**INSTITUTO TECNOLÓGICO DE COSTA RICA**

**ESCUELA DE COMPUTACIÓN**

**LICENCIATURA DE ADMINISTRACIÓN DE TECNOLOGÍAS DE INFORMACIÓN**

IC8041 - DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES

**MeetMeAt**

Proyecto de programación  
Entrega final

Prof. Andrei Fuentes Leiva

Estudiantes:

Andrey Bolaños Sandoval

Ariela Vargas Vargas

Diego Vásquez Valerio

29 de setiembre 2014

Cartago, Costa Rica

Tabla de contenidos

[2 Resumen ejecutivo 3](#_Toc399710960)

[3 Funcionalidades 4](#_Toc399710961)

[3.1 Administrativas 4](#_Toc399710962)

[3.2 Cliente 4](#_Toc399710963)

[4 Requerimientos 5](#_Toc399710964)

[4.1 Requerimientos funcionales 5](#_Toc399710965)

[4.2 Requerimientos no funcionales 6](#_Toc399710966)

[5 Interfaz Gráfica, Experiencia y Usabilidad (UI/UX) 8](#_Toc399710967)

[5.1 Wireframes 8](#_Toc399710968)

[5.2 Fuentes y Esquema Colores 10](#_Toc399710969)

[5.3 Mockups 11](#_Toc399710970)

[6 Descripción de diseño de alto nivel 12](#_Toc399710971)

[6.1 Diagrama de despliegue 13](#_Toc399710972)

[6.2 Diagrama de clases 14](#_Toc399710973)

[6.3 Diagrama de base de datos 15](#_Toc399710974)

[7 Descripción detallada 16](#_Toc399710975)

[7.1 Frontend 16](#_Toc399710976)

[7.1.1 Onsen UI 16](#_Toc399710977)

[7.1.2 Angular js 16](#_Toc399710978)

[7.2 BackEnd 16](#_Toc399710979)

[7.2.1 Cakephp 16](#_Toc399710980)

[7.2.2 PHP REST SQL 16](#_Toc399710981)

[7.3 Patrones de diseño 17](#_Toc399710982)

[7.3.1 Patrón de navegación 17](#_Toc399710983)

[7.3.2 Patrón de navegación secuandaria 18](#_Toc399710984)

[7.3.3 Patrón de formularios 19](#_Toc399710985)

[7.3.4 Patrón de búsquedas 19](#_Toc399710986)

[8 Interacción con sistemas externos 21](#_Toc399710987)

[8.1 API Facebook 21](#_Toc399710988)

[8.1.1 Método de login 21](#_Toc399710989)

[8.2 API Foursquare 21](#_Toc399710990)

[8.2.1 Método de obtención de coordenadas geográficas 21](#_Toc399710991)

[9 Manual de usuario 22](#_Toc399710992)

# Resumen ejecutivo

MeetmeAt reúne a tus amigos y contactos en una sola aplicación para compartir las actividades sociales que desees realizar. Desde crear una lista de posibles actividades hasta elegir de manera aleatoria el próximo sitio para seguir la fiesta, eso sí, ajustándose a tu presupuesto y ubicación geográfica, para localizar los lugares de mayor proximidad.

Así también, MeetMeAt considera tus gustos, e incluye un filtro de los sitios que deseas visitar según tus preferencias de música, ambientes sociales, comidas, entre otros.

Salí de la rutina con MeetMeAt, donde tendrás la facilidad de planear tus actividades sociales.



# Funcionalidades

Entiéndase el uso de la palabra gestionar como crear, consultar, actualizar y eliminar la entidad respectiva.

## Administrativas

1. Login/logout de usuarios
2. Gestionar usuarios
3. Gestionar tags
   1. Los tags son entidades que almacenan datos relacionados con los eventos para clasificarlos por tipos de música, bebidas, comidas, ambiente, entre otros.

