

---

This is the **published version** of the bachelor thesis:

Valdés Ochoa, Katia Camila; Ferrero Beato, Ignacio, dir. SITUA : Aplicació per estudiar les desigualtats socials amb una perspectiva interseccional. 2020. (1373 Màster Universitari en Geoinformació)

---

This version is available at <https://ddd.uab.cat/record/231426>

under the terms of the  license

# Situa

*Aplicación para estudiar las desigualdades sociales con una perspectiva interseccional.*

**Proyecto de Final de Máster**  
*Máster en Geoinformación,  
Desarrollo de geoaplicaciones, Universitat Autònoma de Barcelona*

*Katia Valdés Ochoa*

## Contenido

RESUMEN .....	1
1 INTRODUCCIÓN .....	2
1.1. Antecedente .....	2
1.2. Estado de la cuestión .....	3
1.3. Marco Institucional .....	5
1.4. Glosario de Términos .....	6
2 OBJETIVOS .....	7
2.1. Objetivo General: .....	7
2.2. Objetivos Específicos: .....	7
3 DESARROLLO DEL PROYECTO .....	8
3.1. Propósito .....	8
3.2. Alcance .....	8
3.3. Visión general .....	8
3.4. Capacidades y condiciones .....	9
3.5. Restricciones .....	9
3.6. Suposiciones y dependencias .....	9
3.7. Planificación inicial .....	9
4 ANÁLISIS DE REQUERIMIENTOS .....	13
4.1. Actores .....	13
4.2. Requerimientos funcionales: .....	13
4.3. Casos de uso .....	18
4.4. Requerimientos no funcionales .....	27
5 DISEÑO .....	29
5.1. Prototipo de interfaz usuario .....	29
5.2. Diseño de la base de datos .....	29
5.2.1. Diagrama EER (Enhanced entity-relationship) .....	30
5.2.2. Descripción de las entidades .....	30
5.2.3. Descripción de las asociaciones .....	32
5.3. Diseño Funcional .....	33
5.3.1. Arquitectura del sistema .....	33

5.3.2.	Mapa de navegación .....	34
5.4.	Diseño de vistas.....	34
5.4.1.	Aplicación móvil .....	34
6	IMPLEMENTACIÓN .....	38
6.1.	Tecnologías utilizadas.....	38
6.2.	Aplicación Cliente.....	38
6.2.1.	Tecnologías y lenguajes de programación .....	38
6.2.2.	Librerías de JavaScript y Framework:.....	40
6.2.3.	Configuración del entorno de la aplicación.....	40
6.2.4.	Programación de la aplicación .....	41
6.3.	Servidor .....	41
6.3.1.	Tecnologías y lenguajes de programación .....	41
6.3.2.	Framework .....	42
6.3.3.	Programación del servidor .....	42
6.4.	Entorno de desarrollo .....	42
6.5.	Implementación de base de datos .....	43
6.5.1.	Creando modelo de datos.....	43
6.5.2.	Crear base de datos.....	44
7	Pruebas Funcionales y puesta en marcha .....	45
7.1.	Pruebas.....	45
7.2.	Compilación de la aplicación.....	46
8	Conclusiones.....	47
9	Bibliografía .....	48
10	Anexos .....	1

## **RESUMEN**

El concepto de interseccionalidad defiende que no puede entenderse la experiencia de la opresión con un solo marco explicativo (ya sea el género, la clase social o la etnicidad), sino que hay que comprender que los diferentes ejes se relacionan y que es su intersección lo que condiciona nuestras experiencias. Así, se contribuye a una comprensión más compleja y dinámica de las estructuras de poder que parte de la diferencia entre categorías, intentando evitar exclusiones. En este contexto se desarrolla SITUA como una aplicación para estudiar las desigualdades sociales con una perspectiva interseccional, geográfica y emocional, la cual será una herramienta útil para recoger, analizar y exportar datos en formatos compatibles con clientes SIG (ARGIS, QGIS).

## **RESUM**

El concepte de interseccionalitat defensa que no es pot entendre l'experiència de l'opressió amb un sol marc explicatiu (ja sigui el gènere, la classe social o l'etnicitat), sinó que cal comprendre que els diferents eixos es relacionen i que és la seva intersecció el que condiciona les nostres experiències. Així, es contribueix a una comprensió més complexa i dinàmica de les estructures de poder que part de la diferència entre categories, intentant evitar exclusions. En aquest context se desenvolupa SITUA com una aplicació per estudiar les desigualtats socials amb una perspectiva interseccional, geogràfica i emocional, la qual serà una eina útil per recollir, analitzar i exportar dades en formats compatibles amb clients SIG (ARGIS, QGIS).

## **ABSTRACT**

The concept of intersectionality defends that the experience of oppression cannot be understood with a single explanatory framework (gender, social class or ethnicity), but rather that it is necessary to understand that the different axes are related and that their intersection is what that conditions our experiences. Thus, it contributes to a more complex and dynamic understanding of power structures that starts from the difference between categories, trying to avoid exclusions. In this context, SITUA is developed as an application to study social inequalities with an intersectional, geographic and emotional perspective, which will be a useful tool to collect, analyze and export data in formats compatible with GIS clients (ARGIS, QGIS).

# 1 INTRODUCCIÓN

## 1.1. *Antecedente*

En la actualidad los cartógrafos han visto la importancia del componente emocional de la relación humana con el espacio. Esto permite poder mapear los espacios a medida que las personas los experimentan. Sin embargo, debido a las dificultades en la recopilación de datos y a la falta sistemática de datos medidos sobre otros aspectos de la experiencia humana, solo existe una pequeña proporción de mapas construidos para representar las emociones y los espacios que no sean de censos y otros conjuntos de datos administrativos. (véase A.L. Griffin, J. McQuoid ,2012)

Las emociones son una de las características definitorias de ser humano, pero su presencia en los mapas y los datos espaciales no son comunes. Estas han sido históricamente despreciadas en el estudio sobre las desigualdades sociales, pero existen muchas formas de discriminación que solo pueden identificarse a partir de esta dimensión: la humillación, el miedo o la percepción de exclusión sirven como indicadores fundamentales de desigualdades. En relación con los lugares, la perspectiva geográfica tampoco suele estar presente en los debates sobre interseccionalidad, pero es justamente el lugar el que actúa como motor de las dinámicas interseccionales y lo que ayuda a comprenderlas de forma más fluida y cambiante. No implica la misma situación ser mujer joven cuando estás en casa de la familia, volviendo a casa sola de noche o de escalada con las amigas. Y seguramente la dinámica es muy distinta si esta chica es blanca, migrada o trans. Es justamente este dinamismo entre posiciones, lugares y emociones lo que se pretende recoger a través de nuestra aplicación. (véase RODÓ-DE-ZÁRATE, 2014)

El concepto de interseccionalidad fue introducido a finales de los años 80 por Kimberlé Crenshaw (1989) como una forma de describir las interconexiones existentes entre raza y género.

Este proyecto surge para complementar la herramienta de Relief Map o Mapas de Relieve de la Experiencia, la cual sirve para el estudio de las geografías de la interseccionalidad mostrando las relaciones entre tres dimensiones: las estructuras de poder (la social), las experiencias vividas (la psicológica) y los lugares (la geográfica). Estos mapas son una forma visual de mostrar las experiencias interseccionales en los espacios, yendo desde los espacios de opresión hasta los lugares de alivio o relief, los lugares dónde no se experimentan discriminaciones o malestares. Pero con estos mapas surge la idea de que se pudiera ir un paso más allá, que tal si también podemos obtener la posición estimada del usuario, no solo se representaría en relación a un espacio, las experiencias y relación de poder, también se incluye dónde se encuentra la persona. Esto nos permite generar mapas de calor respecto a las experiencias vividas por los usuarios en diferentes zonas, además conociendo el tipo de espacio y si fue discriminado. Estos mapas podrían ser tanto personales como globales. Con el simple detalle de agregar la localización aproximada de la persona se puede lograr un Sistema de Información Geográfica muy potente. Con esta

idea en mente surge SITUA, un aplicación que nos permite el estudio de las desigualdades sociales con una perspectiva interseccional sobre mapas cartográficos.

## 1.2. Estado de la cuestión

En esta sección se analizarán los productos existentes en el mercado actualmente y las funcionalidades que poseen cada una de ellas. De este modo, se puede determinar si existen en el mercado aplicaciones completas que logren englobar las desigualdades sociales con una perspectiva interseccional relacionando tres dimensiones: la social (posiciones o identidades de género, clase social, etnicidad, edad, etc.), la geográfica (lugares de la vida cotidiana) y la psicológica (efectos sobre las emociones) (véase RODÓ-DE-ZÁRATE, 2014)

La interseccionalidad es una teoría que pretende capturar las múltiples relaciones entre las estructuras de poder como el género, la raza, la clase, la sexualidad o la edad. En el campo de la geografía, recientemente se está consolidando como una aproximación relevante, pero las propuestas teóricas y metodológicas son aún escasas.

A continuación se muestran las principales aplicaciones que se han encontrado:

### No es No:

El objetivo de esta aplicación es contribuir a denunciar los casos de violencia machista ofreciendo un servicio sencillo de emergencias, ya que nos permite avisar con rapidez si nos encontramos en una situación de riesgo o si vemos a alguien en una situación similar llamando, en tan solo un click, al teléfono de emergencias. Actualmente no se encuentra disponible en google play. [W1]



Figura 1. APP No es No

### Pillada por ti:

Orientada a la sensibilización contra la violencia de género en adolescentes, utilizando el cómic como un elemento para atraer la atención de los jóvenes. El Ministerio de Sanidad, Política Social e Igualdad, la Secretaría de Estado de Igualdad y la Delegación de Gobierno para la Violencia de Género están detrás de esta propuesta [W1].



Figura 2. APP Pillada por ti

## Todas Unidas:

Proporciona diversas herramientas para ayudar y apoyar a las víctimas. Entre las funciones disponibles cuenta con un acceso rápido de llamada al teléfono 016, contra la violencia de género. Además, ofrece un widget que se puede colocar en la página principal del teléfono móvil para llamar al 016 de forma que no aparezca en el registro de llamadas [W1].

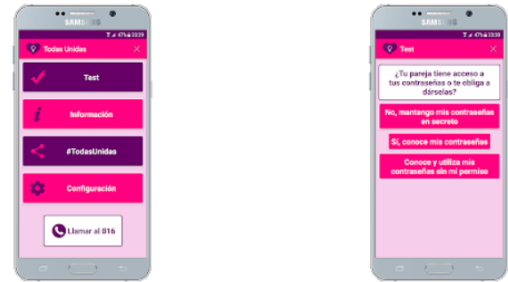


Figura 3. APP Todas Unidas

## Alerta Discriminación

Si crees que has sido víctima o has sido testigo de un hecho discriminatorio o delito de odio por cualquier motivo CUENTÁNOSLO

Mediante esta aplicación creada por RED ACOGE, puedes explicar lo sucedido y el contexto, de manera anónima o aportándonos tus datos si quieres. Siempre y cuando lo solicitéis, contamos con un equipo legal que podrá ayudarte a poner una denuncia o asesorarte sobre tu caso [W2].



Figura 4. APP\_Alerta Discriminación

Como se puede observar actualmente existen disímiles aplicaciones para registrar violencia de género, de las cuáles muchas funcionan como un botón de urgencia, es decir, si te sientes que tu vida pueda estar en peligro porque vives con un maltratador a través de la aplicación se hace una llamada de emergencia y la aplicación adquiere tu geolocalización. Otras que se han creado son para a través de cuestionarios puedas saber que tipo de relación de pareja tienes, o si sufres algún tipo de maltrato en la misma. La aplicación más interesante que se encontró fue alerta discriminación, desde la cual se puede registrar cualquier tipo de discriminación y explicar la causa y el lugar donde ha ocurrido la misma. Esta aplicación es la mas parecida a nuestra aplicación final, la diferencia fundamental es que esta no incluye posicionamiento o geolocalización del usuario y tampoco el registro de las emociones (factor psicológico). Esto hace que la nueva aplicación sea viable a desarrollar debido a que recoge datos sistematizados sobre la experiencia a partir de diferentes posiciones, en función del lugar y teniendo en cuenta las emociones.

El sistema que se propone desarrollar puede ser utilizado en diferentes ámbitos ya que la cuestión de las desigualdades sociales, las emociones y los lugares es un tema transversal. Los datos recopilados pueden usarse para tratar cuestiones como la LGTBIfobia, la islamofobia, el clasismo, el poder adulto, el racismo, la xenofobia, la discriminación por razón de diversidad funcional, de nacionalidad. (véase RODÓ-DE-ZÁRATE, 2014)



A continuación se enumeran algunas posibles aplicaciones del proyecto a desarrollar por ámbitos:

- **Investigación en ciencias sociales:** estudio de la experiencia vivida de las desigualdades sociales en temas como el derecho en la ciudad, la violencia machista, la discriminación laboral, el acoso escolar (*bullying*), el miedo del espacio público, el racismo institucional, etc.
- **Investigación en ciencias de la salud:** estudios sobre el efecto de ciertas enfermedades sobre las emociones en diferentes espacios o los efectos sobre la salud que tiene el hecho de sufrir discriminaciones.
- **Docencia:** uso como herramienta pedagógica para trabajar cuestiones como la propia interseccionalidad, las relaciones de poder, las geografías sociales y culturales o las relaciones concretas entre ejes.
- **Consultoría:** elaboración de diagnosis para planes de igualdad, sobre temas LGTBI (lesbianas, gais, bisexuales, transexuales e intersexuales) o sobre interculturalidad para administraciones u organizaciones.
- **Gestión de empresas u organizaciones:** elaboración de proyectos sobre salud laboral, planes internos o reflexión sobre las dinámicas internas de gestión de conflictos.
- **Acción social:** elaboración de proyectos orientados a la acción sobre colectivos concretos o sobre la ciudadanía en general. Pueden ser tanto para la reflexión sobre la propia posicionalidad en organizaciones que trabajan por el cambio social como para campañas de sensibilización.

### **1.3. Marco Institucional**

El máster oficial de Geoinformación es organizado por el departamento de Geografía de la Universidad Autónoma de Barcelona y el Instituto Cartográfico y Geológico de Cataluña. El presente proyecto constituye el Trabajo Final de Máster (TFM) asociado a las prácticas laborales con el objetivo de realizar un proyecto de aplicación real que permita afianzar y aplicar los conocimientos obtenidos durante el máster.

El proyecto ha sido realizado para el *Grupo de Geografía i Gènere del Departament de Geografia de la UAB*. El principal reto científico del grupo es el enfoque de género en la Geografía, y hacerlo principalmente, a través de dos líneas amplias de investigación: las Geografías de la Ciudad desde la perspectiva de Género y el Género y la Ruralidad. El segundo reto científico global del grupo es innovar en la utilización de la metodología cualitativa en la que el grupo tiene demostrada experiencia desde sus orígenes, con la exploración de nuevos métodos y técnicas, con el desarrollo de métodos colaborativos y de acción participativa. También en esta línea, nos planteamos consolidar la dimensión de género en el análisis de la ciudad a todas las escalas geográficas e incrementar el análisis interseccional a nivel teórico, empírico y metodológico en la investigación sobre el fenómeno de la inclusión social en el entorno urbano, así como analizar los fenómenos

migratorios teniendo en cuenta tanto la perspectiva espacial como la configuración del sentido del lugar

## 1.4. *Glosario de Términos*

**Aplicación:** Las aplicaciones son programas que se instalan en un dispositivo móvil, diseñadas para realizar una función específica que amplía o mejora la capacidad de funcionamiento del dispositivo.

**Sistema:** Un sistema es un conjunto de elementos relacionados entre sí que funciona como un todo. Cuando se habla de sistema en el documento es para referirse a toda la implementación desde el servidor

**Interseccionalidad:** Teoría que defiende que no puede entenderse la experiencia de la opresión con un solo marco explicativo (ya sea el género, la clase social o la etnicidad), sino que hay que comprender que los diferentes ejes se relacionan y que es su intersección lo que condiciona nuestras experiencias.

**WMS (servicio Web Map):** Definido por el OGC (Open Geospatial Consortium) produce mapas de datos referenciados espacialmente, de forma dinámica a partir de información geográfica. Este estándar internacional define un "mapa" como una representación de la información geográfica en forma de un archivo de imagen digital, sin los propios datos. Los mapas producidos por WMS se generan normalmente en un formato de imagen como PNG, GIF o JPEG, y, opcionalmente, como gráficos vectoriales en formato SVG (Scalable Vector Graphics) o WebCGM (Web Computer Graphics Metafile). El servicio WMS también permite la creación de una red de servidores distribuidos de mapas, a partir de los cuales los clientes pueden construir mapas a medida.

**OpenStreetMaps (OSM):** Proyecto de colaboración de código abierto y libre, encaminado a la creación, publicación y visualización de mapas para su uso en diferentes plataformas y dispositivos. La principal característica y lo que distingue a OSM, es que los mapas son creados y editados por los propios miembros, aportando cada uno su conocimiento. El proyecto de OpenStreetMap no puede ofrecer una API para facilitar que los mapas sean mostrados en las páginas de sitios web. Esto es asumido por varios servicios (openlayers, leaflet, googleMap). Es uno de los más populares servicios de mapas online de internet.

**Leaflet:** Proporciona una librería en JavaScript, libre y de código abierto con muchos recursos.

Posibilita insertar mapas de OSM y de otros servicios de mapas de internet, en las páginas de un blog o sitio web. Es usado por muchos sitios importantes de internet.

**Geolocalización:** Capacidad para obtener la ubicación geográfica real de un dispositivo de manera precisa. Actualmente esta geolocalización precisa es capaz de ubicar una dirección en el mapa y no sólo coordenadas.

## 2 OBJETIVOS

### 2.1. *Objetivo General:*

Diseñar, desarrollar e implementar una aplicación móvil que permita recoger datos sistematizados sobre la experiencia, a partir de diferentes posiciones y en función del lugar, para hacer estudios sobre desigualdades sociales con una perspectiva interseccional relacionando tres dimensiones: la social (posiciones o identidades de género, clase social, etnicidad, edad, etc.), la geográfica (lugares de la vida cotidiana) y la psicológica (efectos sobre las emociones).

### 2.2. *Objetivos Específicos:*

- I. Desarrollar una aplicación móvil que permita a los usuarios registrar cualquier tipo de discriminación. A través de un cuestionario se relacionan 3 dimensiones: geográfica (dónde ha pasado, puede ser por geolocalización directa o a posteriori, la social (cuál fue la causa: género, orígenes, sexo, entre otros) y la psicológica (efectos sobre las emociones).
- II. Diseñar e implementar la base de datos del sistema para almacenar la información de los usuarios y poder hacer estudios sobre desigualdad social.
- III. Desarrollar un servidor web para la gestión y análisis de la información. Exportar esta información en un Excel para realizar estudios con la misma.
- IV. Obtener una aplicación que tenga un buen funcionamiento y desempeño en gran parte de los dispositivos móviles y realizar los respectivos ajustes si son necesarios.
- V. Realizar pruebas para verificar el correcto funcionamiento de la aplicación.

