

BLOQUE III

Realización de la aplicación, pruebas y resultados

Capítulo 7

Descripción de la aplicación

En el siguiente capítulo se explica la forma en que se ha implementado la funcionalidad de la que debe disponer el terminal para poder detectar al inhibidor. Para ello, primero se realiza un análisis sobre la elección de las herramientas que se han usado para el desarrollo de la aplicación. Segundo, se describe las funciones programadas y las clases más importantes.

7.1 Análisis de alternativas de implementación

Tal y como se concluyó en el Capítulo cuarto, para el diseño de esta aplicación se ha optado por el sistema operativo Symbian, principalmente por ser el más usado entre los smartphones y tener una variada y extensa colección de bibliotecas. Gracias a estas bibliotecas se ha podido acceder a las funcionalidades que se necesitan en este proyecto. Además de esto, al ser un sistema operativo muy extendido, existen importantes foros web para el desarrollo de aplicaciones, que facilitan significativamente la comprensión y el diseño de las aplicaciones.

También se ha optado especialmente por el kit de desarrollo de software de la Serie S60 de Nokia, como se ha visto en la sección 6.5 del Capítulo sexto. Con este SDK se permite acceder a muchas de las funciones de los teléfonos móviles, como ha sido acceder a los valores de la intensidad recibida en los teléfonos y también nos permite realizar llamadas telefónicas.

La serie S60 es el SDK más extendido entre los teléfonos móviles con Symbian y se encuentra respaldada por Nokia, que es la principal empresa propietaria de Symbian. La serie UIQ también se ha extendido mucho, pero en la actualidad parece estar en declive.

Para la programación se ha usado el entorno de desarrollo Carbide v.2 C++, que es una herramienta gratuita, y como lenguaje de programación Symbian C++. Esta versión modificada de C++ es el lenguaje nativo de Symbian y permite prolongar la duración de la memoria y ahorrar batería. Siendo una de sus principales cualidades su gran acceso a las funcionalidades del sistema operativo.

Frente a Java se puede destacar que C++ es más rápido en la compilación que Java y consigue un mejor aprovechamiento de los recursos y capacidades del sistema operativo Symbian. Con C++ se encuentran accesibles muchas de las características de los smartphones, disponiendo de librerías para manejar cada una de estas, ya que se proporcionan un conjunto extenso de APIs que ofrecen acceso a todas las funcionalidades de los smartphone.

Todas estas elecciones y sus justificaciones también se han ido aportando en los capítulos anteriores. Para acceder al código se pueden mirar los anexos al proyecto.

7.2 Funcionamiento de la aplicación

El programa diseñado consiste en una aplicación que controla la intensidad de la señal GSM recibida en el teléfono desde la estación base. Cuando la aplicación detecta que la intensidad de la señal es nula o no se puede medir se realizan varias llamadas de voz a un número de teléfono que se ha programado, aunque estas llamadas no se podrán terminar por la ausencia de cobertura.

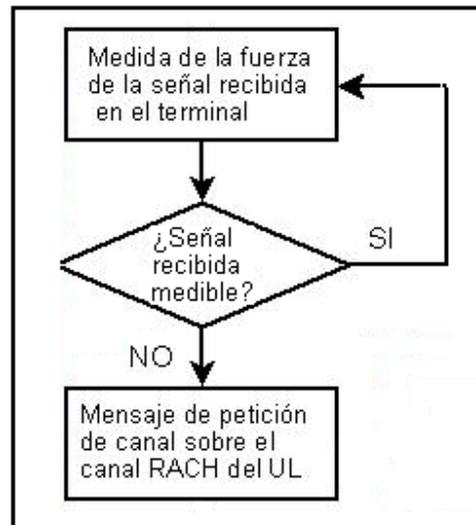


Figura 29. Diagrama de la aplicación.

Una vez instalada correctamente la aplicación en el terminal, se podrá iniciar a través de unos de los menús de su interfaz de usuario. Pulsando sobre "Cheking Signal" se lanza la aplicación. Se mostrará por pantalla el valor de la intensidad de la señal, volviéndose a mostrar cada vez que cambie, siempre y cuando la señal sea medible. En este caso, cuando la señal no se pueda medir, se realizan múltiples intentos de llamadas a un número programado, con el fin de que el receptor de seguimiento pueda detectar un funcionamiento anormal del terminal.