## Cliente

1. Registrar usuario
   1. Un usuario se registra a través de facebook.
2. Getionar eventos
   1. Al registrarse un usuario este puede crear eventos para planificar sus actividades sociales.
3. Getionar actividades
   1. La planificación del evento puede ser dividido en actividades específicas.
   2. Así también el usuario puede crear un reto para la actividad.
4. Gestionar ubicación del lugar
   1. Para determinar la ubicación de los eventos es necesario registrar los lugares, para ellos la entidad Venues obtiene de foursquare las cordenadas geográficas de latitud y longitud.
5. Invitar contactos
   1. El usuario puede invitar a sus contactos a su evento para que notifiquen si se unen a este o no.

# Requerimientos

Para el diseño de MeetMeAT es necesario definir los requerimientos funcionales y no funcionales que guiarán el desarrollo de la aplicación.

Los requerimientos funcionales indican lo que el sistema deben hacer y no deben hacer. Estas declaraciones de los servicios indican al sistema cómo éste debe reaccionar a entradas y situaciones que se presenten para el flujo continuo del programa.

Por otro lado, los requerimientos no funcionales indican cómo se deben llevar a cabo las funcionalidades en términos de calidad para que la solución final sea aceptada por el usuario.

## Requerimientos funcionales

|  |  |
| --- | --- |
| RF-1 | Gestionar usuarios |
| Funcional | El administrador podrá agregar, consultar, editar, y eliminar un usuario del sistema. Atributos: username, email, password, role, created, modified, status. |

|  |  |
| --- | --- |
| RF-2 | Gestionar eventos |
| Funcional | El usuario podrá agregar, consultar, editar, y eliminar un evento de la aplicación.  Atributos: title, theme, date, photo, description, budget, startTime, endTime. |

|  |  |
| --- | --- |
| RF-3 | Gestionar actividades |
| Funcional | El usuario podrá agregar, consultar, editar, y eliminar una o más actividades asociadas a un evento de la aplicación.  Atributos: name, description, budget, status. |

|  |  |
| --- | --- |
| RF-4 | Gestionar ubicación del lugar |
| Funcional | El usuario podrá agregar, consultar, editar, y eliminar la ubicación de un lugar en relación con un evento de la aplicación. Los datos para agregar la ubicación son obtenidos por medio de uso del API de Foursquare.  Atributos: latitude, longitude. |

|  |  |
| --- | --- |
| RF-5 | Gestionar etiquetas |
| Funcional | El administrador podrá agregar, consultar, editar, y eliminar una etiqueta asociada a un evento de la aplicación. Las etiquetas son de datos relacionados con música, bebidas, dress code, animación, mood, ambientes.  Atributos: name, description. |

|  |  |
| --- | --- |
| RF-6 | Asignar una actividad a un evento |
| Funcional | El usuario podrá asignar una o más actividades al evento creado por él en la aplicación. |

|  |  |
| --- | --- |
| RF-7 | Autenticación de usuarios con facebook |
| Funcional | Los usuarios pueden tener acceso a la aplicación al loguearse por medio de facebook. |

|  |  |
| --- | --- |
| RF-8 | Importar los contactos de facebook del usuario |
| Funcional | La aplicación debe permitir al usuario importar sus contactos de facebook para que estos sean invitados del evento que desee crear. |

|  |  |
| --- | --- |
| RF-9 | Invitar contactos a un evento |
| Funcional | El usuario podrá asignar uno o más invitados al evento creado por él. |

|  |  |
| --- | --- |
| RF-10 | Calcular presupuesto del evento |
| Funcional | El sistema deberá calcular el presupuesto total del evento según la suma del costo de las actividades, si existen actividades relacionadas al evento. |

