

UNIVERSIDAD OBERTA DE CATALUÑA

2º Ciclo Ingeniería Informática

PFC – Redes de Computadores

localiCEA: Localización de Centros Educativos en Andalucía

[Memoria del proyecto]

Autora: María Urbano Olmo
Consultor: Victor Carceler Hontoria

Enero 2013

Índice de contenido

1. DEFINICIÓN DEL PROYECTO.....	6
1.1. Descripción del proyecto.....	6
1.2. Justificación de la tecnología elegida.....	6
1.3. Objetivos generales.....	7
1.4. Recursos.....	8
Recursos hardware.....	8
Recursos software.....	8
2. ANÁLISIS DE REQUISITOS.....	9
2.1. Funciones del producto.....	9
2.2. Características del usuario.....	9
2.3. Restricciones.....	10
2.4. Requisitos específicos.....	10
Requisitos funcionales.....	10
Requisitos de interfaces externas.....	11
Requisitos de rendimiento.....	11
2.5. Comportamiento del sistema.....	11
Caso de uso Seleccionar provincia.....	13
Caso de uso Seleccionar municipio.....	13
Caso de uso Seleccionar enseñanzas.....	14
Caso de uso Seleccionar servicios y programas.....	15
Caso de uso Visualizar centros en el mapa.....	15
Caso de uso Listar centros en el mapa.....	16
Caso de uso Visualizar mi ubicación.....	17
Caso de uso Consultar información de un centro.....	18
Caso de uso Llamar a un centro.....	19
Caso de uso Visualizar ruta de centro a mi ubicación.....	20
3. DISEÑO DEL PROYECTO.....	21
3.1. Modelo de datos.....	21
Análisis de los tipos de entidad.....	22
Análisis de los tipos de interrelación.....	23
Esquema Entidad-Interrelación.....	23

3.2. Arquitectura general de la aplicación.....	25
Paquete raíz com.localicea.....	26
Paquete actividades.....	27
Paquete bd.....	27
Paquete mapa.....	28
3.3. Diseño de la interfaz.....	28
Pantalla inicial.....	28
Selección de la provincia.....	29
Selección del municipio.....	30
Selección de enseñanza.....	31
Selección de servicios y programas.....	32
Visualizar centros en el mapa.....	34
Información de un centro seleccionado.....	35
Mi ubicación.....	36
Listado de centros resultado.....	37
4. PLANIFICACIÓN.....	39
4.1. Entregables.....	39
PEC1 - Plan de trabajo.....	39
PEC2 - Análisis y diseño de la aplicación. Prototipo.....	39
PEC3 - Desarrollo de la aplicación.....	39
Entrega Final.....	39
4.2. Planificación temporal.....	40
5. PLAN DE PRUEBAS.....	42
6. LICENCIA Y CRÉDITOS DE LOS RECURSOS GRÁFICOS.....	44
7. CONCLUSIONES Y FUTURAS MEJORAS	49
BIBLIOGRAFÍA.....	50
Consultas bibliográficas.....	50
Consultas web.....	50
ANEXO I. SCRIPT PYTHON PARA GENERAR LA CODIFICACIÓN GEOGRÁFICA DE LOS CENTROS.....	51
ANEXO II. ESTRUCTURA DE UN PROYECTO ANDROID.....	54
Carpeta src.....	54
Carpeta res.....	54
Carpeta gen.....	54
Carpeta assets.....	55

Carpeta bin.....	55
Carpeta libs.....	55
Fichero AndroidManifest.xml.....	55

Índice de ilustraciones

Ilustración 1: Diagrama de casos de uso.....	12
Ilustración 2: Diagrama Entidad-Interrelación.....	24
Ilustración 3: Diagrama de paquetes.....	26
Ilustración 4: Diseño pantalla “Inicio”.....	29
Ilustración 5: Diseño pantalla “Provincia”.....	30
Ilustración 6: Diseño pantalla “Municipio”.....	31
Ilustración 7: Diseño pantalla “Enseñanzas”.....	32
Ilustración 8: Diseño pantalla “Servicios y Programas”.....	33
Ilustración 9: Diseño diálogo “Búsqueda con todos/algunos de los requisitos”.....	34
Ilustración 10: Diseño pantalla “Centros en el mapa”.....	35
Ilustración 11: Diseño del panel “Información de un centro”.....	36
Ilustración 12: Diseño pantalla “Mi ubicación”.....	37
Ilustración 13: Diseño pantalla “Lista centros resultado”.....	38
Ilustración 14: Diagrama de Gantt de la planificación realizada.....	41

1. DEFINICIÓN DEL PROYECTO

1.1. Descripción del proyecto

Este proyecto lleva por nombre “**localiCEA: Localización de Centros Educativos en Andalucía**”. Se trata de una aplicación Android que pretende ayudar a los padres de la Comunidad Autónoma de Andalucía a conocer los centros educativos de su entorno, tanto en lo referente a las enseñanzas, servicios y proyectos que ofrecen, como en la localización de cada uno de ellos.

Se está observando a través de diversos estudios, que el tiempo que se dedica a las apps, se está aproximando cada vez más al tiempo que se dedica a navegar por internet. Por eso, es muy interesante llevar a las familias este tipo de información de manera rápida y fácil, a través de un smartphone o tableta Android; a la hora de escolarizar a los niños, es fundamental para los padres disponer de toda esta información para realizar la elección que más le interesa.

Para ello nos basaremos en los datos publicados desde la Junta de Andalucía en su [Catálogo de Datos Abiertos](#)¹. Este es un proyecto que se inició en 2011 y que consiste en la difusión de información del sector público de la Junta de Andalucía, siendo uno de sus objetivos fomentar la innovación tecnológica en el sector TIC andaluz. Entre los datos publicados encontramos el [directorio de centros docentes de Andalucía](#)², que nos permite conocer toda la relación de centros públicos, privados y concertados de ésta comunidad, con datos como su dirección, teléfono, enseñanzas, servicios, etc.

1.2. Justificación de la tecnología elegida

Este proyecto consistirá en el desarrollo de una aplicación Android con uso de mapas basado en la API de Google Maps. Hay muchas plataformas móviles disponibles en el mercado, pero la mayoría de las empresas prefieren el desarrollo Android, ya que ofrece grandes ventajas que no se pueden encontrar en cualquier otra plataforma móvil:

- Es un sistema para móviles gratuito y no hace falta pagar nada para adquirir las herramientas necesarias para programar aplicaciones para él.
- Al ser una tecnología de código abierto, los desarrolladores pueden reutilizar el código de otros programadores y aplicaciones y así ahorrar algo de tiempo que se puede utilizar para mejorar el *desarrollo Android* de una aplicación desde un punto de vista creativo.
- Ofrece excelentes gráficos de apoyo. Los usuarios de los dispositivos móviles siempre quedan impresionados y son atraídos por la alta calidad de los gráficos que la plataforma ofrece. Android proporciona soporte incorporado para potentes gráficos en 2D y 3D que permiten a las empresas atraer a los usuarios a través de aplicaciones originales.
- No está atado a un único fabricante de dispositivos.

1 <http://www.juntadeandalucia.es/datosabiertos/portal.html>

2 <http://www.juntadeandalucia.es/datosabiertos/portal/catalogo/detalle/13897.html>

- Campo de pruebas más ágil. Si tienes tu cuenta de desarrollador en Android, y creas tu aplicación, subirla al Play Store es tan sencillo como un par de clicks. Si quieres hacerlo en la tienda de iOS es necesario pasar un proceso de aprobación que tarda en torno a una semana, si todo va bien y sale aprobada a la primera.

Por otro lado, las ventajas de usar la API de Google Maps son las siguientes:

- Rapidez. El poder integrar en cuestión de segundos mapas en nuestras aplicaciones.
- Robustez. Está tan sumamente extendida que ha sido probada en toda clase de situaciones de estrés, por lo que podremos asegurar que su respuesta será adecuada en la inmensa mayoría de los casos.
- Imagen. Cuando los usuarios usan a menudo un buen servicio lo acaban reconociendo, y el asociar nuestra imagen a la de un grande como Google, puede ayudarnos a contagiarnos de la buena percepción que los usuarios tienen de este gigante.
- Mantenimiento. Los servicios accedidos a través de la API están en una constante mejora.

1.3. Objetivos generales

Como ya se ha comentado, el objetivo principal de este proyecto es dar respuesta a la necesidad de información que tienen las familias andaluzas a la hora de tomar una decisión tan importante como es dónde escolarizar a sus hijos.

Para ello, se pretende realizar una aplicación Android que permita a los interesados alcanzar los siguientes objetivos:

- Situar y localizar en el mapa los centros educativos de su municipio.
- Poder visualizar los centros que cumplan las características que ellos necesitan de un centro, con respecto a tres criterios:
 - Enseñanzas que se imparten en el centro.
 - Servicios que ofrece el centro: aula matinal, comedor y actividades extraescolares.
 - Programas educativos con los que cuenta el centro: bilingüe español-inglés, bilingüe español-francés, bilingüe español-alemán.
- Consultar la información relativa al centro seleccionado.
- Situar y centrar al usuario en el mapa para que pueda comprobar los centros que tiene más cercanos.
- Calcular y visualizar la ruta desde la ubicación del usuario al centro seleccionado.
- Llamar al centro seleccionado.

1.4. Recursos

Los recursos que inicialmente se van a utilizar para el desarrollo del proyecto se detallan a continuación.

Recursos hardware

El material hardware del que se dispone es el siguiente:

- Dell Studio 15 (1555). Intel Core 2 Duo 2.4GHz, 3GB de RAM con Ubuntu 12.04.1 LTS.
- Móvil HTC Desire donde se irá probando la aplicación.

Recursos software

El material software que se utilizará para el desarrollo de la aplicación es el siguiente:

- Eclipse Juno (versión 4.2 de la plataforma), con el plugin eGit como ayuda para gestionar nuestro repositorio.
- Android SDK Revisión 20.0.3
- Paquete con las APIs de Google Maps correspondiente al anterior SDK.
- LibreOffice 3.5.
- Gantt Project 2.5.5.
- Dia v0.97.2: editor de diagramas.
- <http://www.balsamiq.com/> : Balsamiq es una herramienta interactiva de wireframing.
- <https://bitbucket.org/> : servicio de alojamiento basado en web, usando repositorio Git.

2. ANÁLISIS DE REQUISITOS

En este apartado realizaremos una descripción completa del comportamiento del sistema que se va a desarrollar. Se definirá de manera clara y precisa las funcionalidades y restricciones que tendrá el sistema que se desea construir, así como todas las interacciones que tendrán los usuarios con el software. A partir de esta especificación estaremos en condiciones de establecer un diseño que se ajuste a los requerimientos aquí expuestos.

