# Especificación de requerimientos de software

# para

# FCEFyN maps

Version 1.0

Preparada por Jeremías Benítez

cmd

02/05/2017

Tabla de contenidos

Tabla de contenidos [2](#__RefHeading___Toc20732946)

Historia de revi**siones 4**

**1. Introducción 5**

**1.1 Propósito 5**

**1.2 Audiencia Objetivo y Sugerencias de Lectura 5**

**1.3 Scope del Producto 5**

**1.4 Referencias** **5**

2. **Descripción general 7**

**2.1 Definición 7**

**2.2 Funciones del Producto 7**

**2.3 Clases de Usario y Características 8**

**2.4 Ambiente de Operación 8**

**2.5 Reestricciones de Diseño e Implementación 8**

**2.6 Documentación de Usuario 9**

**2.7 Supociciones y Dependencias** 9

**3. Listado de Requerimientos Funcionales** 10

**4. Detalle de los Requerimientos 10**

**4.1 REQ 1: Exploración** 10

4.1.1 **Objetivo**: 10

4.1.2 Input: 10

4.1.3 Output: 10

4.1.4 Pre-condition: 10

4.1.5 Steps: 11

4.1.6 Post-condition 11

4.1.7 **Diagrama** 11

**4.2 REQ 2: Recorrido más corto** 11

4.2.1 **Objetivo**: 11

4.2.2 Input: 11

4.2.3 Output: 11

4.2.4 Pre-condition: 11

4.2.5 **Pasos**: 11

4.2.6 Post-condition 11

4.2.7 **Caso Excepcional** 12

4.2.8 **Diagrama** 12

**4.3 REQ 3: Búsqueda** [1](#__RefHeading___Toc20732969)3

4.3.1 **Objetivo**: 13

4.3.2 Input: 13

4.3.3 Output: 13

4.3.4 Pre-condition: 13

4.3.5 **Pasos**: 13

4.3.6 Post-condition 13

4.3.7 **Caso Excepcional** 13

4.3.8  **Diagrama** 13

**4.4 REQ 3: Zoom** 1[4](#__RefHeading___Toc20732969)

4.4.1 **Objetivo**: 1[4](#__RefHeading___Toc20732971)

4.4.2 Input: 1[4](#__RefHeading___Toc20732972)

4.4.3 Output: 1[4](#__RefHeading___Toc20732973)

4.4.4 Pre-condition: 1[4](#__RefHeading___Toc20732975)

4.4.5 **Pasos**: 1[4](#__RefHeading___Toc20732976)

4.4.6 Post-condition 15

4.4.7 **Caso Excepcional** 15

4.4.8  **Diagrama** [15](#__RefHeading___Toc20732979)

5. **Requerimientos No Funcionales** [1](#__RefHeading___Toc20732981)5

**5.1 Requerimientos de** Performance 15

**5.2 Requerimientos de Calidad de Software** 1[5](#__RefHeading___Toc20732982)

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Fecha** | **Razón de los cambios** | **Version** |
| **Jeremías Benítez** | **02/05/17** | Creación del documento. | V1.0.0 |
| **Jeremías Benítez** | **03/05/17** | Se agregó toda la sección 2.  Se cambió el nombre de la aplicación.  Se actualizó el índice. | V1.0.1 |
| Jeremías Benítez | 06/05/17 | Se eliminó lo relacionado con la base de datos.  Se agregaron los requisitos funcionales. | V1.1.0 |
| Jeremías Benítez | 07/05/17 | Agregado requerimiento funcional 4.  Agregados requerimientos no funcionales.  Agregados diagramas.  Corregido el índice. | V1.1.1 |

# Introducción

## Propósito

El presente documento ha sido redactado para especificar en su totalidad los requerimientos de las versiones v1.X de la aplicación móvil FCEFyN maps.

## Audiencia Objetivo y Sugerencias de Lectura

La audiencia objetivo de este SRS está compuesta por los siguientes individuos:

* Desarrolladores, quienes deben implementar el software basandosé en estos requerimientos.
* Testers, quienes deben asegurarse que dichos requerimientos sean validables y cuyos tests deben realizar dicha validación.
* Project Managers.
* Personal encargado de la redacción de la documentación.
* Ingenieros de la Calidad, que deben serciorarse que los requerimientos de calidad sean cumplidos.

Este documento comienza ofreciendo una visión de alto nivel de la aplicación FCEFyN maps. Luego se enumeran las restricciones de software y hardware, como así también las dependencias necesarias y cualquier dato que brinde información que facilite la comprensión del documento.