## Requerimientos no funcionales

|  |  |
| --- | --- |
| RNF-1 | Interrupciones, notificaciones y multitasking |
| No Funcional | Cuando el usuario responda a la interrupción (como aceptar una llamada o la lectura de un SMS), la aplicación tendrá la capacidad para funcionar normalmente y reanudar el servicio después de que el usuario termina el manejo de la interrupción o después de que prefieren ignorar la interrupción. |

|  |  |
| --- | --- |
| RNF-2 | Resolución de pantalla y factores de forma |
| No Funcional | La aplicación debe ser capaz de ajustarse a la resolución de pantalla y factores de forma de los dispositivos móviles. |

|  |  |
| --- | --- |
| RNF-3 | Tiempos de procesamiento |
| No Funcional | El tiempo aceptable para realizar funciones claves o datos de exportación/importación es de 10 segundos. |

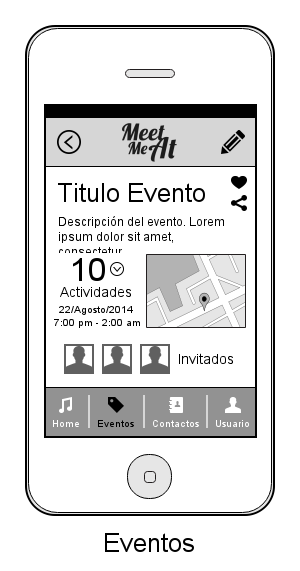
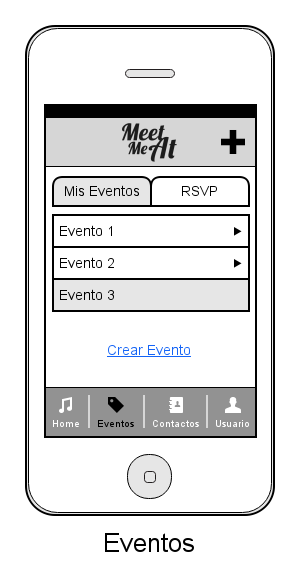
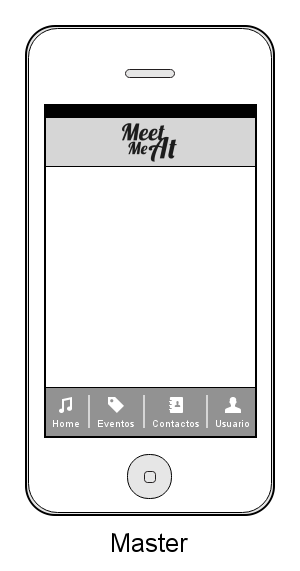
|  |  |
| --- | --- |
| RNF-4 | Consultas y tiempos de reportes |
| No Funcional | Las consultas o tiempos de reportes con sistemas externos deben considerar los tiempos de consulta aceptables (máximo 1 minuto), proporcionados por una API. |

|  |  |
| --- | --- |
| RNF-5 | Funcionamiento de la aplicación considerando horarios de disponibilidad |
| No Funcional | La aplicación debe estar disponible 24/7. |

|  |  |
| --- | --- |
| RNF-6 | Funcionamiento de la aplicación considerando la ubicación |
| No Funcional | La ubicación geográfica, los requisitos de conexión y las restricciones de una red local prevalecen, por tanto deben ser considerados para lograr alcanzar las funcionalidades de la aplicación. |

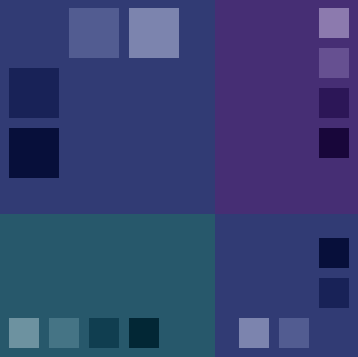
# Interfaz Gráfica, Experiencia y Usabilidad (UI/UX)

