

Título	TTT_Android_Control_Documentación_Final		
Proyecto	Práctica de la asignatura Proyectos Informáticos: Tiempos del Transporte de Teruel (TTT)		
Equipo	<p>Jorge Navarrete Argilés</p> <p>Jesús Fuentes Romero</p> <p>Óscar Munárriz Sesma</p> <p>Ángel Sánchez Espílez</p>		
Código	TTT_Andr_Con_Docu	Fecha	16/06/11
Profesor	Félix Serna Fortea	Versión	1.0

0. ÍNDICE

0. ÍNDICE.....	2
1. INTRODUCCION.....	4
1.1 Propósito.....	4
1.2 Alcance.....	4
1.3 Definiciones, acrónimos y abreviaturas.....	5
2. DESCRIPCIÓN GLOBAL.....	6
2.1 Perspectiva del producto.....	6
2.1.1 Interfaz de sistema.....	6
2.1.2 Interfaz software.....	6
2.1.3 Diagramas de Casos de Uso.....	7
3. CARACTERÍSTICAS DEL PRODUCTO.....	8
3.1 Características específicas.....	8
3.2 Restricciones de sistema	8
3.3 Restricciones de usabilidad.....	8
4. REQUISITOS ESPECÍFICOS.....	9
4.1 Requisitos generales.....	9
4.2 Requisitos específicos.....	9
4.2.1 General.....	9
4.2.2 Vista lista de líneas.....	9
4.2.3 Vista detalle de línea.....	9
4.2.4 Vista detalle de autobús.....	9
4.2.5 Vista de mapa.....	9
4.2.6 Lista de autobuses.....	10
4.3 Requisitos de sistema.....	10
5. DIAGRAMAS UML.....	11
5.1 Diagramas Clases.....	11
5.2 Diagramas Actividad.....	13
5.3 Diagramas Secuencia.....	16

6. INTERFACES DE USUARIO.....	17
6.1 Análisis de requisitos.....	17
A.- Desarrollo del concepto de producto.....	17
A.1.- Escenarios / storyboards.....	17
A.2.- Análisis competitivo.....	18
B.- Análisis de requisitos. Informe de usuarios / roles de la aplicación.....	18
B.1.1.- Posibles usuarios de la aplicación.....	18
B.1.2.- Fichas de usuarios.....	18
C.- Tareas.....	19
C.1.- Especificar las tareas.....	19
C.2.- Diagrama HTA de cada tarea.....	19
D.- Especificaciones	21
D.1.- De usabilidad.....	21
E.- Prototipado.	23
E.1. - Card Sorting	23
E.2.- Prototipos	25
E.3.- Recorrido Pluralístico.....	27
F.- Factor tecnológico.....	28
6.2 Diseño.....	29
1.- Guía de estilo.	29
Look & feel del proyecto.	29
Colores (paleta de colores)	29
Disposición de menús.	30
Formato de los listados.	31
Pautas de lenguaje y diccionario propio (vocabulario específico).....	32
2.- Diseño del prototipo no funcional.....	33
3.- Árbol de pantallas de nuestra aplicación.....	37
4.- Requerimientos para su uso e instalación.....	38
7. VTP.....	39
7.1. Introducción.....	39
7.2. Primeras pruebas.....	39
7.3. Planificación de pruebas.....	39
7.4. Pruebas que se van a realizar para la aplicación viajero.....	40
7.5. Necesidades para realizar las pruebas.....	40

1. INTRODUCCION

El presente documento contiene una Especificación de Requisitos de Software (ERS) para la aplicación informática que irá en los teléfonos y tabletas Android del personal encargado de controlar la puntualidad del servicio de autobuses que componen el sistema “Tiempos de Transporte de Teruel”, a realizar durante la asignatura Proyectos Informáticos. La estructura del presente documento está inspirada en las directrices referidas en el estándar IEEE Recommended Practice for Software Requirements Specifications IEEE 830-1998.

El objetivo principal de esta aplicación es proporcionar al personal información sobre el estado del servicio de autobuses.

1.1 Propósito

El objeto de la especificación de requisitos es definir de manera clara y precisa tanto las funcionalidades como las posibles restricciones del sistema de información a construir. Dicho documento está dirigido tanto a los miembros del equipo de desarrollo como al cliente del proyecto. En él, se establecen las bases sobre las cuales el equipo de desarrollo procederá al diseño y posterior construcción del nuevo sistema. También servirá de modelo en el cual basar las pruebas funcionales de aceptación por parte del cliente y la evaluación final del producto por parte del mismo.

1.2 Alcance

El objeto de la especificación de requisitos es definir de manera clara y precisa tanto las funcionalidades como las posibles restricciones del sistema de información a construir. Dicho documento está dirigido tanto a los miembros del equipo de desarrollo como al cliente del proyecto. En él, se establecen las bases sobre las cuales el equipo de desarrollo procederá al diseño y posterior construcción del nuevo sistema. También servirá de modelo en el cual basar las pruebas funcionales de aceptación por parte del cliente y la evaluación final del producto por parte del mismo.

1.3 Definiciones, acrónimos y abreviaturas

Línea	Secuencia de paradas donde un viajero puede coger un autobús para llegar a otra de las paradas de la línea
Sentido	Cada una de las direcciones en las que se puede recorrer una línea
Ruta	El trazado que une las paradas de una línea
Parada	Punto localizado geográficamente donde los autobuses de ciertas líneas pueden dejar o coger viajeros
Vista	Cada una de las pantallas que la aplicación puede mostrar
Mapa	Representación gráfica de las vías de circulación a partir de una vista aérea.
Android	Sistema operativo para dispositivos móviles compatible con aplicaciones

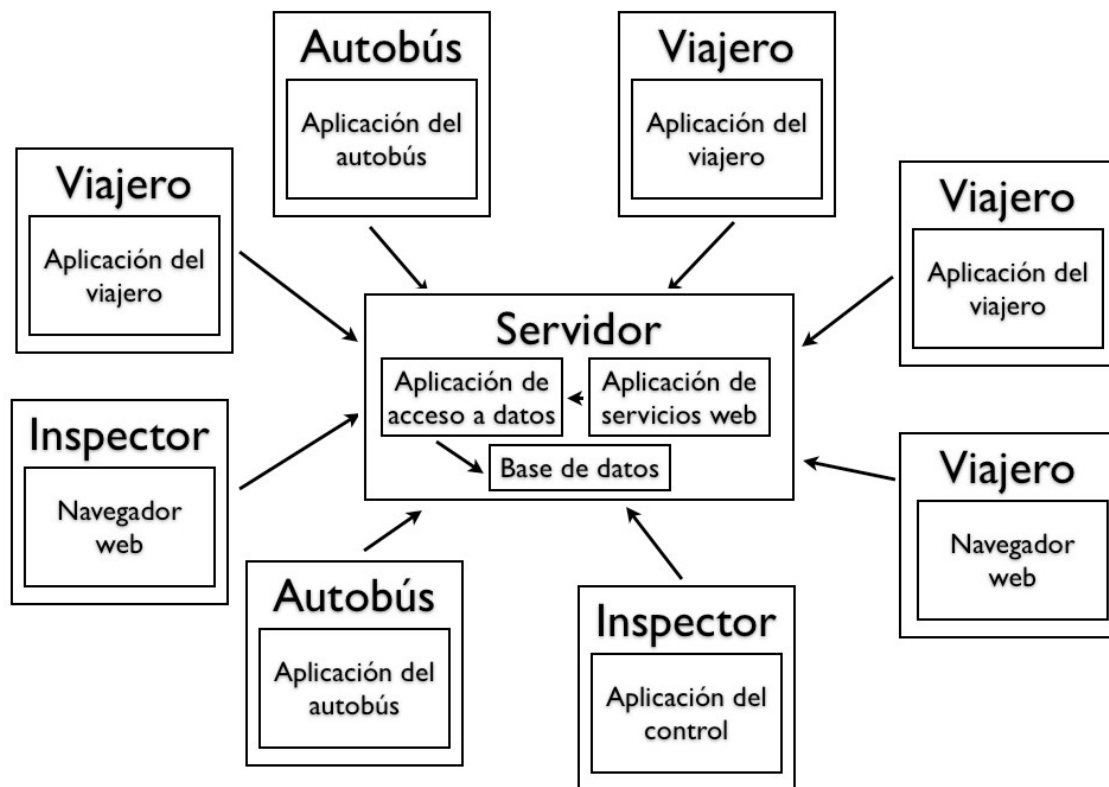
2. DESCRIPCIÓN GLOBAL

2.1 Perspectiva del producto

La aplicación obtiene del sistema la información necesaria, por lo que se necesitará conectar a una aplicación central. Para ello se debe seguir un protocolo por definir.