## **3 DESARROLLO DEL PROYECTO**

### **3.1. Propósito**

El propósito del proyecto es desarrollar una aplicación que permita el estudio sobre las desigualdades sociales con una perspectiva interseccional, geográfica y emocional utilizando la tecnología para facilitar las tareas de recogida de datos, codificación, transcripción y análisis de resultados. Logrando poder visualizar de forma simplificada el dinamismo entre posiciones, lugares y emociones.

### **3.2. Alcance**

Por cuestión de tiempo se necesita simplificar el desarrollo de la aplicación, por lo tanto, en su primera versión solo podrán crear proyectos los administradores con la información provista por el gestor del proyecto. Una vez creado el proyecto se genera un código de proyecto, con el cual los usuarios implicados en el estudio pueden registrarse en la aplicación. Además, los estudios se harán solo sobre la región de Cataluña, pero en versiones posteriores se puede incluir el resto de España o incluso otros países.

Esta aplicación, aunque se desarrollará para que sea multiplataformas, solo se va a probar su correcto funcionamiento para celulares con sistema operativo Android en versiones superior a la V.4.0. También será una aplicación sin coste ninguno para el usuario, la cual, puede ser usada por cualquier persona estén o no asociados a algún proyecto.

### **3.3. Visión general**

La aplicación que se desarrollará permite el estudio de las desigualdades sociales de una forma muy sencilla. Cualquier usuario, tanto a título individual como asociado a un proyecto, puede crearse un perfil y registrar: la discriminación, el tipo de lugar (casa, espacios públicos, transporte, trabajo, entre otras opciones), el lugar donde ocurrió (puede ser la posición actual o registrar un punto en el mapa) o qué tipo de emociones sintió. Esta información que se obtiene, puede servir tanto para investigaciones académicas, proyectos relacionados con la consultoría para administraciones, la gestión de organizaciones y empresas, para la docencia o, incluso, para la acción social de asociaciones o colectivos. Mediante la geolocalización de las discriminaciones, pueden generarse mapas de calor donde se evidencia las zonas donde más desprotegidos se sientan los ciudadanos o realizar estudios de las experiencias emocionales en cada lugar, o el grado de bienestar/malestar, de los usuarios según su género, orientación sexual, clase social, identidad religiosa, la etnicidad o la diversidad funcional. Es importante mencionar que los datos provistos en el perfil no revelan la identidad del usuario y son de carácter general. Estos datos no aparecerán en los mapas porque no quiere hacerse uso de las categorías sociales (como mujer, hombre, trans) para la visualización sino de las estructuras de poder (género). Sin embargo, para generar filtros de búsqueda por el administrador si puede ser necesario saber cómo se identifican las personas.

Al desarrollar una aplicación móvil se puede hacer uso de las funcionalidades de los teléfonos inteligentes actuales, por lo que a través del GPS el usuario podrá registrar su posición actual o registrar una manualmente, ya sea mediante los filtros de búsqueda por dirección que se incorporarán o pulsando un punto en el mapa.

### **3.4. Capacidades y condiciones**

El sistema se podrá descargar de forma gratuita y será accesible para todo tipo de público. El sistema hará uso de algunas funcionalidades de tu teléfono inteligente (GPS) pero al instalar la aplicación se pedirá el previo autorizo del usuario para acceder a esas funcionalidades.

Para el uso de la aplicación es necesario estar conectado a internet

### **3.5. Restricciones**

✓ Sistema modulable y extensible:

El sistema deberá ser modular y fácilmente extensible. La inclusión de nuevas funcionalidades deberá ser sencilla.

✓ Rapidez:

El sistema deberá ser suficientemente rápido y eficaz.

✓ Responsive:

El sistema debe tener un diseño responsive, de esta manera se verá bien desde cualquier dispositivo electrónico (Tablet, móvil, PC...).

✓ Conexión a internet:

La principal restricción a la que se enfrenta la aplicación tiene relación con la conexión a Internet por parte del dispositivo móvil del usuario, sin una conexión a internet será imposible realizar el envío de información desde el dispositivo móvil al servidor, con la consecuente imposibilidad de almacenar los registros en la base de datos.

### **3.6. Suposiciones y dependencias**

Las dependencias del sistema:

- 1- Como plataforma de desarrollo de la aplicación se usará Phonegap.
- 2- Las librerías de mapas a usar son: Leaflet y GoogleMap.

### **3.7. Planificación inicial**

Para la planificación inicial se hizo un desglose de las fases que serán llevadas a cabo para el desarrollo del proyecto:

Fases del proyecto:

1. Gestión del proyecto:

En esta etapa se define el alcance que tiene el proyecto y la planificación temporal que se debe seguir para un buen desarrollo.

2. Análisis de los requisitos del sistema:

Se determinan los requerimientos funcionales y no funcionales del sistema, por tanto, se describen las necesidades del sistema en cuanto a las funcionalidades, propiedades, comportamiento, rendimiento e interconexión.

Esta etapa de análisis permite establecer los objetivos que debe cumplir el sistema, lo cual permite organizar el desarrollo del mismo usando esquemas y modelos.

3. Recopilación y evaluación de la información existente:

Esta fase consiste en un paso previo obligatorio para el diseño de la base de datos, para asegurar que la base de datos cumplirá con los objetivos del sistema, por lo que se analizan distintos factores, entre los cuales:

- ✓ Los datos que necesitamos almacenar y de dónde provienen.
- ✓ La información que los datos describen.
- ✓ Los usuarios de la base de datos y sus necesidades a la hora de acceder a los datos.

4. Diseño arquitectura del sistema:

Luego del análisis de los requisitos se puede realizar un diagrama que exponga todos los módulos del sistema y su interrelación. Esto permite organizar el sistema para los desarrolladores. La arquitectura del sistema se basa en MVC (Modelo → Vista → Controlador), que separa la definición y acceso a los datos (el modelo), de la interfaz de usuario (la vista) y la lógica del negocio (el controlador), ofreciendo como principal ventaja la independencia de cada uno de los componentes, sin afectar la implantación general.

5. Diseño de la base de datos:

Se realizan los modelos conceptual, lógico y físico para la organización y representación de los datos. Es importante también elegir el sistema de gestión de base de datos que se adapte mejor al proyecto. Para el mismo se usó MySQL.

6. Diseño de interfaz de usuario:

Es de vital importancia en esta etapa hacer el diseño de la aplicación atractiva y que su interacción con el usuario sea lo más intuitiva posible, en todo momento el diseño debe estar centrado en el usuario. Se realiza un prototipo de interfaz de usuario que permita que el cliente valide la interacción de la aplicación.

7. Implementación:

En esta fase se requiere seleccionar las herramientas y las tecnologías que permitan cumplir con los objetivos del sistema en general, estas deben ser alternativas de software libre o de código abierto. (aquí se podrá el lenguaje y framework que se usarán)

Esta fase se divide en varias etapas de desarrollo:

- i. En la primera etapa se implementará el servidor y se hará la conexión con la base de datos creada para el sistema, esto permitirá almacenar los datos de los usuarios para su posterior uso en el módulo de análisis de la información.
- ii. Se comenzará a desarrollar la aplicación móvil de acuerdo a los requerimientos de diseño establecidos y se hará la conexión con el servidor.
- iii. Implementar el módulo para registrar usuarios, por la ley de protección de datos estos usuarios no podrán ser identificados con la información que nos provean.
- iv. Crear módulo para registrar discriminaciones, que constará de distintos cuestionarios que el usuario rellenará. Además, el usuario también podrá indicar el lugar donde se ha producido la discriminación durante el día (a través de un mapa) o capturar su posición actual si quiere registrarlo directamente en el lugar del suceso.
- v. Implementar las restantes funcionalidades y probar el correcto funcionamiento de todos los módulos.

#### 8. Evaluación:

Hay que tener en cuenta que en cada fase de la implementación estos módulos se prueban por separado. En esta etapa se prueba el funcionamiento del sistema como un todo. Además, que se hacen pruebas con diferentes usuarios para que verifiquen el correcto funcionamiento de la aplicación. Se realizan pruebas de rendimiento y de seguridad también.

Fases	Descripción	Duración(días)
1	Gestión del proyecto	7
1.1	Alcance	1
1.2	Metodología	2
1.3	Planificación	2
2	Análisis y diseño	22
2.1	Requerimientos	3
2.2	Casos de Uso	3
2.3	Diseño base de datos	6
2.4	Diseño arquitectura	7
3	Implementación	39
3.1	Implementar base de datos	4

3.2	Desarrollar servidor y crear la conexión con la base de datos	4
3.3	Comenzar a desarrollar aplicación móvil con los requerimientos de diseño definidos	6
3.4	Desarrollar módulo de gestión de usuarios	7
3.5	Desarrollar módulo para registrar discriminaciones	7
3.6	Programación de funcionalidades adicionales en el sistema	6
4	Pruebas de control de calidad	8
4.1	Test de velocidad	2
4.2	Comprobación de todas las funcionalidades y control de errores	3
4.3	Pruebas de seguridad	1

Tabla 1. Fases del proyecto con su duración en días

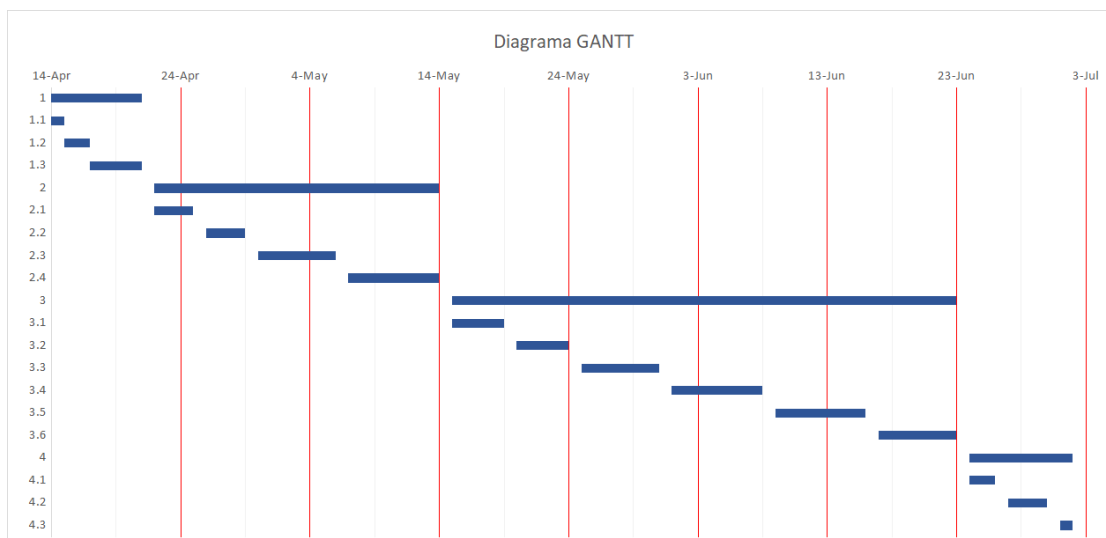


Figura 5. Diagrama de GANTT



## 4 ANÁLISIS DE REQUERIMIENTOS

### 4.1. Actores

Existen solo dos tipos de actores en el sistema:

- **Administrador:** Tendrá acceso a todas las funcionalidades del sistema. y puede crear proyectos y gestionar la información referente a los mismos.
- **Usuario:** Crea su perfil y puede registrar discriminaciones tanto por geolocalización como añadiendo manualmente la posición.

### 4.2. Requerimientos funcionales:

#### Usuario

Código: REQ_FUN_001	Usuario	Verificación:
El sistema gestionar los usuarios que interactuaran con la aplicación. Se gestionarán las diferentes funcionalidades asociadas a los diferentes tipos de usuarios.		
Requerimiento padre:		

#### **Buscar:**

Código: REQ_FUN_001_001	Buscar	Verificación:
El sistema permitirá hacer búsqueda de usuarios por los campos: <u>nombre de usuario</u> , <u>categoría</u> , <u>posición identitaria</u> , <u>tipo de lugar</u> , <u>medio transporte</u> , <u>emociones</u> , <u>código de proyecto</u> .		
Este filtro solo lo usará desde el administrador		
Requerimiento padre: REQ_FUN_001		

#### **Gestión:**

Código: REQ_FUN_001_002	Gestión	Verificación:
El sistema permitirá añadir, eliminar información de un usuario. Esto sería de acuerdo al nivel de acceso que tengan según su rol. Los administradores solo podrán eliminar entradas, por si hay errores, pero no modificar lo que ha introducido un usuario, mientras que los usuarios solo podrán añadir su información, ya que para el posterior análisis de los datos es importante que sean estables, no que puedan sufrir modificaciones.		
Requerimiento padre: REQ_FUN_001		

**Listado:**

Código: REQ_FUN_001_003	Listado	Verificación:
El sistema permitirá visualizar un listado de usuarios. Solo se podrá visualizar según los niveles de acceso que se les otorgué. Los administradores y los gestores de proyectos solo podrán visualizar los usuarios que participen en sus estudios.		
Requerimiento padre: REQ_FUN_001		

**Registro:**

Código: REQ_FUN_001_004	Registro	Verificación:
Un usuario debe registrarse en el sistema llenando un formulario con sus datos. La información que se le pide será nombre de usuario, contraseña. Solo existen dos roles, los usuarios que se registran desde la aplicación siempre cumplen el ROLE_USER.		
Requerimiento padre: REQ_FUN_001		

**Login:**

Código: REQ_FUN_001_005	Login	Verificación:
Un usuario accede al sistema con un formulario de acceso introduciendo usuario, contraseña. Desde la aplicación en el caso que participe en un proyecto ponen el código del proyecto al que están asociado.		
Requerimiento padre: REQ_FUN_001		

**Logout:**

Código: REQ_FUN_001_006	Logout	Verificación:
Un usuario podrá cerrar su sesión.		
Requerimiento padre: REQ_FUN_001		

**Perfil:**

Código: REQ_FUN_001_007	Perfil	Verificación:
El administrador tendrá un perfil donde visualizará toda la información que puede gestionar o la que pueda solo visualizar y exportar a excell.		
En esta primera versión los usuarios desde la aplicación no gestionan su perfil.		

Requerimiento padre: REQ_FUN_001
----------------------------------

**Proyecto:**

Código: REQ_FUN_002	Proyecto	Verificación:
El sistema permite gestionar proyectos para realizar estudios de desigualdad social. Solo el administrador del sistema puede gestionar esta información. Este se encarga de dar su código de proyecto a los usuarios que participen en su estudio.		
Requerimiento padre:		

**Gestión:**

Código: REQ_FUN_002_001	Gestión	Verificación:
El sistema permitirá añadir, eliminar o editar información de un proyecto. Los administradores pueden gestionar cualquier tipo de proyecto, mientras que los usuarios solo podrán participar en un proyecto teniendo el código del proyecto.		
Requerimiento padre: REQ_FUN_002		

**Buscar:**

Código: REQ_FUN_002_002	Buscar	Verificación:
El sistema permitirá hacer búsquedas proyectos por los campos: <u>código de proyecto</u> , <u>nombre del gestor</u> , <u>título</u> o <u>proyecto público</u> . Solo los administradores podrán realizar las búsquedas desde el panel de administración.		
Requerimiento padre: REQ_FUN_002		

**Perfil (Vista en detalle):**

Código: REQ_FUN_002_003	Perfil	Verificación:
Un proyecto tendrá una vista en detalle donde visualizará toda la información del proyecto.		
Requerimiento padre: REQ_FUN_002		

**Listado:**

Código: REQ_FUN_002_004	Listado	Verificación:
El sistema permitirá visualizar un listado de proyectos. Solo los administradores podrán visualizar todos los proyectos y exportar la información en formato Excell.		

Requerimiento padre: REQ_FUN_002
----------------------------------

### **Localizaciones:**

Código: REQ_FUN_003	Incidencia	Verificación:
El sistema permite gestionar localizaciones para expresar las emociones que sintió o si sufrió algún tipo de discriminación.		
Requerimiento padre:		

### **Gestión:**

Código: REQ_FUN_003_001	Gestión	Verificación:
Un usuario podrá añadir localizaciones desde la aplicación. Mientras que el administrador podrá solo visualizar y exportar la información.		
<b>Campos:</b>		
<u>posición</u> (latitud, longitud). Puede ser la posición actual del usuario o una posición que escoja en el mapa.		
<u>emociones</u> (se escogen de un listado, puede escogerse más de una)		
<u>grado bienestar/malestar</u>		
<u>tipo de discriminación</u> (opciones para escoger, se puede marcar de una, indica el tipo de discriminación que sentiste)		
<u>tipo de lugar</u> (opciones para escoger, solo se selecciona uno, si se escoge la opción medio de transporte, te sale para escoger el <u>tipo de transporte</u> )		
Requerimiento padre: REQ_FUN_003		

### **Tipo de lugar**

Código: REQ_FUN_003_002	Espacio público	Verificación:
El sistema permite gestionar (agregar, editar o eliminar) los tipos de lugares requerida en el sistema, solo el administrador tiene esta funcionalidad.		
Tipos de lugares:		
<ul style="list-style-type: none"><li>• Espai públic (carrer, plaça)</li><li>• Espai domèstic (casa)</li><li>• Comerç o servei (botiga, supermercat, gimnàs, centre comercial, entitat bancària)</li></ul>		

<ul style="list-style-type: none"> <li>• Espai laboral (feina)</li> <li>• Espai formatiu (escola, institut, centre formatius, universitat)</li> <li>• Centre sanitari (CAP, hospital)</li> <li>• Lloc de lleure (bar, restaurant, discoteca, cinema, entitat)</li> <li>• Transport públic (metro, bus, tren, taxi)</li> <li>• Oficina o servei de l'administració pública (oficina, presó, policia, jutjat)</li> </ul>
Requerimiento padre: REQ_FUN_003

### **Tipos de discriminación**

Código: REQ_FUN_003_003	Espacio público	Verificación:
<p>El sistema permite gestionar (agregar, editar o eliminar) las categorías requeridas en el sistema, solo el administrador tiene esta funcionalidad.</p> <p>Tipo de discriminación:</p> <ul style="list-style-type: none"> <li>• Gènere</li> <li>• Orientació sexual</li> <li>• Origen</li> <li>• Etnicitat</li> <li>• Religió</li> <li>• Classe social</li> <li>• Edat</li> <li>• Discapacitat</li> </ul>		
Requerimiento padre: REQ_FUN_003		

### **Tipo de transporte**

Código: REQ_FUN_003_004	Espacio público	Verificación:
<p>El sistema permite gestionar (agregar, editar o eliminar) los tipos de transporte requeridos en el sistema, solo el administrador tiene esta funcionalidad.</p> <p>Tipos de transporte:</p> <ul style="list-style-type: none"> <li>• Metro (si escoges está opción, tienes que indicar la línea)</li> <li>• Autobús (si escoges está opción, tienes que indicar la línea)</li> <li>• Tren (si escoges está opción, tienes que indicar la línea)</li> <li>• Taxi</li> </ul>		
Requerimiento padre: REQ_FUN_003		

### **Listado:**

Código: REQ_FUN_003_006	Listado	Verificación:
-------------------------	---------	---------------

El administrador podrá visualizar un listado de localizaciones y exportar la información por cada usuario.

Requerimiento padre: REQ\_FUN\_003

### **Geolocalización**

Código: REQ\_FUN\_002

Geolocalización

Verificación:

Existe un botón que te permite mostrar tu ubicación en el mapa. Esto es posible al activar la localización del dispositivo mediante la funcionalidad del GPS del teléfono inteligente.