2.1. Funciones del producto

La aplicación tendrá funciones tales como:

- Localizar en el mapa: se mostrarán en el mapa los centros educativos del municipio seleccionado, que cumplan con las condiciones que se hayan especificado por parte del usuario.
- Datos de un centro: se trata de mostrar los datos conocidos del centro seleccionado; su nombre, dirección, que tipo de titularidad tiene, enseñanzas que imparte, etc.
- Llamar al centro: a partir de los datos del centro, dispondremos del teléfono del mismo, por lo que se podrá establecer una llamada en caso de requerirlo.
- Localizar la ubicación del usuario en el mapa: se podrá situar y centrar al usuario en el mapa para ver los centros que tenemos alrededor.
- Calcular y mostrar ruta: se podrá mostrar la ruta desde un centro seleccionado hasta la ubicación del usuario.

2.2. Características del usuario

Los usuarios de esta aplicación van a ser muy heterogéneos, ya que existen varios colectivos que pueden estar interesados en conocer los datos que se ofrecen:

- Padres que están en el momento de escolarizar a sus hijos, o cuyos hijos van a cambiar de centro por cambio de etapa educativa (por ejemplo, pasan de un colegio a un instituto) o por intereses particulares.
- Jóvenes y adultos que deciden continuar su formación a través de los ciclos formativos de grado medio o superior, o a través de la enseñanza de adultos; y que están interesados en saber qué centros las ofertan.
- Jóvenes y adultos que desean completar su formación con enseñanzas de idiomas, música, danza, etc.

Como vemos, los usuarios candidatos a usar la aplicación puede variar mucho en edad, por lo que puede ser que el usuario esté poco o muy acostumbrado a usar aplicaciones de este tipo. Esto es un rasgo a tener en cuenta, ya que debemos diseñar una aplicación lo más intuitiva posible, fácil de usar y que ofrezca de manera sencilla y visual la información que les interesa.

El único requisito que deben cumplir los usuarios es tener a su alcance un dispositivo con el sistema operativo Android; ya sea un smartphone o una tableta; cosa que hoy día es muy común.

2.3. Restricciones

La restricción más importante de esta aplicación es la información con la que podemos trabajar, ya que depende directamente de los datos que la Junta de Andalucía publica para su explotación por este tipo de aplicaciones. Esta información se puede consultar desde el [Catálogo de Datos Abiertos](#) de la Junta de Andalucía, donde ponen a nuestra disposición el [directorio de centros docentes de Andalucía](#); es decir, la relación de centros docentes públicos, privados y concertados de esta comunidad.

Sobre cada centro educativo podremos consultar los siguientes datos:

- ✓ Código del centro.
- ✓ Denominación del centro.
- ✓ Nombre.
- ✓ Dependencia.
- ✓ Domicilio.
- ✓ Localidad.
- ✓ Municipio.
- ✓ Provincia.
- ✓ Código postal.
- ✓ Teléfono.
- ✓ Enseñanzas que imparte.
- ✓ Servicios que ofrece.
- ✓ Programas educativos con los que cuenta el centro.

2.4. Requisitos específicos

Pasamos ahora a profundizar un poco más en los requisitos con los que debe cumplir nuestra aplicación:

Requisitos funcionales

- REQ01 Selección de la provincia y municipio: el usuario podrá seleccionar la provincia y municipio sobre la que quiere consultar los centros educativos.
- REQ02 Selección de las características deseadas: el usuario tendrá la posibilidad de seleccionar las enseñanzas, servicios y programas educativos que desea que tengan los centros a mostrar por la aplicación.
- REQ03 Localizar centros en el mapa: el usuario podrá visualizar la localización de

los centros educativos que cumplan con los requisitos que él ha establecido.

- REQ04 Descripción del centro: el usuario tendrá la posibilidad de visualizar la información disponible del centro, tras pulsar con el dedo sobre él.
- REQ05 Llamar por teléfono al centro: el usuario podrá llamar al centro si se dispone de su número de teléfono entre los datos conocidos de dicho centro.
- REQ06 Situar al usuario en el mapa: el usuario podrá situar y centrar su localización en el mapa, de modo que pueda ver los centros que tiene en sus cercanías.
- REQ07 Calcular la ruta desde un centro a la ubicación del usuario: el usuario podrá visualizar la ruta desde su ubicación al centro seleccionado, de modo que disponga de la posibilidad de ver cómo se llega al centro.

Requisitos de interfaces externas

- REQ08 Logotipo: El logotipo será propio y permitirá identificar el principal objetivo de la aplicación. Este logotipo será además el que se use para señalar la localización en el mapa de los centros educativos.
- REQ09 Interfaces del usuario: Se podrá comunicar con el usuario para aprovechar los requisitos del sistema, el usuario indicará al sistema las operaciones que debe realizar e introducirá los datos que el sistema le pida.
- REQ10 Interfaces del software: La comunicación entre los módulos del sistema se realizará mediante bases de datos relacionadas.

Requisitos de rendimiento

- REQ11 Tiempo de respuesta: La respuesta que dará el sistema con respecto a la petición del usuario deberá ser en tiempo real.

2.5. Comportamiento del sistema

En este apartado se describe y organiza el comportamiento deseado del sistema; es decir, la interacción del sistema con los usuarios y los procesos internos que se llevan a cabo para las distintas operaciones del sistema. Para realizar este análisis se hará uso de los diagramas de caso de uso.

Un caso de uso se define como una descripción de un conjunto de secuencias de acciones que ejecuta un sistema para producir un resultado observable de interés para un actor.

A continuación, se desarrollan los distintos casos de uso identificados para el dominio del problema que estamos tratando.

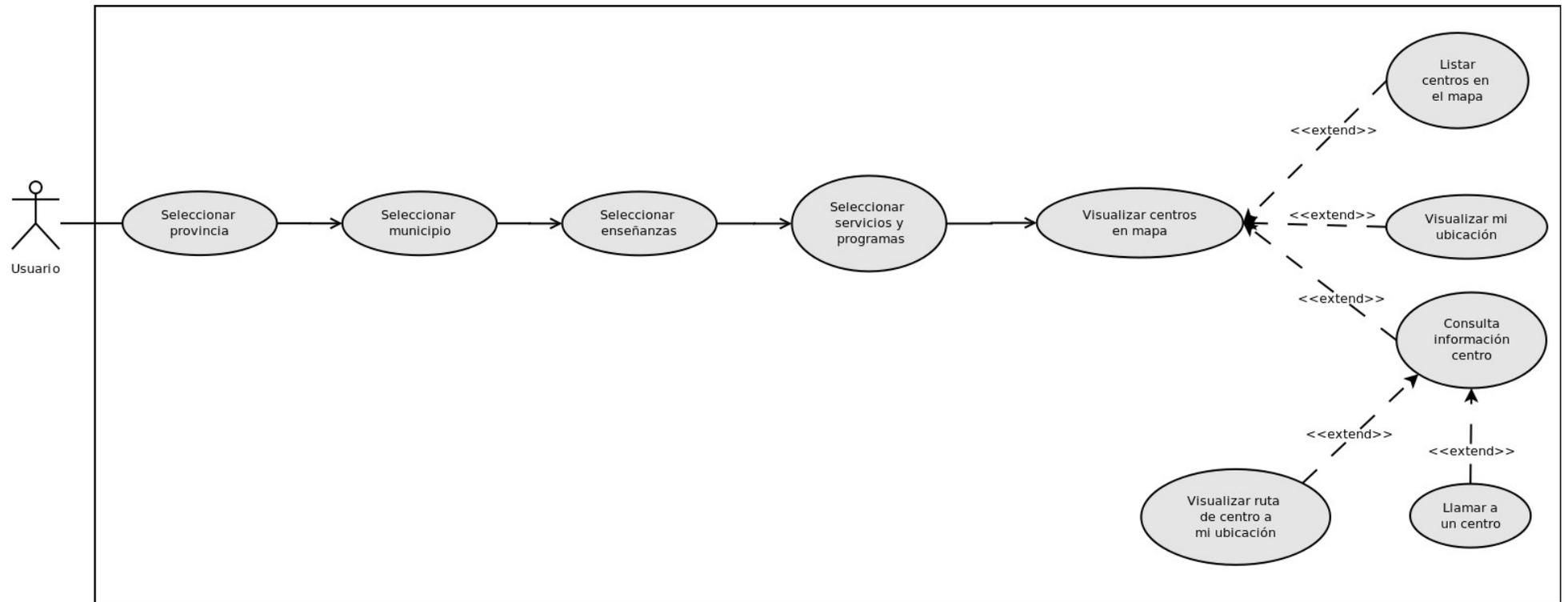


Ilustración 1: Diagrama de casos de uso

Caso de uso Seleccionar provincia

Caso de uso	Seleccionar provincia
Actor	Usuario final de la aplicación
Descripción	<p>El usuario desea iniciar la aplicación para poder hacer uso de las diferentes opciones que ésta ofrece. Las principales son:</p> <ul style="list-style-type: none"> ➤ Localizar en el mapa los centros que cumplen los requisitos establecidos por el usuario. ➤ Consultar la información correspondiente a un centro seleccionado. ➤ Realizar una llamada a un centro seleccionado. ➤ Conocer la ruta desde la ubicación del usuario hasta el centro seleccionado. <p>Por tanto, para comenzar a usar la aplicación, se le solicita al usuario que seleccione la provincia en la que quiere realizar la búsqueda de centros.</p>
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ La aplicación debe dar la opción de elegir una de las provincias de la Comunidad Autónoma de Andalucía.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide seleccionar una provincia.
Postcondiciones	No aplican.

Caso de uso Seleccionar municipio

Caso de uso	Seleccionar municipio
Actor	Usuario final de la aplicación
Descripción	<p>El usuario continua utilizando la aplicación y se le ofrecen por tanto dos opciones:</p> <ul style="list-style-type: none"> ➤ Seleccionar uno de los municipios pertenecientes a la provincia seleccionada anteriormente. ➤ Volver atrás para seleccionar una provincia distinta.

Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado la provincia donde desea buscar los centros. ✓ La aplicación debe dar la opción de elegir uno de los municipios de la provincia seleccionada anteriormente.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide seleccionar un municipio.
Postcondiciones	No aplica.

Caso de uso Seleccionar enseñanzas

Caso de uso	Seleccionar enseñanzas
Actor	Usuario final de la aplicación
Descripción	<p>El usuario continua utilizando la aplicación y puede actuar de varias maneras:</p> <ul style="list-style-type: none"> ➤ Seleccionar la/s enseñanza/s que desea que tenga el centro que busca. ➤ No seleccionar ninguna enseñanza. ➤ Volver atrás para seleccionar un municipio distinto.
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. ✓ La aplicación debe dar la opción tanto de no seleccionar ninguna enseñanza, como de seleccionar una o varias enseñanzas entre todas las que se imparten en los centros de Andalucía.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación.
Postcondiciones	No aplica.

Caso de uso Seleccionar servicios y programas

Caso de uso	Seleccionar servicios y programas
Actor	Usuario final de la aplicación
Descripción	<p>El usuario continua utilizando la aplicación y tiene las siguientes posibilidades:</p> <ul style="list-style-type: none"> ➤ Seleccionar uno o varios servicios y/o programas. ➤ No seleccionar nada. ➤ Volver atrás para rehacer la selección de enseñanzas previa.
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. La selección de enseñanzas no es obligatoria, si no se ha seleccionado ninguna, no se aplica este filtro. ✓ La aplicación debe dar la opción tanto de no seleccionar ningún servicio o programa, como de seleccionar uno o varios servicios y/o programas que se ofertan en los centros de Andalucía.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación.
Postcondiciones	No aplica.