2.1.1 Interfaz de sistema

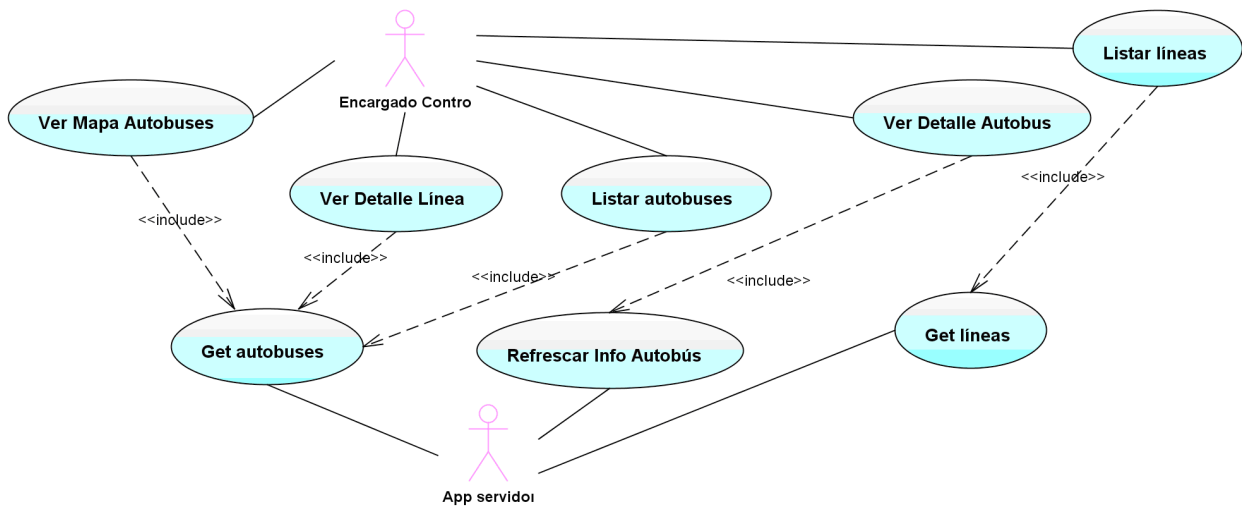
La aplicación de control actúa como una aplicación cliente en el sistema, es decir, se debe conectar a otra aplicación central para su correcto funcionamiento.



2.1.2 Interfaz software

Se requiere Android 2.1 o superior.

2.1.3 Diagramas de Casos de Uso



Ver Mapa Autobuses: La aplicación mostrará al controlador un mapa donde se podrá ver la ubicación geográfica de todos los autobuses que están dando servicio en ese momento.

Ver Detalle Línea: La aplicación mostrará al controlador un mapa donde se podrá ver la ubicación geográfica de los autobuses de la línea.

Listar autobuses: La aplicación mostrará al controlador una vista con un listado de todos los autobuses que están prestando servicio en ese momento.

Ver Detalle Autobús: La aplicación mostrará al controlador una vista con los datos del autobús solicitado: Número de autobús, línea en la que está asignado, ubicación geográfica y velocidad media.

Listar líneas: La aplicación mostrará al controlador una vista con un listado de todas las líneas de Teruel.

Get autobuses: El servidor debe responder devolviendo un listado con todos los autobuses que están cubriendo líneas en ese momento.

Refrescar info Autobús: El servidor debe responder con la información de localización, velocidad media y línea asignada del autobús indicado.

Get líneas: La aplicación del servidor debe responder devolviendo un listado con todas las líneas del sistema.

3. CARACTERÍSTICAS DEL PRODUCTO

Las características del producto definen las funcionalidades que tendrá la aplicación. Las características se detallan en el punto 4 “Requisitos específicos”.

3.1 Características específicas

CAR_CON_01: El personal de control puede ver la lista de las líneas de la ciudad en la vista de Líneas.

CAR_CON_02: El usuario puede ver el detalle de las líneas de la ciudad con la situación de los autobuses en la vista Detalle de línea.

CAR_CON_03: El personal dispone de la opción de obtener el detalle de cada autobús en funcionamiento accediendo desde otras vistas a la de Detalle de autobús.

CAR_CON_04: El usuario puede ver la posición de los autobuses en un mapa en la vista Mapa.

CAR_CON_05: El usuario puede listar los autobuses que están cubriendo líneas en la vista Lista de autobuses.

3.2 Restricciones de sistema

RES_GEN_01: Se requiere Android 2.1 o superior.

3.3 Restricciones de usabilidad

RES_INT_01: Para la visualización de mapas se debe usar la API de Google Maps para Android.

4. REQUISITOS ESPECÍFICOS

4.1 Requisitos generales

REQ_GEN_01_01: La aplicación obtendrá la información cuando el usuario lo solicite o bien pasados unos segundos desde la anterior actualización.

4.2 Requisitos específicos

Los requisitos están agrupados por la vista a la que se refiere.

4.2.1 General

REQ_CON_00_01: La aplicación presenta una pantalla de bienvenida como pantalla inicial. Después se muestra la vista del Mapa.

REQ_CON_00_02: Desde cualquier pantalla el usuario dispone de la opción de saltar a la vista de Mapa, Lista de líneas o Lista de autobuses.

4.2.2 Vista lista de líneas

REQ_CON_01_01: La aplicación presenta en una pantalla la lista de líneas donde se enumeran junto a sus sentidos. El personal de control dispone la opción de elegir una de ellas para saltar a su vista Detalle de línea.

4.2.3 Vista detalle de línea

REQ_CON_02_01: La aplicación debe ubicar gráficamente la situación de los autobuses que están recorriendo la línea de forma que al pulsar sobre uno de ellos se accede a su vista Detalle de autobús.

REQ_CON_02_02: Periódicamente la aplicación actualizará la posición de los autobuses. Si el usuario desea forzarlo, dispondrá de la opción de refrescar.

4.2.4 Vista detalle de autobús

REQ_CON_03_01: La vista debe mostrar el número de autobús, la línea a la que está sirviendo y la velocidad media en el último tramo.

REQ_CON_03_02: Existe un botón atrás que nos devuelve a la vista que nos ha remitido a este detalle de autobús (Mapa, Detalle de línea o Lista de autobuses).

REQ_CON_03_03: El personal de control dispone de la posibilidad de saltar a la vista de Mapa centrando el plano en este autobús.

REQ_CON_03_04: Periódicamente la aplicación actualizará la información del autobús. Si el usuario desea forzarlo, dispondrá de la opción de refrescar.

4.2.5 Vista de mapa

REQ_CON_04_01: El mapa muestra todos los autobuses circulando por la ciudad.

REQ_CON_04_02: Cuando el personal de control pulse sobre un autobús se muestra su vista Detalle de autobús.

REQ_CON_04_03: Periódicamente la aplicación actualizará la posición de los autobuses. Si el usuario desea forzarlo, dispondrá de la opción de refrescar.

4.2.6 Lista de autobuses

REQ_CON_05_01: El listado presenta todos los autobuses en servicio de la ciudad ordenados por su número de autobús y la línea en la que está prestando servicio.

REQ_CON_05_02: Cuando el personal pulse sobre un autobús se salta a su vista Detalle de autobús.

REQ_CON_05_03: Periódicamente la aplicación actualizará el listado de autobuses. Si el usuario desea forzarlo, dispondrá de la opción de refrescar.

4.3 Requisitos de sistema

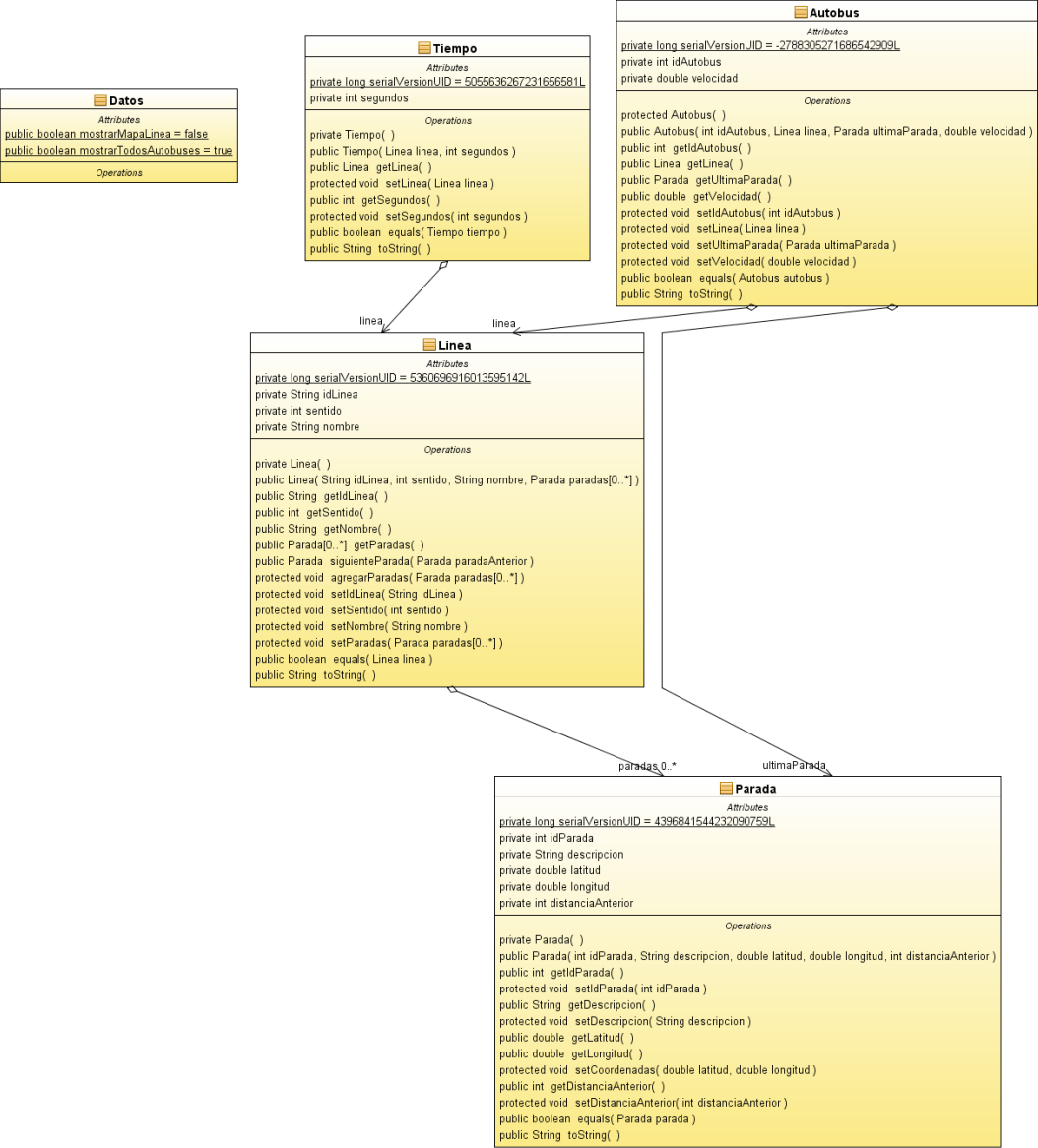
REQ_SIS_01: Se requiere Android 2.1 o superior.