Requerimiento padre:

## **4.3. Casos de uso**

### **DIAGRAMA 1: GESTIÓN DE USUARIO**

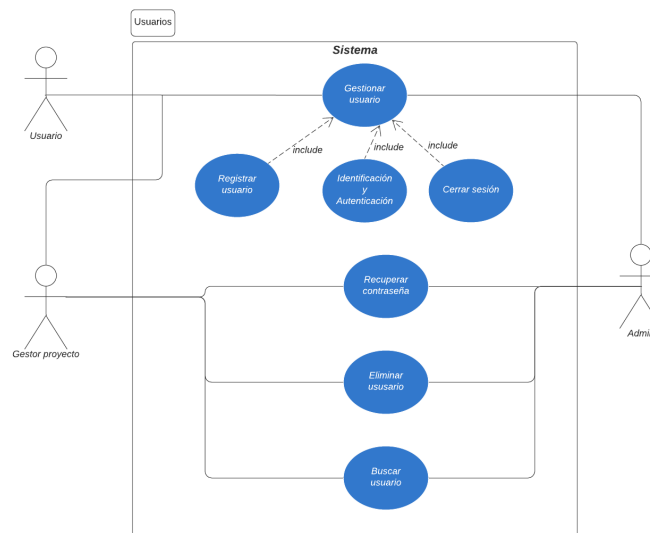


Figura 6. Diagrama 1. Gestión de usuario

### **CU1: Registrar usuario**

**Actores:** Todos los usuarios

**Precondición:** El sistema se iniciará sin problemas críticos: Acceso a la base de datos de forma correcta.

**Disparador:** El usuario instala la aplicación y solicita registrarse en la misma.

**Flujo principal:**

1. La aplicación mostrará el formulario de registro y solicitará nombre de usuario, contraseña y aceptar los términos y condiciones.
2. El usuario introduce los campos solicitados.
3. El sistema validará que el valor de cada campo es correcto.
4. El sistema accederá a la base de datos e insertará el nuevo usuario.
5. El sistema redireccionará a la página de autenticación de usuario.

**Flujos alternativos:**

1. En el paso 3, si los campos introducidos son incorrectos se mostrara un mensaje de error indicando que se deben corregir y el proceso volverá al paso 1.
2. En el paso 3, si al verificar si el usuario existe se mostrara un mensaje de error indicando que el usuario ya existe y que se debe modificar el nombre de usuario. Además el proceso volverá al paso 1.

**Postcondición:** -

**CU2: Identificación y Autenticación**

**Actores:** Todos los usuarios

**Precondición:** Debes estar registrado en el sistema.

**Disparador:** El usuario ya registrado quiere entrar en la aplicación.

**Flujo principal:**

1. El sistema mostrará el formulario para autenticarse y solicitará los campos nombre de usuario, contraseña y código de proyecto.
2. El usuario introducirá los campos solicitados.
3. El sistema validará que el valor de cada campo es correcto.
4. El sistema accederá a la base de datos y comprobará que los datos introducidos existen en la base de datos.
5. El sistema autenticará al usuario con sus datos y lo redireccionará a la vista correspondiente según el tipo de usuario.

**Flujos alternativos:**

1. En el paso 3, si los campos introducidos son incorrectos se mostrará un mensaje de error indicando que se deben corregir y el proceso volverá al paso 1.
2. En el paso 3, si al verificar los datos el usuario no existe en el sistema entonces el proceso volverá al paso 1 y le indica al usuario que debe registrarse en el sistema.

**Postcondición:**

El sistema quedara configurado para habilitar al usuario el acceso a los módulos según el tipo de usuario. Todos los otros módulos que no tengan permisos con este rol deberán desactivarse.

### **CU3: Cerrar sesión**

**Actores:** Todos los usuarios

**Precondición:** Debes estar registrado y autenticado en la aplicación.

**Disparador:** El usuario quiere cerrar su sesión.

#### **Flujo principal:**

La aplicación mostrará en el menú la opción de cerrar sesión.

El usuario interactuará con esa opción.

El sistema cerrará la sesión del usuario.

**Flujos alternativos:** -

**Postcondición:** -

### **CU5: Buscar usuario**

**Actores:** Administrador

**Precondición:** Debe existir al menos un usuario para que el buscador funcione.

**Disparador:** Se solicita una búsqueda de los usuarios registrados en la aplicación

#### **Flujo principal:**

1. El gestor de proyecto y el administrador dispondrán de un pequeño buscador, únicamente orientado a la búsqueda de usuarios.
2. El gestor de proyecto o el administrador escribirá algún dato del usuario nombre de usuario, categoría, entre otros y el buscador mostrará los que haya coincidido con lo escrito.

#### **Flujos alternativos:**

1. En el caso que no se haya encontrado nada revisar que se haya escrito correctamente

**Postcondición:** -

### **CU6: Eliminar usuario**

**Actores:** Administrador

**Precondición:** Debe existir al menos un usuario para poder eliminar.

**Disparador:** Se solicita eliminar uno o más usuarios registrados en la aplicación

#### **Flujo principal:**

1. El administrador visualizará al usuario y dispondrá de un botón que le permitirá



eliminar usuarios del sistema.

2. El administrador afirmará y el usuario será eliminado

**Flujos alternativos: -**

**Postcondición: -**

## DIAGRAMA 2. GESTIÓN DE PERFIL

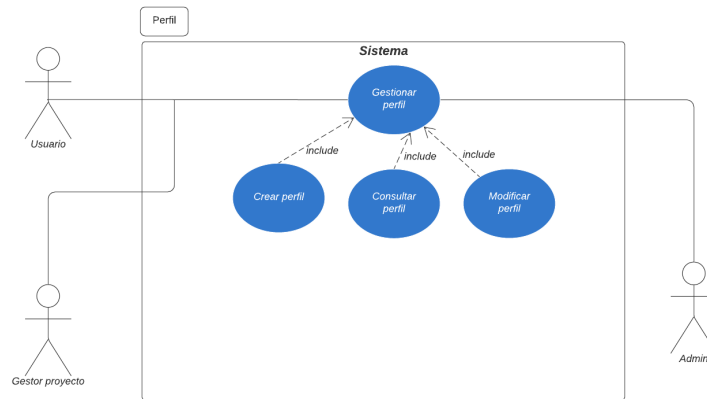


Figura 7 Diagrama 2. Gestión de perfil

### CU4: Crear perfil

**Actores:** Usuario

**Precondición:** El usuario debe estar registrado y autenticado en la aplicación.

**Disparador:** Para realizar el estudio es necesario que el usuario provea la información básica en su perfil

**Flujo principal:**

1. Una vez registrado y autenticado el sistema carga un formulario para que el usuario provea los datos de perfil
2. El usuario rellena el formulario y al terminar presiona el botón añadir para crear el perfil
3. El sistema lo redirigirá a la vista para registrar lugar.

**Flujos alternativos: -**

**Postcondición: -**

## DIAGRAMA 3: GESTIÓN DE PROYECTO

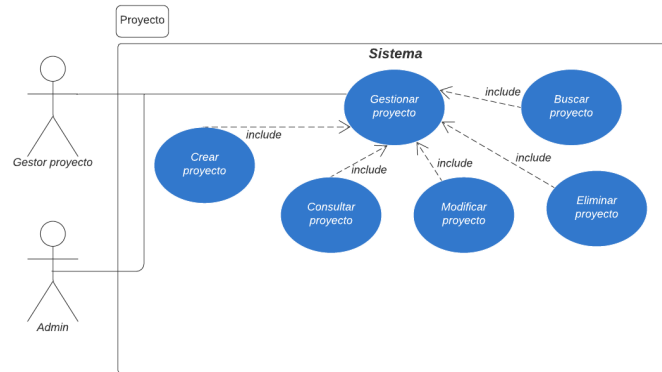


Figura 8. Diagrama 3. Gestión de proyecto

### **CU7: Crear proyecto**

**Actores:** Administrador

**Precondición:** El administrador debe estar registrado y autenticado en el sistema.

**Disparador:** El administrador quieren crear un proyecto para comenzar a hacer un estudio social.

#### **Flujo principal:**

1. El administrador desde su perfil tiene un botón para crear proyecto y redirecciona a un formulario
2. El administrador introduce los campos solicitados y al terminar presiona el botón añadir para crear el proyecto
3. El sistema validará que el valor de cada campo es correcto.
4. El sistema accederá a la base de datos e insertará el nuevo proyecto
5. El sistema lo redirigirá a su perfil desde donde puede ver todos los proyectos que tiene o crear otro nuevo.

#### **Flujos alternativos:**

1. En el paso 3, si los campos introducidos son incorrectos se mostrara un mensaje de error indicando que se deben corregir y el proceso volverá al paso 2.

**Postcondición:** -

### **CU8: Modificar proyecto**

**Actores:** Administrador

**Precondición:** El administrador debe estar autenticado en el sistema.

**Disparador:** El administrador solicita actualizar la información de un proyecto.

**Flujo principal:**

1. El administrador desde su perfil tienen todos los proyectos que ha creado y la opción para editar cada proyecto
2. Presiono la opción editar del proyecto a modificar el cual redirecciona a un formulario
3. El administrador realizará los cambios pertinentes y tendrá un botón guardar, para mantener los cambios realizados del proyecto.
4. El sistema validará que el valor de cada campo es correcto
5. El sistema lo redirigirá a su perfil desde donde puede ver todos los proyectos que tiene.

**Flujos alternativos:**

**Postcondición:** El sistema habrá modificado los datos del proyecto que se haya editado.

#### **CU9: Consultar proyecto**

**Actores:** Administrador

**Precondición:** El administrador autenticado en el sistema.

**Disparador:** El administrador quiere visualizar la información detallada de un proyecto

**Flujo principal:**

1. El administrador desde su perfil tiene todos los proyectos que ha creado y la opción para ver la vista en detalle de cada proyecto
2. El administrador escoge esa opción en un proyecto y lo redirigirá a su vista en detalle donde verá toda la información del proyecto. También desde esa vista podrá editar la información del proyecto.

**Flujos alternativos:** -

**Postcondición:** -

#### **CU10: Buscar proyecto**

**Actores:** Administrador

**Precondición:** Debe existir al menos un proyecto para que el buscador funcione.

**Disparador:** El gestor de proyecto o el administrador solicita una búsqueda de los proyectos pertenecientes a la aplicación

**Flujo principal:**

1. El administrador dispondrá de un pequeño buscador, únicamente orientado a la búsqueda de proyectos.
2. El administrador escribirá algún dato del proyecto título, código de proyecto, entre otros y el buscador mostrará los que haya coincidido con lo escrito.

**Flujos alternativos:**

1. En el caso que no se haya encontrado nada revisar que se haya escrito correctamente

**Postcondición:** -

**CU11: Eliminar proyecto**

**Actores:** Administrador

**Precondición:** Debe existir al menos un proyecto para poder eliminar.

**Disparador:** Se solicita eliminar uno o más proyectos del sistema.

**Flujo principal:**

1. El administrador visualizará todos sus proyectos y dispondrá de un botón que le permitirá eliminar proyectos del sistema.
2. El administrador presionará el botón y le saldrá un aviso conforme está eliminando ese proyecto
3. El administrador afirmará y el proyecto será eliminado

**Flujos alternativos:**

En el paso 3 si el administrador elige la opción “No”, no se borrará ningún proyecto

**Postcondición:** -

**DIAGRAMA 4: GESTIÓN DE LOCALIZACIÓN**

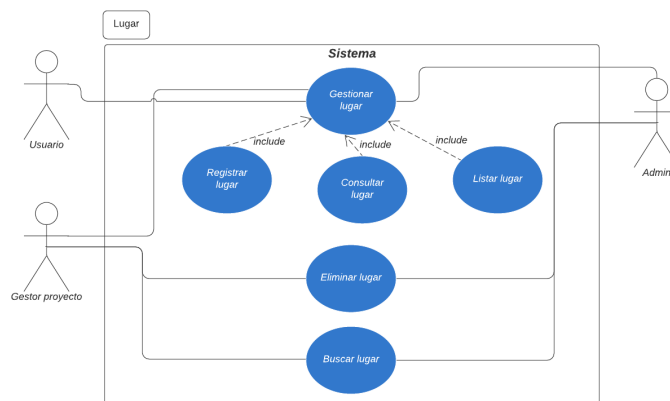


Figura 9. Diagrama 4. Gestión de localización

## **CU12: Registrar lugar**

**Actores:** Todos los usuarios

**Precondición:** Debe estar autenticado en la aplicación

**Disparador:** Se solicita registrar un nuevo lugar

**Flujo principal:**

1. El usuario solicita registrar un nuevo lugar y escoge el tipo de lugar.
2. El usuario registra la posición del lugar puede ser por geolocalización o manualmente
3. El usuario registra las emociones que sintió en el lugar
4. El usuario registra si sintió algún tipo de discriminación y si es cierto indica que tipo de discriminación sufrió.
5. Una vez rellenado los datos termina el registro y sale de la aplicación o puede seguir registrando lugares.

**Flujos alternativos:**

En el paso 1 si el tipo de lugar es un medio de transporte, se escoge el tipo de transporte y también se registra la línea en caso de que sea bus, metro o tren.

**Postcondición:** -

## **CU13: Listar lugares**

**Actores:** Administrador

**Precondición:** Debe estar autenticado en el sistema

**Disparador:** Se solicita listar los lugares según el permisos de acceso que tengas.

**Flujo principal:**

1. El administrador indica al sistema que quiere ver todos los lugares registrados.
2. El sistema muestra todos los lugares que se han encontrado por fecha de registro.
3. El sistema permite ver una vista en detalle de cada lugar.

**Flujos alternativos:**

**Postcondición:** Teniendo el listado de lugares se puede acceder a la vista en detalle de cada uno.

## **CU14: Consultar lugar**

**Actores:** Administrador

**Precondición:** Debe estar autenticado en el sistema

**Disparador:** Se solicita listar los lugares según los permisos de acceso que tengas.

**Flujo principal:**

1. El administrador indica al sistema que quiere ver la información en detalle de un lugar escogido de la lista de lugares.
2. El sistema muestra la información en detalle del lugar.

**Flujos alternativos:** -

**Postcondición:** -

### **CU14: Buscar lugar**

**Actores:** Administrador

**Precondición:** Debe existir al menos un lugar para que el buscador funcione.

**Disparador:** El administrador solicita una búsqueda de los lugares pertenecientes a la aplicación

**Flujo principal:**

1. El administrador dispondrá de un pequeño buscador, únicamente orientado a la búsqueda de lugares.
2. El administrador escribirá algún dato del proyecto tipo de lugar, emociones, categoría y el buscador mostrará los que haya coincidido con lo escrito.

**Flujos alternativos:**

1. En el caso que no se haya encontrado nada revisar que se haya escrito correctamente

**Postcondición:** -

### **DIAGRAMA 5: EXPORTAR INFORMACIÓN**

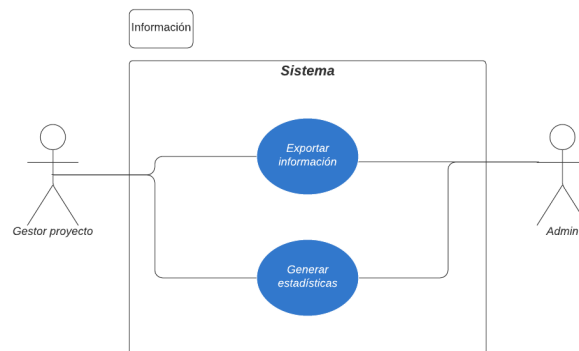


Figura 10. Diagrama 5. Información a exportar

### **CU15: Exportar información**

**Actores:** Administrador

**Precondición:** Debe existir al menos un usuario para que exportar información.

**Disparador:** El gestor de proyecto o el administrador solicita exportar la información del estudio que están realizando

**Flujo principal:**

1. El administrador tendrá el listado de proyectos y por proyecto la información de los usuarios que participaron
2. El administrador con un botón podrá descargar la información por proyecto en un fichero csv.

**Flujos alternativos:**

**Postcondición:** El sistema descargara el fichero.

**4.4. Requerimientos no funcionales****Servidor de Base de Datos:**

Código: REQ_NOFUN_002	Carga	Verificación:
El sistema debe permitir la conexión con un servidor de base de datos MySQL		
Requerimiento padre:		

**Responsive:**

Código: REQ_NOFUN_003	Carga	Verificación:
El sistema debe ser responsive, es decir, se debe visualizar bien en celulares, tablets y computadoras.		
Requerimiento padre:		

**Accesible Web:**

Código: REQ_NOFUN_004	Carga	Verificación:
Es necesaria una conexión a internet para que sea accesible desde cualquier parte.		
Requerimiento padre:		

**Sistema Operativo:**

Código: REQ_NOFUN_005	Carga	Verificación:
-----------------------	-------	---------------

El sistema debe ser compatible para Android, a partir de la versión 4.0.
Requerimiento padre:

**Framework:**

Código: REQ_NOFUN_006	Carga	Verificación:
La aplicación móvil se desarrolla usando el framework PhoneGapp, este permite crear aplicaciones para teléfonos inteligentes de distintas plataformas.		
Requerimiento padre:		

**Lenguajes de programación:**

Código: REQ_NOFUN_007	Carga	Verificación:
El sistema debe desarrollarse usando HTML, CSS, Javascript todo esto para el lado del cliente y PHP para implementar el servidor.		
Requerimiento padre:		

**Velocidad y latencia:**

Código: REQ_NOFUN_009	Carga	Verificación:
El sistema ha de ser capaz de dar respuesta rápida a las peticiones del usuario, ya que sino el usuario podría cansarse y dejar de utilizar la aplicación. El mismo debe responder en menos de 1segundo.		
Requerimiento padre:		

**Estilo:**

Código: REQ_NOFUN_010	Carga	Verificación:
El sistema ha de tener un diseño simple para que los usuarios se sientan cómodos utilizándolo.		
Requerimiento padre:		



## 5 DISEÑO

### 5.1. *Prototipo de interfaz usuario*

Hacer un prototipo de aplicación móvil es una de las técnicas más sencillas para previsualizar cómo va a quedar la aplicación a desarrollar, en cuestión de diseño y de sus funcionalidades básicas, y sin tener que empezar a programar ni una línea de código. Además, te va a permitir ver de manera muy fiel cómo va a quedar la aplicación y si es necesario optimizar el diseño, apariencia y usabilidad de la misma, y a ampliar las ideas iniciales.

En función de su grado de complejidad o semejanza con el diseño final, podemos distinguir entre prototipos de baja, media o alta fidelidad. En el caso concreto del presente proyecto se ha decidido optar por las representaciones de media fidelidad, también denominados *Mockup*. Estos muestran la parte más visual del proyecto. Presentan la estructura de la información, los contenidos y las funcionalidades de forma estática. Representan muy bien cómo va a ser la aplicación, pero con una inversión mucho menor de lo que supondría un prototipo propiamente dicho.

Para realizar el prototipo se usó el programa *Balsamiq Mockups 3*. Esta herramienta permite elaborar y exportar el prototipo para compartirlo con el cliente. En el anexo 9 se muestran estas vistas. En total se diseñaron 21 interfaces diferentes, de acuerdo a las solicitudes del cliente.