Caso de uso Visualizar centros en el mapa

Caso de uso	Visualizar centros en el mapa
Actor	Usuario final de la aplicación
Descripción	<p>El usuario sigue adelante en el uso de la aplicación. Tras establecer el tipo de enseñanzas que busca y los servicios y programas que le interesan; se le mostrará un mapa con los centros resultantes de la búsqueda marcados en el mapa. Llegado este punto, el usuario tendrá varias opciones:</p> <ul style="list-style-type: none"> ➤ Mostrar un listado con los centros que se han localizado en el mapa, para que el usuario pueda dirigirse a uno en concreto, centrándolo

	<p>en el mapa.</p> <ul style="list-style-type: none"> ➤ Consultar la información correspondiente a los centros localizados, seleccionando cada uno de ellos. ➤ Visualizar y centrar en el mapa la ubicación del usuario, viendo así cuales son los centros más cercanos a su posición.
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. La selección de enseñanzas, servicios o programas no es obligatoria, si no se ha seleccionado nada, no se aplican estos filtros. ✓ La aplicación debe dar la opción de localizar los centros resultado de la búsqueda en el mapa.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide visualizar en el mapa los centros que cumplan con los requisitos que él ha establecido.
Postcondiciones	No aplica.

Caso de uso *Listar centros en el mapa*

Caso de uso	Listar centros en el mapa
Actor	Usuario final de la aplicación
Descripción	<p>Una vez el usuario ha llegado al punto de visualizar en pantalla el mapa del municipio seleccionado, con los centros que cumplen los requisitos que ha marcado previamente; una de las opciones que se le ofrecen es la de generar un listado con los nombres de los centros que se han encontrado y marcado en el mapa.</p> <p>A partir de este listado, el usuario puede seleccionar uno de los nombres de los centros, y automáticamente se le centrará dicho centro en el mapa, facilitándole así su consulta.</p>
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente.

	<ul style="list-style-type: none"> ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. ✓ El usuario ha tenido la opción de seleccionar las enseñanzas, servicios y programas que le interesan, aunque no es obligatorio en ninguno de los casos. ✓ El usuario ha decidido visualizar los centros resultantes de la búsqueda en el mapa. ✓ La aplicación debe dar la opción de generar un listado con el nombre de los centros que aparecen marcados en el mapa.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide generar un listado de los centros localizados en el mapa, resultado de la búsqueda que ha realizado.
Postcondiciones	No aplica.

Caso de uso Visualizar mi ubicación

Caso de uso	Visualizar mi ubicación
Actor	Usuario final de la aplicación
Descripción	<p>Una vez el usuario ha llegado al punto de visualizar en pantalla el mapa del municipio seleccionado, con los centros que cumplen los requisitos que ha marcado previamente; tiene la posibilidad de visualizar a la vez su localización.</p> <p>Esto le va a permitir al usuario, comprobar fácilmente cuales son los centros más cercanos a su ubicación.</p>
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. ✓ El usuario ha tenido la opción de seleccionar las enseñanzas, servicios y programas que le interesan, aunque no es obligatorio en ninguno de los casos. ✓ El usuario ha decidido visualizar los centros resultantes de la búsqueda en el mapa.

	<ul style="list-style-type: none"> ✓ La aplicación debe dar la opción de localizar en el mapa la ubicación del usuario.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide situar y centrar su ubicación en el mapa para hacerse una idea de cuales son los centros que tiene más cercanos.
Postcondiciones	No aplica.

Caso de uso Consultar información de un centro

Caso de uso	Consultar información de un centro
Actor	Usuario final de la aplicación
Descripción	<p>Partimos de que el usuario se encuentra visualizando en pantalla el mapa del municipio seleccionado, con los centros que cumplen los requisitos que él ha establecido.</p> <p>Una de las posibilidades que tiene el usuario es pulsar sobre uno de los centros que tiene en el mapa, y se le abrirá un diálogo con toda la información disponible del centro, y la posibilidad de realizar dos acciones:</p> <ul style="list-style-type: none"> ➤ Establecer llamada telefónica con el centro que se ha seleccionado. ➤ Mostrar la ruta desde la ubicación del usuario a dicho centro.
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. ✓ El usuario ha tenido la opción de seleccionar las enseñanzas, servicios y programas que le interesan, aunque no es obligatorio en ninguno de los casos. ✓ El usuario ha decidido visualizar los centros resultantes de la búsqueda en el mapa. ✓ La aplicación debe dar la opción de seleccionar cualquiera de los centros que aparecen localizados en el mapa, para consultar toda la información disponible del mismo.

Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide consultar la información disponible de alguno de los centros localizados en el mapa.
Postcondiciones	No aplica.

Caso de uso Llamar a un centro

Caso de uso	Llamar a un centro
Actor	Usuario final de la aplicación
Descripción	Una vez que el usuario se encuentra consultando la información de uno de los centros resultantes de su búsqueda; una de las opciones que se le ofrecen es la de llamar al centro (siempre y cuando, el número de teléfono sea uno de los datos disponibles).
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. ✓ El usuario ha tenido la opción de seleccionar las enseñanzas, servicios y programas que le interesan, aunque no es obligatorio en ninguno de los casos. ✓ El usuario ha decidido visualizar los centros resultantes de la búsqueda en el mapa. ✓ El usuario ha pulsado sobre uno de los que se le muestran en el mapa para consultar su información. ✓ La aplicación debe dar la opción de realizar una llamada al centro seleccionado, en caso de disponer de su número de teléfono.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide realizar una llamada al centro que ha seleccionado, y se dispone de su número de teléfono.
Postcondiciones	No aplica.

Caso de uso Visualizar ruta de centro a mi ubicación

Caso de uso	Visualizar ruta de centro a mi ubicación
Actores	Usuario final de la aplicación
Descripción	Una vez que el usuario se encuentra consultando la información de uno de los centros resultantes de su búsqueda; otra de las opciones que se le ofrecen es la de visualizar la ruta para llegar desde su ubicación al centro seleccionado.
Precondiciones	<ul style="list-style-type: none"> ✓ La aplicación debe funcionar correctamente. ✓ El usuario ha seleccionado previamente una provincia y un municipio donde desea buscar los centros. ✓ El usuario ha tenido la opción de seleccionar las enseñanzas, servicios y programas que le interesan, aunque no es obligatorio en ninguno de los casos. ✓ El usuario ha decidido visualizar los centros resultantes de la búsqueda en el mapa. ✓ El usuario ha pulsado sobre uno de los que se le muestran en el mapa para consultar su información. ✓ La aplicación debe dar la opción de mostrar al usuario la ruta desde su ubicación al centro seleccionado.
Escenario de éxito principal	<ul style="list-style-type: none"> ✓ La aplicación está disponible para su uso. ✓ El usuario decide utilizar la aplicación. ✓ El usuario decide visualizar la ruta para llegar desde su ubicación al centro seleccionado.
Postcondiciones	No aplica.

3. DISEÑO DEL PROYECTO

El diseño es el primer paso en la fase de desarrollo de cualquier producto o sistema de ingeniería. El objetivo de esta fase es realizar el diseño del sistema, para lo cual, se usarán técnicas de modelado que nos ayudarán a comprender mejor el sistema que estamos desarrollando. Se pretende definir completamente la solución que se va a implementar.

3.1. Modelo de datos

Para empezar, vamos concretar el dominio de la información que se tratará en el sistema. Para ello partiremos de los siguientes supuestos:

- S1 – Se mantendrá la siguiente información de cada centro educativo: código del centro, denominación del centro, nombre, dependencia, domicilio, localidad, municipio, provincia, código postal, teléfono, enseñanzas que imparte, servicios que ofrece, programas educativos con los que cuenta el centro, coordenadas latitud/longitud.
- S2 – Se considera que no existen dos códigos de centro iguales.
- S3 – Un centro no podrá existir en nuestro sistema si no tiene un código de centro asociado.
- S4 – Las enseñanzas que se pueden impartir en los centros educativos son las que a continuación se detallan (junto con su representación abreviada correspondiente):
 - INF: infantil
 - PRI: primaria
 - SEC: secundaria
 - BAC: bachillerato
 - PCPI: programa de cualificación profesional inicial
 - EE: educación especial
 - CFGM: ciclo formativo de grado medio
 - CFGS: ciclo formativo de grado superior
 - ADUL: adultos
 - MUS: música
 - DANZ: danza
 - IDI: idiomas
 - ADR: arte dramático
 - APD: artes plásticas y diseño

- S5 – Los servicios que pueden ofertar los centros educativos son los que se muestran a continuación (junto con su representación abreviada correspondiente):
 - AM: aula matinal
 - COM: comedor
 - AEX: actividades extraescolares
- S6 – Los programas educativos que pueden ofertar los centros son los siguientes (junto con su representación abreviada correspondiente):
 - CBEI: centro bilingüe español-inglés
 - CBEF: centro bilingüe español-francés
 - CBEA: centro bilingüe español-alemán
- S7 – Los municipios que consideremos en el sistema tendrán al menos un centro educativo.
- S8 – Se deben obtener las coordenadas latitud/longitud de los centros a partir de su dirección, para poder geolocalizarlos en el mapa, ya que este dato no está disponible en el [directorio de centros docentes de Andalucía](#).

Análisis de los tipos de entidad

Tipo de entidad *provincia*:

Representa a las provincias con las que podremos trabajar en nuestro sistema, en este caso, las pertenecientes a la Comunidad Autónoma de Andalucía.

Se consideran los siguientes atributos:

- Nombre: nombre la provincia.

Tipo de entidad *municipio*:

Representa a los municipios sobre las que podremos visualizar los centros del sistema, en este caso, serán los pertenecientes a las provincias de Andalucía.

Se consideran los siguientes atributos:

- Nombre: nombre del municipio.

Tipo de entidad *centro*:

Representa a un centro educativo del mundo real, dado de alta en la Red de Centros de la Consejería de Educación de la Junta de Andalucía.

Se consideran los siguientes atributos:

- Código: código identificativo del centro. Será de tipo numérico y debe ser único (supuesto 2 y 3).
- Denominación: indica el tipo de centro del que se trata, por ejemplo: de educación

infantil, instituto de educación secundaria, etc (supuesto S1).

- Nombre: representa el nombre de dicho centro (supuesto S1).
- Dependencia: indica si el centro es de tipo público o privado (supuesto S1).
- Domicilio: representa la dirección donde se ubica el centro (supuesto S1).
- Localidad: localidad en la que se sitúa el centro (supuesto S1).
- Municipio: municipio en el que se sitúa el centro (supuesto S1).
- Provincia: provincia en la que se sitúa el centro (supuesto S1).
- Cod_postal: indica el código postal del centro (supuesto S1).
- Teléfono: teléfono de dicho centro (supuesto S1).
- Enseñanzas: lista de las enseñanzas que se imparten en el centro (supuesto S4).
- Servicios: lista de los servicios que ofrece el centro (supuesto S5).
- Programas: lista de los programas educativos con los que cuenta el centro (supuesto S6).
- Latitud y longitud: coordenadas latitud/longitud del centro (supuesto S8).

