**Nearby**

**Desarrollo De Aplicaciones Para Dispositivos Móviles**

**Preparado por:**

**José Miguel Mora Rodríguez**

**Karina Zeledón pinell**

**Dylan Rodríguez barboza**

**Abril, 2018**

# Propósito

Nearby es una aplicación desarrollada para dispositivos android. Consiste en un medio para la creación de eventos, en donde se pueda especificar el lugar en el cual se llevará a cabo el evento. Consta de quince ventanas, mediante las cuales se pueden hacer acciones como: ver eventos, ver grupos que poseen eventos, ver el perfil del usuario, registrarse en la aplicación, visualizar los grupos y eventos de un usuario. En la aplicación, se hace uso de: el GPS y acceso a la memoria interna del dispositivo. Se hace conexión con el API de Facebook, Google Places y Google Maps. La aplicación tiene conexión con un API propio, desarrollado en django rest framework, en el cual se encuentra la lógica y el consumo de los datos de la aplicación.

La aplicación, inicialmente tuvo un alcance alto, indebidamente validado, que consistía de una gran cantidad de funcionalidades, por lo que se decidió mostrar la primera versión, sin la implementación de todas las funcionalidades planteadas en el prototipo, pero con el propósito de ampliar las funcionalidades en versiones posteriores, incluso para otras plataformas diferentes a android.

## Requerimientos

Se requería que la aplicación cumpliera con los siguientes requisitos:

* Desarrollo en Android nativo (Java o Kotlin).
* Desarrollo de backend propio (no Baas), que sea accesible por dominio público.
* El backend debe tener un backoffice web de administración, con su propia autenticación.
* Debe usar user login.
* Deben manejar autenticación y autorización a nivel de API.
* Conexión con al menos dos APIs externos (sin contar el backend propio).
* Usar una de las siguientes características del teléfono: GPS, acelerómetro, cámara, compás.
* El app debe tener al menos 15 activities.
* Deben manejar métricas de uso con Mixpanel o Fabric.

# Funcionalidades

.Para esta versión se la aplicación, se tomaron en cuenta las siguientes funcionalidades:

* Registrarse en la aplicación.
* Listar eventos, dada una categoría.
* Crear un evento, en el cual se incluya una imagen que lo ilustre y la ubicación del evento.
* Visualizar un evento, en el cual se puedan ver sus comentarios respectivos e información general.
* Visualizar un grupo, en el cual se pueda ver su información general.
* Listar grupos.
* Poder votar como usuario, si se va a asistir o estar interesado en un evento.
* Filtrar los eventos por el voto que hizo un usuario en el evento, es decir, que se puede separar la visualización de los eventos a los que asistirá de los eventos en los que está interesado.
* Visualizar notificaciones, estas serán los nuevos eventos que se agreguen a un grupo al que pertenezca un usuario.
* Visualizar el perfil.
* Cambiar foto y nombre del perfil.
* Visualizar los eventos de un usuario en su perfil.
* Visualizar los grupos de un usuario en su perfil.

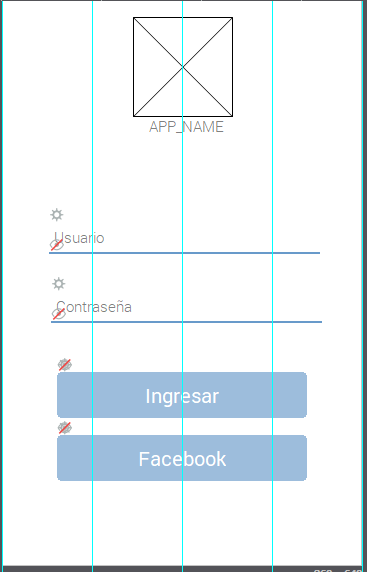
# UX, UI (wireframes)

A continuación, se muestran los wireframes que se propusieron originalmente para la aplicación:



