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

|  |
| --- |
| OBJETIVO |
| Comprender la funcionalidad de Mapas en los dispositivos Android mediante la creación de una aplicación que permita regresar a un punto especifico desde otra localización. |

|  |
| --- |
| MARCO TEÓRICO |
| La API de Google Maps se integró con los Google Play Services allá por finales de 2012. Este cambio trajo consigo importantes mejoras, como la utilización de mapas vectoriales y mejoras en el sistema de caché, lo que proporcionaba mayor rendimiento, mayor velocidad de carga, y menor consumo de datos.  También llegó con un cambio en la forma en que los desarrolladores interactuaríamos con los mapas, pasando de los antiguos MapActivity y MapView a un nuevo tipo de fragment llamado MapFragment, con las ventajas que conlleva el uso de este tipo de componentes.  Tu aplicación necesita una clave de API para acceder a los servidores de Google Maps. El tipo de clave que necesitas es una clave de API con restricción para Android apps. La clave es gratuita. Puedes usarla con aquellas de tus aplicaciones que llamen a la Google Maps Android API y admitan un número ilimitado de usuarios.  De manera predeterminada, el archivo XML que define el diseño de la aplicación se encuentra en res/layout/activity\_maps.xml.  De manera predeterminada, el archivo Java que define la actividad de mapas lleva el nombre MapsActivity.java.  Con Google Maps Android API v2, puede incrustar mapas en una actividad como un fragmento con un simple fragmento XML. Los nuevos mapas ofrecen características interesantes como mapas en 3D; mapas interiores, satelitales, de terreno e híbridos; mosaicos basados ​​en vectores para el almacenamiento en caché y el dibujo eficiente; transiciones animadas; y mucho más.  Google Maps Android API es parte de la plataforma de servicios de Google Play. Para usar Google Maps, configure el SDK de servicios de Google Play en su proyecto de desarrollo de aplicaciones. |

|  |
| --- |
| DESARROLLO |
| Para sincronizar el proyecto con los servicios de Google Maps, estos te pedirán que crees un proyecto en Google con el cual crearas una llave para poder acceder a sus servicios, para esto Google ofrece ayuda en la página:  <https://developers.google.com/maps/documentation/android-api/start?hl=es-419>  Al crear una aplicación de Android se selecciona directamente la opción al momento de crear el proyecto.  *Creamos una clase la cual nos ayudara a trabajas con los objetos que se encuentran en formato JSON, esta clase constara de dos métodos:*  *El método parse se encargará de decodificar un objeto JSON compuesto por una gran cantidad de JSONs en arreglo con los objetos separados, cada objeto tiene dos valores de tipo String.*      *En la clase principal en el método onCreate se evalua si la apliacion tiene los permiso, ya que ocupa el permiso para saber la localización de dispositivo.*    *El método onMapReady se ejecuta una vez que el fragmento de mapas se completa, en este método se vuelve a evaluar si la aplicación tiene los permisos necesarios, si los tiene el mapa se enfoca en la ubicación del usuario y se crea la localización del destino (la casa del usuario), la posición del destino se diferencia mediante el uso de un marcador. Este método también se encarga de crear la URL con la cual se consultará al servicio de mapas y descargará la ruta recomendada hacia la ubicación destino.*    *El método download\_Url retorna una cadena con el resultado de una consulta Http usanto un objeto de tipo HttñConnection.*    *La clase privada DownloadTask se encarga de llamar l método anterior y descargar la información de la consulta realizada.*    *La clase privada ParserTask se encarga de convertir el resultado de la consulta la cual se encuentra en formato JSON a un arreglo de tipo <<string, string>>, para ello utiliza la clase creada en los pasos anteriores.*    *El método onPostExecute se encarga de convertir la lista anterior en polilíneas las cuales son pintadas en el mapa, estas polilíneas representan una serie de puntos que se encuentran entre los puntos de origen (la hubicacion del usuario) y los puetos de desitno (la casa del usuario).* |

|  |
| --- |
| CONCLUSIONES |
| Google Map tiene una gran variedad de ventajas ya que permite al usuario poder localizar un lugar en específico, para implementarlo en Android se debe crear un proyecto en Google y conseguie una llave para poder utilizarlo, esto permite mantener un nivel de seguridad al trabajar con Android maps ya que las llaves solo funcionan con sus respectivos proyectos.  Google maps implementa el formato JSON lo que proporciona una gran ventaja a los desarrolladores ya que es un formato ligero y fácil de entender. |

|  |
| --- |
| BIBLIOGRAFÍA |
| <https://developer.android.com/training/maps/index.html>  <http://www.sgoliver.net/blog/mapas-en-android-google-maps-android-api-1/>  <https://developers.google.com/maps/documentation/android-api/start?hl=es-419> |