Análisis de los tipos de interrelación

Tipo de interrelación P-M:

Relaciona el tipo de entidad *provincia* con el tipo de entidad *municipio*, de forma que una provincia puede tener uno o varios municipios, y un municipio sólo puede pertenecer a una provincia.

Tipo de interrelación M-C:

Relaciona el tipo de entidad *municipio* con el tipo de entidad *centro*, de forma que un municipio puede tener uno o varios centros, mientras que un centro puede situarse en un único municipio.

Esquema Entidad-Interrelación

Partiendo de la descripción del dominio de la información comentada anteriormente, se ha realizado un esquema conceptual usando la metodología propuesta por Peter Chen, para definir las estructuras de datos que necesitaremos en el sistema.

Como anteriormente se ha mencionado, en este esquema participan tres entidades: *provincia*, *municipio* y *centro*. A pesar de que en la entidad *centro* ya se recoge la información acerca de la provincia y municipio donde se localiza el centro, se ha decidido redundar esta información a través de las correspondientes entidades, ya que se va a trabajar con un volumen considerable de centros, y para comenzar con la aplicación, será necesario mostrar un listado de provincias y municipios. La obtención de estos listados será mucho más rápida a través de estas entidades, que lanzando una consulta a la entidad *centro*.

El diagrama Entidad-Interrelación que se ha obtenido es el siguiente:

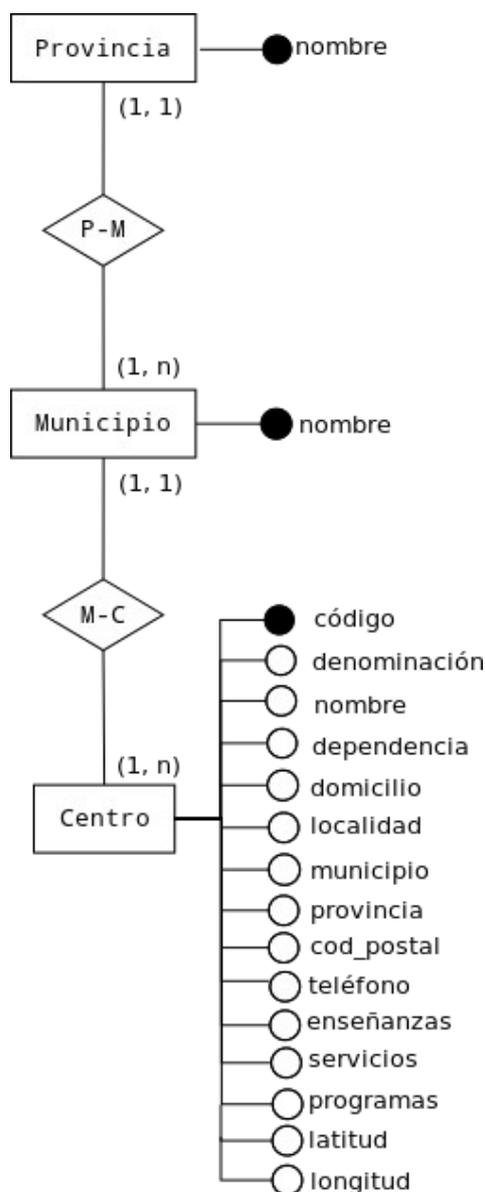


Ilustración 2: Diagrama Entidad-Interrelación

Como se ha comentado, los datos de latitud y longitud son necesarios para poder geolocalizar a un centro en el mapa. Estos datos no son proporcionados inicialmente en el [directorio de centros docentes de Andalucía](#). Se ha realizado una petición a la Junta de Andalucía, para que incluya estos datos junto con los ya publicados, pero por el momento, se han tenido que obtener a través de un script Python que se adjunta en el “**Anexo I. Script Python para generar la codificación geográfica de los centros**”.

Se ha decidido incorporar esta información a la base de datos por dos motivos:

- *Como medida para mejorar el rendimiento*: Debido al volumen de centros con los que vamos a trabajar, y a que se pueden mostrar muchos de ellos como resultado

de una búsqueda en una de las capitales, se ha estimado conveniente hacer una carga previa de la latitud y longitud de cada centro en la base de datos. De este modo, en vez de ir obteniendo estos datos a partir de la dirección del centro cada vez que se vaya a pintar un centro en el mapa (en tiempo de ejecución), simplemente se lanzarán consultas contra la base de datos.

- *Solicitudes de codificación geográfica limitadas por día*: Otro motivo no menos importante, es que hay un límite en el número de solicitudes de codificación geográfica por día (15.000). Por tanto, de esta manera evitaremos que un usuario se encuentre en la situación de no poder utilizar la aplicación porque se han agotado las solicitudes de codificación geográfica de ese día.

También hay que mencionar que para gestionar esta base de datos se utilizará un sistema de gestión de bases de datos relacional SQLite 3.0.

A diferencia de los sistema de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host.

3.2. Arquitectura general de la aplicación

El objetivo de este apartado es realizar el diseño del sistema, para lo cual, se usarán técnicas de modelado que nos ayudarán a comprender mejor el sistema que estamos desarrollando. Al igual que para analizar el comportamiento del sistema, para aplicar estas técnicas se usará UML (Lenguaje Unificado de Modelado)³, ya que nos aporta la sintaxis y semántica necesaria para alcanzar este fin.

Para ello se va a realizar el modelo de paquetes, donde se mostrará la agrupación de los elementos del sistema, pudiendo manejarlos como grupo y facilitando la comprensión del mismo. Esto se representará mediante diagramas de paquetes.

3 Consulta bibliográfica (1).

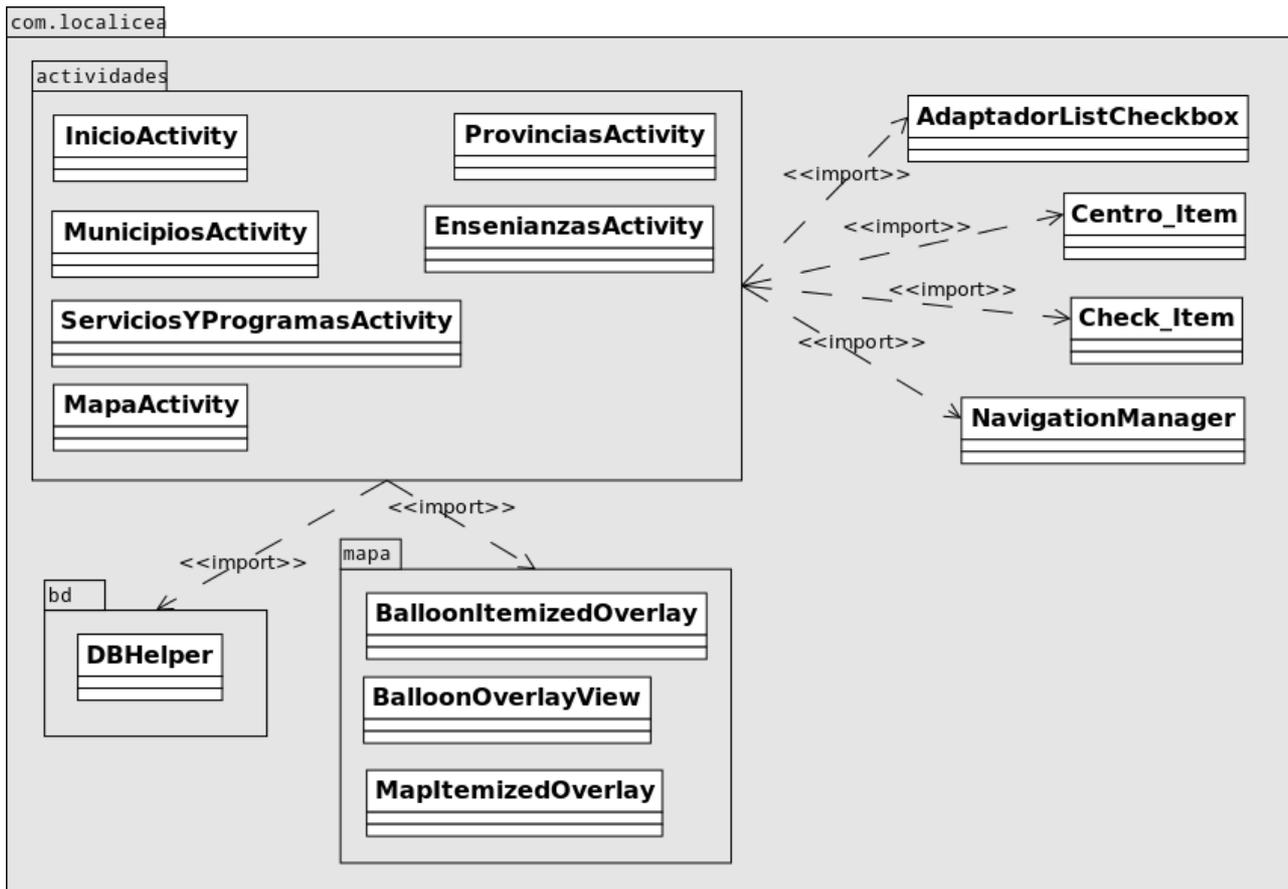


Ilustración 3: Diagrama de paquetes

Paquete raíz com.localicea

En este paquete raíz de la aplicación, es donde vamos a englobar todos los componentes del sistema. En él encontraremos los paquetes de la aplicación que detallaremos a continuación, además de las siguientes clases genéricas:

- **NavigationManager**: Clase que se encarga de gestionar la navegación entre las Activities; es decir, en ella encontraremos los métodos que utilizaremos para pasar de una pantalla a otra de la aplicación, así como de los métodos necesarios para obtener los elementos seleccionados en ellas (provincia, municipios, enseñanzas, etc.).
- **AdaptadorListCheckbox**: Clase que usaremos para crear una lista de checkbox a nuestra medida. Cada elemento de esta lista será del tipo *Check_Item*.
- **Check_Item**: Clase en la que se establece como va a ser cada elemento de la lista de checkbox. En este caso, en cada elemento tendremos un texto y un estado según lo que marque el usuario.
- **Centro_Item**: Clase que establece las características del objeto Centro. En ella se fijan los atributos que vamos a manejar con respecto a cada centro, así como los correspondientes getters y setters.

Paquete actividades

Las Activities son el motor fundamental de cualquier aplicación Android.

Una Activity es una pieza de código ejecutable que siendo iniciada por el usuario o por el sistema operativo, se ejecuta mientras es necesitada.

La característica principal de un Activity es que puede interactuar con el usuario, aunque también puede pedir datos o servicios de otras activities por medio de queries o Intents.