### 5.2. *Diseño de la base de datos*

La principal causa de fracaso en los sistemas de información es la poca confianza en las metodologías de diseño de base de datos. El diseño de la base de datos es una fase crucial pues es donde se define la estructura de nuestro modelo de datos

En este apartado se ilustra el diagrama de entidad, con sus atributos, las relaciones entre ellas, su cardinalidad del sistema a desarrollar.

### 5.2.1. Diagrama EER (Enhanced entity-relationship)

Con el gestor de base de datos MySQL Workbench se hizo el diagrama de entidades que se muestra a continuación. Consiste en un modelo relacional con una descripción de tablas, columnas y sus relaciones.

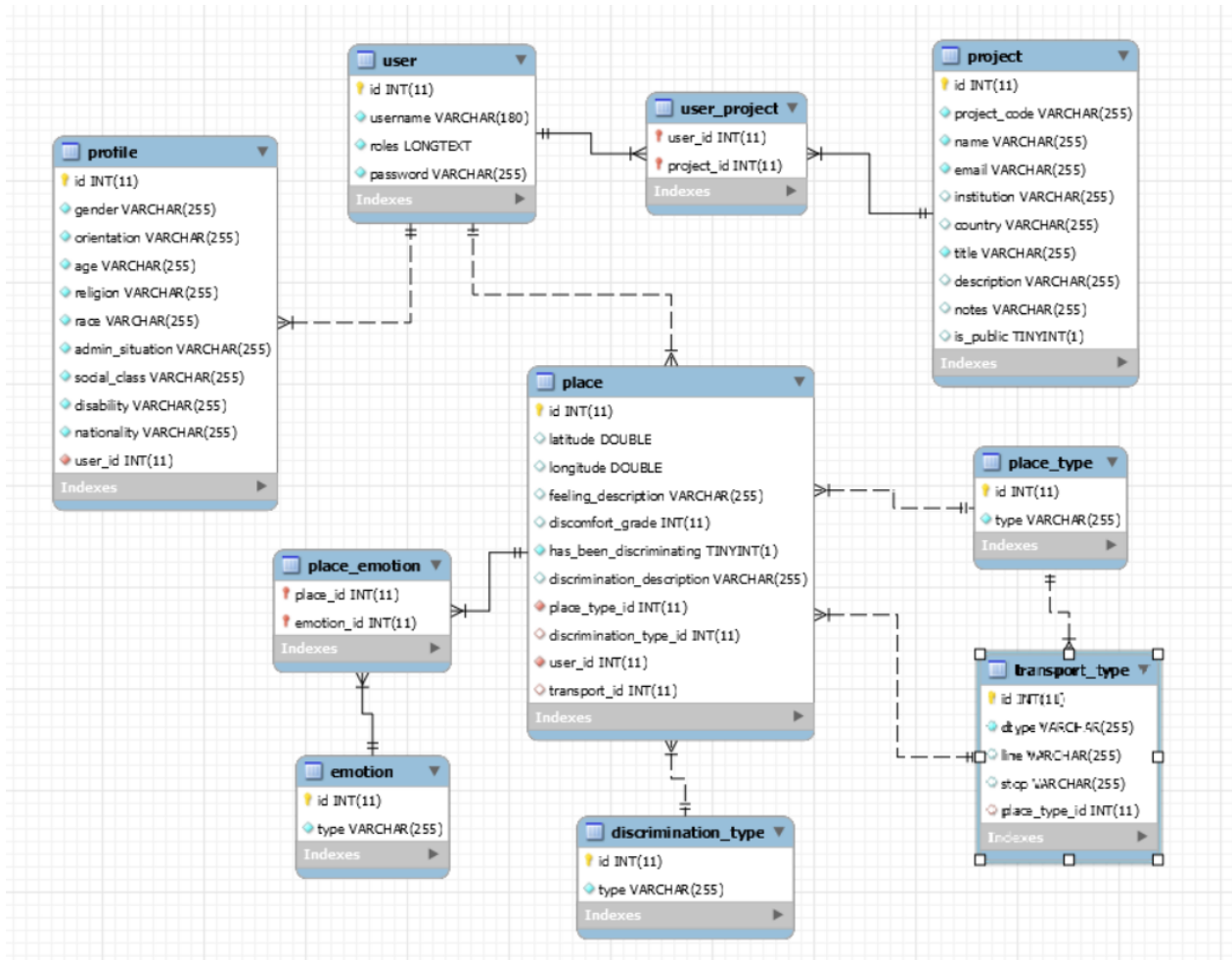


Figura 11. Diagrama EER

### 5.2.2. Descripción de las entidades

#### User:

- Descripción: Representa los usuarios que se registran en la aplicación.
- Atributos:
  - username: Define el nombre con que se registra el usuario dentro de la aplicación. Este atributo es único.
  - password: Define la contraseña que introduce el usuario

- role: Define el role de los usuarios dentro de la aplicación. Existen dos roles ROLE\_USER o ROLE\_ADMIN. Por defecto todos los usuarios al loguearse tienen ROLE\_USER, este campo el usuario no lo rellena

### **Profile:**

- Descripción: Representa la información de perfil del usuario. Esta información la rellena solo cuando se registra en la aplicación. Todos los campos son obligatorios a rellena.
- Atributos:
  - gender: Identidad de género. Es un campo de selección
  - orientation: Orientación sexual. Es un campo de selección
  - age: Rango de edad. Es un campo de selección
  - religion: Creencia religiosa que practica. Es un campo de selección
  - race: Origen étnico. Es un campo de selección
  - admin\_situation: Situación legal administrativa. Es un campo de selección.
  - social\_class: Nivel socio-económico. Es un campo de selección.
  - disability: Diversidad funcional. Es un campo de selección
  - nationality: Nacionalidad. Es un campo de selección, escoge el país.

### **Emotion:**

- Descripción: Los distintos tipos de emociones. El administrador puede agregar nuevas
- Atributos:
  - type: tipo de emoción (rabia, angustia, alegría, entre otras.)

### **PlaceEmotion:**

- Descripción: Representa las emociones que sintió el usuario en un lugar.

### **DiscriminationType:**

- Descripción: Representa los distintos tipos de discriminaciones. El administrador puede agregar nuevas
- Atributos:
  - type: tipo de discriminación (identidad de género, racial, edad, etc)

### **PlaceType:**

- Descripción: Representa los distintos tipos de lugares. El administrador puede agregar nuevas
- Atributos:
  - type: tipo de lugar (trabajo, casa, transporte público, entre otros)

### **TransportType:**

- Descripción: Representa los distintos tipos de transportes públicos.
- Atributos:
  - type: tipo de transporte (metro, tren, autobús, taxi)
  - línea: En caso de metro, tren y autobús puedes incluir la línea
  - parada: En caso de metro, tren y autobús puedes incluir la parada

### **Place:**

- Descripción: Representa la ubicación de un determinado lugar, el grado de bienestar-malestar, la descripción de cómo se sintió y de la discriminación que sufrió.
- Atributos:
  - latitude: Define la coordenada geográfica latitud. Este campo no lo rellena el usuario, se obtiene automáticamente del punto que el usuario registra
  - longitude: Define la coordenada geográfica longitud. Este campo no lo rellena el usuario, se obtiene automáticamente del punto que el usuario registra
  - feeling\_description: Describe como se sintió en el lugar
  - discomfort\_grade: Grado de bienestar/malestar
  - has\_been\_discriminating: Pregunta si se sintió discriminado
  - discrimination\_description: Describir un poco lo que pasó para que la persona se sintiera discriminada.

### **Project:**

- Descripción: Permite crear nuevos proyectos con una perspectiva interseccional
- Atributos:
  - project\_code: Define el código de un proyecto. Este valor se genera automáticamente
  - name: Define el nombre del encargado del proyecto
  - email: Correo electrónico
  - institution: Define la institución encargada del proyecto
  - country: Define el país donde se realiza el estudio
  - title: Define el nombre del proyecto
  - description: Pequeña descripción donde se expone en qué consiste
  - note: Notas adicionales que se quieran incluir
  - is\_public: Declara si el proyecto puede ser público o es una investigación privada

### **UserProject:**

- Descripción: Representa todos los usuarios y los proyectos a los que pertenecen.

## **5.2.3. Descripción de las asociaciones**

A continuación, se explican las relaciones existentes entre las diferentes entidades de la base de datos.

**User-Profile:** La relación es 1 a 1 ya que un usuario tiene un único perfil.

**User-Project:** La relación es \* a \*, un usuario puede participar en varios proyectos mientras que en un proyecto hay varios usuarios realizando el estudio. Se crea la nueva clase user-project.

**User-Place:** La relación es 1 a \*, un usuario puede registrar muchos lugares en los cuales define cómo se sintió y si sufrió algún tipo de discriminación.

**Place-Emotion:** La relación es \* a \*, en un lugar el usuario puede sentir varias emociones. Se crea la nueva clase place\_emotion.

**Place-DiscriminationType:** La relación es \* a 1, un tipo de discriminación puede sentirse en varios lugares por diferentes usuarios.

**Place-PlaceType:** La relación es \* a 1, un tipo de lugar puede ser incluido por varios usuarios al registrar un lugar.

**Place-TransportType:** La relación es \* a 1, un tipo de transporte puede ser incluido por varios usuarios al registrar un lugar donde el tipo de lugar sea medio de transporte.

### 5.3. Diseño Funcional

#### 5.3.1. Arquitectura del sistema

La arquitectura del sistema será cliente-servidor. El cliente será la aplicación móvil que se conecta, a través de Internet, a un servidor que será el encargado de almacenar y procesar la información recibida por el usuario. El dibujo de esta arquitectura se muestra en la figura 12.

En lo referente a la información, estará compuesta esencialmente por registros de lugares con las emociones y la discriminación que tuvo almacenados en una base de datos de tipo alfanumérico que será gestionada en un entorno único mediante MySQL Workbench. Dicha información será visualizable en dos plataformas diferentes, como son un cliente web móvil y el panel de administrador en el servidor, únicamente será explotable a través del panel de administración, mientras que solamente se adquirirá y recogerá información a través de la plataforma móvil. Desde el panel de administración se crean los proyectos.

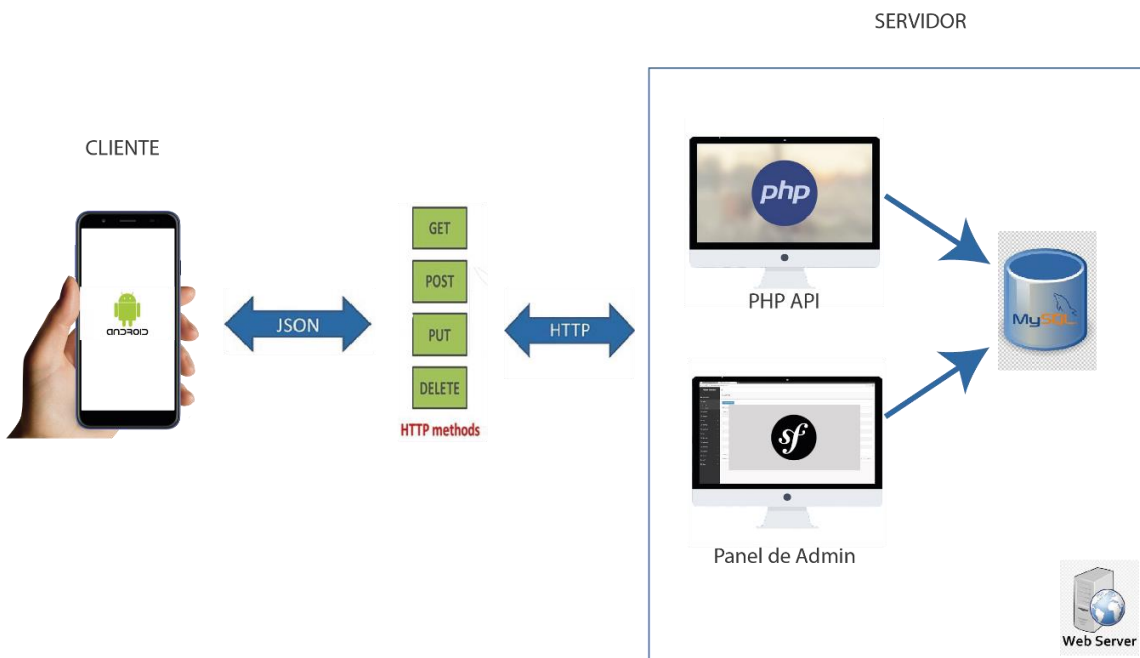


Figura 12. Arquitectura Cliente-Servidor del sistema

### 5.3.2. Mapa de navegación

En esta sección, se muestra el mapa de navegación de la aplicación móvil, con el que se representa de forma esquemática el funcionamiento de la aplicación.

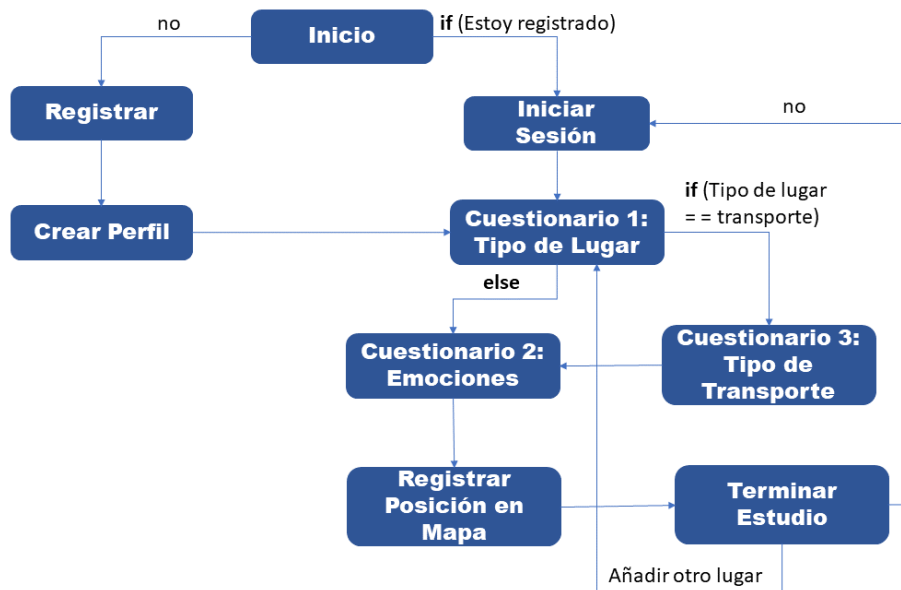


Figura 13. Mapa de navegación

## 5.4. Diseño de vistas

### 5.4.1. Aplicación móvil

Para que un usuario pueda acceder a todas las funcionalidades que se han especificado previamente, se ha realizado una aplicación Android con la que el usuario puede interactuar.

La aplicación móvil cuenta con un total de 9 vistas distintas. De esta manera, la totalidad de las vistas del cliente móvil se contemplan a continuación mayoritariamente ordenadas según su orden de aparición conforme el usuario hace uso de la aplicación.

En la figura 14 de muestra la pantalla de inicio, siempre que entras a la aplicación debe autenticarse o registrarse en caso de que sea la primera vez que acceda. Las figuras 15 y 16 respectivamente muestran las pantallas para registrarse o autenticarse en el sistema, cuando haga el registro debe aceptar los términos y condiciones y solo necesita introducir un nombre de usuario y contraseña, el nombre de usuario es único en el sistema, en caso de exista ya, se le pedirá introducir otro nombre de usuario, por cuestiones de seguridad la contraseña se almacena encriptada en la base de datos. Para autenticarse además de introducir su nombre de usuario y contraseña, también tiene la posibilidad de poner el código de proyecto al cual pertenece, en caso de no pertenecer a ningún proyecto puede

dejar este campo en blanco. Cuando se registra en el sistema y se autentica por primera vez debe rellenar la información de su perfil, todos los campos son obligatorios ya que esta opción solo aparece una vez y luego no se puede modificar. Se implementó de esta forma por la necesidad de tener estos datos para la fiabilidad de los estudios y realmente es improbable que el usuario regrese a modificar la información, esta pantalla se muestra en la figura 17.

Las figuras 18 a la 20 muestran los cuestionarios que debe rellenar el usuario para registrar un lugar, primero se le pregunta por el tipo de espacio (figura 18), luego las emociones y el grado de bienestar-malestar que sintió (figura 19). Si el tipo de lugar que quiere registrar es un medio de transporte aparece otro cuestionario donde indica el tipo de transporte (metro, tren, bus o taxi) y en caso de que recuerde puede poner la línea y la parada (figura 20).

Una vez completado los cuestionarios debe registrar la posición, puede ser por geolocalización o haciendo búsqueda por direcciones en el mapa. Para registrar la posición debe mantener apretado por 3 segundos la posición en el mapa a registrar y aparece una alerta la cual debe aceptar para terminar el proceso. En la figura 21 y 22 se muestra lo descrito anteriormente.

Al culminar el registro del lugar aparece una pantalla que te indica que ya se culminó el proceso y te permite hacer otro registro nuevo. En la figura 23 se muestra esta pantalla. Un detalle que faltó, pero se va a agregar es que también te da la opción de salir de la aplicación por si no quieres hacer ningún otro registro.