REQ_SIS_02: La aplicación debe mantener la conexión con el servidor el mínimo tiempo posible.

REQ_SIS_03: La aplicación solamente puede hacer consultas a la base de datos, no modificar la información.

5. DIAGRAMAS UML

5.1 Diagramas Clases



Explicación diagrama de Clases:

La clase Linea contiene la información del identificador de la línea, el sentido de la marcha en el que se circula, y el nombre de la línea. Tiene un método para comprobar cual es la siguiente parada a la que se debe ir (siguienteParada()). El resto de métodos son get/set y constructores.

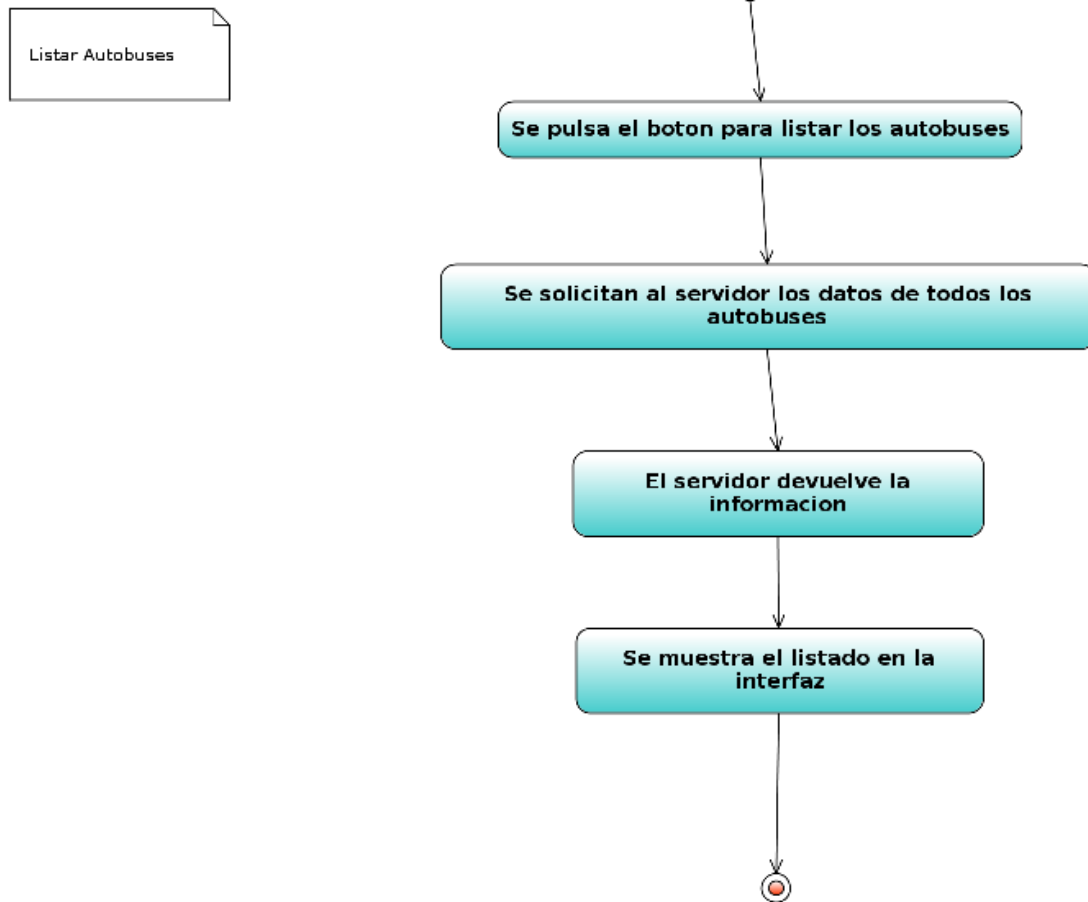
La clase Parada por su parte almacena todos los datos de una parada como su identificador, su descripción, su posición y la distancia a la anterior parada. Esta clase contiene métodos get y set junto con sus constructores.

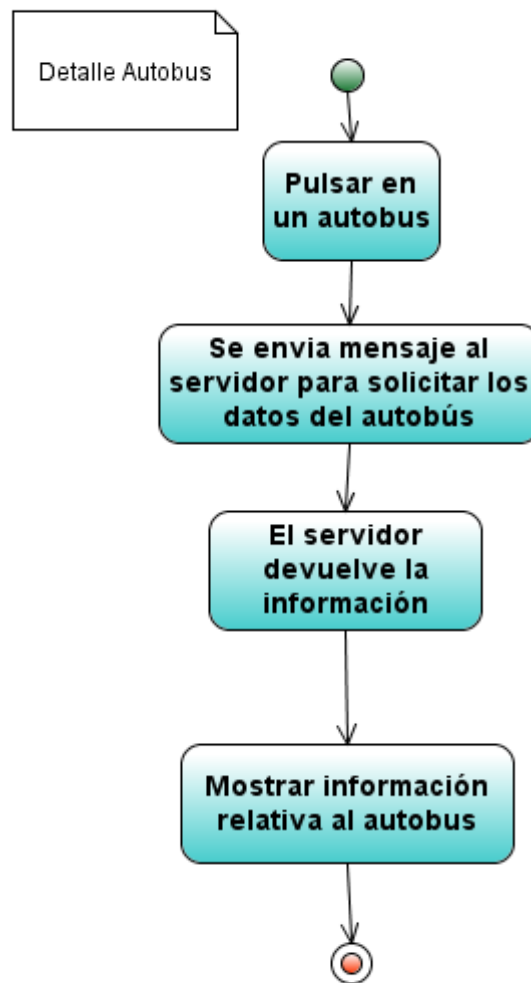
La clase Autobús almacena el identificador del autobús, y la velocidad a la que circula. Tiene métodos get/set como para obtener la velocidad o la última parada.

La clase Tiempo almacena el tiempo de paso de un autobús por una parada y sirve para poder procesar las peticiones del servidor. Pide las diferentes peticiones necesarias para la gestión del controlador, como obtener todas las paradas de una determinada linea. Esta clase contiene métodos get/set además de los constructores.

La clase datos contiene dos booleanos que son utilizados como banderas para hacer determinadas acciones, por ejemplo al arrancar la aplicación en el primer mapa deben aparecer todos los autobuses que se encuentren en funcionamiento, otro caso sería si queremos que en el mapa nos muestre todos los autobuses o sólo los que se encuentren recorriendo una línea determinada.

