|  |  |  |  |
| --- | --- | --- | --- |
| **Alumno(s):** | Daniel Francisco Franco Ortiz | **Fecha:** |  |

|  |
| --- |
| OBJETIVO |
| Crear una aplicación que permita al usuario las opciones de bloquear y desbloquear contactos telefónicos, así como implementar un servicio para la eliminación del historial de llamadas y notificar al usuario cuando se bloqueó una llamada. |

|  |
| --- |
| MARCO TEÓRICO |
| *Un receptor de difusión (receptor corto) es un componente de Android que le permite registrarse para eventos de sistema o aplicación. Todos los receptores registrados para un evento son notificados por el tiempo de ejecución de Android una vez que ocurre este evento.* |

|  |
| --- |
| DESARROLLO |
| *Para esta práctica se utilizará un archivo aidl el cual utilizará para manipular las llamadas, el archivo se creará en la carpeta aidl.*    *Dentro del archivo ITelephony que se creo anteriormente se colocaran los métodos que ayudaran a la manipulación de las llamadas.*    *En el archivo manifest se colocarán los permisos necesarios que requerirá la aplicación para la manipulación de las llamadas y la consulta de los contactos del teléfono.*    *Se creará una clase que extenderá de la clase ArrayAdapter la cual se utilizara para representar la estructura de los renglones de la lista en la cual se almacenaran los contactos que el usuario desea tener bloqueados, esta clase ara referencia a un layout el cual tendrá los campos que se utilizaran para guardar la información de los contactos bloqueados.*    *Se creara una clase que herede de la clase ListActivity, la cual representara la lista de elementos bloqueados., esta clase tendrá la función onListItemSelect() que permitirá eliminar a los contactos de la lista de bloqueados una vez que el usuario lo desee.*    *Se procederá a crear una la clase ICReceiver que permitirá manipular las llamadas al dispositivo mediante la clase ITelephony, esta clase tendrá varios métodos como el método onReceive en el cual evaluara las llamadas recibidas y verificará los números en la lista de números bloqueados para terminar la llamada en caso de que el numero este en la lista de bloqueo.*    *La clase ICReceiver también tendrá el método showNotification(), el cual se utilizara para enviar notificaciones al usuario cada vez que el sistema bloquee una llamada.*    *En la actividad principal en el método onCreate se iniciará el servicio que consultar la base de datos y también se solicitaran a usuario los permisos.*  *La Actividad principal constara de varios métodos por ejemplo que agregara un nuevo numero a la lista de bloqueo, este método utilizara un piker de Android para permitir al usuario buscar entre los contactos los que desea bloquear.*    *En el método onActivityResult se recibirá el contacto seleccionado por el usuario y se agregara a la base de datos, si el usuario no selecciono un contacto, la actividad no hara nada.*    *El método showSelectNumber muestra un mensaje al usuario cuando un contacto es agregado a la lista,*  *Los últimos dos métodos representan las funciones de los dos botones en la actividad principal, uno permite agregar un nuevo contacto y el otro muestra una lista con los contactos agregados* |

|  |
| --- |
| CONCLUSIONES |
| La manipulación de aplicaciones nativas como son la aplicación de llamadas o la aplicación de contactos requieren el uso de varios permisos, en el caso de la creación de una lista negra también se debe implementas una clase adll como ITelephony ya que esta facilita en gran medida el control de las llamadas, |

|  |
| --- |
| BIBLIOGRAFÍA |
| <https://blog.previewtechs.com/blocking-call-and-sms-in-android-programmatically> |