Figura 14. Inicio

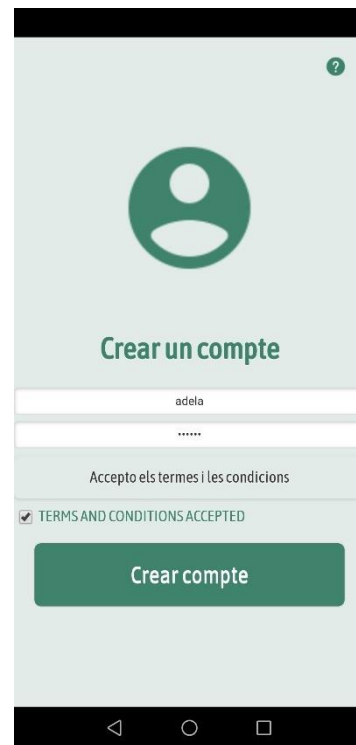


Figura 15. Registrarse



Figura 16. Autenticarse



Figura 17. Crear Perfil



Figura 18. Tipo de Lugar

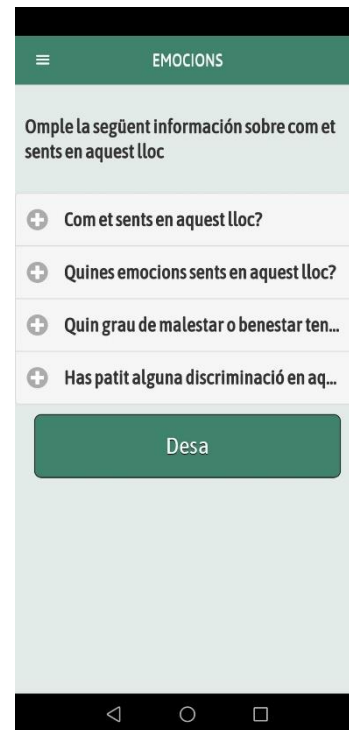


Figura 19. Cuestionario de Emociones





Figura 20. Transporte público

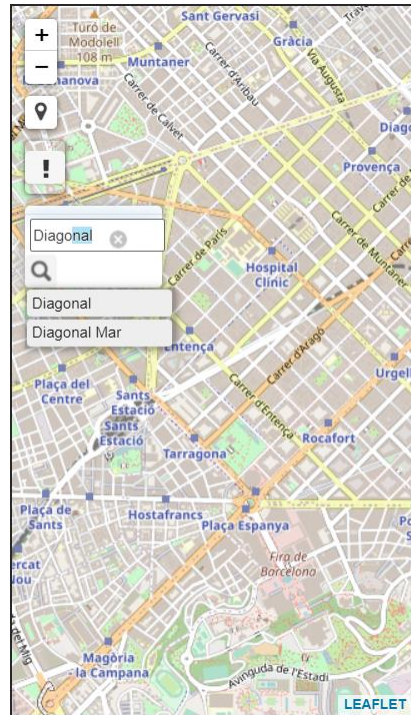


Figura 21. Mapa

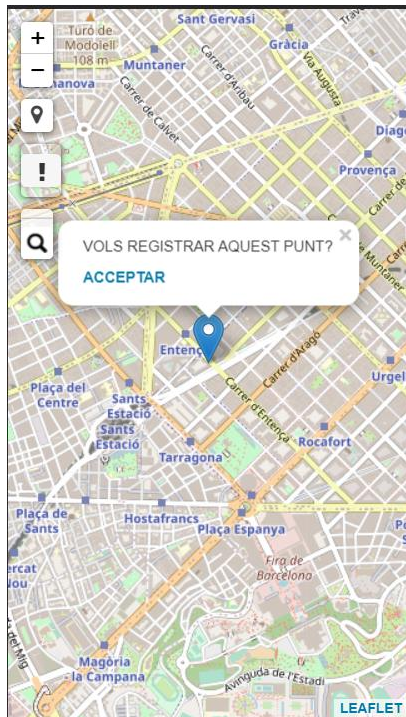


Figura 22. Registrar posició

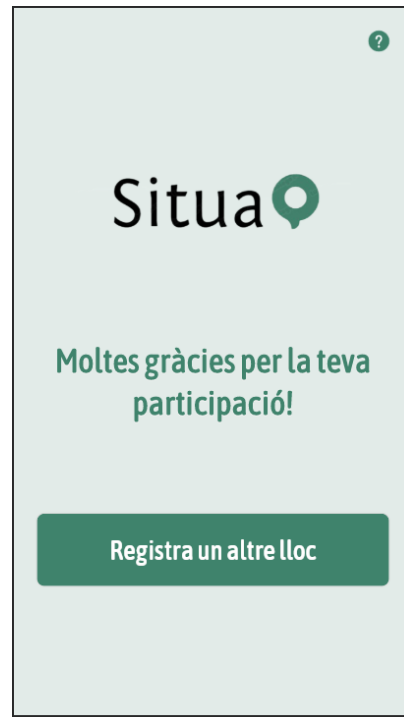


Figura 23. Terminar

## 6 IMPLEMENTACIÓN

### 6.1. Tecnologías utilizadas

En este apartado se explicarán las tecnologías que se utilizan para implementar el proyecto. En primer lugar, se citarán las tecnologías utilizadas en la parte de aplicación al cliente (desarrollo *front-end*) y posteriormente las utilizadas en la parte del servidor (desarrollo *back-end*). En ambas dos, se mencionarán también las librerías que se utilizan para poder obtener la implementación deseada.

En este proyecto se ha llevado a cabo tanto el desarrollo *front-end* como el *back-end*. Primero es necesario conocer estos términos. El *front-end* es la parte de una web que conecta e interactúa con los usuarios que la visitan. Es la parte visible, la que muestra el diseño, los contenidos y la que permite a los visitantes navegar por las diferentes páginas mientras lo deseen, mientras que el *back-end* es el polo completamente opuesto, es la capa que accede a datos y software en general para la comunicación con el usuario. Se encarga de que toda la lógica de una página web funcione. Su función principal es acceder a la información que se solicita, a través de la app, para luego combinarla y devolverla al usuario final [W3] [W4].

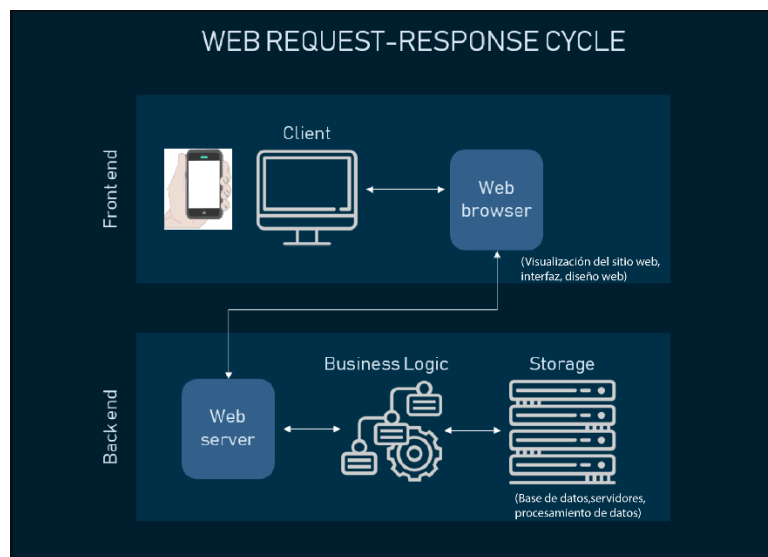


Figura 24. Estructura de un sitio web, con front-end y back-end. Fuente: <https://www.altexsoft.com>

### 6.2. Aplicación Cliente

#### 6.2.1. Tecnologías y lenguajes de programación

La aplicación cliente será una aplicación móvil que funcione en dispositivos Android en la primera versión porque se pretende que en versiones posteriores sea multiplataforma. Para implementar la aplicación se utilizaron las siguientes tecnologías y lenguajes de programación:

- ✓ HTML (*HyperText Markup Language*): Es la pieza más básica para la construcción de la web ya que describe la estructura y el contenido semántico de la página web [W5].
- ✓ CSS (*Cascading Style Sheets*): Es utilizado para diseñar y dar estilo a las páginas web. Define la parte de la presentación, es decir, cómo deben mostrarse los elementos de la página, su posición, forma, espaciados, colores y, en resumen, toda la parte estética [W6].
- ✓ JavaScript: Es un lenguaje de programación que te permite crear contenido nuevo y dinámico, controlar archivos de multimedia, crear imágenes animadas y muchas otras cosas más [W7].
- ✓ AJAX (*Asynchronous JavaScript and XML*): Es un conjunto de técnicas de desarrollo web que permiten que las aplicaciones web funcionen de forma asíncrona, procesando cualquier solicitud al servidor en segundo plano. Cualquier aplicación web que use AJAX puede enviar y recuperar datos del servidor sin la necesidad de volver a cargar toda la página [W8].



En el esquema que se muestra a continuación se explica con detalle lo que ocurre desde que se envía una solicitud AJAX desde el cliente, los procedimientos que se realizan en el servidor para luego mandar la respuesta en este caso en formato JSON. La comunicación entre cliente y servidor se hace a través del protocolo HTTP. Estas solicitudes usando AJAX no requieren recargar la página para actualizar la información, esto hace que la aplicación sea más rápida y con mayor capacidad de respuesta a los pedidos de los usuarios.

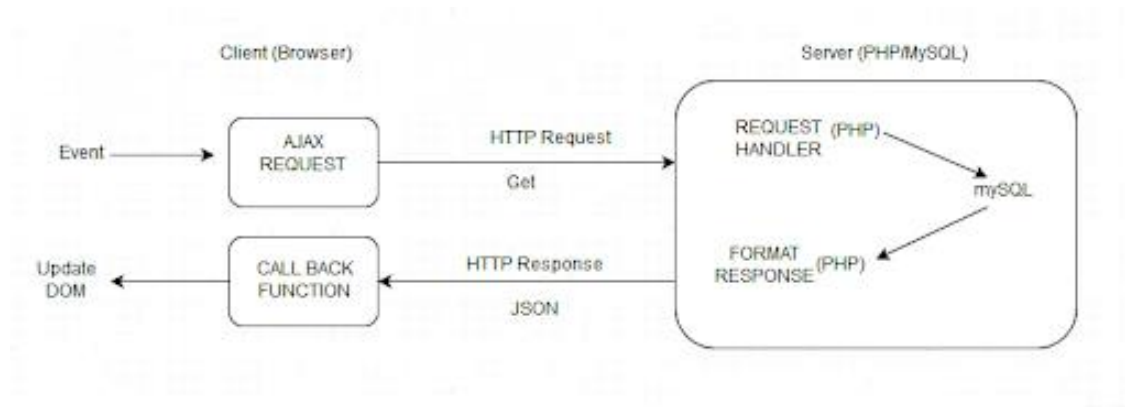


Figura 25. Esquema de pedidos AJAX entre cliente-servidor. Fuente: <http://www.cs.uregina.ca>

## 6.2.2. Librerías de JavaScript y Framework:

Lo más importante de JavaScript es que se trata de un lenguaje de programación idóneo del lado del navegador. Sin embargo, para muchos programadores resulta problemática la interfaz de la página web: el DOM (Modelo de objetos del documento). Aquí es precisamente donde entran en juego los frameworks JS y sus librerías, que facilitan el trabajo de desarrolladores en este y otros campos de la programación. Las extensiones de JavaScript que aparecen a continuación pueden descargarse de forma gratuita y son las que se usaron para el desarrollo de la aplicación cliente [W9].

- ✓ **JQuery:** Es la más utilizada de todas las librerías JavaScript. Permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web [W10].
- ✓ **JQuery Mobile:** Es un Framework optimizado para dispositivos táctiles (también conocido como Framework móvil) que está siendo desarrollado actualmente por el equipo de proyectos de jQuery. El desarrollo se centra en la creación de un Framework compatible con la gran variedad de teléfonos inteligentes y tabletas, algo necesario en el creciente y heterogéneo mercado de tabletas y teléfonos inteligentes. El Framework de jQuery Mobile es compatible con otros frameworks móviles y plataformas como PhoneGap y Worklight entre otros [W11].
- ✓ **Leaflet:** Es una librería en JavaScript, libre y de código abierto con muchos recursos destinada a facilitar el desarrollo de mapas en entornos web. Los plugins de leaflet permiten incorporar al mapa características tales como superponer un mapa de calor, animar marcadores, cargar archivos de datos csv, dibujar formas complejas, medir la distancia, manipular capas y mostrar coordenadas.
- ✓ **PhoneGapp:** Es un framework para el desarrollo de aplicaciones móviles utilizando herramientas genéricas tales como JavaScript, HTML5 y CSS3. Las aplicaciones resultantes son híbridas, es decir que no son realmente aplicaciones nativas al dispositivo. La principal ventaja es que se codifica la aplicación una sola vez y la adaptamos a múltiples plataformas, en vez de escribir una aplicación diferente para cada una. El principal inconveniente, no obstante, es que no es tan eficiente en términos de uso de los recursos y herramientas del dispositivo móvil. En resumen, PhoneGap es una buena opción cuando queremos desarrollar, en muy poco tiempo, una aplicación nativa de complejidad moderada para múltiples plataformas. Pero si queremos alcanzar mejores resultados y contamos con tiempo, tendremos que emprender el camino tradicional [W12].

## 6.2.3. Configuración del entorno de la aplicación

La aplicación se va a desarrollar en formato web-app, el cual consiste en una transformación de una aplicación web (creada mediante html, css y javascript) a una app nativa (independientemente del sistema operativo del dispositivo) en formato apk. Para ello se va a utilizar el programa Cordova a través de la interfaz PhoneGap. Mientras Cordova es el software que realiza la transcripción de código web a móvil, PhoneGap es el intermediario que permite crear nuevos proyectos y visualizar el desarrollo de la aplicación directamente en el dispositivo antes de hacer la transformación de código.

Para configurar el entorno es necesario un workspace con PhoneGap donde se almacenará el código, así como otras carpetas y archivos que luego serán necesario para la exportación a apk.

## **6.2.4. Programación de la aplicación**

La aplicación funciona como un recolector de datos. Un usuario perteneciente o no a un proyecto se registra en la aplicación llena su perfil y comienza el estudio agregando un espacio, las emociones y si sufrió algún tipo de discriminación. En si los detalles más importantes de la aplicación son que el usuario se registre o una vez que esté registrado habrá su sesión y rellene los distintos cuestionarios hasta que agregue la posición del lugar.

Por tanto, la aplicación presenta 5 pantallas principalmente: la pantalla de registro donde debe aceptar los términos y condiciones, la de iniciar sesión que es donde se agregar el código del proyecto al que pertenece si se va a hacer un estudio específico, el formulario de perfil que solo lo rellena cuando se registra en la aplicación, el cuestionario para agregar un lugar, el cual está compuesto por distintas pantallas de acuerdo a las respuestas y por último un mapa interactivo que permite registrar la posición del lugar.

### **Creación del mapa**

Para insertar el mapa se utiliza la librería de Leaflet, ya que esta se adapta muy bien a entornos móviles.

Como mapa base se han utilizado el OSM que por defecto aparece con botones para acercar y alejar el mapa.

También se ha añadido la posibilidad de activar la localización del dispositivo mediante un botón y mostrarla en el mapa. Para esta funcionalidad hace uso del gps del móvil y el usuario da permiso para acceder a la posición.

Si el usuario mantiene apretado en una posición te sale un mensaje para registrar la posición, este paso es necesario para completar el registro del lugar.

Se incluye un buscador por direcciones, para facilitar el registro de la posición.

## **6.3. Servidor**

### **6.3.1. Tecnologías y lenguajes de programación**

En el lado del servidor se ha utilizado básicamente PHP (Hypertext Preprocessor) ya que un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web.

Para el desarrollo del back-end se utilizaron algunos protocolos o formatos de datos los cuales se explican a continuación:

- ✓ API (Application Programming Interface): Es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. En el caso del sistema es el protocolo de comunicación entre la aplicación cliente con el servidor de base de datos, lo cual, permite insertar o verificar la información que el usuario provee [W14].

- ✓ JSON (JavaScript Object Notation): Es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo [W15].
- ✓ YAML (Ain't Markup Language): Es un formato para guardar objetos de datos con estructura de árbol. Se utiliza para archivos de configuración, traducciones y representar información [W16].

### **6.3.2. Framework**

**Symfony**: Es un framework diseñado para desarrollar aplicaciones web basado en el patrón Modelo Vista Controlador. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. Symfony está desarrollado completamente en PHP y es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server [W17].

### **6.3.3. Programación del servidor**

El desarrollo del back-end se dividió en dos funcionalidades principales. Primero se desarrolló una API usando PHP para comunicarnos con la aplicación cliente. Esta API recibe la solicitud del cliente, se conecta con la base de datos, procesa la información y responde al cliente mandando la respuesta en formato JSON.

También se implementó un panel de administración usando el framework de PHP Symfony 4. Se usó el bundle EasyAdmin el cual te genera de una forma sencilla un panel de administración que se encarga de todas las tareas repetitivas (listar información, paginar, ordenar, crear y editar entidades, etc). En un simple fichero YAML se configura este bundle y solo para funcionalidades más avanzadas o muy personalizadas se utilizar código PHP. En este caso para exportar los ficheros en formato csv se tuvo que personalizar el bundle usando código PHP al igual se creó una función para generar automáticamente el código de proyecto al crearse un nuevo proyecto.

## **6.4. Entorno de desarrollo**

Para el proceso de desarrollo en la aplicación cliente se han utilizado dos editores de código distintos. Mayormente se ha utilizado Visual Studio Code, es un editor de código fuente que permite trabajar con diversos lenguajes de programación, admite gestionar tus propios atajos de teclado y refactorizar el código. Es gratuito, de código abierto y nos proporciona una utilidad para descargar y gestionar extensiones con las que podemos personalizar y potenciar esta herramienta [W18].

También se usó Notepad++, un editor de código muy simple con soporte para varios lenguajes de programación.

Mientras que para el servidor se utilizó PhpStorm, es un editor de código que compatible con todas las funcionalidades del lenguaje PHP y Bases de Datos/SQL.

Con el objetivo tanto de instalar PHP como de construir un servidor de datos, se ha utilizado el paquete de software XAMPP. Este paquete consiste en un único programa que permite descargar e instalar, de manera cómoda y simultánea, el servidor web Apache, por un lado, el intérprete de scripts de PHP y un gestor de base de datos PhpMyAdmin, entre otros componentes que no se han utilizado en el proyecto. Así, este paquete ha resultado de vital importancia puesto que todo el proceso de desarrollo, implementación y pruebas del sistema se ha visto soportado por la existencia de un servidor como es Apache para permitir la comunicación entre la base de datos y los clientes.

Para la gestión de la base de datos además de usar el PhpMyAdmin que nos provee XAMPP también se utilizó MySQLWorkbench, desde estos dos gestores de puede acceder a todas las funcionalidades de la base de datos.

## **6.5. Implementación de base de datos**

Una de las tareas más comunes y difíciles en cualquier aplicación es el tratamiento de información con una base de datos. Symfony basa el guardado de los datos, la persistencia de los datos, en Doctrine, librería cuyo objetivo es proporcionar herramientas potentes para el tratamiento de la información la cual permite mapear objetos a una base de datos relacional, como MySQL, PostgreSQL o MicrosoftSQL. Esto hace que desde Symfony sea muy sencillo crear un modelo de datos con solo simples líneas de comando te crea una estructura de datos en minutos.

### **6.5.1. Creando modelo de datos**

En Symfony la mayoría de los datos, o casi la mayoría, se guardan usando objetos de PHP que llamamos entidades simplemente serán unas clases de PHP con sus getters y setters y unas cuantas anotaciones a modo de comentarios que hacen muchas cosas de forma automática.

Esta abstracción de los datos te permitirá listar, almacenar, editar y borrar datos de forma muy fácil y en muy pocas líneas de código. A Doctrine le diremos, “toma estos objetos y guárdalos en la base de datos”, “dame todos los objetos con estas características”, podremos obtener un objeto en concreto, modificarlo y decirle a Doctrine: “guarda este objeto modificado”. Todo esto en muy pocas líneas de código fuente, gracias a usar estas clases en PHP. Evidentemente, puedes seguir usando el SQL tradicional si con Doctrine no alcanza a todo lo que necesitamos

Para nuestro proyecto se crearon 8 entidades y las relaciones entre sí, las cuales se detallaron en el capítulo 5 Diseño, donde se describe cada una de las entidades y asociaciones de la base de datos.

Para crear las entidades se usa el comando, al ejecutarlo van insertando cada campo que incluye la entidad y al dar enter se crea automáticamente la entidad con los campos que insertaste y sus getters y setters.

```
php bin/console make:entity <nombre de entidad>
```

Para modificar una entidad que existe, sólo tenemos que volver a lanzar el comando de crear sobre la misma entidad. Symfony detectará que ya existe, y en vez de crearla de nuevo, la modificará.