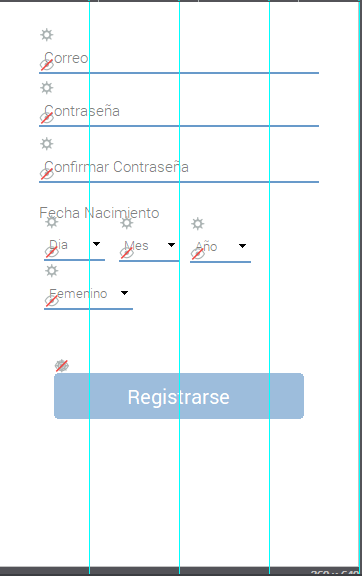
**Principal**

En esta primer activity, se muestra la pantalla de inicio de la aplicación, la cual está delimitada por el nombre de la aplicación (parte superior de la pantalla) y dos botones: uno encargado de ingresar, que dirigirá a otro activity para digitar los datos del usuario, y otro encargado de registrar, en el cual saldrá un pop up para inscribirse con facebook o manualmente.



**Inicio de sesión**

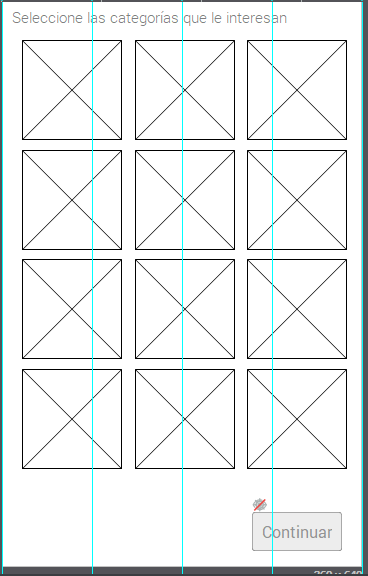
En esta segunda activity, se permite ingresar los datos, para ingresar a la aplicación, puede ingresar con credenciales de la aplicación o bien, con la cuenta de facebook registrada.



**Registro**

En este activity, el usuario será capaz de registrarse en el sistema, deberá ingresar datos como el correo electrónico, contraseña, fecha de nacimiento y género.

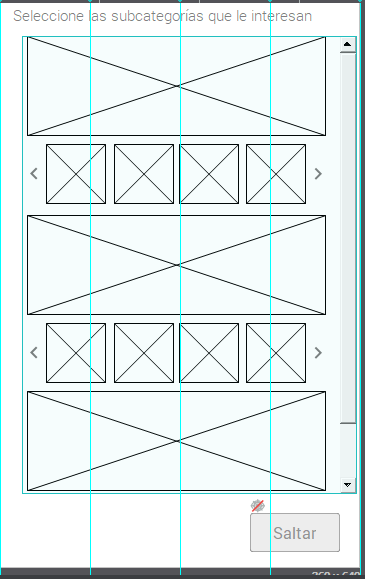




**Categorías de interés**

En este activity, el usuario podrá seleccionar los temas de interés, de manera que la aplicación los pueda tomar en cuenta para presentarle eventos en su perfil. Se permite que el usuario evite esta sección de la aplicación, mediante el botón de saltar.





**Subcategorías de interés**

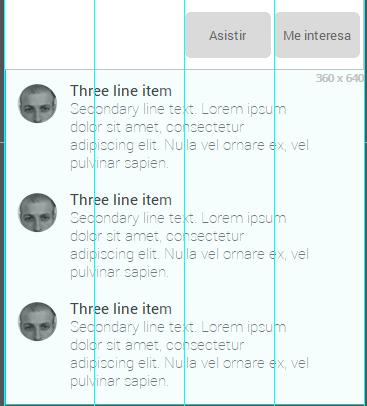
En este activity, el usuario podrá seleccionar las subcategorías de los temas de interés, mostrados anteriormente, de manera que la aplicación los pueda tomar en cuenta para presentarle eventos en su perfil. Se permite que el usuario evite esta sección de la aplicación, mediante el botón de saltar. Se puede deslizar hacia la derecha o izquierda en los cuadros de subcategorías (más pequeños), para mostrar más contenido.