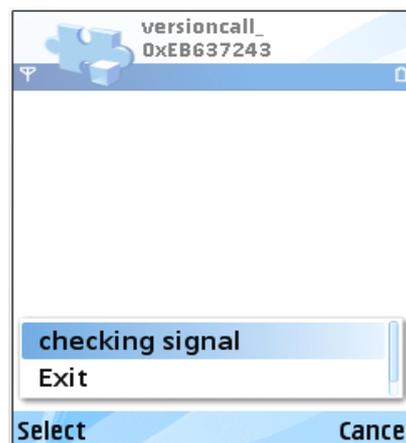


Figura 30. Visualización del menú de la aplicación en el emulador.

Cuando se hayan finalizado las llamadas, la aplicación vuelve al estado inicial, comprobando la fuerza de la señal recibida y volviendo a llamar en caso de ser necesario.

7.3 Clase CTelephony

Para el desarrollo de la aplicación se ha utilizado la API Telephony que se encuentra disponible en la librería `Etel3rdParty.h` del SDK 3rd edición de S60.

La clase `CTelephony` provee de una interfaz para el terminal del sistema de telefonía. Ofrece dos servicios:

- 1- Se puede encontrar información sobre el teléfono, como información de red.
- 2- Se puede marcar, responder y controlar llamadas de voz. No se puede usar para llamadas de datos o de fax.

Información sobre el teléfono

Se puede agrupar dicha información de la siguiente forma:

- Teléfono, batería e información de abonado: puede informarse sobre la marca, modelo y número de serie del teléfono. También se puede averiguar el estado de la batería y sobre el estado flight mode.
- Información de línea y de llamada: estado de las llamadas de voz en proceso.
- Información de red: se puede obtener información acerca de la red a la que se encuentra actualmente conectado, incluyendo el nombre, el estado y la fuerza de la señal.
- Servicios de información de configuración: información sobre el desvío de llamadas, bloqueo de llamadas, llamada en espera e identificación de llamadas.

Pudiéndose acceder a la información de dos maneras:

- Se puede ver el valor actual, como la batería, la fuerza de la señal, o si es una llamada que se está realizando actualmente.
- Se puede solicitar una notificación cuando la información cambia.

Muchas llamadas a `CTelephony` son asíncronas. Por ello, se necesita conocer los objetos activos, para poder manejarlas con facilidad, como ya se vio en

capítulos anteriores. En particular, habrá que asegurarse de que el objeto activo ha comenzado antes de usar la clase CTelephony.

CTelephony es una API pública que está disponible para su uso. También es la forma más sencilla que tiene Symbian OS de acceder al sistema de telefonía, pero proporciona servicios limitados. Algunos fabricantes de teléfonos Symbian pueden acceder a servicios más avanzados que no están disponibles para cualquiera que desee desarrollar una aplicación. Sólo podrán acceder a estos servicios avanzados los titulares de licencias de Symbian.

7.4 Fuerza de la señal

Para controlar la intensidad de la señal obtenida en el teléfono móvil procedente de la estación base, se tiene que acceder a la información de red en la clase CTelephony.

Al usar la función GetSignalStrength la fuerza de la señal es devuelta en un paquete que contiene dos variables:

- Fuerza de la señal medida en dBm. Un valor de cero indica que no se puede medir.
- Número de barras de señal. Un valor de -1 indica que no está disponible.

Cada vez que la intensidad cambie de valor, el paquete con las dos variables se modifica a su valor actual, esto se puede realizar gracias a la función Notifychange.

7.5 Petición de canal sobre el canal RACH del enlace ascendente

Para hacer la petición del canal se hace un intento de establecimiento de conexión, a través de una llamada de voz. Para realizar esta llamada se usa el servicio de CTelephony para controlar las llamadas de voz. Se podrá especificar el número de teléfono al que se dirige la llamada en los argumentos de la función DialNewCall.

Para facilitar la detección de mensajes Immediate Assignment usando el receptor, como se ha visto en el Capítulo 3 (sección 3.2), se realizan 50 llamadas en un bucle, con una separación entre llamadas de un segundo. De este modo la estación base debe responder con 50 Immediate Assignment correspondientes a todas las llamadas.