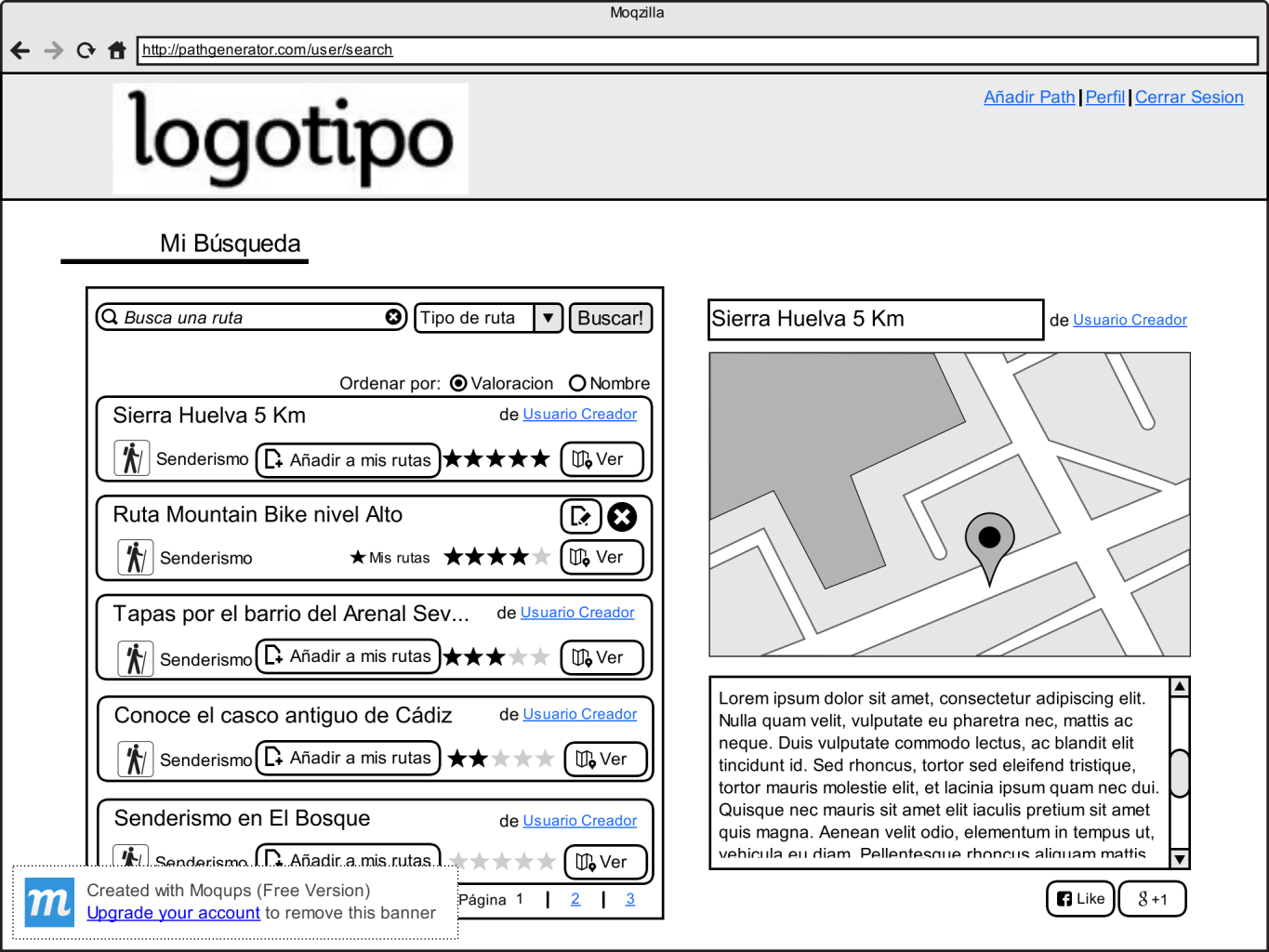
#### 7.3.7.- Pantalla Búsqueda (Usuario logado)

Texto



#### 7.3.8.- Pantalla Resultados Búsqueda (Usuario logado)

Texto



#### 7.3.9.- Pantalla Resultados Búsqueda (Usuario NO logado)

Texto