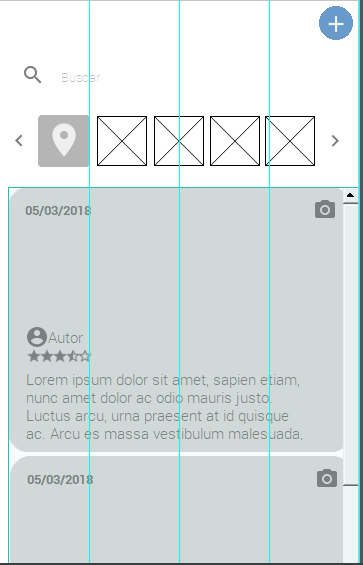
**Menú principal**

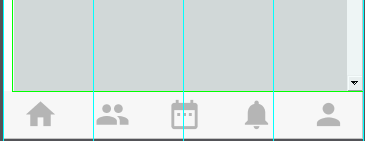
En este activity se muestra un menú de navegación en la parte inferior, en el resto del activy se muestra un buscador de eventos, un filtro de eventos (por día, hora y distancia), una sección para ver eventos cerca del usuario (debajo del filtro) y categorías que ha gustado a la derecha de este. Debajo de estos, aparecen eventos, en el cuál se presentarán sus respectivos datos, el símbolo de la cámara representará la subcategoría a la que pertenece. Finalmente, se pueden agregar nuevos eventos (ya sea individual o grupalmente) mediante el botón con el símbolo “+”.



**Detalle de evento**

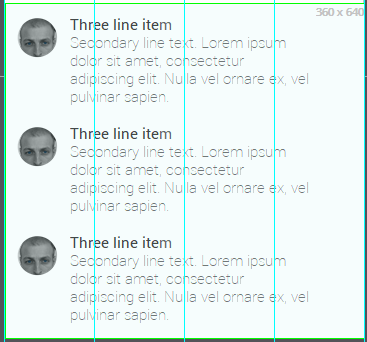
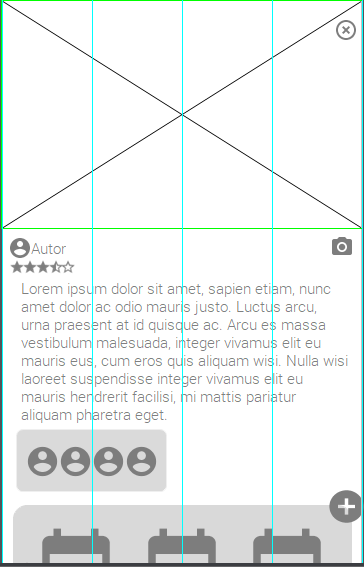
En este activity se muestra el detalle de un evento, con datos como el autor, fecha, lugar (uso de google maps), descripción, una imagen, calificación, comentarios. Y una sección que muestra los participantes (cuadro con perfiles) el cual mostrará un pop up con los participantes del evento. Además, se presenta la oportunidad de asistir o mostrarse interesado por el evento. El símbolo de “x” dentro de la imagen, permite volver al menú principal.





**Grupos**

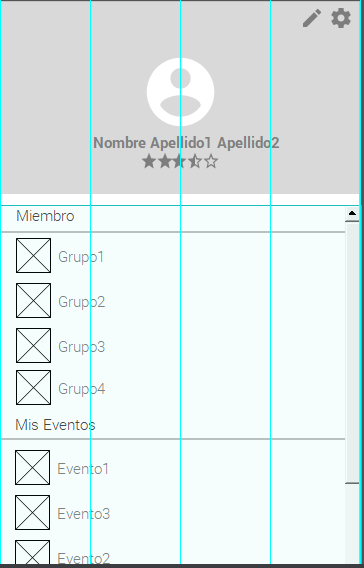
En este activity se muestra los grupos, al inicio aparecen los grupos a los que se pertenece y el buscador funciona como el activity de menú principal, solo que este buscará grupos. Este activity se activa cuando se da click al segundo botón del menú de desplazamiento en la parte inferior. El símbolo de “+” permite agregar un nuevo grupo.