## Wireframes



## Fuentes y Esquema Colores

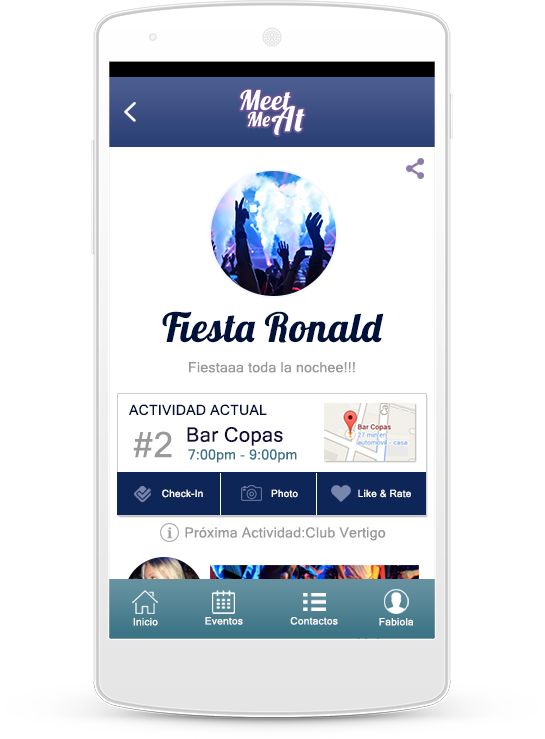
El diseño visual seguirá la tendencia “azul” de las aplicaciones sociales. Se ha comprobado que este color se relaciona psicológicamente a conceptos como: comunicación, confianza y seguridad.[[1]](#footnote-1) Se desea que MeetMeAt simplifique estos conceptos en la creación de eventos.



Link: <http://paletton.com/#uid=53T0u0kllllaFw0g0qFqFg0w0aF>

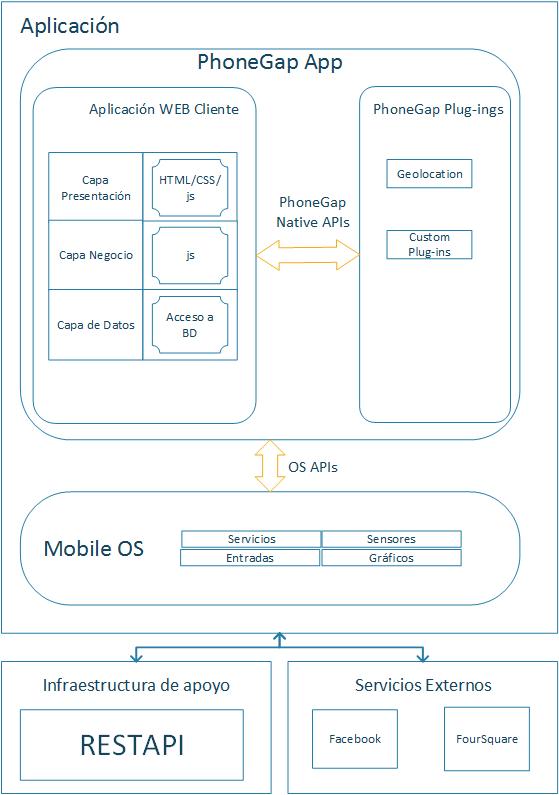
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Tipo Letra | Tamaño | Estilos | Color |
| Logo | Lobster | -- | Bold | #070F3A / #7C84AE |
| Titulos 1 | Sans, Droid Sans | 18px | Bold | #070F3A |
| Títulos 2 | Sans, Droid Sans | 14px | Bold | #182257 |
| Contenido | Sans, Droid Sans | 12px | Normal | #000000 |
| Formulario | Sans, Droid Sans | 12px | Normal | #000000 |
| Errores | Sans, Droid Sans | 12px | Normal | #FF0000 |