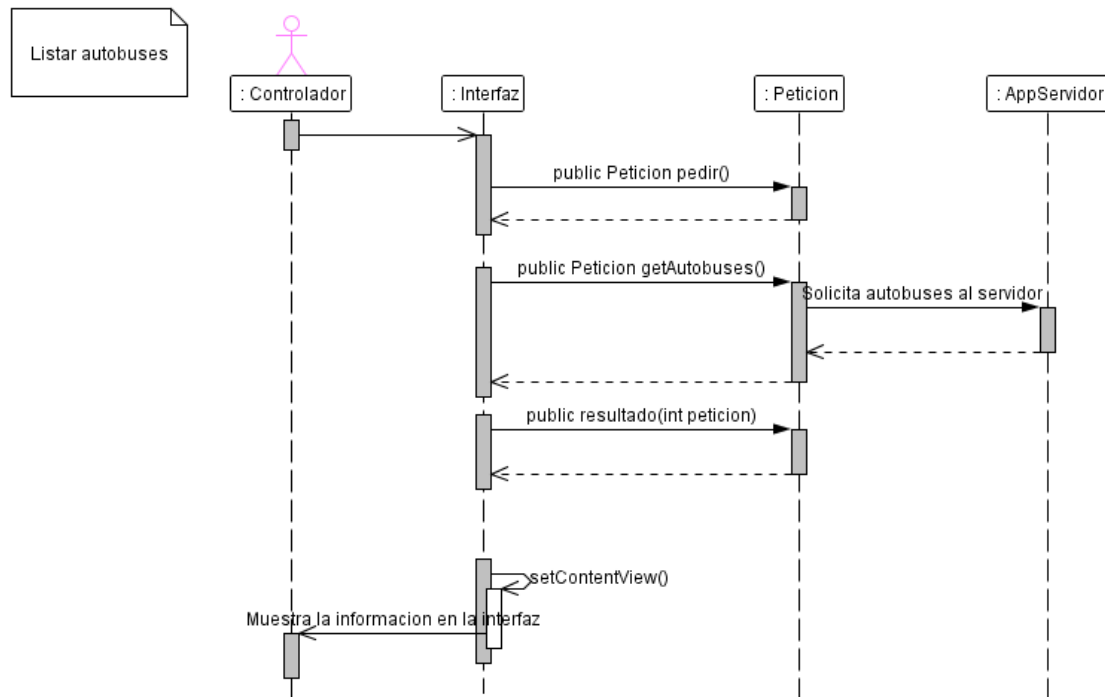
5.2 Diagramas Actividad



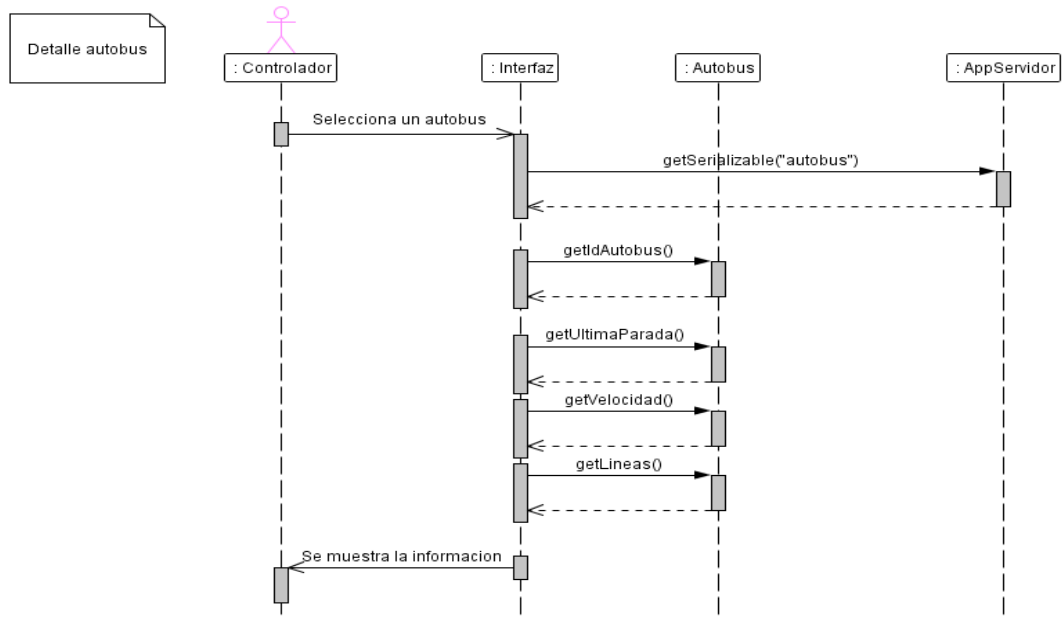




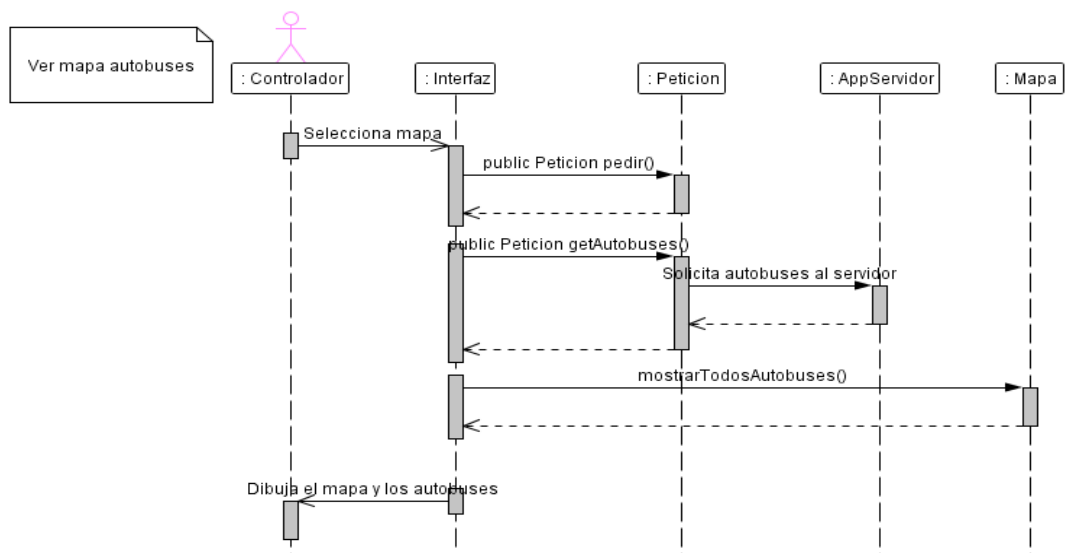
5.3 Diagramas Secuencia



Si el controlador pulsa la pestaña Autobuses, se mostrará el listado de todos los autobuses que recorren la ciudad. La interfaz hace una petición al servidor para que le de toda la información de la base de datos correspondiente a los autobuses. Una vez obtiene la información la muestra en forma de listado. Si el controlador pulsa uno de los autobuses de la lista nos llevará al mapa de la ciudad en el que se dibujarán todos los autobuses.



Si el controlador selecciona un autobús, se mostrará la información correspondiente a ese autobús. La interfaz hace una petición al servidor para que le de toda la información de la base de datos correspondiente a ese autobús. Una vez obtiene la información va leyendo el identificador del autobús, la última parada, la velocidad y la líneas en la que está. Posteriormente esta información se mostrará en pantalla. Una vez mostrada la información si el controlador desea volver a la pantalla anterior tendrá que pulsar el botón de volver.



Para ver el mapa con los autobuses, se realiza una petición al servidor en la que nos devuelve todos los autobuses, se procesa esta información y se muestra el mapa con los autobuses.

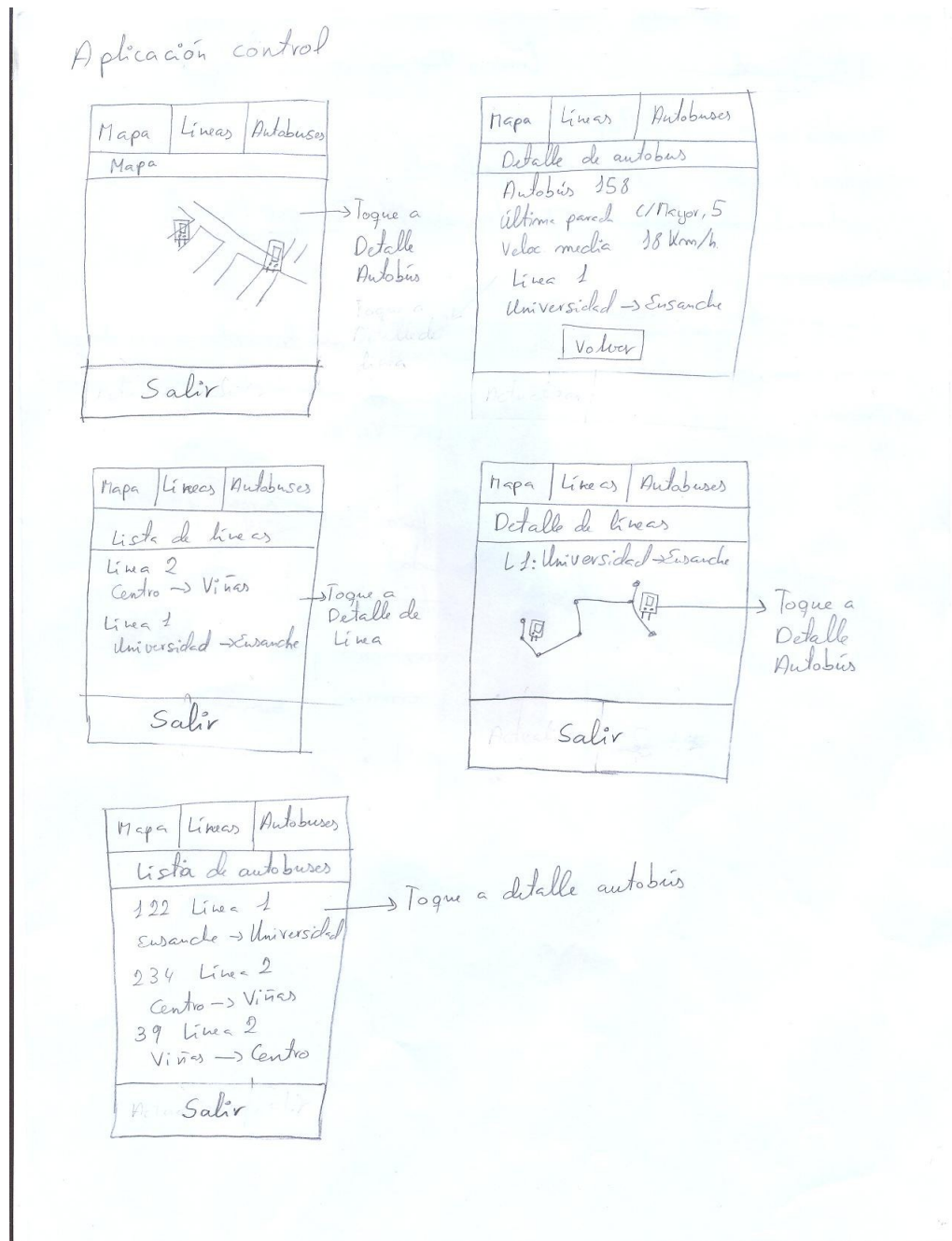
6. Interfaces de Usuario

6.1 Análisis de requisitos

A.- Desarrollo del concepto de producto.

A.1.- Escenarios / storyboards.

Se muestran en las siguientes páginas.



A.2.- Análisis competitivo.