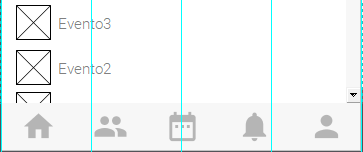




**Detalle del grupo**

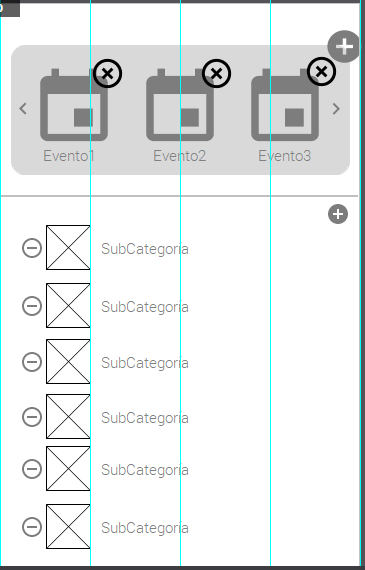
En este activity se muestra la imagen del grupo, el autor del grupo, calificación del autor, descripción del grupo, lista de miembros, subcategoría a la que pertenece y eventos que posee el grupo, con la posibilidad de añadir más, mediante el símbolo de “+”, el autor debe aprobarlos para que sean publicados Además, se muestran comentarios para dicho grupo. El símbolo de “x” en la imagen de la parte superior, hace que se devuelva al menú principal.

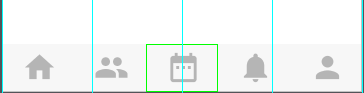




**Perfil**

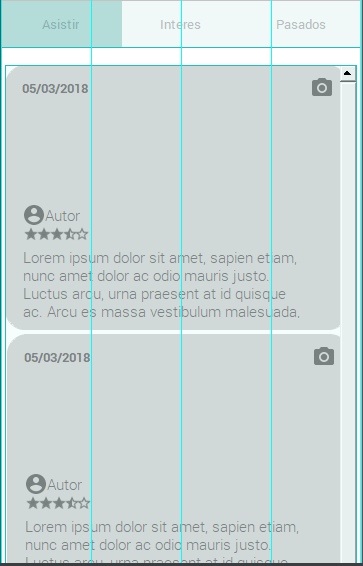
En este activity se muestran los datos del usuario, como lo es su foto, nombre, apellidos, calificación, grupo de los que es miembro, eventos que ha creado, además, la opción de editar la opción de editar el perfil (lápiz) y configuración para configurar las categorías (a la par del lápiz). Este activity se muestra al seleccionar el quinto elemento del menú de navegación de la parte inferior.

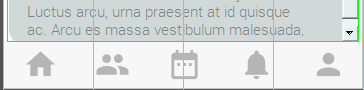




**Configuración de perfil**

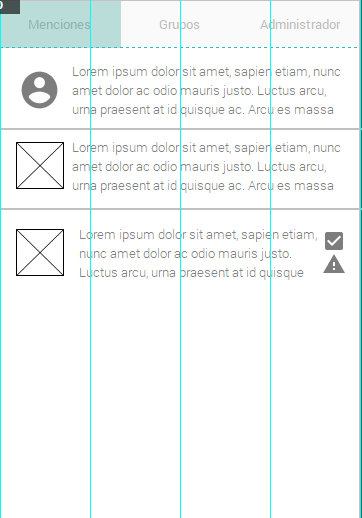
En este activity se muestra la lista de categorías en la parte superior (se pueden eliminar o agregar). Al seleccionar una categoría, se presentará una lista de subcategorías de la categoría, estas se pueden agregar o eliminar. En el prototipo se muestran como eventos, fue un error, en realidad son categorías.