## Mockups

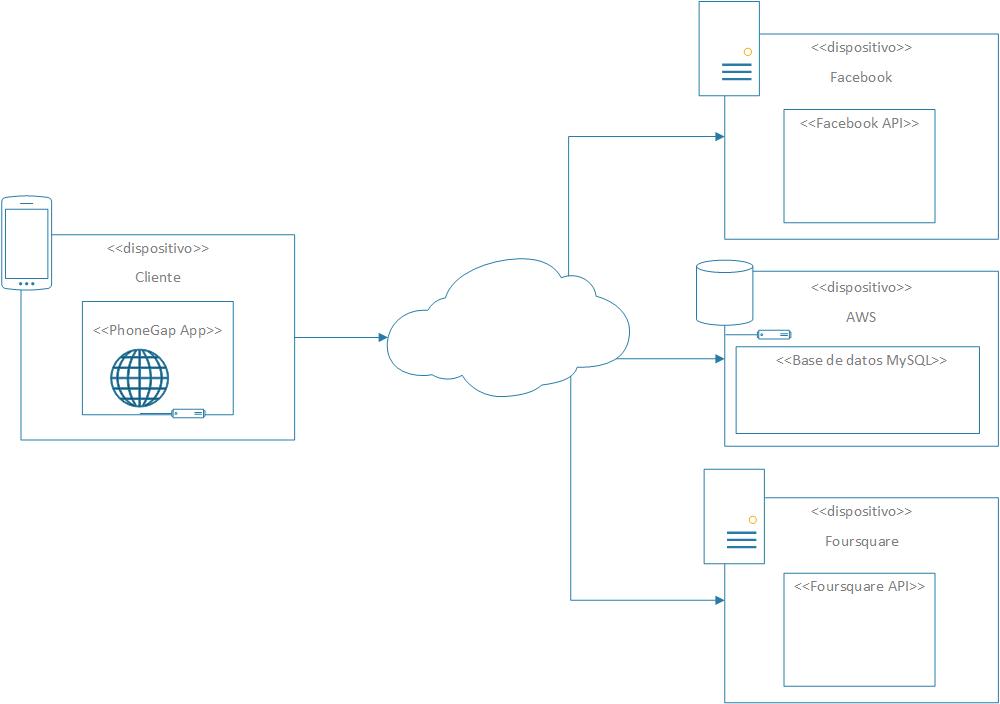
# Descripción de diseño de alto nivel

La aplicación se va a desarrollar utilizando el framework PhoneGap tomando ventaja de nuestros conocimientos de HTML, css y java script. La base de datos se va a hospedar utilizando los servicios proporcionados por Amazon.com con AWS.