La mayor parte del mismo se ocupa de listar los requerimientos. Cada uno de los mismo es descripto utilizando casos de uso, así como diagramas de secuencia donde sea necesario.

Al final se encuentral los requerimientos no funcionales establecidos para este sistema.

## Scope del Producto

La aplicación deberá ayudar tanto a alumnos como a profesores a desenvolverse en el edificio de la cede Ciudad Universitaria de la FCEFyN. A ambos les permitirá acceder a un mapa de la institución con las aulas propiamente identificadas. Además, el sistema les indicará, de forma visual, cuál es el camino más corto desde el aula o lugar identificado en el que se encuentre el usuario y el aula o lugar identificado al que desea ir. La idea es que el sistema le permita a los usuarios un manejo más eficiente del tiempo al permitir las averiguaciones pertinentes al mismo desde el celular del usuario, sin necesidad de estar el mismo en el edificio.

El sistema no se funcionará con GPS, por lo que no será posible dar el recorrido desde la posición real del usuario hasta la posición destino.

## Referencias

Templates de documentos de requerimientos:

* [https://drive.google.com/file/d/0B6Lbceno346BVloxWUlfUUFRTG8/](https://drive.google.com/file/d/0B6Lbceno346BVloxWUlfUUFRTG8/view)
* [https://drive.google.com/file/d/0B6Lbceno346BYU1wOXV4Q01WSkk/](https://drive.google.com/file/d/0B6Lbceno346BYU1wOXV4Q01WSkk/view)
* <http://www2.latech.edu/~box/ase/srs_template.doc>

Gráficos:

* Casos de uso, diagramas de actividades: <https://yuml.me/diagram/scruffy/activity/draw>
* Diagramas de secuencias: <https://www.websequencediagrams.com/>

Sommerville, I., (2011), *Ingeniería de software*, México, Pearson.

# Descripción General

Esta sección provee una descripción de alto nivel de la aplicación FCEFyN maps.

## Definición

La aplicación FCEFyN maps es un nuevo sistema, el cual busca ofrecer a sus usuarios (los cuales serán los alumnos, personal docente y no docente de la institución) una forma más simple de desarrollarse en el ambiente universitario, permitiendo la posibilidad de navegar mejor dentro de la institución.

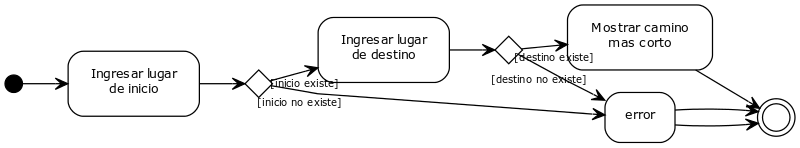
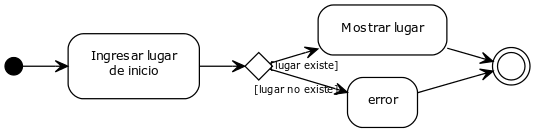
Diagrama de actividades para la acción de buscar camino más corto.

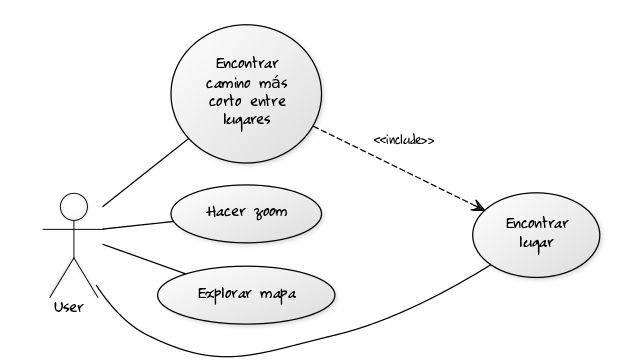
Diagrama de actividades para la acción de buscar un lugar.

## Funciones del Producto

El sistema debe permitirle a los usuarios realizar las siguientes acciones:

**A todos los usuarios:**

* Encontrar el camino más corto para llegar desde la posición de inicio provista, hasta la de destino deseada.
* Acceder al mapa de la facultad, donde estarán listados todos los edificios, pisos y entrepisos de la cede correspondiente.



## Diagrama de casos de uso para la aplicación.

## Clases de Usuario y Características

Hay dos grupos principales de usuarios que se espera interactúen con este sistema. El primer grupo está compuesto por los Administradores de Sistema encargados de mantener la integridad de los datos y la estabilidad del sistema. Este grupo tiene un gran conocimiento técnico. Si interacción con el sistema es mínima y solamente necesaria para hacer backups de las bases de datos, como también para actualizar los datos según sea necesario.