### 6.5.2. Crear base de datos

Ahora para guardar en la base de datos estos cambios tendremos que modificar el fichero `.env` para decirle donde tenemos la BD. En mi caso tengo una BD MySQL instalada en local con la siguiente cadena de conexión:

```
DATABASE_URL=mysql://situa:password@127.0.0.1:3306/situa?serverVersion=5.7
```

Simplemente en el gestor se creó un usuario y una contraseña y se le otorgaron los privilegios, ya que desde Symfony es muy simple crear la base de datos usando el comando.

```
php bin/console doctrine:database:create // se crea la BD
```

Doctrine Migrations es un sistema de versionado de la base de datos. Con esto podremos ir guardando versiones de la base de datos, o, mejor dicho, scripts de SQL que hacen que una base de datos pase de una versión a otra. En caso de problemas podremos echar atrás también los cambios en la base de datos

Para guardar la siguiente modificación tenemos el primer comando siguiente:

```
php bin/console make:migration
```

El siguiente ejecuta los cambios en la base de datos, a través de este comando se crea o se actualiza el esquema de la base de datos:

```
php bin/console doctrine:migrations:migrate
```

Como se pudo evidenciar con solo 4 comandos se ha creado la base de datos desde symfony. Incluso cualquier modificación que se requiera en la base de datos se hace de manera muy rápida y evitando errores en la implementación.



## 7 Pruebas Funcionales y puesta en marcha

### 7.1. Pruebas

Para comprobar la eficiencia de la aplicación es importante realizar pruebas que nos aseguren que la aplicación es óptima y que funciona correctamente en cualquier caso de uso permitido por la aplicación.

Se realizaron una serie de pruebas las cuales se describen a continuación:

#### **Aplicación Cliente:**

**Visualización correcta del mapa centrado en Barcelona:** El usuario tiene que ser capaz de ver el mapa completo y centrado en Barcelona, en caso de registrar un medio de transporte se usará un mapa que muestra las líneas de bus, tren y metro. Las de metro están resaltadas con los colores de cada línea. **SUPERADA**

**Correcto funcionamiento del zoom en el mapa:** El usuario tiene que ser capaz de acercar o alejar el mapa usando los botones de zoom. **SUPERADA**

**Correcto funcionamiento del buscador de direcciones:** El usuario debe ser capaz de encontrar direcciones usando el buscador. *El usuario puede encontrar las direcciones, pero al incluir el geojson con las direcciones se demora mucho en cargar el mapa esto queda pendiente por optimizar*

**Correcto funcionamiento y visualización del menú en todas las pantallas:** El usuario debe ser capaz de abrir o cerrar el menú y todos los enlaces dentro deben funcionar correctamente. **SUPERADA, falta por generar la información para añadir**

**Comprobar la opción de añadir posición en el mapa:** El usuario dejando apretado sobre una posición en el mapa debe poder añadirla. **SUPERADA**

**Comprobar que funciona la geolocalización en el mapa:** El usuario haciendo uso del botón de geolocalización debe encontrar su posición actual. Para poder registrar esa posición debe dejar apretado sobre esa posición en el mapa. **SUPERADA**

**Correcto funcionamiento de los cuestionarios:** El usuario puede rellenar los formularios correctamente y se hagan las validaciones correctas según la información que se requiera rellenar. **SUPERADA**

**Comprobar que se un usuario se puede registrar correctamente:** El usuario se puede registrar con un usuario, contraseña y aceptando los términos y condiciones. **SUPERADA**

**Comprobar que un usuario pueda loguearse correctamente:** El usuario puede acceder a la aplicación con su usuario, contraseña y en caso de pertenecer a un proyecto se asocia. **SUPERADA**

### **Panel de Administración:**

**Comprobar la gestión de proyectos:** El administrador puede agregar, modificar y eliminar los proyectos. ***SUPERADA***

**Comprobar la información que puede visualizar solamente:** Existen otras entidades que el administrador solo puede acceder a la información, pero no la puede modificar, sería el caso del perfil de usuario, de la información de los usuarios y de los lugares que se registran. ***SUPERADA***

**Comprobar la exportación de datos a formato csv:** El administrador desde la entidad *Users* puede exportar la información de los usuarios en formato csv. ***Esta opción se necesita ajustar, porque ahora mismo exporta la información de todos los usuarios, cuando lo que realmente se quiere es exportar la información de los usuarios por proyectos, queda pendiente de desarrollo***

## **7.2. Compilación de la aplicación**

Una vez comprobado el correcto funcionamiento de la aplicación se pasa a compilar el código y a crear un archivo apk que puede ser instalado como una aplicación nativa en el dispositivo. Como se comentó anteriormente, aunque a través de phonegap se puede crear apk para distintas plataformas, este proyecto solo estará orientado para dispositivos Android.

Por último, se instala la aplicación en varios dispositivos Android y se vuelve a comprobar el funcionamiento de la aplicación para distintas versiones de Android.

## 8 Conclusiones

Durante el desarrollo de este proyecto se pretendía desarrollar una aplicación móvil que permita recoger datos sistematizados sobre la experiencia, a partir de diferentes posiciones y en función del lugar, para hacer estudios sobre desigualdades sociales. Este propósito se ha cumplido y como resultado final se obtiene tanto una aplicación, una API para comunicarnos con la aplicación y un panel de administración que permite tanto gestionar proyectos como interactuar con la información provista por el usuario.

Han sido el propio cliente final el que ha puesto en marcha la aplicación. En general han quedado satisfechos con el trabajo realizado y las funcionalidades del sistema, pero nos quedan pendientes algunos ajustes para lograr que la experiencia del usuario sea más intuitiva y algunas mejoras del funcionamiento de la aplicación ya que se han detectado algunos problemas al no ser una aplicación nativa con temas de actualización, ya que te solicita actualizar la aplicación cuando son versiones de Android muy modernas por lo que se demora en abrirse.

Se le pueden agregar nuevas funcionalidades al mismo en versiones posteriores, como pueden ser un perfil de usuario donde pueda ver mapas de calor y se muestra todos los registros que ha hecho como estudios estadísticos que se han hecho en zonas aledañas.

En mi opinión personal este proyecto me ha servido para consolidar y aplicar todos los conocimientos adquiridos durante el máster cursado. Además de que ha sido necesario asimilar nuevas tecnologías. Ha sido una experiencia muy enriquecedora poder tener un contacto cercano en ámbito del desarrollo de software.

## 9 Bibliografía

- [1] RODÓ-DE-ZÁRATE, Mireia Baylina. Universitat Autònoma de Barcelona (defensa: 22 d'abril, 2014). Tesi Doctoral: "Geografies de la Interseccionalitat: l'accés de la joventut a l'espai públic de Manresa"
- [2] A.L. Griffin, J. McQuoid (December 2012): At the Intersection of Maps and Emotion
- [3] Sarah Elwood & Agnieszka Leszczynski (2018) Feminist digital geographies, Gender, Place & Culture, 25:5, 629-644, DOI: 10.1080/0966369X.2018.1465396
- [4] Eduarda Ferreira & Regina Salvador (2015) Lesbian collaborative web mapping: disrupting heteronormativity in Portugal, Gender, Place & Culture, 22:7, 954-970, DOI:10.1080/0966369X.2014.917276
- [5] Nazgol Bagheri (2014) Mapping women in Tehran's public spaces: a geo-visualization perspective, Gender, Place & Culture, 21:10, 1285-1301, DOI:10.1080/0966369X.2013.817972
- [6] Mei-Po Kwan (2002) Introduction: Feminist geography and GIS,9:3, 261-262, DOI: 10.1080/0966369022000003860
- [7] Mei-Po Kwan (2002) Is GIS for Women? Reflections on the critical discourse in the 1990s, Gender, Place and Culture: A Journal of Feminist Geography, 9:3, 271-279, DOI:10.1080/0966369022000003888
- [8] Nadine Schuurman & Geraldine Pratt (2002) Care of the Subject: Feminism and Critiques of GIS, Gender, Place and Culture: A Journal of Feminist Geography, 9:3, 291-299, DOI: 10.1080/0966369022000003905

## Sitios Web

- [W1] <https://www.periodismociudadano.com/4-apps-imprescindibles-contr-la-violenciadegenero>
- [W2] <https://www.alertadiscriminacion.org>
- [W3] <https://neoattack.com/neowiki/front-end/>
- [W4] <https://rafarionilla.com/que-es/backend/>
- [W5] <https://developer.mozilla.org/es/docs/Web/HTML>
- [W6] <https://developer.mozilla.org/es/docs/Web/CSS>
- [W7] [https://developer.mozilla.org/es/docs/Learn/JavaScript/First\\_steps/Qué\\_es\\_JavaScript](https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qué_es_JavaScript)
- [W8] <https://www.hostinger.mx/tutoriales/que-es-ajax>

[W9] <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/frameworks-javascript-y-librerias-populares/>

[W10] <https://es.wikipedia.org/wiki/JQuery>

[W11] [https://es.wikipedia.org/wiki/JQuery\\_Mobile](https://es.wikipedia.org/wiki/JQuery_Mobile)

[W12] <https://es.wikipedia.org/wiki/PhoneGap>

[W13] <https://www.4rsoluciones.com/blog/aplicaciones-nativas-con-tecnologia-web-ventajas-y-desventajas-de-phonegap/>

[W14] <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>

[W15] <https://www.json.org/json-es.html>

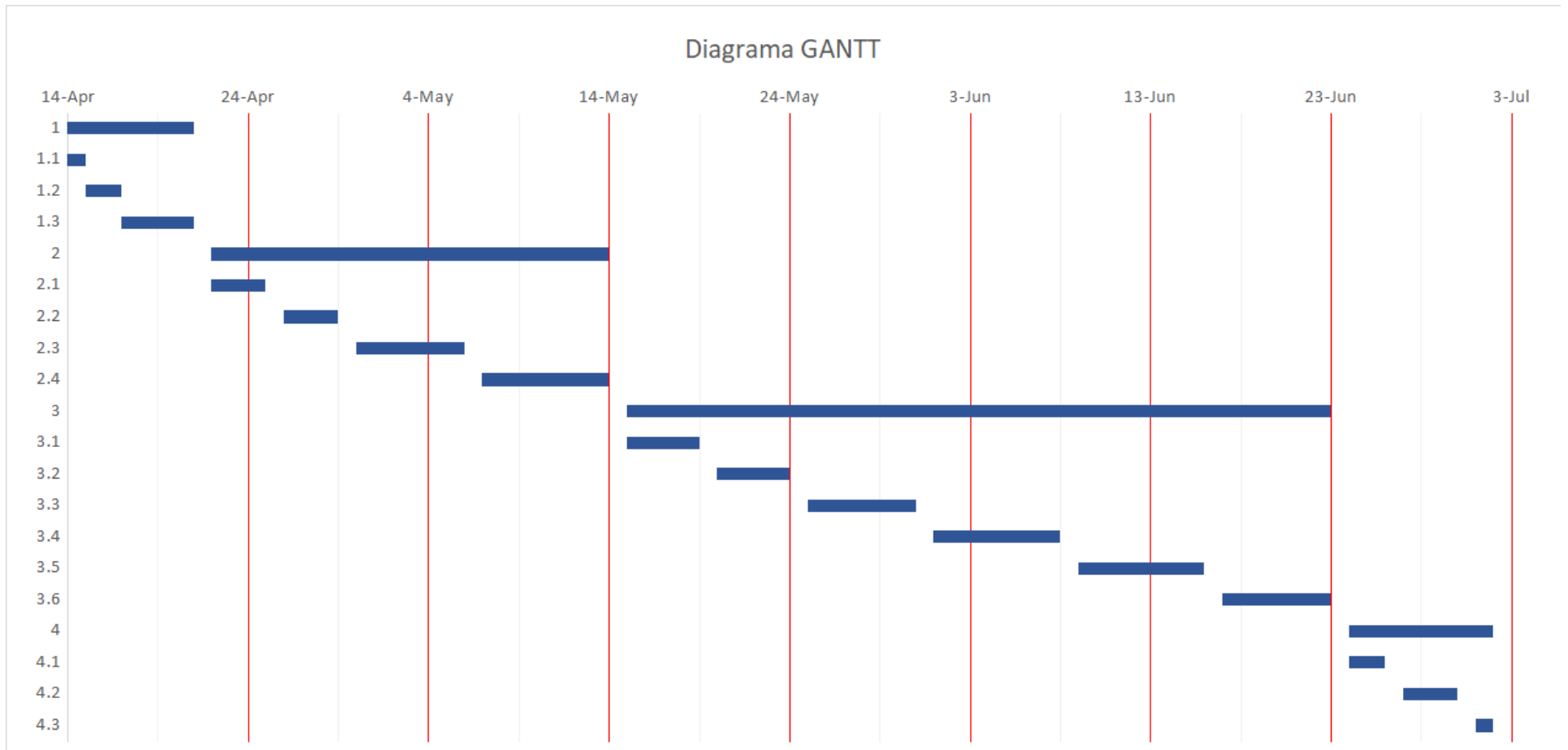
[W16] <https://fercontreras.com/conoce-que-es-un-yaml-e18e9d21ade4>

[W17] <https://es.wikipedia.org/wiki/Symfony>

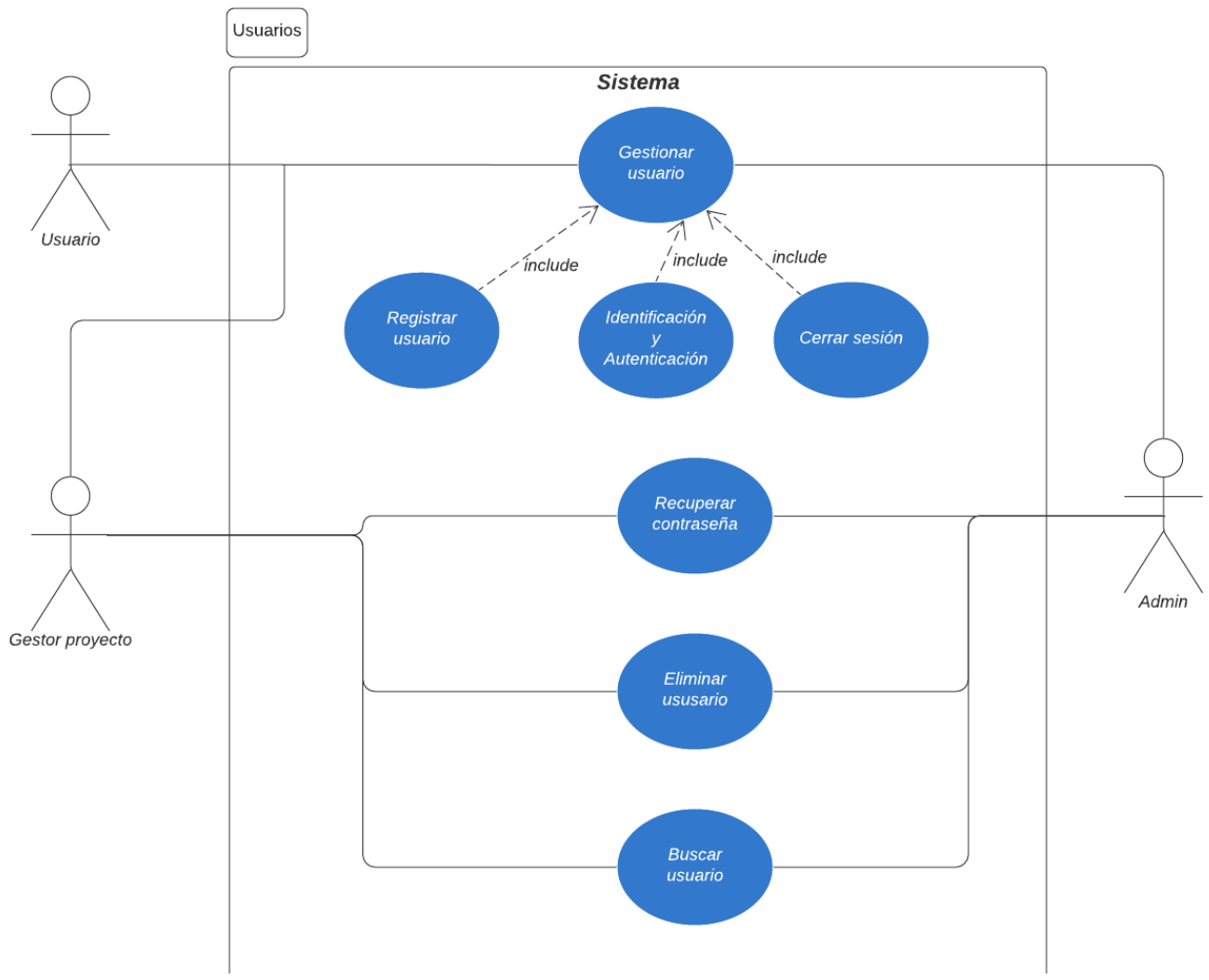
[W18] <https://blog.aitana.es/2018/10/16/visual-studio-code/>