Para la realización de este apartado hemos realizado una búsqueda por Internet intentando encontrar alguna aplicación similar pero no hemos tenido éxito para poder probar su funcionamiento, las aplicaciones que existen estarán instaladas en las salas de las diferentes empresas y estas empresas no son partidarias de que nadie conozca su sistema de información.

Pero reflexionando un poco llegamos a la conclusión de que las aplicaciones de este estilo serán todas más o menos similares y deberán mostrar la información necesaria para el correcto funcionamiento del sistema.

B.- Análisis de requisitos. Informe de usuarios / roles de la aplicación.

B.1.1.- Posibles usuarios de la aplicación,

Para esta aplicación solo existe un rol de usuario, los operarios encargados del control del sistema de gestión de los autobuses, estos operarios pueden visualizar la posición de los autobuses, obtener listados de los diferentes autobuses y líneas que recorren la ciudad, así como también conocer información más detallada de los autobuses y las diferentes líneas.

B.1.2.- Fichas de usuarios

En esta aplicación solo existe un rol de usuario, el conductor del autobús que como tal solo puede introducir los datos del autobús, iniciar la ruta y una vez en ruta alterar el sentido de la marcha y comunicarlo al servidor.

C.- Tareas.

C.1.- Especificar las tareas.

El operario de control podrá realizar las siguientes tareas:

- Ver los autobuses que se encuentran en circulación en un determinado instante.
- Ver listado de líneas de autobús.
- Ver listado de autobuses que recorren la ciudad.
- Ver detalle de cada línea de autobús.
- Ver detalle de cada autobús.

C.2.- Diagrama HTA de cada tarea.

Tareas del operario de control:

- Tarea 1: Ver los autobuses que se encuentran en circulación en un determinado instante.

Diagrama:

1. Ver los autobuses que se encuentran en circulación en un determinado instante.

1.1. Seleccionar Mapa.

1.2. Se muestran los autobuses en circulación.

1.3. Actualizar si fuera necesario.

- Tarea 2: Ver listado de líneas de autobús.

Diagrama:

2. Ver listado de líneas de autobús.

2.1. Seleccionar Líneas.

2.2. Se muestra un listado de líneas.

- Tarea 3: Ver listado de autobuses que recorren la ciudad.

Diagrama:

3. Ver listado de autobuses que recorren la ciudad.

3.1. Seleccionar autobuses.

3.2. Mostrar autobuses en el mapa.

3.3. Actualizar si fuera necesario.

- Tarea 4: Ver detalle de cada línea de autobús.

Diagrama:

4. Ver detalle de cada línea de autobús.

4.1. Seleccionar Líneas.

4.2. Seleccionar líneas a ver con detalle.

4.3. Se muestra la información solicitada.

- Tarea 5: Ver detalle de cada autobús.

Diagrama:

5. Ver detalle de cada autobús.

5.1. Seleccionar Autobús.

5.2. Seleccionar autobús a visualizar.

5.3. Se muestra la información.

D.- Especificaciones

D.1.- De usabilidad.

Sobre los usuarios:

- Los usuarios de aplicaciones móviles se desenvuelven en entornos donde van a estar expuestos a multitud de **estímulos simultáneos** fuera de la aplicación. La gran diferencia con respecto a entornos telemáticos tradicionales es que la atención del usuario va a estar influida en gran medida por el entorno. Por consiguiente, los productos destinados a dispositivos móviles han de tener en cuenta estas circunstancias y crear aplicaciones que sean sencillas, de navegación extremadamente simple y cuyos objetivos sean fácilmente alcanzables con la mínima carga cognitiva.
- Debido a que la mayoría de las ocasiones en las que se utiliza el dispositivo móvil van a implicar entornos cambiantes o, al menos, abiertos, un factor a tener muy en cuenta es la **luminiscencia que sufrirá la interfaz inicial**. Esta sufrirá variaciones drásticas incluso en escasos periodos de tiempo imposibilitando la fijación de la atención en tareas largas o que requieran una atención excesiva.
- En el caso de que sea necesario, aplique técnicas de **internacionalización** y localización a fin de acomodar los entornos que cree a las distintas culturas que puedan actuar potencialmente con el desarrollo que usted está llevando a cabo.
- Es crítico saber el **tipo de población** a la que el programa va ir destinado. Diversos usuarios utilizando un mismo entorno, pueden provocar grandes inconvenientes, en tanto en cuanto su manera de interactuar varía. Por ese motivo, es recomendable hacer los desarrollos de software lo más estándares posible, sin particularizar hacia poblaciones específicas.

Sobre la interacción Usuario – Máquina:

- Debido a que la movilidad es el aspecto que va a imperar en el desarrollo, existe un factor determinante para el terminal: su **reducido tamaño**. Desde el punto de vista de la Usabilidad ello va a generar consecuencias críticas a tener en cuenta, como puede ser el reducido tamaño de la pantalla, por ejemplo. Por ese motivo, la simplificación de la terminología, la asignación de extrema relevancia a la información icónica o la elección de iconos auto-explicativos son pilares fundamentales sobre los que planificar el diseño y estructura de la interfaz.
- Adicionalmente al punto anterior, la **interactividad** que el usuario va a tener con el terminal va a ser **reducida**, ya que, en la mayoría de los casos, lo va a sostener con una mano y con la otra va a sostener un puntero o con el propio dedo por medio del cual interactuará con la interfaz. Por ese modo, la versatilidad de movimientos va quedar reducida en gran medida.

- Otro aspecto a tener en cuenta y que también se encuentra ligado a este punto es que debido a que estamos hablando de terminales que se utilizan con una sola mano y que, generalmente, quedan alineados visualmente hacia el lado de la mano que está sosteniendo dicho terminal, la percepción de los elementos de la interfaz será distinta si es observada por un usuario diestro o por un usuario zurdo. Especialmente, esto es crítico en usuarios **zurdos**, pues estos, asimilando la manera de coger el bastón como suelen hacerlo con los bolígrafos cuando escriben, en muchos casos adoptarán una posición de pinza, la cual, al interactuar con la interfaz y realizar desplazamientos, puede provocar frecuentes bloqueos visuales de elementos de la interfaz. La mano de este modo, taparía parte de la pantalla, dificultando el uso y la efectividad de las acciones realizadas.

- Mientras que **seleccionar opciones de la pantalla con el puntero** del ratón de un ordenador de sobremesa viene a ser algo sencillo y fácilmente adquirible como destreza, emular las mismas actuaciones en el mundo de los dispositivos móviles es algo mucho más complicado. La destreza de los usuarios a la hora de lograr un impacto con un bastón de smartphone en un determinado punto de la pantalla es ostensiblemente inferior a la que logran ejecutando la misma tarea con un puntero de ratón en un ordenador de sobremesa. Ello, queda agravado por el hecho de que muchos usuarios tienden a utilizar sustitutos del bastón si este se ha extraviado. De modo que reemplazan dicho elemento de navegación por bolígrafos, palillos, lapiceros, etc...

Sobre la máquina:

- Si bien los ordenadores personales se han convertido en herramientas cotidianas del entorno profesional y doméstico, los smartphones aún no han alcanzado ese estatus. Por ese motivo no se les ‘perdonan’ ciertos fallos que si bien están presentes en los terminales de sobremesa, pueden provocar el desuso o el abandono del smartphone. Por ello, es imprescindible desarrollar las aplicaciones y los terminales para que presenten los mínimos inconvenientes posibles.

- Minimice todo lo que pueda las interacciones que deba hacer el usuario con el terminal. Ello, se ha de hacer partiendo de la base que la pantalla es de reducidas dimensiones, que solo podrá interactuar con una mano, sosteniendo un bastón (el cual no les es familiar y, a veces, ni siquiera cómo), que lo hará en movimiento y que el entorno presentará gran cantidad de estímulos distractores. En consecuencia, las acciones, o navegaciones deben disponer de un ratio máximo de ‘clicks’ de tres o cuatro.

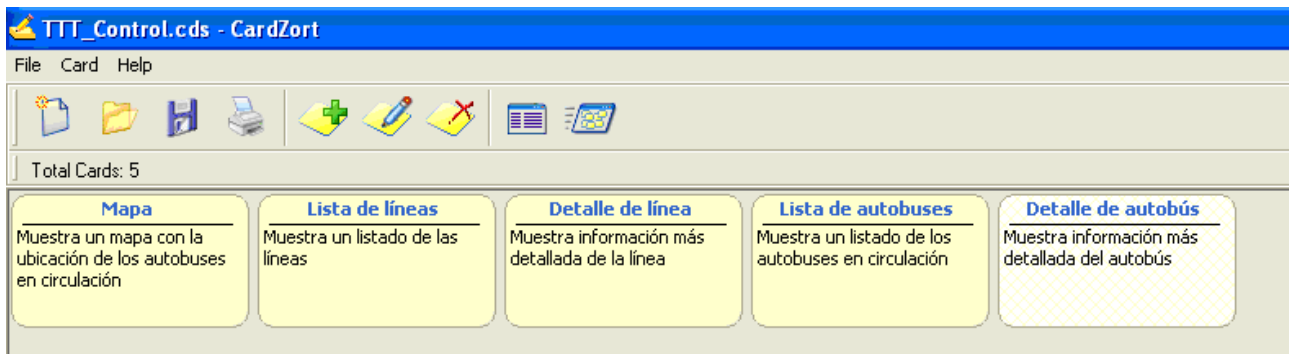
- Aquellas tareas no necesariamente relacionadas con la interfaz o con el producto de software en sí, han de ser igualmente minimizadas. De este modo, se ha de buscar una simplicidad casi infantil en tareas tales como establecer conexiones WIFI, sincronizar los datos e informaciones de los ordenadores de sobremesa con la información contenida en el smartphone o las actualizaciones de software requeridas por parte de los paquetes de software contenidos en los smartphones.