En la mayoría de los casos cada Activity corresponde a una vista (layout). Esto quiere decir que por cada pantalla debe crearse un Activity, aunque no es requisito obligatorio vincular la parte visual a un Activity, en nuestro caso se puede plantear así. Concretamente los Activity que necesitaremos son los siguientes:

- **InicioActivity**: Actividad principal con la que se inicia la aplicación y donde se muestra una descripción de la aplicación.
- **ProvinciasActivity**: Actividad que mostrará un listado de las provincias de Andalucía, y que recogerá la acción del usuario al seleccionar una de las provincias para mostrar en la siguiente actividad sus municipios correspondientes.
- **MunicipiosActivity**: Actividad que mostrará un listado de los municipios de la provincia seleccionada previamente. Al igual que antes, recogerá la acción del usuario al pulsar uno de los municipio para almacenar el municipio seleccionado.
- **EnsenanzasActivity**: Esta actividad mostrará un listado de checkbox con las distintas enseñanzas que se pueden encontrar en los centros educativos, para que el usuario seleccione si le interesa alguna/s en concreto. Esta actividad recogerá la selección realizada por el usuario.
- **ServiciosYProgramasActivity**: Actividad que mostrará dos listados de checkbox con los servicios y programas respectivamente, que se ofertan en los centros educativos. La actividad recogerá la selección que realice el usuario para filtrar la búsqueda de centros en los que está interesado el usuario.
- **MapaActivity**: Esta actividad extiende de MapActivity, la cual encapsula toda la funcionalidad para mostrar un MapView. Todas las actividades anteriores extendían de Activity, pero en este caso necesitamos del MapActivity para poder visualizar el mapa. Esta actividad es la encargada de pintar en el mapa los centros resultantes de la búsqueda de centros realizada. También será la que ofrezca al usuario las demás funcionalidades planteadas, como visualizar la ubicación del usuario, y mostrar un listado de los centros localizados en el mapa.

Paquete bd

En este paquete encontramos la clase **DBHelper**, la cual extiende de **SQLiteOpenHelper**, clase que nos va a permitir crear la base de datos.

A grandes rasgos, la clase **DBHelper** va a ser la que nos va a permitir crear la base de datos, y donde vamos a configurar las consultas que vamos a realizar a dicha base de datos.

Paquete mapa

En este paquete encontraremos las clases que utilizaremos para mostrar los “*balloon*” (o globos) donde mostraremos la información de cada centro, así como las acciones que podemos realizar a través de ellos.

Concretamente, tendremos las siguientes clases:

- ***MapItemizedOverlay***: Clase que irá recogiendo una capa por centro resultante de la búsqueda, en la cual insertaremos la información correspondiente de ese centro, y la localización del centro para situar el marcador. Esta clase extiende de *BalloonItemizedOverlay*.
- ***BalloonItemizedOverlay***: De esta clase hereda la clase *MapItemizedOverlay*, por tanto nos ayudará a manejar la colección de capas que tendremos, como resultado de generar una capa por centro a representar en el mapa. En esta clase se definen los métodos para poner un marcador en primer plano al pulsar sobre él y mostrar un “*balloon*” con su información. También se encargará de detectar si hay otro *balloon* abierto, para cerrarlo si se decide consultar la información de otro centro.
- ***BalloonOverlayView***: Con esta clase se definen las características y funcionalidad de cada “*balloon*” en concreto. En ella se pintará el “*balloon*” con la imagen y demás características establecidas, así como la funcionalidad para llamar al centro que representa en concreto, y para mostrar la ruta desde ese centro a la ubicación del usuario.

3.3. Diseño de la interfaz

En este apartado diseñaremos las pantallas que se van a presentar al usuario y la información que se le muestra o solicita; en definitiva, se diseñará la navegación que se pretende obtener en la aplicación a través de bocetos.

A continuación se muestra pantalla a pantalla la interfaz que se va a construir:

Pantalla inicial

Esta será la pantalla de inicio de la aplicación. En la parte superior de la pantalla tendremos una barra de herramientas, donde encontraremos el logo de la aplicación y el nombre. Con esta pantalla se presenta al usuario una breve descripción de la aplicación a través de un texto y una imagen de presentación. También se le ofrece un botón para que comience a utilizar la aplicación si lo desea.

Con ella, simplemente se pretende dar al usuario una idea de lo que se va a encontrar, antes de pasar a usar la aplicación.



Ilustración 4: Diseño pantalla “Inicio”

Selección de la provincia

En esta pantalla es donde realmente se comienza a utilizar la aplicación. Se comienza mostrando un listado (por orden alfabético) de las provincias de la Comunidad Autónoma de Andalucía. Al seleccionar cualquiera de ellas, pasaremos a la siguiente pantalla.



Ilustración 5: Diseño pantalla “Provincia”

Selección del municipio

En esta tercera pantalla, se nos mostrará un listado por orden alfabético con todos los municipios de la provincia escogida anteriormente. Tendremos una barra de desplazamiento en el lateral para facilitar la búsqueda del municipio deseado, ya que este listado será de gran dimensión y ocupe más del tamaño de la pantalla.

Si queremos volver a la pantalla anterior, tenemos un icono en la esquina inferior izquierda para retroceder a la pantalla previa, y volver a seleccionar una provincia distinta si se desea.

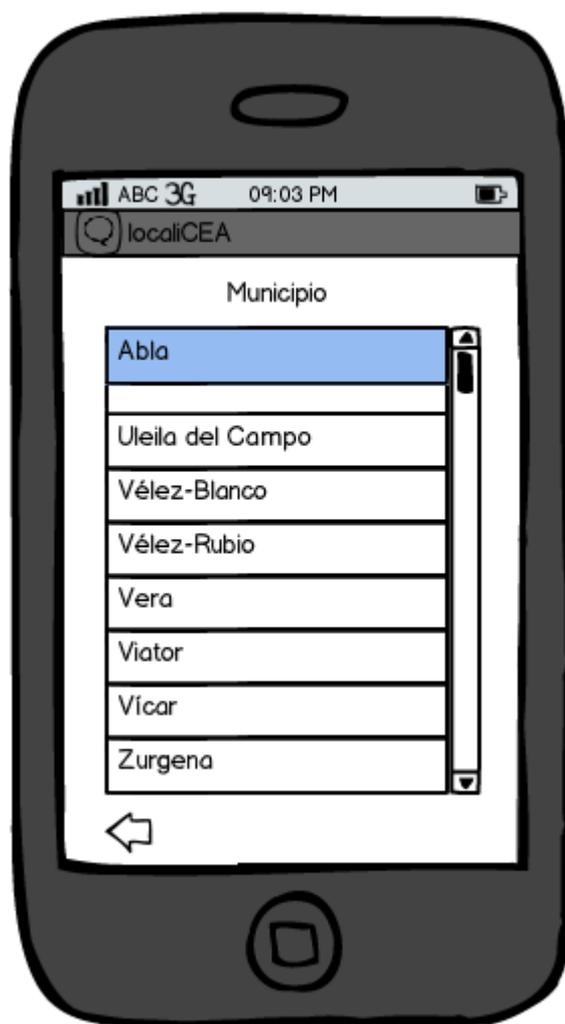


Ilustración 6: Diseño pantalla “Municipio”

Selección de enseñanza

Llegados a este punto, sabemos la zona geográfica que al usuario le interesa consultar. Ahora, vamos a refinar un poco la búsqueda de centros según los intereses del usuario. En esta pantalla se le solicita la/s enseñanza/s que desea que tenga el centro que busca. Se ofrece una lista de checkbox, ya que es posible seleccionar más de una enseñanza. En caso de no seleccionar ninguna enseñanza, no se filtrará la búsqueda de centros por este criterio.

Vemos en las esquinas inferiores de la pantalla dos iconos de flechas. El de la izquierda será para pasar a la pantalla anterior a seleccionar otro municipio de la provincia que habíamos seleccionado previamente, y el de la derecha para pasar a la pantalla siguiente.

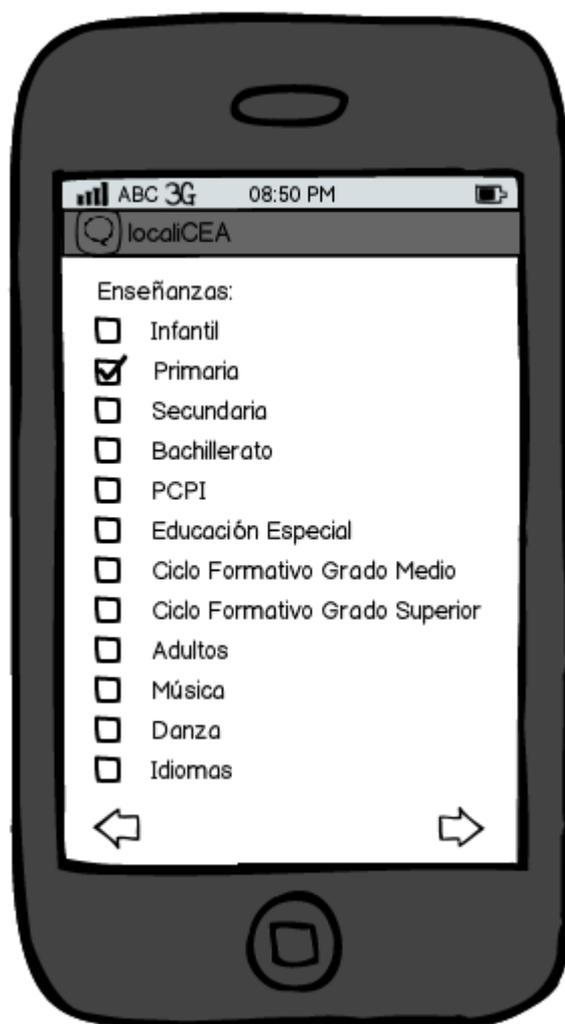


Ilustración 7: Diseño pantalla “Enseñanzas”

Selección de servicios y programas

En esta pantalla, el usuario podrá establecer los servicios y programas educativos que desea que tenga el centro que busca. Nuevamente, podrá seleccionar más de un elemento de la lista, o ninguno. Al igual que en la pantalla anterior, podremos navegar a través de los iconos de flechas a la pantalla anterior (para modificar la selección de enseñanzas) o a la siguiente. En caso de volver a la pantalla anterior, se mantendrá la selección previa del usuario para que éste pueda modificarla.