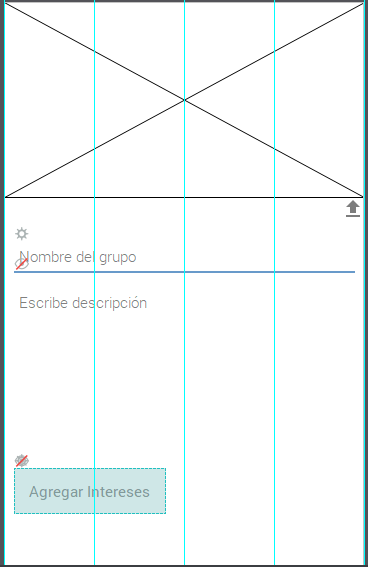
**Calendario**

En este activity se muestra la lista de eventos a los que el usuario asistirá, a los que el usuario está interesado y los que han pasado. Este activity se muestra al seleccionar el tercer elemento del menú de navegación de la parte inferior.



**Notificaciones**

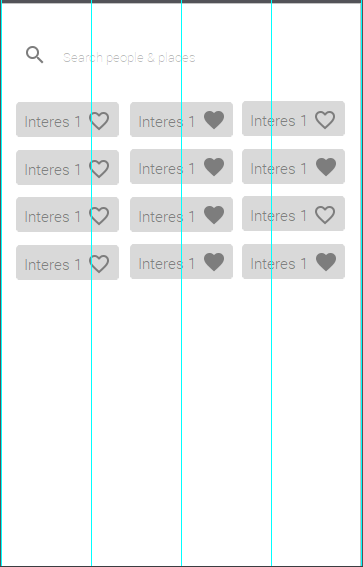
En este activity se muestran notificaciones. La primera notificación es relacionada con las menciones, la segunda notificación está relacionada con grupos y la tercera notificación está relacionado con administradores (nuevos eventos, donde se aprueban o no).





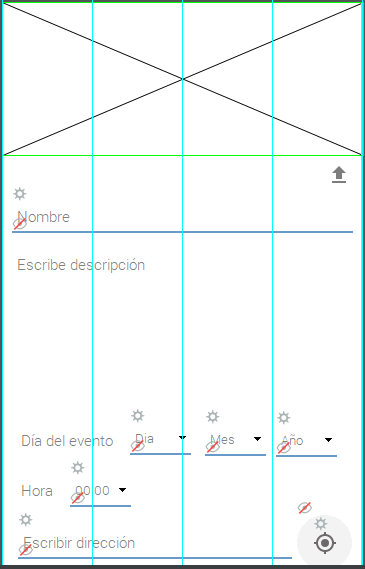
**Agregar nuevo grupo**

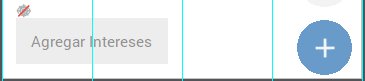
En este activity se permite agregar un nuevo grupo, al dar datos como la imagen, nombre de grupo, descripción, y agregar intereses. Se agrega mediante el símbolo de “+”.



**Agregar intereses a los eventos o grupos**

En este activity se muestra una lista de intereses que se pueden agregar a un evento, puede usarse un buscador para facilitar la búsqueda.



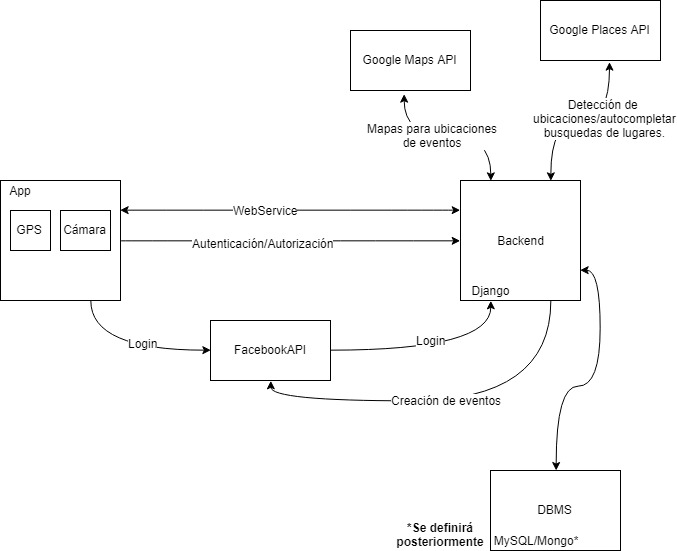
****

**Agregar nuevo evento**

En este activity se permite agregar un nuevo evento, al dar datos como la imagen, nombre de evento, descripción, y agregar intereses, fecha, hora, dirección (se utiliza google maps). Se agrega mediante el símbolo de “+”.