El segundo grupo es el de los usuarios. Se espera que tengan un conocimiento promedio de manejo de aplicaciones móviles. La interacción de los mismos con el sistema va a depender de sus necesidades, pero podemos asumir que van a interactuar con el mismo por lo menos una vez por día.

## Ambiente de Operación

El software correrá en dispositivos móviles que operen con el sistema operativo Android 5 o superior.

## Restricciones de Diseño e Implementación

**Limitaciones de software:**

* El sistema necesitará Android versión 5 o superior.
* Andorid Studio 2.3 será usado para el desarrollo del software.

**Limitaciones de hardware:**

* Cada usuario necesitará un dispositivo móvil que corra Android 5 o superior.

## Documentación de Usuario

Junto con el sistema se proveerá un link que dirija al usuario a una página on-line donde se encuentre el manual de uso.

## Suposiciones y Dependencias

El sistema depende de Java y del sistema operativo móvil Android (versión 5 o superior) para funcionar.

# Listado de Requerimientos Funcionales

**REQ 1:** Exploración – el usuario deberá poder recorrer los mapas disponibles.

**REQ 2:** Recorrido más corto – el sistema deberá encontrar el camino más corto desde la posición indicada de inicio y la posición de llegada.

**REQ 3:** Búsqueda – el sistema deberá poder localizar el aula u oficina indicada por el usuario y devolverle al usuario el piso en donde dicho lugar se encuentra. Además actualizará la vista del mapa, centrando dicho lugar.

# Detalle de los Requerimientos

## REQ 1: Exploración

### Objetivo:

El usuario debe poder recorrer libremente los mapas de todos los pisos y edificios de la intitución que estén disponibles.

### Input:

N/A

### Output:

En la pantalla del dispositivo móvil se deberá mostrar el mapa de la planta baja del edificio principal, el cual se deberá poder recorrer en cualquier dirección, así como también se podrá cambiar de piso ya sea mediante un menú deslizable que se encuentre a la izquierda de la pantalla; o haciendo clic en los íconos correspondientes a “subir/bajar las escaleras que se encuentren en las mismas.

### Pre-condición:

N/A

### Pasos:

#### Paso 1: El usuario abre la aplicación.

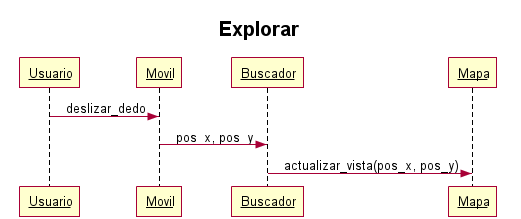
#### Paso 2: El usuario desliza el mapa con el dedo.

#### Paso 3 (opcional): El usuario cambia de piso utilizando el menú o haciendo clic en los íconos.

### Post-condition

N/A

### Diagrama



## REQ 2: Recorrido más corto

### Objetivo:

El sistema debe encontrar el camino más corto desde la posición indicada de inicio y la posición de llegada. Esta opción estará disponible a través de un botón flotante ubicado en la esquina inferior derecha, el cual abrirá el menú de búsqueda.

### Input:

El usuario deberá ingresar el lugar desde el cual desea partir, y el lugar hasta el cual desea llegar. Se entiende por lugar a cualquier aula, oficina, u otros que tengan asignada una identificación en el mapa.

### Output:

Sobre el mapa mostrado en la pantalla principal se mostrará con una línea azul el camino que se deberá recorrer para llegar, recorriendo la menor distancia posible, hasta el lugar especificado por el usuario.

### Pre-condición:

N/A

### Pasos:

#### Paso 1: El usuario abre la aplicación.

#### Paso 2: El usuario da tap al botón de generación de recorrido.

#### Paso 3 : El usuario Ingresa ambos lugares en los campos de textos correspondientes.

#### *Paso 4:* El usuario hace clic sobre el botón 'Obtener recorrido'.

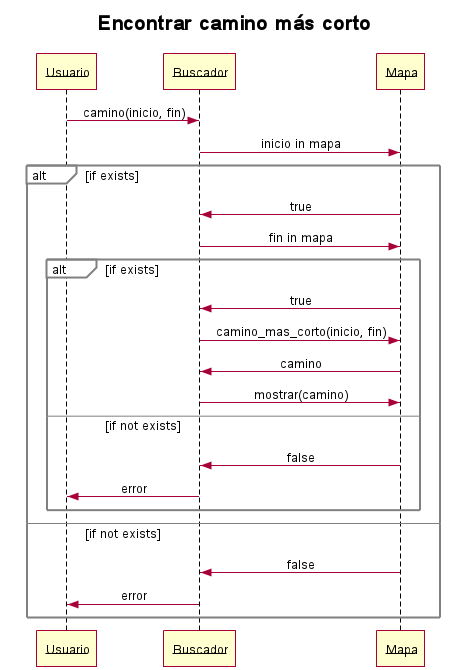
### Post-condition

N/A

### **Caso Excepcional:**

Si alguno de los lugares provistos por el usuario no son encontrados por el sistema, el mismo emitirá un mensaje de error.