E.- Prototipado.

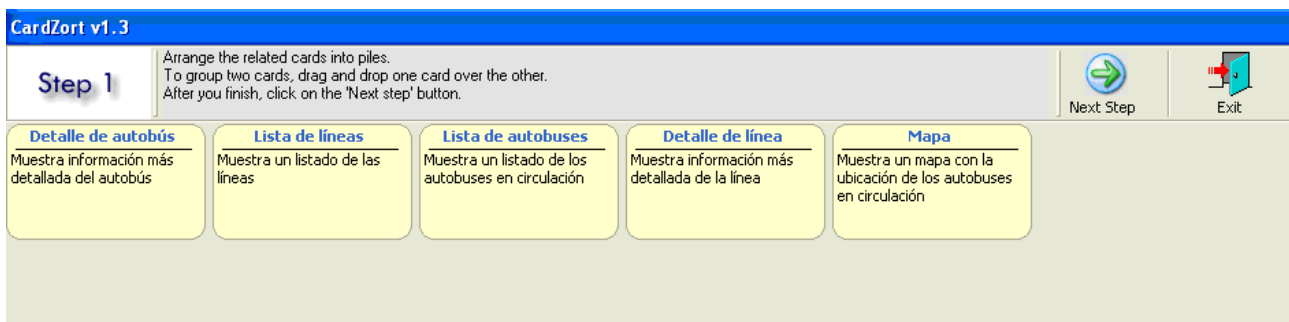
E.1. - Card Sorting

Para la realización de este análisis hemos usado la herramienta CardZort facilitada por el profesor obteniendo las siguientes capturas de pantalla.

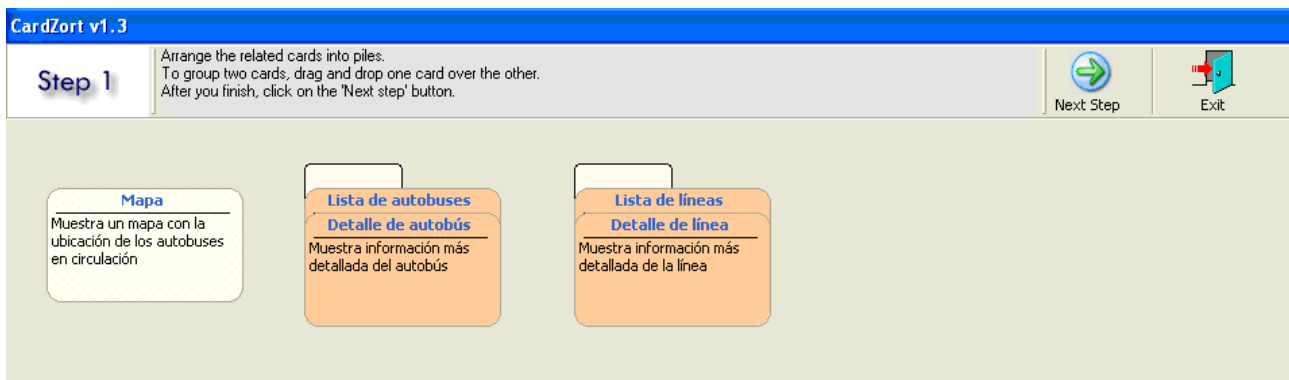
Pantalla que muestra las tarjetas creadas



Pantalla de reordenación de tarjetas



Pantalla en la que se han agrupado ya las tarjetas





Pantalla para nombrar las agrupaciones


CardZort v1.3

Step 2

Give a meaningful name to each one of the groups you created.
To name a group, click on the white group tab and type the name.
After you finish, click on the 'Finish' button.

Prev. Step

Finish

Exit

Mapa

Muestra un mapa con la ubicación de los autobuses en circulación

Autobuses

Lista de autobuses

Detalle de autobús

Muestra información más detallada del autobús

Líneas

Lista de líneas

Detalle de línea

Muestra información más detallada de la línea

E.2.- Prototipos

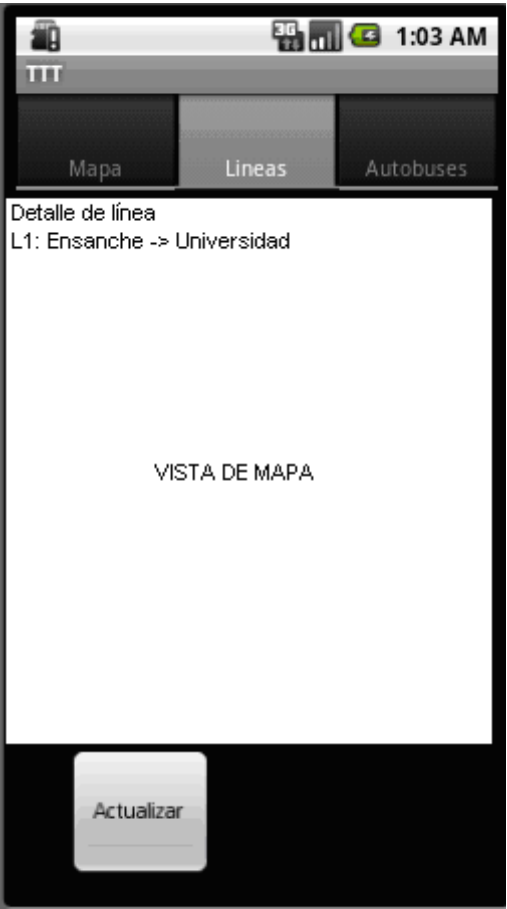
Mapa



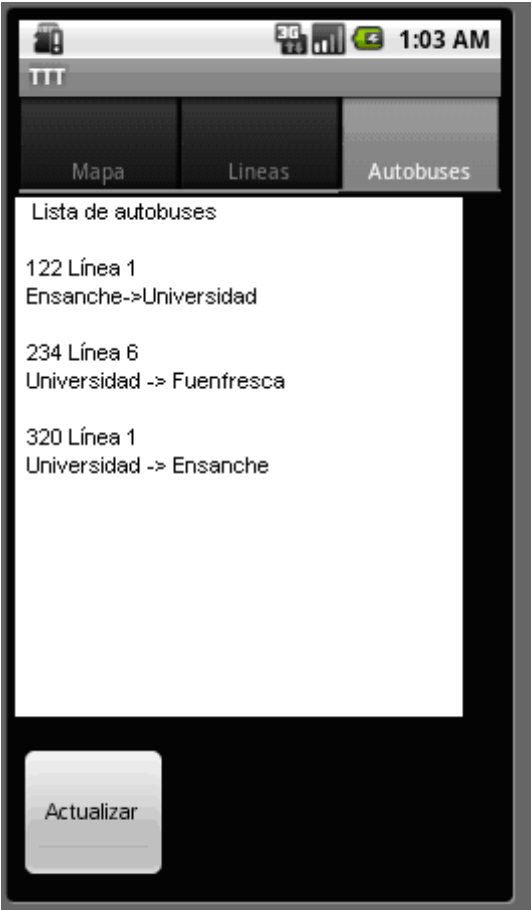
Lista de líneas



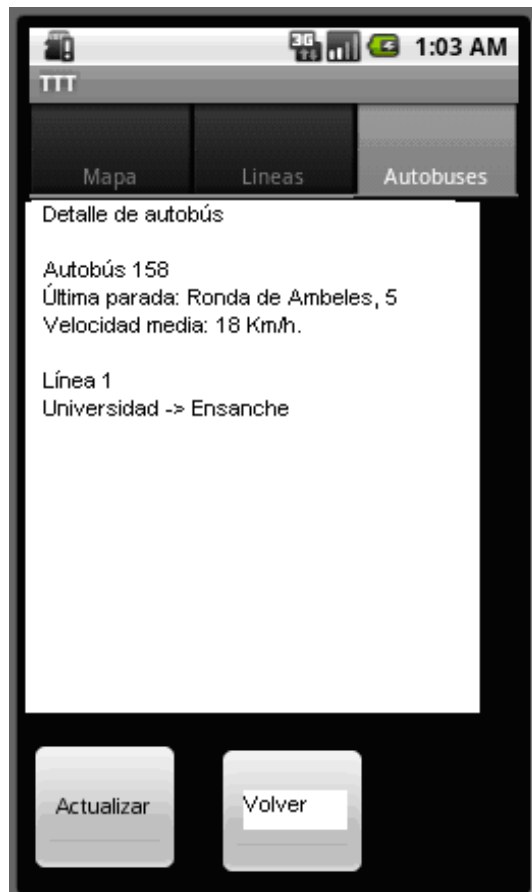
Detalle de línea



Lista de autobuses



Detalle de autobús



E.3.- Recorrido Pluralístico

- Tarea 1: Ver los autobuses que se encuentran en circulación en un determinado instante.

Pasos:

1. Pulsar Mapa.
2. Se muestran los autobuses en circulación.
3. Actualizar si fuera necesario.

- Tarea 2: Ver listado de líneas de autobús.

Pasos:

1. Pulsar Líneas.
2. Se muestra un listado de líneas.

- Tarea 3: Ver listado de autobuses que recorren la ciudad.

Pasos:

1. Pulsar autobuses.
2. Se muestran los autobuses en el mapa.
3. Actualizar si fuera necesario.

- Tarea 4: Ver detalle de cada línea de autobús.

