

NinjaMessaging 2

Rubén Agudo Santos

17 de mayo de 2014

Resumen

En la segunda versión de NinjaMessaging se ha trabajado con notificaciones push, bases de datos remotas, preferencias, widgets y mapas. En las subsecuentes secciones se explicará detalladamente que se ha hecho.

1. Funcionalidades obligatorias

Aquí se presentarán las funcionalidades mínimas para obtener un 5.

1.1. Notificaciones push

Las notificaciones push solo van a funcionar cuando desde la pantalla principal se pinche en el “+” verde en la esquina superior derecha. Al pinchar ahí se mostrarán todos los números IMEI registrados. Es decir, no existe el concepto de amistad, más bien podría considerarse una sala de chat móvil, en la que una vez te has registrado, ya no hay manera de darse de baja.

Una vez seleccionada una conversación (también aparece tu propio dispositivo por motivos de testeo) puedes escribir lo que sea, y llegará una notificación push al dispositivo que estés escribiendo. La notificación, si es de texto plano, al pinchar sobre ella te llevará a la pantalla principal, y NO se mostrará el texto que te ha escrito, ese texto solo será visible a través de la propia notificación.

También se ha usado la clase `SharedPreferences` para guardar el *regid* actual. Idealmente, habría que crear un mecanismo en el cual si Google cambia el *regid* a tu dispositivo pueda actualizarse automáticamente. Lo mas sencillo sería cambiar la versión de la aplicación, y en la base de datos remota, en la tabla registro, guardar como clave primaria el usuario y la versión de la aplicación.

1.2. Bases de datos remotas

La base de datos remota consiste únicamente de una tabla que se compone de dos campos, usuario, que es el IMEI del dispositivo móvil y el *regid*, que es el *registration id* proporcionado por google.

Esa tabla se utiliza para poder enviar mensajes push a través de GCM y curl

Los datos que se obtienen son unicamente los IMEI, que se muestran en la pantalla de Contacts. Es una simple *SELECT* para mostrar todos los usuarios del sistema.

La *INSERT* que se realiza, es para guardar los usuarios registrados, con su numero IMEI + regid.

1.3. Widgets

Se ha creado un widget muy sencillo, con una actividad de configuración, que al lanzarlo, debes introducir el nombre del usuario del que quieres ver el último mensaje.

Una vez has puesto el usuario, se mostrará automáticamente el último mensaje de esa conversación, ya sea tuyo o suyo. El Widget contiene un botón que te permite actualizar el widget y comprobar si el último mensaje ha cambiado.

2. Funcionalidades optativas

Aquí se presentarán las funcionalidades que se utilizan para subir la nota del proyecto.

2.1. Preferencias

En el documento donde se especificaban los requisitos mínimos, se planteaba el uso de preferencias que tuviesen algún tipo de efecto en el funcionamiento del programa.

En mi caso se ha decidido que las preferencias tengan un *checkbox* que inicia o detiene el servicio que notifica cada 15 segundos de que tienes un nuevo mensaje. Como el cambio ha de hacerse de manera instantánea y el *polling* no merece la pena, se ha decidido implementar la interfaz *OnSharedPreferenceChangeListener* que contiene el método *onSharedPreferenceChanged(SharedPreferences sp, String key)* en el que te indica que clave ha cambiado, para que se pueda actuar en consecuencia.

También es importante comentar que se ha utilizado un *PreferenceFragment* embebido dentro de una *Activity* normal, tal y como recomienda Google.

Para acceder a las preferencias, se hace desde la pantalla principal, dando al menú de opciones, y pinchando en preferencias.

2.2. Mapas

También se han implementado los mapas de google, para enviar a un contacto nuestra última posición conocida. Para ello se ha adaptado el laboratorio 7.

Para conseguir el efecto de enviar la localización, lo que se ha hecho es que a través de GCM se envían distintos datos, entre ellos un flag *isMap* en el que si es una localización, el intent de la notificación te lleva a la actividad del mapa,

centrandose en la posición enviada y añadiendo un marker para darle mayor vistosidad al conjunto.

La única manera de entrar al mapa es recibiendo una notificación de mapa.