### **Diagrama**



## REQ 3: Búsqueda

### Objetivo:

El sistema debe poder localizar el aula u oficina indicada por el usuario y devolverle al usuario el piso en donde dicho lugar se encuentra. Además actualizará la vista del mapa, centrando dicho lugar.

### Input:

El usuario deberá ingresar el lugar del cual desea conocer la ubicación.

### Output:

Si el sistema tiene dicho lugar registrado deberá actualizar la vista del mapa para centrarla en dicho lugar. Deberá devolver también un mensaje donde se le indique textualmente al usuario en qué piso se encuentra.

### Pre-condición:

N/A

### Pasos:

#### Paso 1: El usuario abre la aplicación.

#### Paso 2: El usuario da tap al botón de busqueda, ubicado en el menú de la izquierda.

#### Paso 3 : El usuario Ingresa el lugar en el campo correspondiente.

#### *Paso 4:* El usuario hace clic sobre el botón 'Encontrar'.

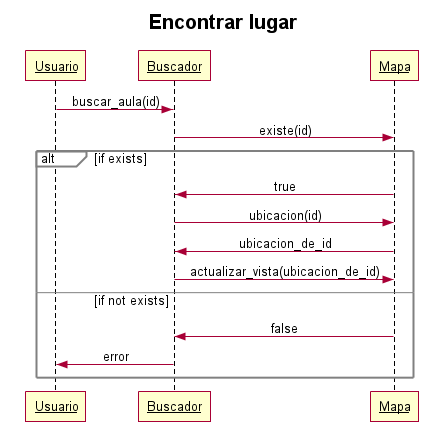
### Post-condition

N/A

### **Caso Excepcional:**

Si el lugar provisto no es encontrado por el sistema, se devolverá un mensaje de error y la vista principal permanecerá sin modificaciones.

### **Diagrama**



## REQ 4: Zoom

### Objetivo:

El sistema debe poder manejar zoom realizado por el usuario sobre el mapa, ampliando o reduciendo la escala del mismo según sea solicitado.

### Input:

El usuario realizará una acción de “pinching” con ambos dedos, ya sea en dirección entrante o saliente.

### Output:

La escala del mapa mostrado variará acorde a la dirección del “pinching”: si la misma es saliente respecto del centro del celular, la escala aumentará; si es, en cambio, entrante, la escala se reducirá-

### Pre-condición:

N/A

### Pasos:

#### Paso 1: El usuario abre la aplicación.

#### Paso 2: El usuario realiza un “pinching” en una dirección (saliente o entrante).

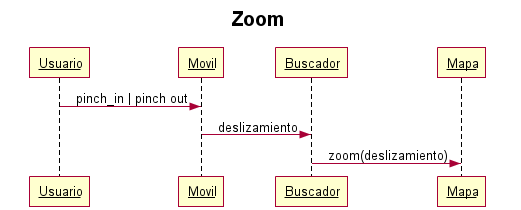
### Post-condition

N/A

### **Caso Excepcional:**

N/A

### **Diagrama**



# Requerimientos No Funcionales

## Requerimientos de Performance

* El sistema debe encontrar la ruta entre dos puntos, o devolver un error si uno de los puntos no es válido, en menos de 0.5 segundos. Esto es así para evitar que la app caiga en desuso por no ser práctica.
* La interfaz de usuario debe ser lo más intuitiva que sea posible, para así volver útil la aplicación a personas mayores.

## Requerimientos de Calidad de Software

**Confiabilidad:**

* El sistema debe estar disponible un 95% de las veces que el usuario lo intente usar.

**Portabilidad:**

* El sistema sólo debe ser portable a otros dispositivos que corran el mismo sistema operativo.

**Facilidad de uso:**

* El sistema no debe requerir entrenamiento especial para ser operado por un usuario promedio.
* El sistema debe requerir entrenamiento de no más de una sesión de 10 minutos para ser operado por un usuario con bajo nivel de manejo de aplicaciones.
* El sistema debe poseer una sola ventana de ayuda, más un link a un manual de usuario on-line.