Pasos:

1. Pulsar Líneas.
2. Seleccionar línea a ver con detalle.
3. Se muestra la información solicitada.

- Tarea 5: Ver detalle de cada autobús.

Pasos:

1. Pulsar Autobús.
2. Pulsar autobús a visualizar.
3. Se muestra la información.

F.- Factor tecnológico.

La aplicación será diseñada para que funcione en terminales Android, con una versión 2.1 ó superior, para ello bastará con lanzar un ejecutable .apk en cada terminal, además, los desarrolladores deberán obtener una APIKey para GoogleMaps.

6.2 Diseño

1.- Guía de estilo.

Look & feel del proyecto.

Las guías de estilo tenidas en cuenta en el desarrollo de esta aplicación se pueden encontrar en la siguiente página:

http://developer.android.com/guide/practices/ui_guidelines/index.html

Dado que en las clases no se especificaban modelos de look and feel orientados a aplicaciones móviles y aún menos para aplicaciones Android hemos tenido que buscar por nuestra cuenta una serie de guías de estilo para Android que propusieron los inventores de la tecnología para que las aplicaciones de esta tecnología tuvieran unos estándares en los que basarse, es por esto que hemos elegido usar estas guías de estilo y no basarnos en las expuestas en clase.

Colores (paleta de colores)

En cuanto a la paleta de colores hemos basado nuestros diseños en el negro de fondo para facilitar la visión de las pantallas incluso en ambientes con alta luminosidad dado que si el fondo es blanco el brillo impide la lectura y comprensión de los elementos de la interfaz en estos dispositivos que están diseñados para su uso en movimiento. Por otro lado para garantizar el contraste de los textos hemos usado color blanco para los textos y así garantizar que se verán bien y se distinguirán del fondo.

Además de estos dos colores para los Dialogs que describiremos posteriormente utilizamos un color gris. También usamos un color gris claro para los botones. Unas muestras de los colores son las que siguen:



En cuanto a las metáforas graficas usadas en el caso de esta aplicación tenemos las siguientes:



Símbolo de autobús que se muestra en el mapa



Autobús



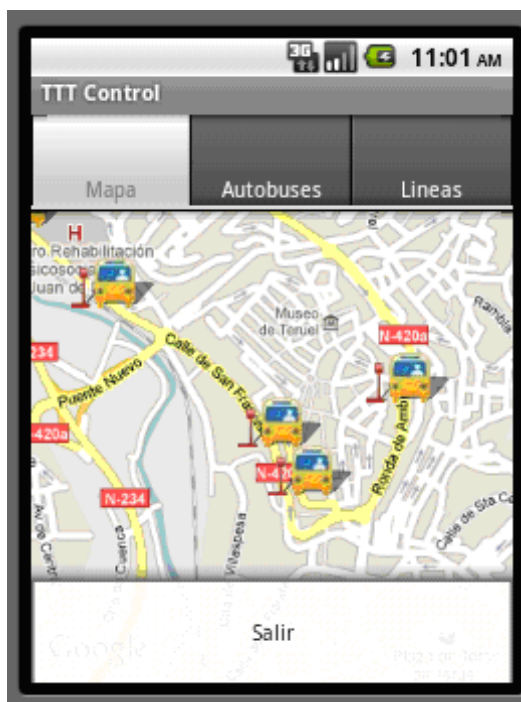
Linea



Parada

Disposición de menús.

En el caso de la aplicación para el control tendremos en cuenta varios menús porque la aplicación está pensada para que el encargado de control pueda navegar entre las diferentes pantallas con diferentes botones y que éstos estén puestos en el menú para que no resten visibilidad en la pantalla, ya que, las pantallas de los dispositivos móviles son de pequeño tamaño. Un ejemplo: menú para abandonar la aplicación.



Tenemos Dialogs en Android que son una especie de ventanas que se disparan con algún evento y permiten generar diálogos con el usuario de progreso. En nuestro caso tenemos ejemplos de Dialogs de progreso que se muestran al cargar diferentes pantallas de la aplicación: las líneas de autobuses que recorren la ciudad, los diferentes autobuses que se encuentran en circulación o las paradas que componen una línea de autobús.

A continuación podemos ver un ejemplo:



Formato de los listados.

El formato de los listados que hemos utilizado se basa en un formato típico de Android que consiste en mostrar la información en elementos unos debajo de otros ocupando todo el ancho de la pantalla y permitiendo interactuar con estos elementos pulsando sobre uno de ellos de forma que toda la pantalla este ocupada y podamos interactuar con los elementos de la lista de forma fácil y efectiva.

En nuestra aplicación hemos colocado dos listados:

El primero se corresponde con la pantalla que muestra los autobuses que componen el servicio de autobuses, este listado con los autobuses en funcionamiento sirve para que el operario de control seleccione el autobús que desee y pueda conocer diversos detalles, pulsando en un autobús, el autobús se seleccionará y le dirigirá a la pantalla que muestra los detalles del autobús.

Listado de autobuses:



El segundo se corresponde con la pantalla que muestra las líneas y el sentido que componen el servicio de autobuses, este listado con las líneas de transporte urbano disponibles para que el operario de control seleccione la línea que desee y pueda ver los autobuses que están recorriendo esa línea y las diferentes paradas de que consta la línea seleccionada, pulsando en una línea, ésta se seleccionará y le dirigirá a la pantalla que muestra el recorrido de la línea.

El aspecto de esta pantalla de listado de líneas es el siguiente:



Pautas de lenguaje y diccionario propio (vocabulario específico).

Por su parte el vocabulario empleado incluye ciertos términos técnicos como Autobús, Favoritos, Línea, Mapa, Parada, Sentido, pero que a su vez se han hecho muy comunes con el paso del tiempo entre la gente normal puesto que el autobús es un medio de transporte comúnmente utilizado por la ciudadanía.

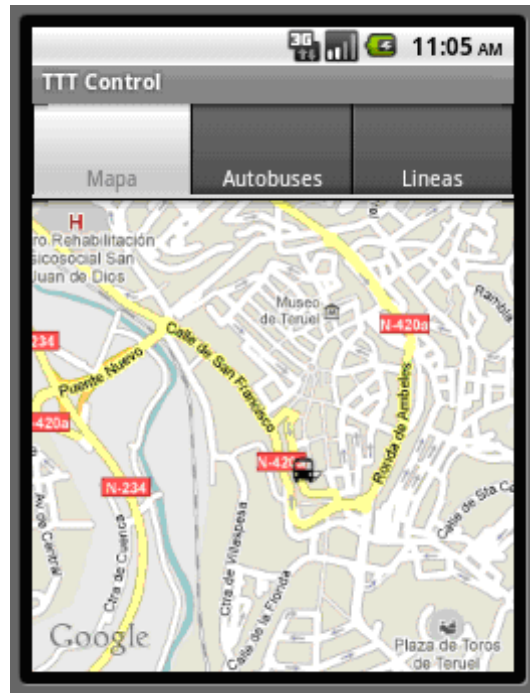
A pesar de todo ello hemos decidido explicar brevemente todos los términos específicos usados en esta aplicación:

- **Autobús:** Se refiere al medio de transporte en si a la máquina que traslada a los pasajeros de un lugar a otro. Cada autobús tiene un número identificador así como una línea asignada durante el trayecto.
- **Línea:** Es el conjunto de paradas por las que debe pasar el autobús para recoger viajeros así como el trayecto entre estas paradas Tiene un número y dos sentidos posibles.
- **Mapa:** Es una representación gráfica de la ciudad en la que se muestran las diferentes calles de la ciudad.
- **Parada:** Se trata de cada uno de los lugares en los que hay situados una marquesina de autobús para que los viajeros lo esperen.
- **Sentido:** Se trata de cada uno de los dos trayectos que puede realizar una línea, habitualmente se corresponden con ida y vuelta.

Así pues quedan definidos los términos más específicos de esta aplicación en cuanto a su lenguaje técnico o vocablos específicos.

2.- Diseño del prototipo no funcional.

Esta pantalla muestra el mapa con los autobuses en circulación a seleccionar por el operario de control.



Si se clicca en un autobús se muestran los detalles del autobús seleccionado en una pantalla diferente, en la siguiente pantalla se puede ver:



Se ha colocado un botón volver en la parte inferior derecha que nos conducirá a la pantalla anterior.

Pulsando en la pestaña de autobuses nos mostrará un listado con todos los autobuses que están circulando en ese momento. Un ejemplo.



Seleccionando cualquier autobús nos mostrará el detalle de autobús visto en la imagen anterior.