# Descripción de diseño de alto nivel

## Diagrama de arquitectura



Como se aprecia en la imagen, la aplicación consiste de un backend, el cual posee sus propias rutinas para el manejo de los datos, en la implementación final, el backend se encarga de guardar los valores de la latitud y la longitud de los lugares en donde se llevará a cabo el evento, mientras que la aplicación hace conexión con los API de Google Places y Google Maps.

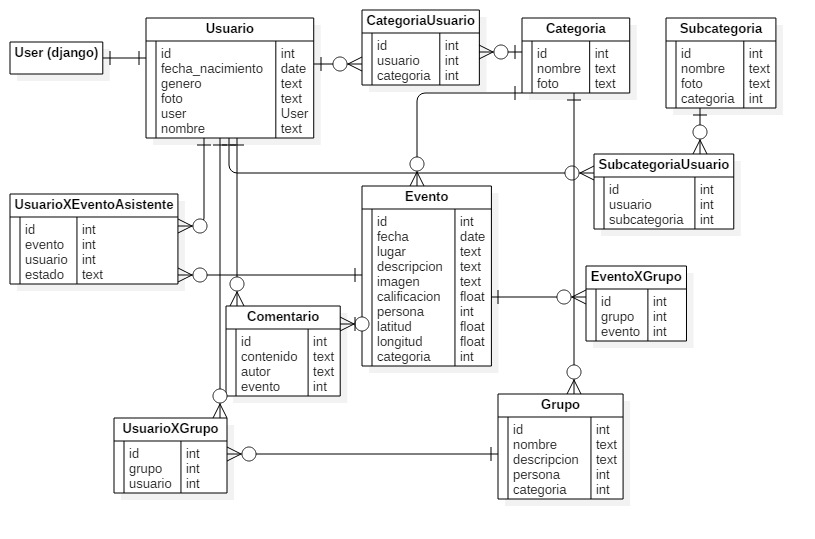
El backend fue desarrollado en python, django rest framework, en el cual se hace el uso de una base de datos sqlite, la cual es utilizada mediante serializadores, los cuales son los encargados de poseer la lógica del manejo de datos del backend, para ser procesados por una vista y posteriormente utilizar la base de datos para guardar cambios o retornar información.

La aplicación también se conecta con el API de Facebook para hacer el inicio de sesión.

Para poder hacer solicitudes al backend desarrollado en django, se debe iniciar sesión en el sistema, para obtener un token, el cual deberá ser utilizado para hacer las peticiones posteriores.

El backend se encuentra disponible en heroku, lugar desde el cual se podrá acceder a los datos desde cualquier dispositivo.

## Diagrama de base de datos



En la imagen anterior, se puede apreciar el diagrama de base de datos implementado para el proyecto, se denota la tabla de User que posee django por defecto, asociada con los usuarios propios definidos. Hay tablas como Categoría y Subcategoría, que se conectan con tablas como Evento, Grupo o Subcategoría usuario. En el modelo se definieron tablas como Comentario, el cual posee un usuario y un evento asociados, de manera que al seleccionar un evento, y que el usuario esté iniciado en el sistema, se puedan ver los comentarios asociados a estos. La tabla Grupo, al igual que evento, posee una referencia a Categoría, además de a UsuarioXGrupo, de manera que se puedan ver los usuarios que pertenecen a un grupo en específico. Finalmente, la tabla UsuarioXEventoAsistente, es una tabla con la cual se quiso guardar a los eventos y usuarios asociados a estos, de manera que puedan representarse como usuarios asistentes.

## Pantallas de la aplicación

## Descripción de los web services

A continuación, se describirán los web services utilizados en la aplicación.

# Interacción con sistemas externos

## Mixpanel