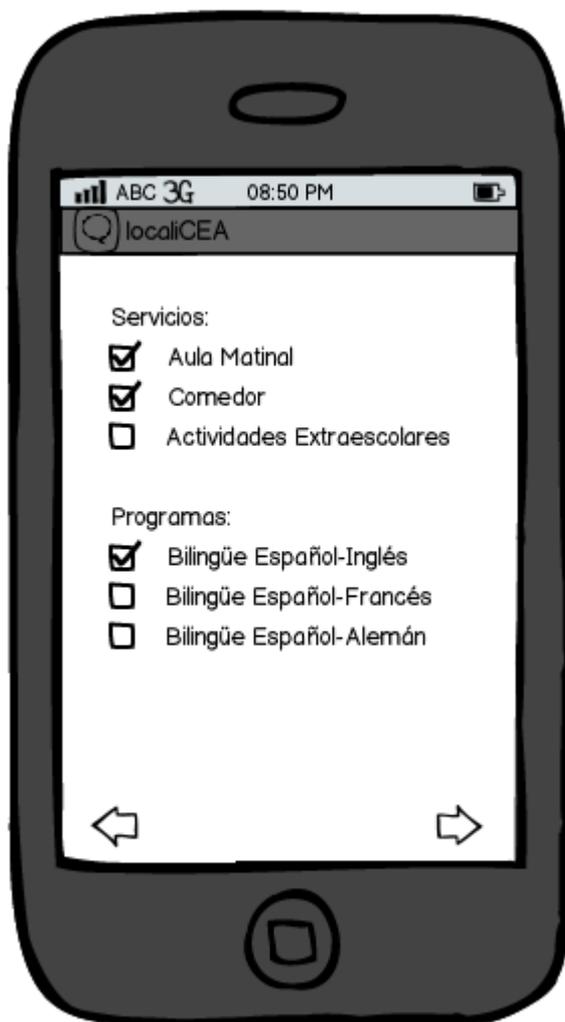


Ilustración 8: Diseño pantalla “Servicios y Programas”

En caso de que el usuario elija pasar a la pantalla siguiente, se le ofrecerá la posibilidad de hacer una búsqueda estricta a partir de los criterios especificados, o algo más flexible. Es decir, se le mostrará un diálogo de selección para que elija si quiere que se cumplan *todos* o *alguno* de los requisitos especificados.

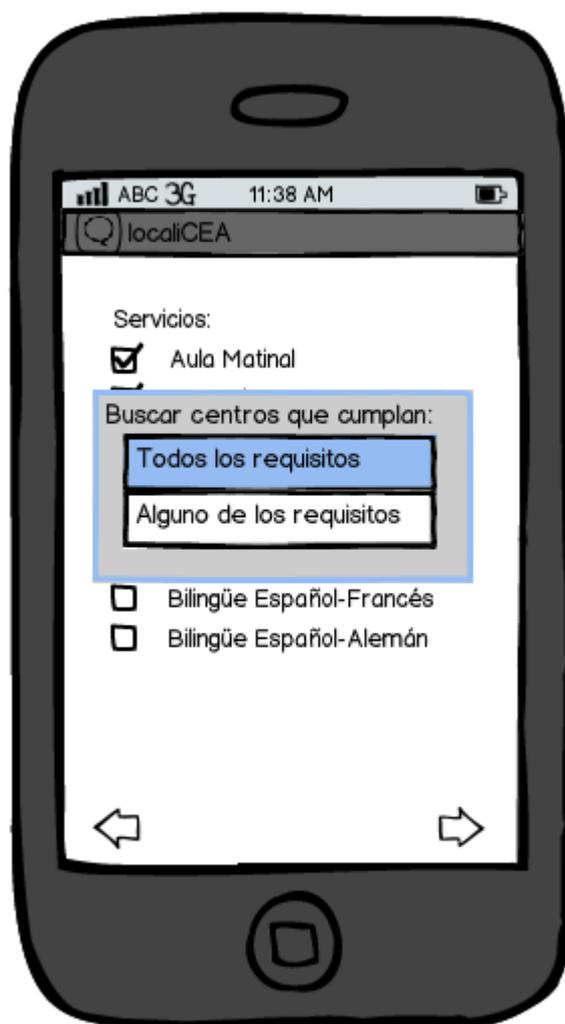


Ilustración 9: Diseño diálogo “Búsqueda con todos/algunos de los requisitos”

Visualizar centros en el mapa

En este punto se presentan en el mapa la localización de los centros que responden a los criterios establecidos, tanto geográficamente, como en enseñanzas, servicios y programas que ofrecen. En el mapa aparecerá un icono marcando la situación de cada centro.

En la parte inferior de la pantalla, tendremos a la izquierda el icono de flecha para volver a la pantalla anterior para modificar los requisitos en cuanto a servicios y programas educativos (el usuario debe encontrar siempre la selección que había realizado previamente).

En la barra superior, en su esquina derecha vamos a encontrar dos iconos:

- Mi ubicación: el cual nos mostrará nuestra ubicación en pantalla. Es el que encontramos en segunda lugar empezando por la derecha.

- Lista de centros resultado de la búsqueda: nos mostrará el listado de centros que tenemos pintados en el mapa. Es el que tenemos en primer lugar empezando por la derecha.



Ilustración 10: Diseño pantalla “Centros en el mapa”

Información de un centro seleccionado

Desde la pantalla anterior, donde se nos muestra en el mapa todos los centros que cumplen los requisitos que el usuario busca; el usuario puede también pulsar sobre el icono del centro que le interese por su situación, para conocer toda la información disponible del mismo. Al pulsar sobre el marcador de un centro se debe centrar automáticamente en pantalla para poder ver su información correctamente.

En el siguiente boceto vemos como se mostrará dicha información. Tanto las enseñanzas, como los servicios y los programas, se mostrarán abreviados por cuestión de espacio. A través del panel que se muestra con la información del centro,

dispondremos de dos botones para realizar las siguientes acciones relacionadas con el centro:

- Llamar al centro.
- Mostrar la ruta desde nuestra ubicación hasta el centro que estamos visualizando.

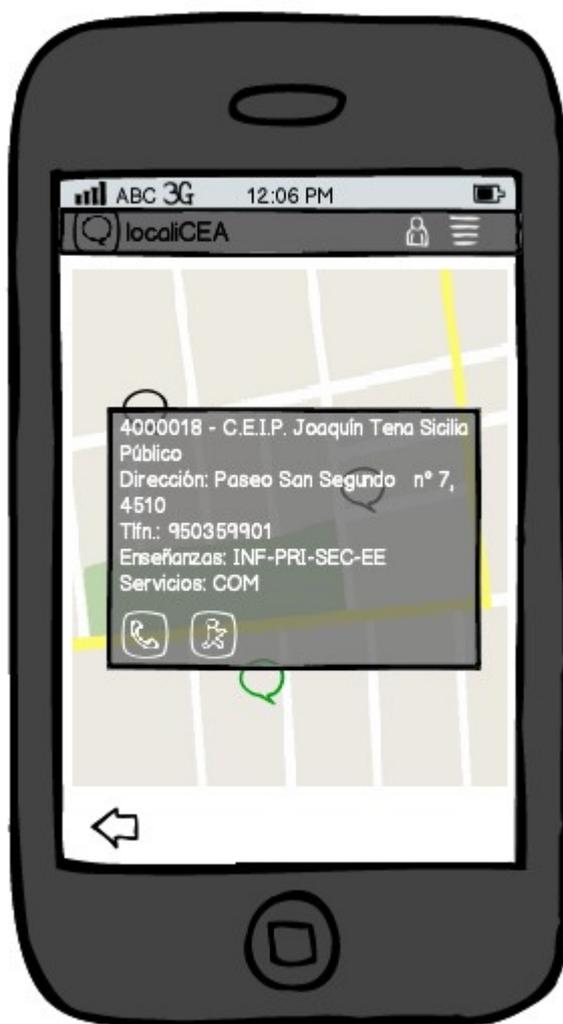


Ilustración 11: Diseño del panel “Información de un centro”

Mi ubicación

Tras pulsar en el botón “Mi ubicación” desde cualquiera de las dos pantallas anteriores, se situará en el mapa la localización actual del usuario y se centrará en la pantalla. El icono utilizado será distinto del usado para localizar a los centros.

Con esta pantalla se consigue ver de un vistazo los centros que el usuario tiene más cercanos, como otro dato que le puede ayudar a la hora de decidir el centro que más le interesa, una vez conocidos todos los centros que cumplen los requisitos deseados.



Ilustración 12: Diseño pantalla "Mi ubicación"

Listado de centros resultado

Tras pulsar en el botón "Lista de centros resultado" desde cualquiera de las pantallas anteriores donde se muestra el mapa con los centros localizados, se mostrará un listado indicando el número de centros mostrados en el mapa, y el nombre de los mismos. El usuario podrá seleccionar un centro de la lista para centrarlo en pantalla y así poder marcarlo y ver su información.

Esto será de utilidad en caso de tener varios centros como resultado de la búsqueda, y querer consultar alguno de ellos en concreto.



Ilustración 13: Diseño pantalla "Lista centros resultado"

4. PLANIFICACIÓN

Para alcanzar los objetivos marcados para este proyecto, vamos a definir las fases y entregables que se van a ir desarrollando, así como la planificación temporal para cada una de ellas.

4.1. Entregables

PEC1 – Plan de trabajo

En este documento se definen las características del proyecto a desarrollar. Se establece el título, una descripción general del proyecto y los objetivos que se pretenden alcanzar.

Teniendo en cuenta todo lo anterior, se definirá también una planificación temporal con las metas a conseguir para cada uno de los hitos o puntos de revisión establecidos.

PEC2 – Análisis y diseño de la aplicación. Prototipo

En este documento se realizará un análisis de los requisitos del usuario, así como el diseño de la aplicación.

Además realizaremos un prototipo base de la aplicación. Para este prototipo nos marcaremos las siguientes metas:

- Mostrar un mapa de Andalucía y poder hacer zoom sobre él.
- Pintar un centro en el mapa.
- Mostrar información del centro a través de un diálogo.

PEC3 – Desarrollo de la aplicación

En esta entrega se pretende tener prácticamente terminado el desarrollo de la aplicación. Se deben de cumplir para entonces los siguientes objetivos.

- Tener una base de datos con todos los centros educativos de Andalucía, y con toda la información necesaria para esta aplicación.
- Disponer de una pantalla inicial donde seleccionar la provincia y municipio a consultar.
- Establecer filtros de búsqueda, para mostrar sólo los centros que cumplan los requisitos deseados: que dispongan de una determinada enseñanza, servicio o proyecto educativo.
- Mostrar los centros educativos que cumplan las características elegidas, así como la información del centro al pinchar sobre uno de ellos.

Entrega Final

Para este momento se entregará la memoria final del proyecto, así como la aplicación

con los últimos objetivos marcados:

- Disponer en el diálogo mostrado al pinchar sobre un centro, además de su información, la opción de llamar al centro y de mostrar la ruta desde ese centro hasta la ubicación del usuario.
- Pintar la ubicación del usuario y centrar en el mapa dicha ubicación, mostrando así los centros más cercanos al usuario.
- Mostrar un listado de los centros resultantes de la búsqueda para que el usuario pueda dirigirse a uno en concreto.
- Refinamiento o mejoras que se consideren oportunas.

4.2. Planificación temporal

El horario de trabajo durante el desarrollo del proyecto será el siguiente: De lunes a sábado de 17:30 a 21:30.

Además, los entregables anteriormente mencionados se considerarán hitos a la hora de realizar la planificación de las tareas:

- PEC 1 – Presentación del plan de trabajo. Día 1 de octubre de 2012.
- PEC 2 – Análisis y diseño de la aplicación. Prototipo. Día 29 de octubre de 2012.
- PEC 3 – Desarrollo de la aplicación . Día 10 de diciembre de 2012.
- Entrega final. Día 7 de enero de 2013.