Pulsando en la pestaña de líneas nos mostrará un Dialog de cargando, para que el usuario sepa que se está realizando alguna acción, como se muestra en la siguiente imagen:



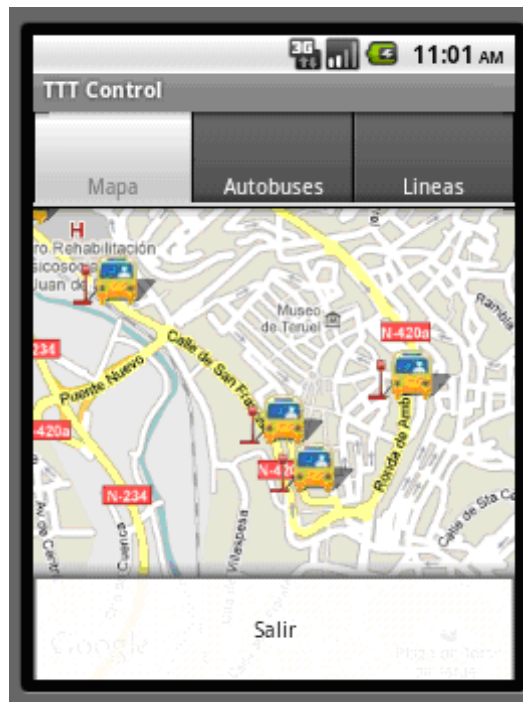
Una vez se ha cargado la información nos mostrará un listado con las líneas y su sentido como se muestra en la imagen inferior:



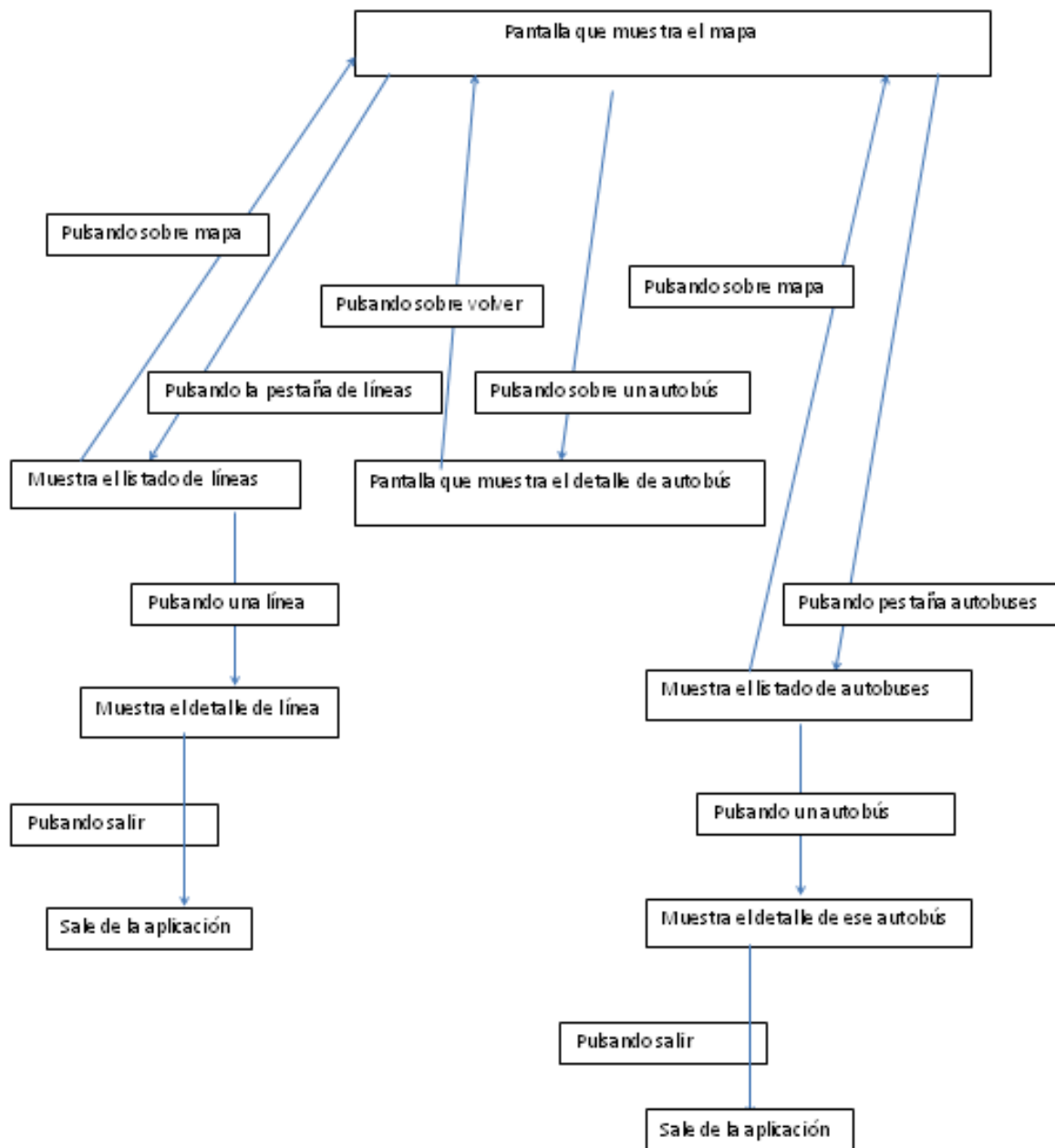
Al seleccionar cualquier línea nos mostrará las paradas que componen esa línea, como se puede ver a continuación:



Pulsando la opción de menú nos aparecerá la opción de salir de la aplicación, si pulsamos esta opción abandonaremos la aplicación, el menú se verá así:



3.- Árbol de pantallas de nuestra aplicación.



4.- Requerimientos para su uso e instalación.

Para la instalación y posterior ejecución de la aplicación se adjunta un fichero ejecutable Android con extensión .apk que contiene la aplicación además de los códigos fuentes necesarios para la aplicación.

Existen dos variantes para su ejecución:

Si se posee un terminal Android: solo se necesita descargar el archivo .apk en el terminal e instalarlo y ejecutarlo normalmente.

Si no se posee un terminal Android:

Se necesitara descargar e instalar el entorno de desarrollo eclipse de la siguiente url:

<http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/heliossr2>

Se necesitara descargar e instalar el plugin de Android de la siguiente url:

<http://developer.android.com/sdk/installing.html>

Se deberá acoplar el plugin al eclipse como se enseña en el siguiente tutorial:

<http://developer.android.com/sdk/eclipse-adt.html#installing>

Se deberá abrir el proyecto en eclipse y proceder a su ejecución para lo cual habrá que definir un emulador con una versión de Android 2.1 o superior con Google APIs. (Es posible que surjan problemas a la hora de mostrar los mapas porque es necesario obtener una clave de Google para usar mapas que se encuentra en la siguiente url:

<http://wiki.androidstartup.com/proyectos/androidnerddinner/gui/api-key>)

A la hora de la instalación solo se puede tener instalada una y no las tres debido a que hemos considerado que es mejor no tener todas las aplicaciones el mismo móvil para mayor seguridad.

7. VTP

7.1. Introducción

El proyecto TTT es una implementación de una aplicación para usuarios que desean conocer el tiempo que falta para la llegada de su autobús urbano y para consulta de las diferentes líneas que circulan por la ciudad. La aplicación será ejecutada en la plataforma electrónica web y Android para dispositivos móviles.

Existe una comunicación entre el usuario y el servidor a través de una conexión TCP/IP, la cual será transparente al usuario. Los autobuses serán los encargados de ir mandando al servidor su localización y toda la información necesaria para que los usuarios puedan obtener sus consultas reales. El usuario desconocerá como es el funcionamiento interno del proyecto, él únicamente accederá a nuestra aplicación y realizará diferentes consultas a un servidor sobre líneas, horarios... y el servidor será el encargado de enviar la información a los usuarios que la soliciten.

7.2. Primeras pruebas

Las primeras pruebas a realizar serán las relacionadas en la conexión entre el cliente y el servidor. Comunicación entre ambos, envío de peticiones y solicitudes, gestión de las mismas y envío de la información solicitada.

Una vez se compruebe que estas primeras pruebas son correctas, se harán pruebas para comprobar el correcto funcionamiento de todos y cada uno de los casos de uso de nuestras diferentes aplicaciones.

7.3. Planificación de pruebas

El desarrollo de las diferentes pruebas que se llevarán a cabo para nuestro proyecto comenzarán a finales de Marzo de 2011, y se irán desarrollando a medida que haya código funcional. Finalmente se hará un conjunto de pruebas finales las cuales comenzarán según la planificación la segunda semana de Abril. Las pruebas se realizarán en la máquina virtual de Android ya que ninguno de los integrantes del grupo disponemos de un dispositivo con dicho sistema operativo.

7.4. Pruebas que se van a realizar para la aplicación viajero

Las diferentes pruebas que se van a llevar a cabo para la aplicación viajero se han obtenido realizando un estudio de todos los requisitos de la aplicación viajero. En este documento se refleja para un determinado requisito las pruebas que se deberán realizar para comprobar el correcto funcionamiento de cada uno de los diferentes prototipos.

4.2.1 General

REQ_VIA_01_01: Cargar las líneas en pantalla.

REQ_VIA_01_01: Si el número de líneas es superior a la pantalla se debe obtener una barra de desplazamiento.

REQ_VIA_01_01: Selección de una línea.

4.2.3 Vista detalle de línea

REQ_VIA_02_02: Colocar parada en el mapa.

REQ_VIA_02_02: Pulsar en parada para ver detalle de la parada.

4.2.4 Vista detalle de parada

REQ_VIA_03_01: Actualizar la información de las líneas con su tiempo.

REQ_VIA_03_02: Pulsar en una línea y ver detalle de la línea.

REQ_VIA_03_03: Introducir favoritos a lista.

4.2.5 Vista de mapa

REQ_VIA_04_01: Mostrar paradas en mapa

REQ_VIA_04_03: Centrar mapa en ubicación actual.

4.2.6 Favoritos

REQ_VIA_05_04: Al pulsar en una parada de la lista de favoritos se ve su detalle de parada.

7.5. Necesidades para realizar las pruebas

Para realizar las diferentes pruebas se necesitara hacer un hipotético entorno como el que sería usado por los usuarios de nuestro sistema. Para ello usaremos el SDK de la máquina virtual de Android sobre un entorno de desarrollo Eclipse.