# 10 Anexos

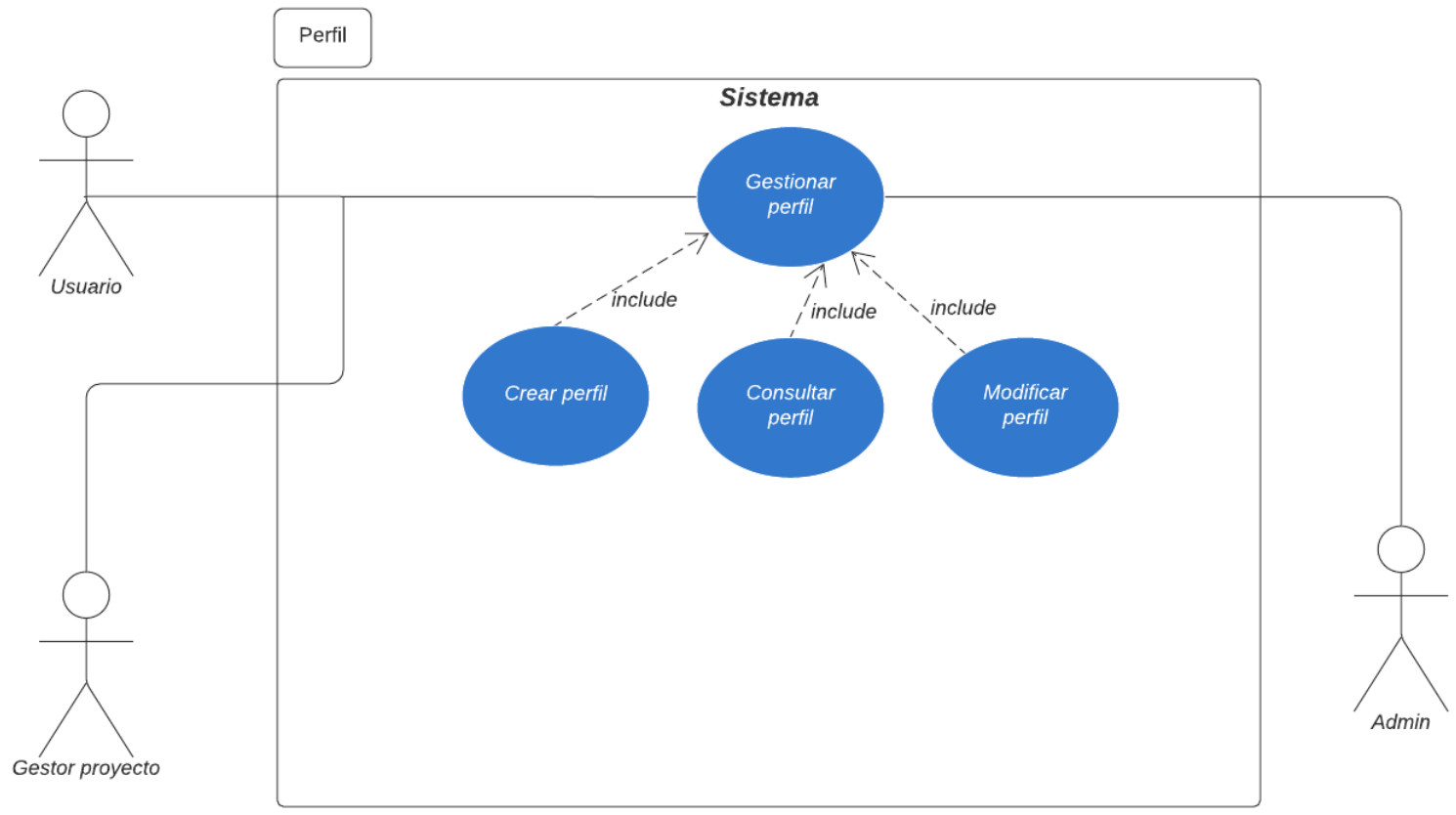
Anexo 1. Diagrama de GANTT



Anexo 2. Caso de uso: Gestión de Usuario

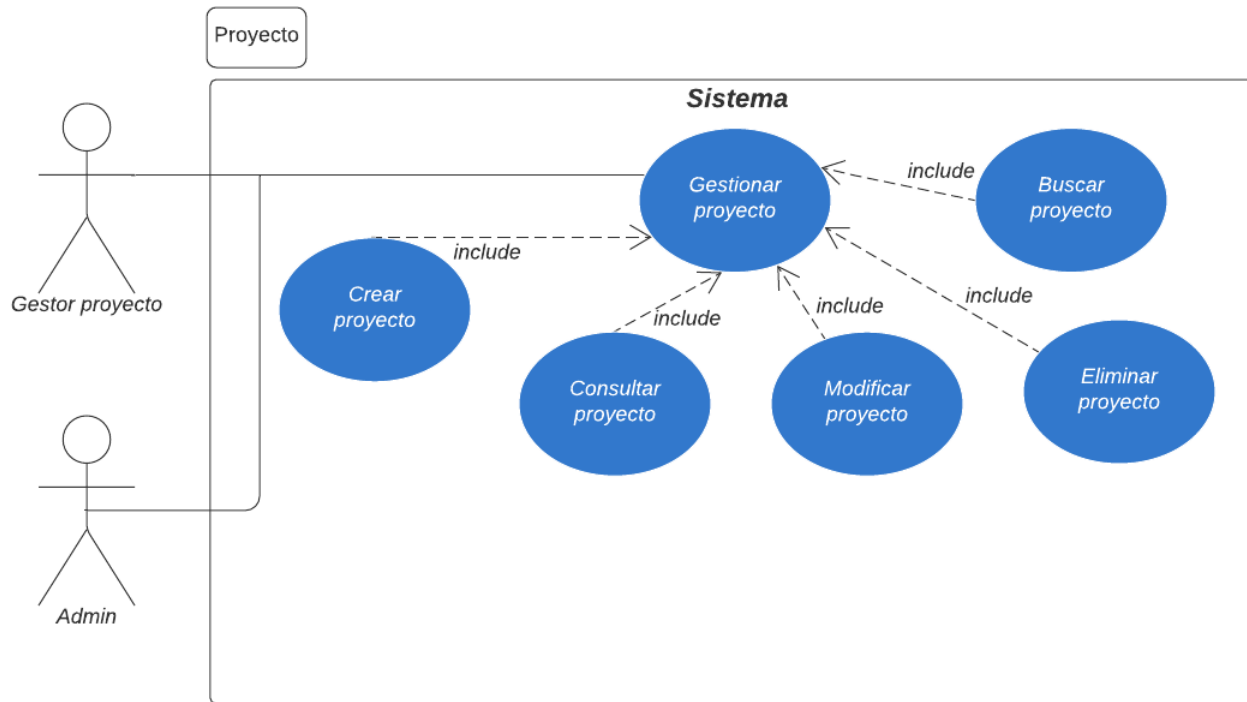


Anexo 3. Caso de uso: Gestión de Perfil

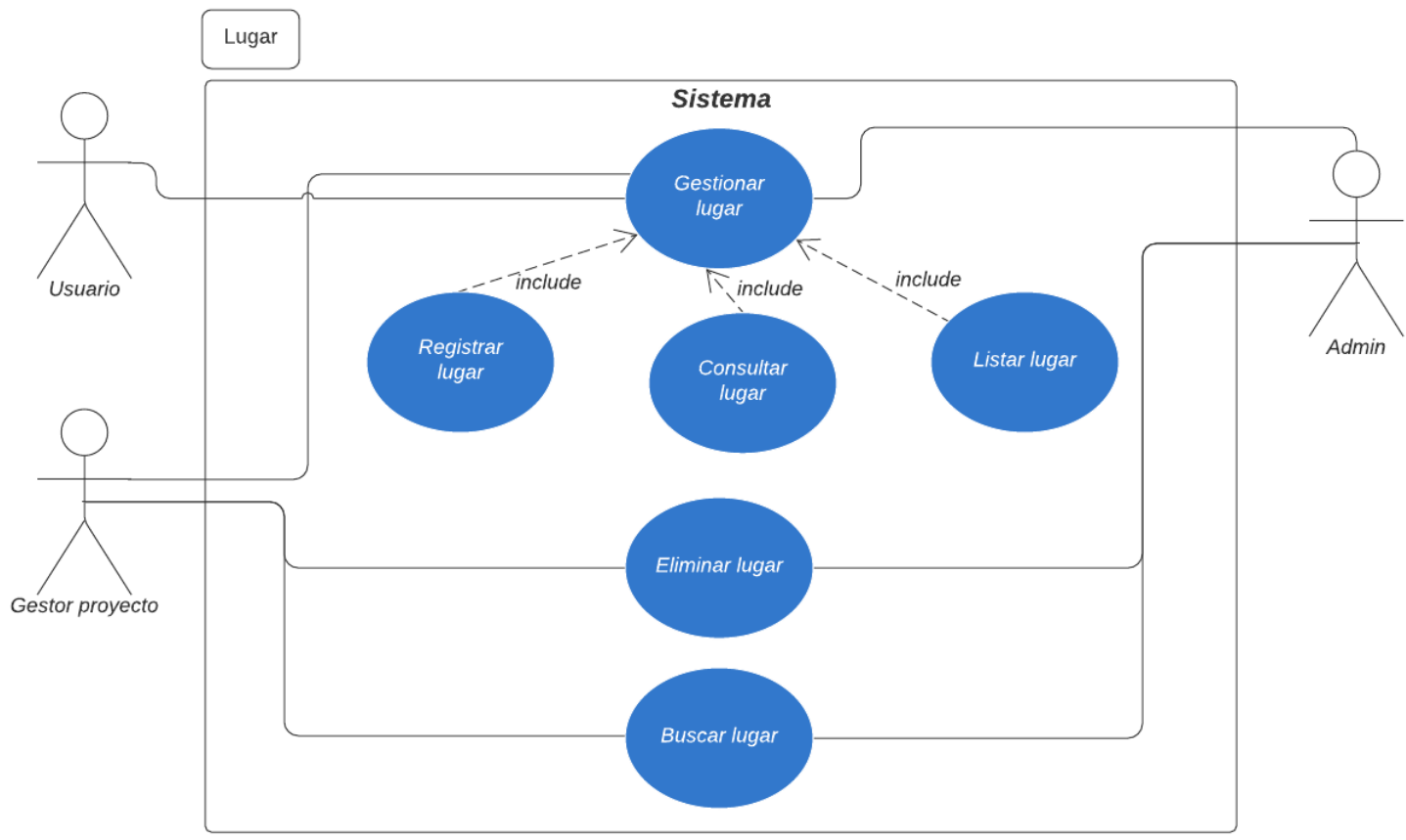




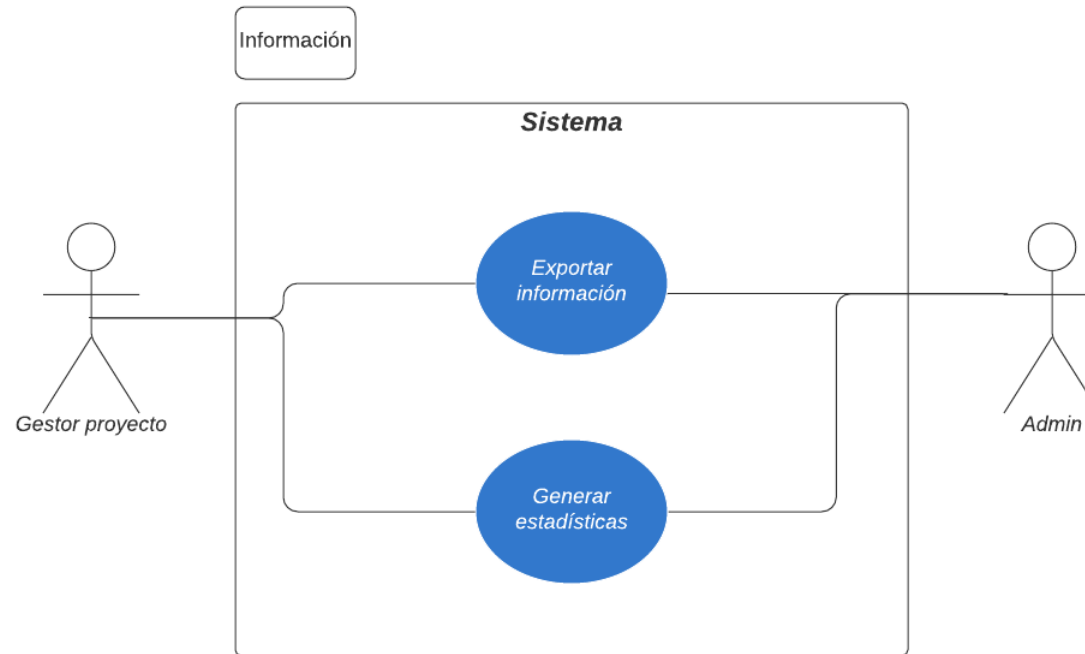
Anexo 4. Caso de uso: Gestión de Proyecto



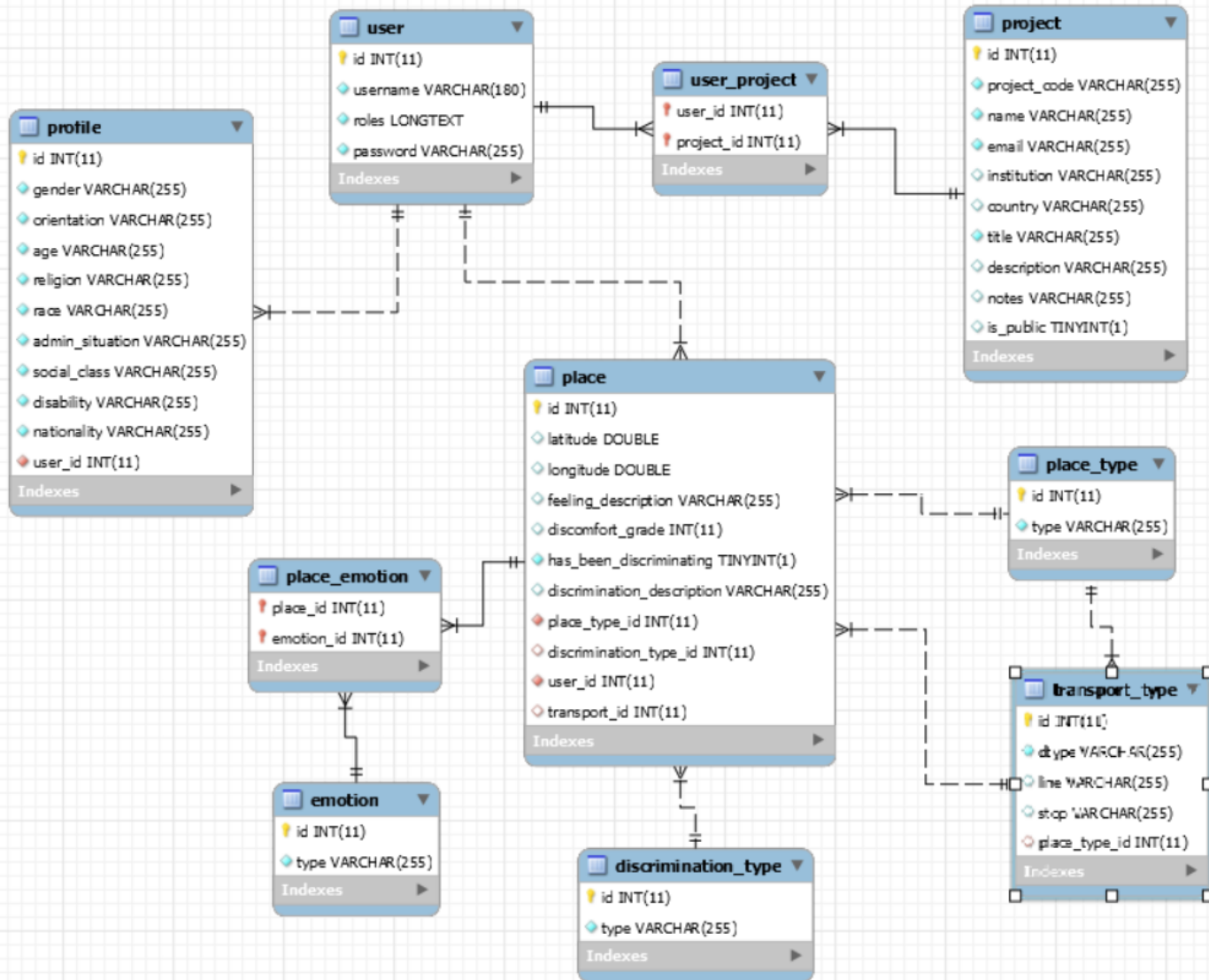
Anexo 5. Caso de uso: Gestión de Lugar



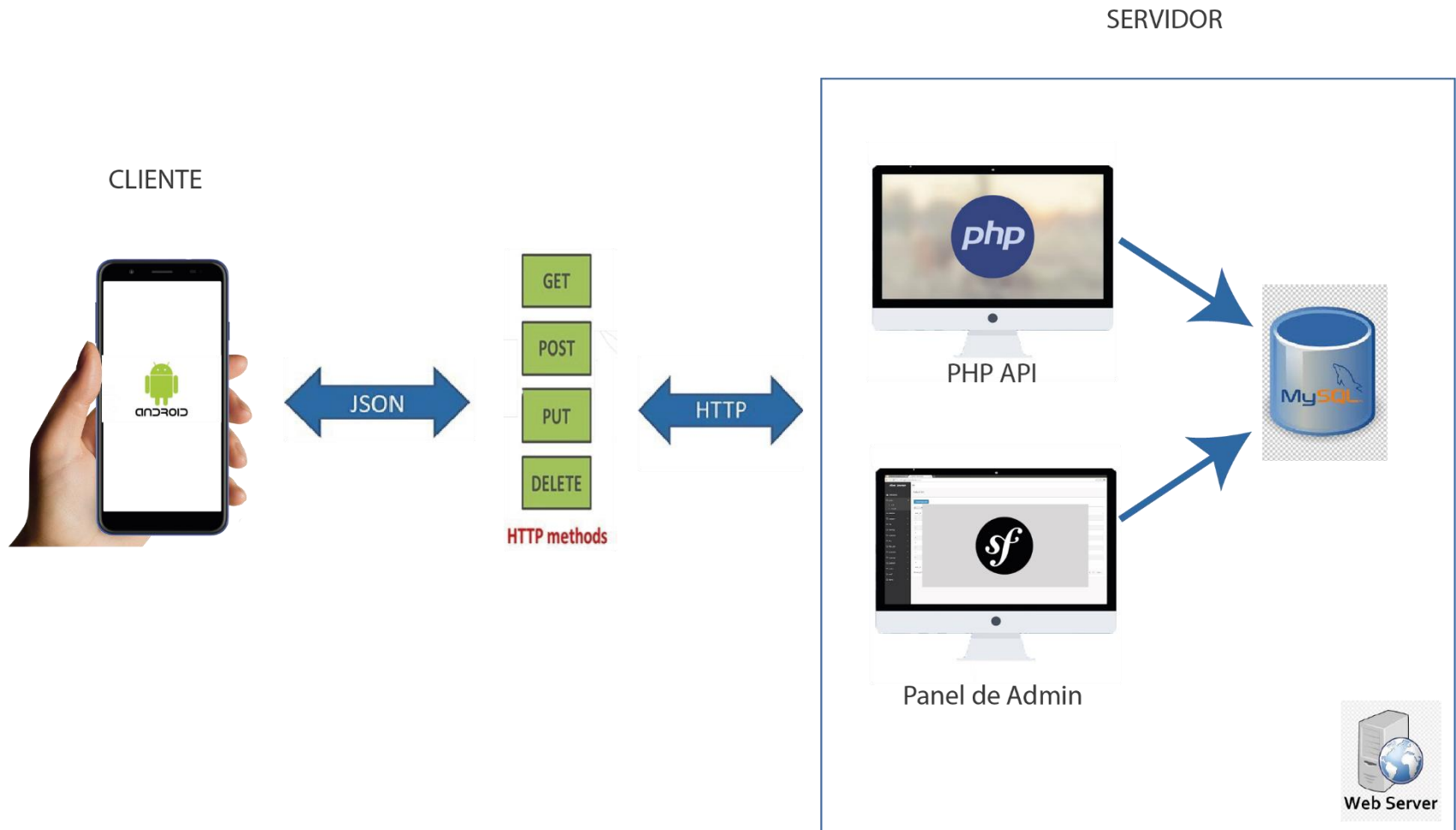
Anexo 6. Caso de uso: Exportar Información