Teniendo en cuenta todo lo anterior, se ha realizado el siguiente diagrama de Gantt donde se plasma la planificación de las tareas y actividades que se llevarán a cabo (aunque durante el desarrollo del proyecto puede sufrir algún cambio):

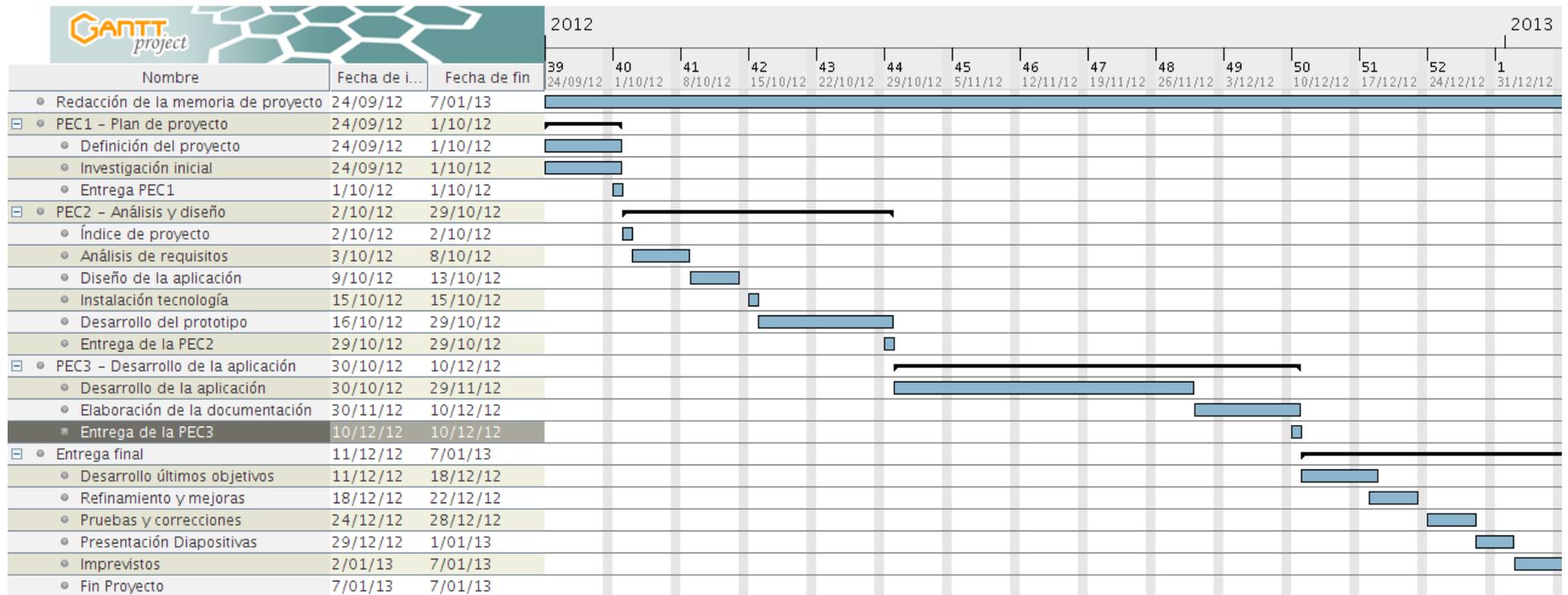


Ilustración 14: Diagrama de Gantt de la planificación realizada

5. PLAN DE PRUEBAS

A continuación se describen las pruebas funcionales realizadas sobre el proyecto:

Nombre	Descripción
Pantalla principal	Comprobar la maquetación y el correcto funcionamiento del botón para comenzar a usar la aplicación.
Listado de provincias	Comprobar que el listado de provincias se muestra correctamente en contenido y maquetación.
Listado de municipios	Comprobar que el listado de municipios se muestra correctamente en cuanto a maquetación, y que los municipios mostrados corresponden a la provincia seleccionada.
Listado de enseñanzas	Comprobar que el listado de posibles enseñanzas de un centro se muestra correctamente en contenido y maquetación.
Listado de servicios y programas	Comprobar que el listado de servicios y programas que puede ofrecer un centro se muestra correctamente en contenido y maquetación.
Navegación entre pantallas	Comprobar que se pasa correctamente a la siguiente pantalla (desde cualquiera de ellas), y que se puede retroceder también a la pantalla previa, visualizando la información marcada con anterioridad si fuera el caso.
Localización de centros resultantes de la búsqueda	Comprobar que los centros mostrados en el mapa corresponden a los criterios de búsqueda establecidos y que su localización es correcta. Comprobar también que ambos tipos de búsqueda (que se cumplan “todos los requisitos” o que se cumplan “algunos de los requisitos”) funcionan correctamente.
Mostrar “mi ubicación”	Comprobar que se muestra la ubicación del usuario correctamente, y en caso de no ser posible, comprobar que se muestra un mensaje informativo.
Mostrar información de un centro seleccionado	Comprobar que al seleccionar cualquiera de los centros mostrados en el mapa, se muestra la información correspondiente al mismo. Comprobar la maquetación del “balloon” donde se muestra la información.
Llamar a un centro	Comprobar que si se tiene disponible el teléfono del centro seleccionado, se establece una llamada con el mismo correctamente.
Mostrar ruta de “mi ubicación” a un centro	Comprobar que se muestra correctamente la ruta desde la ubicación del usuario a un centro seleccionado. En caso de no estar disponible la ubicación del usuario, comprobar que se muestra un mensaje informativo.
Mostrar un listado	Comprobar que se muestra el listado de centros resultantes

con los centros nombres de los centros localizados en el mapa	de la búsqueda correctamente, tanto en formato, como en contenido comprobando que coinciden con los que tenemos en el mapa. Comprobar que se indica correctamente el número de centros mostrados.
Visualización del mapa	Comprobar que el zoom del mapa se ajusta automáticamente para visualizar en pantalla todos los centros resultantes de la búsqueda. Comprobar también que se puede modificar el zoom de la pantalla manualmente.

Se ha llevado a cabo el anterior plan de pruebas y se ha superado con éxito.

6. LICENCIA Y CRÉDITOS DE LOS RECURSOS GRÁFICOS

La licencia que se ha decidido utilizar para este proyecto es la licencia GNU GPLv3.

La Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License (GNU GPL), es una licencia creada por la Free Software Foundation en 1989 (la primera versión, escrita por Richard Stallman), y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

Esta licencia garantiza una serie de libertades al usuario, como el libre uso y adaptación del código fuente del programa, obligando a que las modificaciones de este código estén disponibles para el resto de usuarios si se redistribuye el programa.

Esta licencia se usa para la mayoría de los programas de GNU y para más de la mitad de los paquetes de software libre.

La nueva versión, GNU GPLv3, fue publicada en 2007 y es accesible a través del [Portal de GNU](#)⁴. La nueva versión contempla los siguientes aspectos:

- Las diversas formas en que alguna persona podría quitar libertades a los usuarios.
- Prohibir el uso de software libre en sistemas que utilizan la llamada "gestión digital de derechos" o DRM, sistema criticado por la comunidad del software libre. Con esto se quiere evitar que aún incorporando software bajo copyleft, se haga uso de un hardware preparado para imposibilitar a los usuarios ejecutar versiones modificadas del software en ese hardware.
- Resolver ambigüedades y aumentar la compatibilidad de GPLv3 con otras licencias.
- Facilitar su adaptación a otros países.
- Incluir cláusulas que defiendan a la comunidad de software libre del uso indebido de las patentes de software.
- Mostrar usuarios registrados.

En cuanto a los iconos utilizados en la aplicación, se han hecho uso de recursos con licencia libre. A continuación se muestran las distintas imágenes utilizadas con su licencia correspondiente y una referencia a su autor original.

4 <http://www.gnu.org/licenses/gpl-3.0.html>

Icono original:	
Icono usado:	
Autor:	Daniel Hanly
Licencia:	Creative Commons CC0 - No Rights Reserved
Referencia:	http://thenounproject.com/dhanly/#favorite-icons
Uso:	Icono "Marcador". Icono utilizado como logotipo de la aplicación y para marcar la localización de los centros en el mapa.

Icono original:	
Icono usado:	

Icon set:	Ecommerce and Business Icon Set
Autor:	DesignContest
Licencia:	Creative Commons Attribution 3.0
Referencia:	http://www.designcontest.com/
Uso:	Icono "Inicio". Icono utilizado en la pantalla principal de la aplicación para mostrar una idea del uso de la aplicación.

Icono original:	
Icono usado:	
Autor:	Oxygen Team
Licencia:	Creative Commons (Attribution-NonCommercial-NoDerivs 2.5 Generic)
Referencia:	http://www.oxygen-icons.org/
Uso:	Icono "Cerrar". Icono utilizado para cerrar el diálogo que muestra la información de un centro seleccionado.

Icono original:	
Icono usado:	
Icon set:	Glyphish icons

Autor:	Joseph Wain
Licencia:	Creative Commons Attribution 3.0
Referencia:	http://glyphish.com
Uso:	Icono “Llamada”. Icono utilizado como botón para establecer una llamada con el centro seleccionado.

Icono original:	
Icono usado:	
Icon set:	Glyphish icons
Autor:	Joseph Wain
Licencia:	Creative Commons Attribution 3.0
Referencia:	http://glyphish.com
Uso:	Icono “Ruta”. Icono utilizado para mostrar la ruta desde la ubicación del usuario hasta el centro seleccionado.

Icono original:	
Icono usado:	 
Icon set:	Glyphish icons
Autor:	Joseph Wain
Licencia:	Creative Commons Attribution 3.0

Referencia:	http://glyphish.com
Uso:	<p>Icono “Mi ubicación”.</p> <p>Icono utilizado como botón para que el usuario centre su ubicación en el mapa tras pulsarlo.</p> <p>Se utiliza en dos colores, el modo original en gris, y al pulsarlo se pone en blanco para que el usuario sepa que lo ha pulsado.</p>

Icono original:	
Icono usado:	 
Icon set:	Glyphish icons
Autor:	Joseph Wain
Licencia:	Creative Commons Attribution 3.0
Referencia:	http://glyphish.com
Uso:	<p>Icono “Listado centros”.</p> <p>Icono utilizado como botón para que el usuario pueda visualizar un listado de los nombres de los centros localizados en el mapa.</p>

7. CONCLUSIONES Y FUTURAS MEJORAS

El trabajo desarrollado en este proyecto ha cumplido los objetivos definidos en el punto 1.3 *Objetivos generales*:

- Situar y localizar en el mapa los centros educativos del municipio establecido por el usuario.
- Poder visualizar los centros que cumplan las características que el usuario marque, con respecto a tres criterios:
 - Enseñanzas que se imparten en el centro.
 - Servicios que ofrece el centro: aula matinal, comedor y actividades extraescolares.
 - Programas educativos con los que cuenta el centro: bilingüe español-inglés, bilingüe español-francés, bilingüe español-alemán.
- Consultar la información relativa al centro seleccionado.
- Situar y centrar al usuario en el mapa para que pueda comprobar los centros que tiene más cercanos.
- Calcular y visualizar la ruta desde la ubicación del usuario al centro seleccionado.
- Llamar al centro seleccionado.

Este proyecto ha sido desarrollado durante un semestre y se ha ajustado a la planificación establecida en el punto 4. *Planificación*.