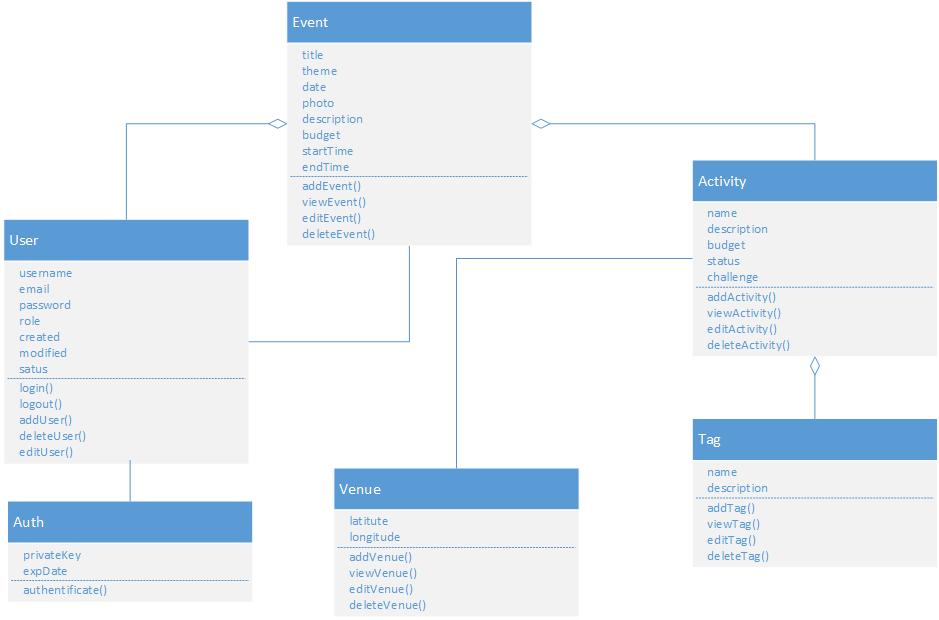


## Diagrama de despliegue

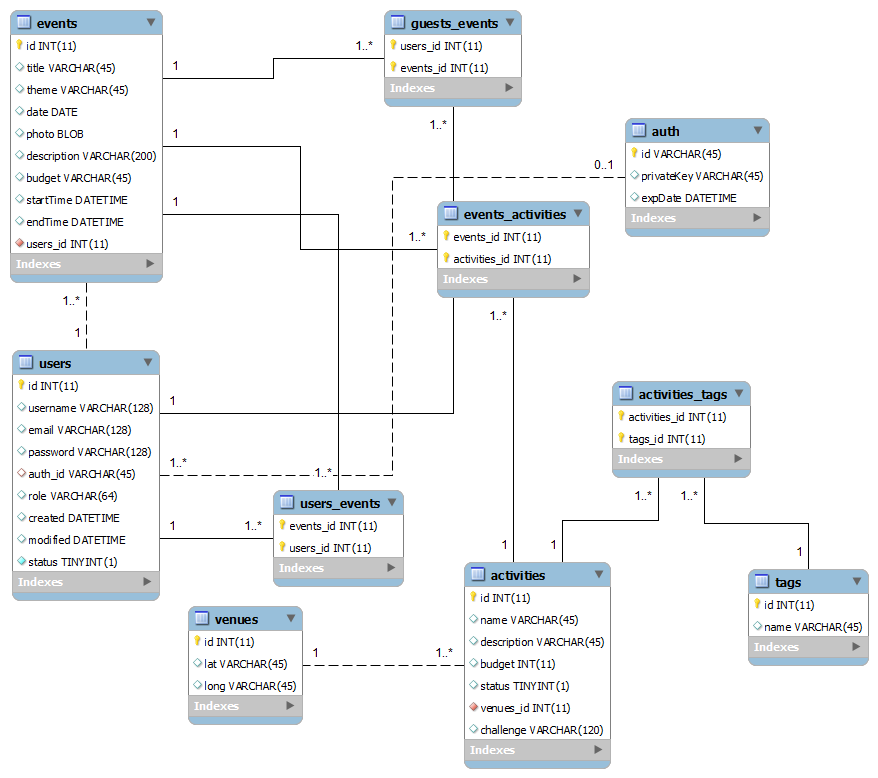
Este diagrama representa los dispositivos involucrados en el funcionamiento del sistema, así como los ambientes de ejecución para nuestra aplicación. El dispositivo móvil interactua con el servidor que contiene la base datos hospedada en Amazon AWS por medio de la red pública. Dicho dispositivo también interactúa con los servidores de facebook y de foursquare para obtener información de los contactos y lugares cercanos.



## Diagrama de clases



## Diagrama de base de datos



# Descripción detallada

A continuación se describen las herramientas utilizadas para desarrollar la aplicación MeetMeApp.