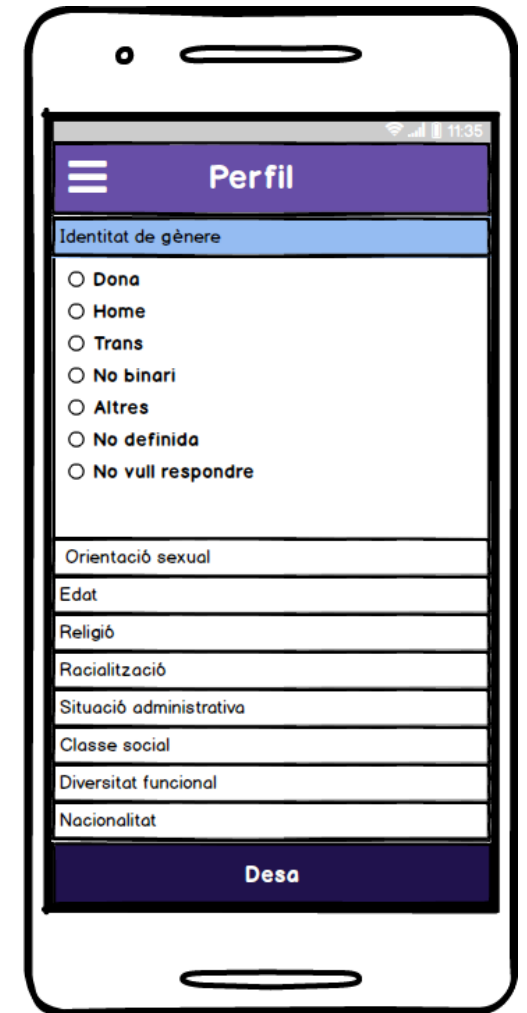
Anexo 7. Modelo Relacional BD



Anexo 8. Arquitectura Cliente-Servidor del Sistema



Anexo 9. Prototipo

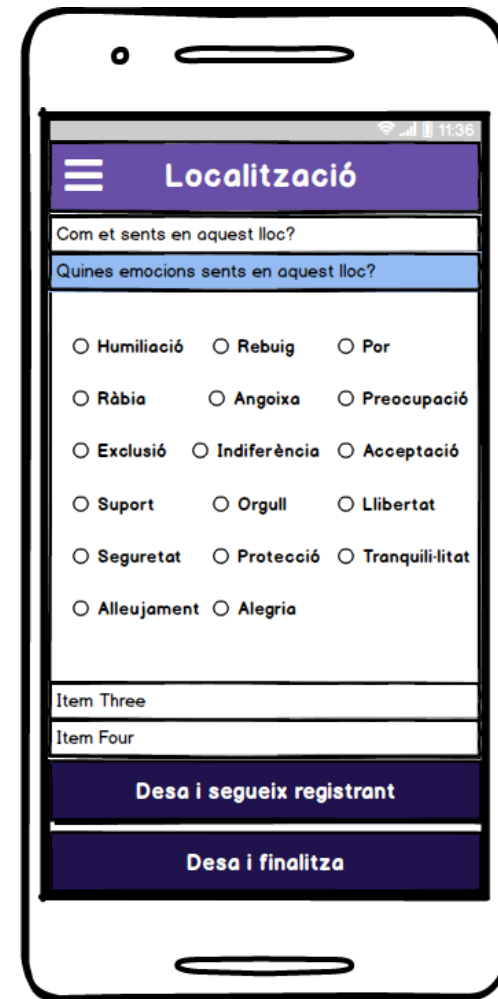


Anexo 9. Prototipo. Continuación

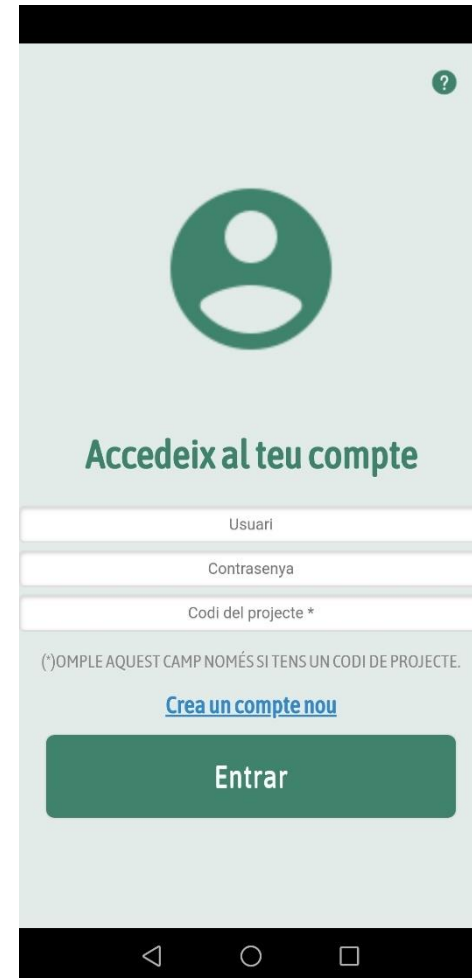
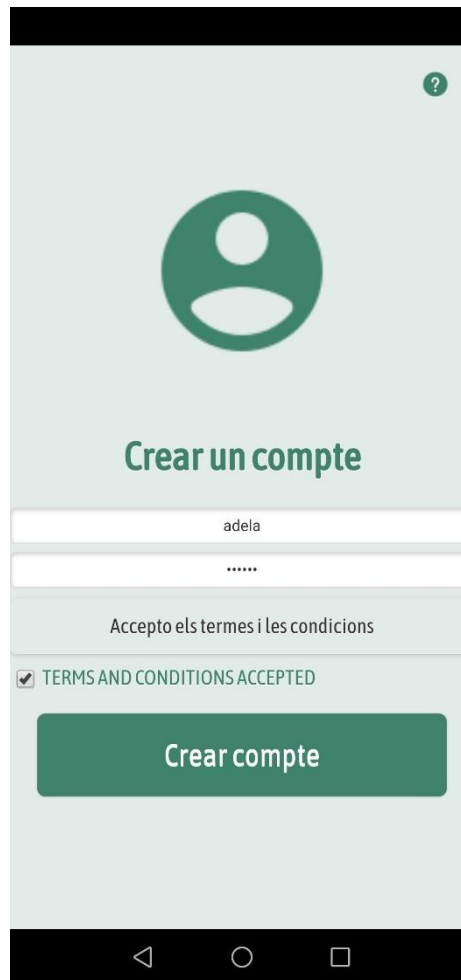


aquí se puede registrar la posición actual o registrar cualquier punto en el mapa

Te da la opción de navegar por el mapa. Cuando pulses 3 segundos a un sitio te aparece una ventada que dice "Quieres registrar este punto?" aceptar es que si te aparece el formulario, sino te sigue mostrando el mapa.

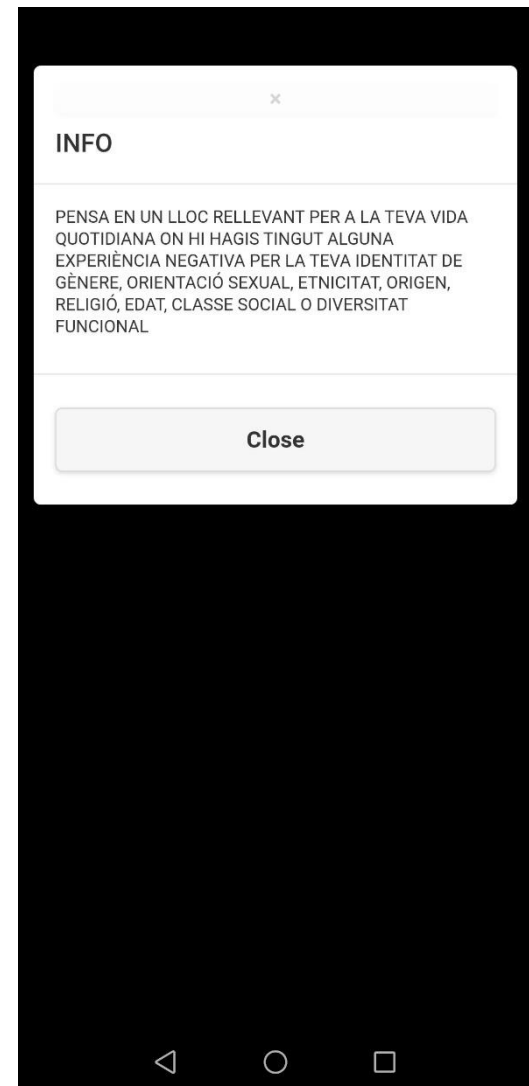


Anexo 10. Pantallas de la aplicación





Anexo 10. Pantallas de la aplicación. Continuación



Anexo 10. Pantallas de la aplicación. Continuación

The screenshot shows the 'TIPUS DE LLOC' screen. At the top, there is a green header with a hamburger menu icon and the text 'TIPUS DE LLOC'. Below the header, the question 'Quin tipus de lloc vols registrar?' is displayed. A list of location types follows, each with a radio button and a description: 'Espai públic (carrer, plaça)', 'Espai domèstic (casa)', 'Comerç o servei (botiga, supermercat, gimnàs, centre comercial, entitat bancària)', 'Espai laboral (feina)', 'Espai formatiu (escola, institut, centre formatiu, universitat)', 'Centre sanitari (CAP, hospital)', 'Lloc de lleure (bar, restaurant, discoteca, cinema, entitat)', 'Transport públic (metro, bus, tren, taxi)', and 'Oficina o servei de l'administració pública (oficina, presó, policia, jutjat)'. The 'Transport públic' option is selected. At the bottom, there is a green button labeled 'Següent'.

The screenshot shows the 'TIPUS DE TRANSPORTE' screen. At the top, there is a green header with a hamburger menu icon and the text 'TIPUS DE TRANSPORTE...'. Below the header, the question 'Quin transport vols registrar?' is displayed. A list of transport types follows, each with a radio button: 'Metro', 'Autobús', 'Tren', and 'Taxi'. The 'Metro' option is selected. Below the list, there are two input fields: 'Quina línia de metro?' with the value 'L5' and 'Quina parada?' with the value 'Diagonal'. Both fields have a close button (X) on the right. At the bottom, there is a green button labeled 'Següent'.

The screenshot shows the 'EMOCIONS' screen. At the top, there is a green header with a hamburger menu icon and the text 'EMOCIONS'. Below the header, the instruction 'Omple la següent informació sobre com et sents en aquest lloc' is displayed. A list of questions follows, each with a plus sign icon: 'Com et sents en aquest lloc?', 'Quines emocions sents en aquest lloc?', 'Quin grau de malestar o benestar ten...', and 'Has patit alguna discriminació en aq...'. At the bottom, there is a green button labeled 'Desa'.

Anexo 10. Pantallas de la aplicación. Continuación

EMOCIONS

Quin grau de malestar o benestar ten...

Has patit alguna discriminació en aq...

Sí

No

Quina és la causa de la discriminació?

Gènere

Orientació sexual

Origen

Etnicitat

Religió

Classe social

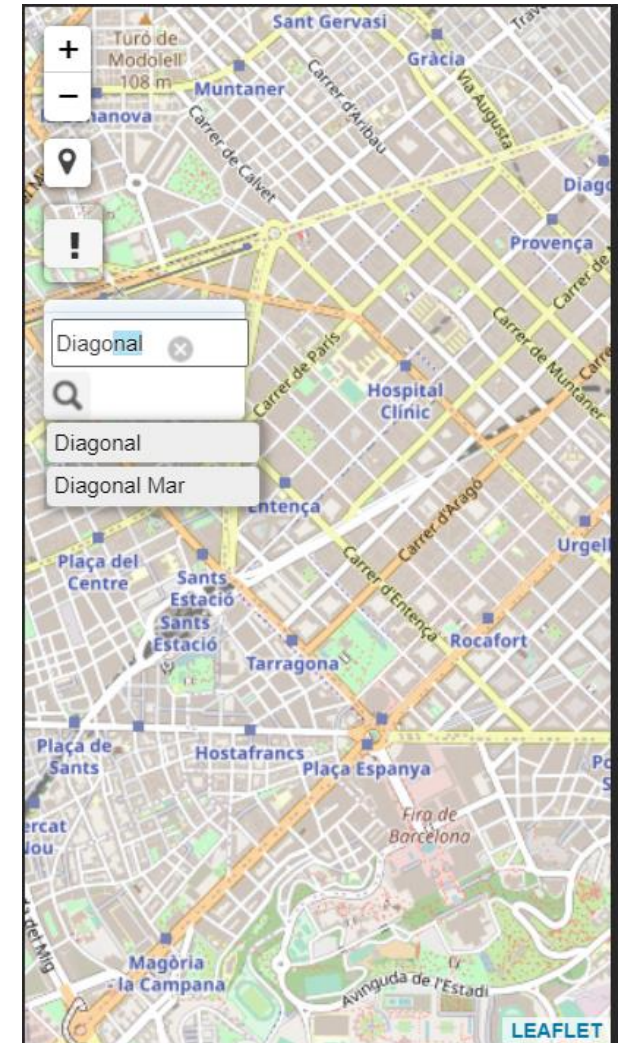
Edat

Discapacitat

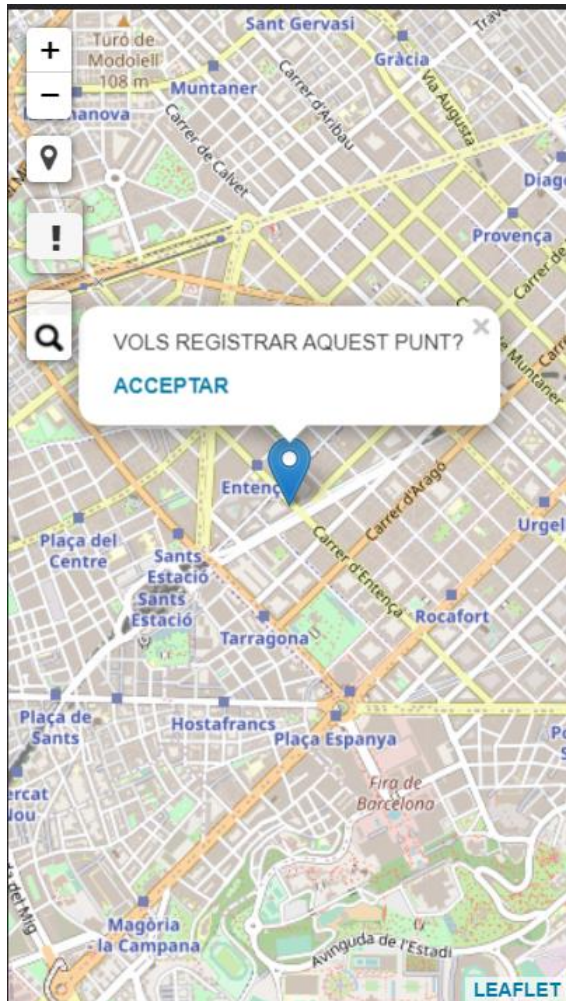
INFO

MARCA EN AQUEST MAPA EL LLOC CONCRET SOBRE EL QUE HAS OMLERT EL FORMULARI. SI ES EL LLOC ON ETS ARA, CLICA SOBRE [ICONA]. SI ES UN ALTRE LLOC, BÚSCA'L AL MAPA I MANTIGUES EL DIT SOBRE EL LLOC CONCRET PER A REGISTRAR-LO

Close



Anexo 10. Pantallas de la aplicación. Continuación



## Anexo 11. Panel de Administración

SITUA Admin admin

ENTITIES

- Projects >
- Emotions >
- Discrimination Types >
- Place Types >

STUDIES

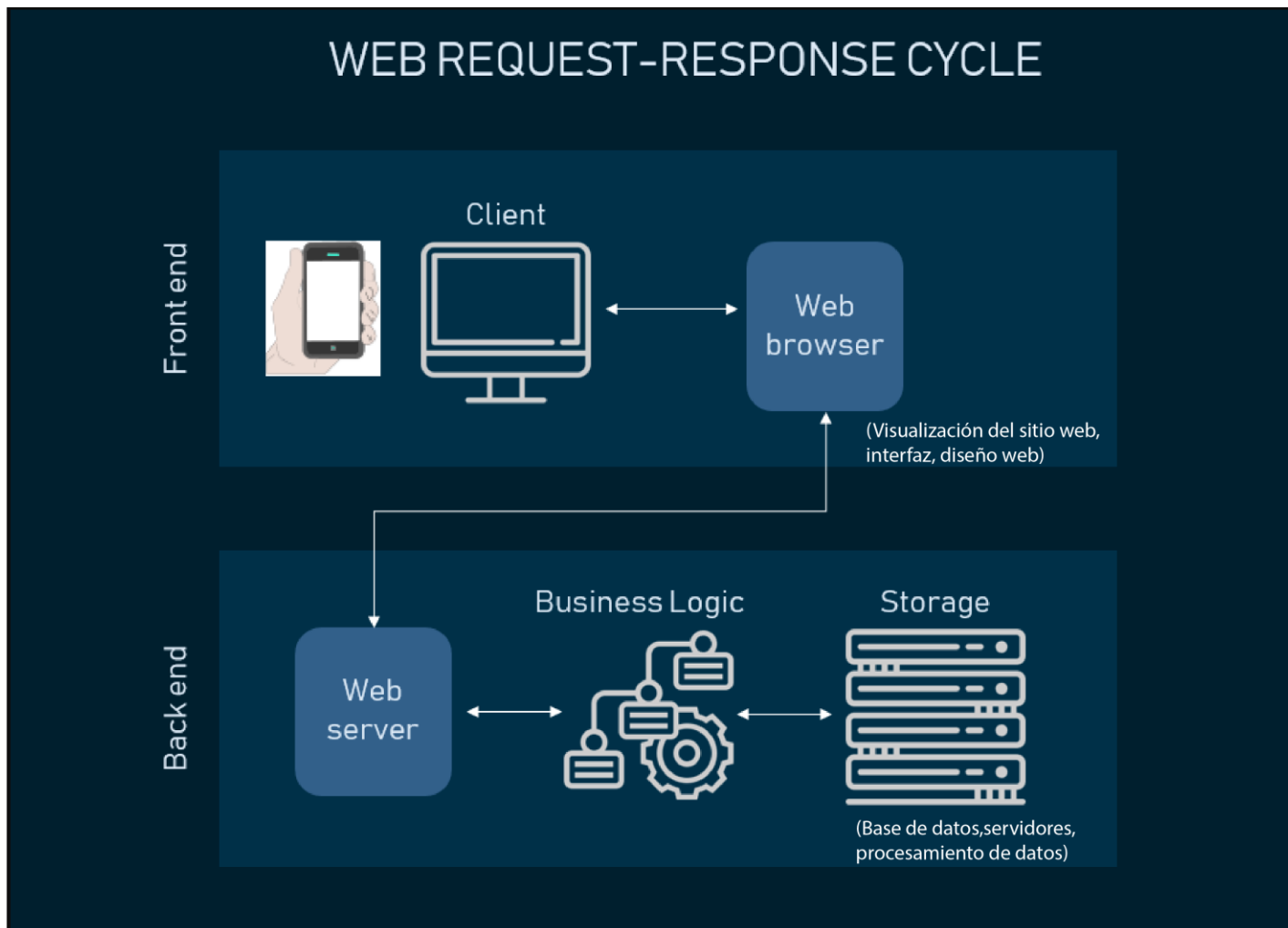
- Users**
- Places
- Profiles

### Users

 [Export](#)

ID ↓	Username ↕	Profile ↕	User project	Places	
21	eric	<a href="#">Profile #30</a>	2	9	<a href="#">Show</a> <a href="#">Export</a>
20	indira	<a href="#">Profile #26</a>	0	0	<a href="#">Show</a> <a href="#">Export</a>
14	ale	<a href="#">Profile #31</a>	1	1	<a href="#">Show</a> <a href="#">Export</a>
6	usd	<a href="#">Profile #19</a>	0	0	<a href="#">Show</a> <a href="#">Export</a>
5	adrian	<a href="#">Profile #22</a>	0	0	<a href="#">Show</a> <a href="#">Export</a>
4	adriana	<a href="#">Profile #20</a>	0	0	<a href="#">Show</a> <a href="#">Export</a>
1	admin	Null	0	0	<a href="#">Show</a> <a href="#">Export</a>

7 results < Previous Next >



Anexo 13. Esquema de pedidos AJAX entre cliente-servidor