Aunque los objetivos del proyecto se hayan cumplido, se proponen una serie de posibles ampliaciones que enriquecerían la funcionalidad del sistema desarrollado:

- Se podría dar la posibilidad al usuario de guardar una búsqueda realizada, para poder consultarla nuevamente sin necesidad de tener que pasar por las pantallas de selección, sino directamente desde la pantalla de inicio.
- Otra posibilidad para guardar centros ya consultados y que le interesan al usuario, sería poder marcarlos como favoritos, y dar la opción desde la pantalla inicial de visualizar directamente los centros señalados como favoritos.
- Poder realizar una búsqueda por el nombre de un centro y que te devuelva un listado con los resultados, señalando además la provincia y municipio donde se encuentran. De este resultado, el usuario podrá pulsar sobre uno de los elementos devueltos para que lo sitúe y lo centre en el mapa.
- También se puede trabajar en la apariencia de la aplicación. Por ejemplo, se pueden configurar botones personalizados con efecto de volumen, en vez de los botones por defecto de Android. También se pueden probar nuevos temas para mostrar la aplicación con más color.

BIBLIOGRAFÍA

Consultas bibliográficas

- (1) Booch, G.; Rumbaugh, J. Et al., El Lenguaje Unificado de Modelado, Addison Wesley, 1999. 464 p. ISBN: 84-7829-028-1.
- (2) Pressman, R., Ingeniería del Software: Un Enfoque Práctico, 4a Edición, 1997. 581 p. ISBN: 84-481-1186-9.
- (3) Luque Ruiz, I.; Gómez-Nieto, M. A., Ingeniería del Software: Fundamentos para el desarrollo de sistemas informáticos. Servicio de Publicaciones de la Universidad de Córdoba. 1999. 304 p. ISBN: 84-7801-486-1.

Consultas web

- (a) Curso Programación Android: Aprende a crear tus propias aplicaciones Android.
 - http://www.sgoliver.net/blog/?page_id=2935
 - Última actualización: Diciembre de 2012.
- (b) Programación Android
 - <http://www.nosinmiubuntu.com/p/programacion-android.html>
 - Última actualización: Noviembre de 2012.
- (c) Documentación de la API de Google Maps
 - <https://developers.google.com/maps/documentation/android/>
 - Última actualización: Diciembre 2012.
- (d) Patrones de diseño para aplicaciones Android
 - <http://www.androidpatterns.com/>
- (e) Artículos varios sobre programación en Android
 - <http://androideity.com/category/programacion/>
 - Última actualización: Noviembre 2012.
- (f) Como programar en Android. Guías 1 a 6.
 - <http://blog.findemor.es/tag/android/>
 - Última actualización: Febrero 2012.

ANEXO I. SCRIPT PYTHON PARA GENERAR LA CODIFICACIÓN GEOGRÁFICA DE LOS CENTROS

Con el script que se muestra a continuación, se ha generado a partir del fichero csv descargado del [directorio de centros docentes de Andalucía](#) con los datos de todos los centros de dicha comunidad, otro fichero csv con los mismos datos, y con dos columnas adicionales correspondientes a su *latitud* y *longitud*; con el objetivo de poder geolocalizar dichos centros en el mapa.

En caso de no poder localizar alguno de los centros, se vuelcan sus datos en un fichero distinto para proceder posteriormente a buscar su geolocalización de manera manual.

Una vez obtenido el fichero csv con los datos completos de cada centro, se ha importado en la base de datos dando lugar a la tabla “centros”.

```
# -*- coding: utf-8 -*-

import csv

from geopy import geocoders
from django.utils.encoding import smart_str

# Partimos de este fichero, donde tenemos todos los centros con su dirección
CSV_INPUT_FILE = 'centros.csv'

# Los centros localizados los iremos volcando a este fichero, ya con dos columnas
# más, una para latitud y otra para longitud.
CSV_OUTPUT_FILE = 'centros_localizados.csv'

# En caso de no conseguir localizar a un centro, lo pasaremos a este fichero, para
# realizar posteriormente una búsqueda manual del mismo.
CSV_NO_LOCATED = 'centros_no_localizados.csv'

# Clave para poder utilizar las API de Google Maps
GOOGLE_MAPS_API_KEY = '01PLCHjAezWyfHKp98c8tfq-zcU_MZJ2O1rw1OA'

if __name__ == "__main__":
    # Obtenemos el objeto para poder lanzar las consultas de geolocalización
```

```
g = geocoders.Google(GOOGLE_MAPS_API_KEY, 'maps.google.es')

# Abrimos para lectura el fichero donde tenemos cargados todos los centros
csv_input_file = open(CSV_INPUT_FILE, 'r')
reader = csv.DictReader(csv_input_file, delimiter=';', restval=None, quotechar='')

# Abrimos para escritura el fichero donde volcaremos los centros con la latitud y
# longitud calculada
csv_output_file = open(CSV_OUTPUT_FILE, 'w')
writer = csv.DictWriter(csv_output_file, delimiter=';',
                        fieldnames=reader.fieldnames + ['lat', 'long'], quotechar='')

# Abrimos para escritura el fichero donde volcaremos los centros lo localizados, para
# buscarlos manualmente
csv_no_located_file = open(CSV_NO_LOCATED, 'w')
writer_nolocated = csv.DictWriter(csv_no_located_file, delimiter=';',
                                  fieldnames=reader.fieldnames, quotechar='')

writer.writeheader()
no_localizados = []

# Recorremos cada centro del fichero de entrada
for row in reader:
    # Reemplazamos algunas abreviaturas para evitar errores
    domicilio = row['Domicilio'].replace('nº', '').replace('C/', 'calle').replace('Avda.',
                                         'avenida').replace('Ctra.', 'carretera').replace('Bda.', 'barriada').replace('Urb.',
                                         'urbanización').replace('s/n', '')

    # Formamos la consulta que vamos a lanzar con los datos que tenemos del centro
    query = "%s, %s, %s, %s Andalucía España" % (domicilio, row['CódPostal'],
                                                  row['Municipio'], row['Provincia'])
    query = smart_str(query.strip())
    lat, lon = None, None
```

```
try:
    # Lanzamos la consulta
    geocodes = g.geocode(query, exactly_one=False)
    place_google, (lat, lon) = geocodes[0]
    print "Localizado: %s: %s %s" % (query, lat, lon)
except Exception as e:
    # Si no se ha encontrado, lo escribimos en el fichero de no localizados
    print "error: %s" % query, e
    no_localizados.append(row["Denominación"])
    writer_nolocated.writerow(row)

if lat and lon:
    # Si tenemos latitud y longitud, lo escribimos en el fichero de centros localizados
    data = {'lat': lat, 'long':lon}
    data.update(row)
    writer.writerow(data)

# Mostramos un resumen de lo sucedido
print "Resumen de no localizados"
print "%d centros no localizados:" % len(no_localizados)
print no_localizados

# Cerramos los ficheros
csv_input_file.close()
csv_output_file.close()
csv_no_located_file.close()
```

ANEXO II. ESTRUCTURA DE UN PROYECTO ANDROID

Para comprender como se construye una aplicación Android vamos a detallar cual es la estructura de carpetas básica que lo compone.

Cuando se crea un nuevo proyecto Android en Eclipse, se generan automáticamente las carpetas necesarias para poder generar más tarde la aplicación. Enumeremos aquellos puntos clave que nos interesa conocer:

Carpeta *src*

Esta es por defecto la carpeta donde se depositará el código fuente Java: clases principales, auxiliares, actividades, servicio, etc.

Todo el código que pongamos en esta carpeta será compilado cuando se requiera. También, como en los proyectos tradicionales de java, el código se organizará en carpetas que resultaran en paquetes.

Carpeta *res*

Esta es una carpeta más compleja ya que se subdivide en múltiples subdirectorios. Esta carpeta contiene los recursos necesarios para el proyecto: imágenes, vídeos, sonidos, cadenas de texto, vistas, etc. Los subdirectorios en los que podemos distribuir los recursos son:

- **Drawable:** Contiene las imágenes [y otros elementos gráficos] usados en por la aplicación. Para definir diferentes recursos dependiendo de la resolución y densidad de la pantalla del dispositivo se suele dividir en varias subcarpetas:
 - `drawable-ldpi` (densidad baja)
 - `drawable-mdpi` (densidad media)
 - `drawable-hdpi` (densidad alta)
 - `drawable-xhdpi` (densidad muy alta)
- **Layout:** Contiene los ficheros de definición XML de las diferentes pantallas de la interfaz gráfica.
- **Menu:** Contiene la definición XML de los menús de la aplicación.
- **Values:** Contiene otros ficheros XML de recursos de la aplicación, como por ejemplo cadenas de texto (*strings.xml*), estilos (*styles.xml*), colores (*colors.xml*), arrays de valores (*arrays.xml*), etc.
- **Values-v11:** los recursos incluidos en esta carpeta se aplicarían tan sólo a dispositivos cuya versión de Android sea la 3.0 (API 11) o superior.

Carpeta *gen*

Contiene una serie de elementos de código generados automáticamente al compilar el proyecto. Cada vez que generamos nuestro proyecto, la maquinaria de compilación de

Android genera por nosotros una serie de ficheros fuente java dirigidos al control de los recursos de la aplicación. Importante: dado que estos ficheros se generan automáticamente tras cada compilación del proyecto es importante no modificarlos manualmente bajo ninguna circunstancia.

A destacar sobre todo el fichero *R.java*, donde se define la clase R. Esta clase R contendrá en todo momento una serie de constantes con los ID de todos los recursos de la aplicación incluidos en la carpeta */res/*, de forma que podamos acceder fácilmente a estos recursos desde nuestro código a través de este dato.

Carpeta assets

Contiene todos los demás ficheros auxiliares necesarios para la aplicación (y que se incluirán en su propio paquete), como por ejemplo ficheros de configuración, base de datos, etc.

La diferencia entre los recursos incluidos en la carpeta *res* y los incluidos en la carpeta *assets* es que para los primeros se generará un ID en la clase R y se deberá acceder a ellos con los diferentes métodos de acceso a recursos. Para los segundos sin embargo no se generarán ID y se podrá acceder a ellos por su ruta como a cualquier otro fichero del sistema. Usaremos uno u otro según las necesidades de nuestra aplicación.

Carpeta bin

Ésta es otra de esas carpetas que en principio no tenemos por qué tocar. Contiene los elementos compilados de la aplicación y otros ficheros auxiliares. Cabe destacar el fichero con extensión “.apk”, que es el ejecutable de la aplicación que se instalará en el dispositivo.

Carpeta libs

Contendrá las librerías auxiliares, normalmente en formato “.jar” que utilizemos en nuestra aplicación Android.

Fichero AndroidManifest.xml

Es importante e imprescindible y describe como se empaquetará la aplicación. Contiene la definición en XML de los aspectos principales de la aplicación, como por ejemplo su identificación (nombre, versión, icono, ...), sus componentes (pantallas, mensajes, ...), las librerías auxiliares utilizadas, o los permisos necesarios para su ejecución. Veremos más adelante más detalles de este fichero.