## Frontend

### Onsen UI

Onsen UI permite desarrollar aplicaciones híbridas y web. En el caso de las aplicaciones híbridas se utiliza la línea de comandos Cordova/PhoneGap. (Asial Corporation, 2014)

### Angular js

AngularJS es un framework JavaScript lanzado en 2012 por Miško Hevery, un empleado de Google, comenzó a trabajar en AngularJS en 2009. Este se distribuye como un archivo de JavaScript, y se puede agregar a una página web con una etiqueta <script>, extiendo atributos HTML con las Directivas, y uniendo datos al HTML con Expresiones. (Google, 2014)

## BackEnd

### Cakephp

CakePHP es un framework o marco de trabajo que facilita el desarrollo de aplicaciones web, utilizando el patrón de diseño MVC (Modelo Vista Controlador). Es de código abierto y se distribuye bajo licencia MIT. Al igual que Ruby On Rails, CakePHP facilita al usuario la interacción con la base de datos mediante el uso de ActiveRecord. Además hace uso del patrón Modelo Vista Controlador. (Wikipedia, 2014)

En el caso práctico de la aplicación MeetMeApp se crearon los siguientes modelos, controladores, y vistas.

* Modelos: User.php, Event.php, Activity.php, Venue.php, Tag.php.
* Controller: UsersController.php, EventsController.php, ActivitiesController.php, VenuesController.php, TagsController.php.
* Vistas: en el caso de las vistas solo se crearon unas específicas para User (add.ctp, edit.ctp, login.ctp, index.ctp), las demás hacen uso de scaffolding.

### PHP REST SQL

Es una interfaz HTTP REST para bases de datos MySQL escrito en PHP.

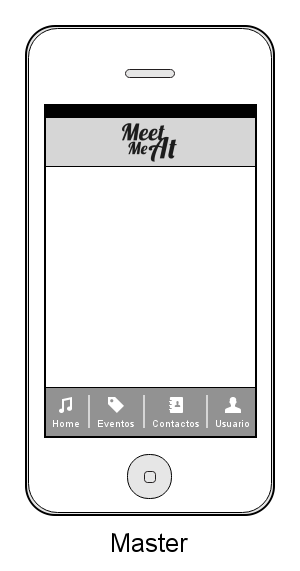
## Patrones de diseño

### Patrón de navegación

#### Springboard



#### Tabs

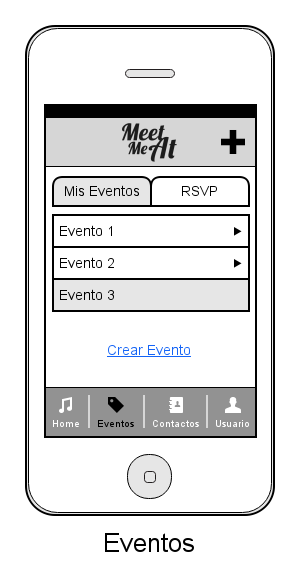


#### Listview



### Patrón de navegación secuandaria

#### Master-detail



### Patrón de formularios

#### Formularios



#### Form de ingreso

PANTALLAZO

### Patrón de búsquedas

#### Búsqueda explícita



# Interacción con sistemas externos

La aplicación tiene interacción con otros sistemas externos para permitir diversas funcionales. Para conocer esos sistemas se realiza una breve cescripción de los web services utilizados, explicando los métodos del API que debieron ser implementados.

## API Facebook

Facebook ofrece un entorno de software para crear aplicaciones y servicios que accedan a datos de la red social. El API permite funcionalidades como la autenticación y plugin sociales, los cuales serán utilizados para conocer la información del perfil del usuario, foto de perfil, así como para obtener una lista de sus amigos y sus nombres.

### Método de login

(descripción, url, params, método http)

## API Foursquare

Foursquare brinda acceso a su base de datos de lugares, así como la posibilidad de interactuar con sus usuarios. Este API nos permite buscar lugares en un área determinada, conectarse con amigos, saber cuando un usuario hace un check-in e incluso obtener datos globales. Este API será implementado para recomendar lugares de interés a sus actividades.

### Método de obtención de coordenadas geográficas

(descripción, url, params, método http)

# Manual de usuario

1. ## Tomado de “Social Media giants – why they use blue as their primary branding colour” http://www.aliciacowan.com/social-media-and-digital-marketing/strategy-and-advice/social-media-giants-branding

   [↑](#footnote-ref-1)