**UNIVERSIDAD DE CUENCA**

Programación 3 G#1

PROYECTO MAGIC ROUTE

William Sánchez

24 de Mayo de 2017

**RESUMEN**

El presente documento trata sobre el desarrollo del proyecto MAGIC ROUTE, proyecto creado para mejorar el turismo en la ciudad de Cuenca. El Documento explicará todo el proceso desarrollo y creación de la aplicación, incluyendo las herramientas usadas, métodos de programación como programación en capas, uso de APIS entre otras más. También se explicara la manera de implementar lo ya mencionado, además de todos los servicios que ofrece el programa y como usarlos.

**INTRODUCCIÓN.**

Cuenca es una de las ciudades turísticas mas importantes del Ecuador, su crecimiento en el área de turismo ha aumentado mucho en los últimos años, es por eso que dia a dia se busca desarrollar aplicaciónes que faciliten y provean al turista una vista previa de la ciudad con todos los lugares turísticos que contiene para que de esta manera pueda elegir mejor que detinos visitar. MAGIC ROUTTE es uno de ellos y surge de la necesidad de proveer paquetes turísticos predefinidos junto con rutas cada una con sus respectivos lugares turísticos.

Magic Route es una aplicación con una interfaz muy amigable y de fácil uso la cual provee rutas turísticas. El usuario podrá elegir entre las diversas rutas y mediante un mapa podrá ubicar los lugares que contiene la misma, además podrá visualizar imágenes de cada lugar mediante un visor de imágenes.

El programa incluye Street View para que a medida que avance por la ruta elegida pueda apreciar como son las calles en 360 hasta llegar a cada uno de los destinos, también incluye un botón que le permitirá al usuario apreciar en el navegador la ruta en Google Maps, además de un botón que le enviara a un video en la plataforma Youtube relacionado con el lugar turístico que el usuario escoja.

Magic Route busca faciltar al turista la elección de los lugares que quiera visitar ya que muchos son turistas de corto tiempo de tal forma que no podrán visitar todos los lugares.

Materiales y métodos.

Para la construcción de este programa se usó una de las herramientas más usadas para la creación de aplicaciones, la programación por capas.

Programación por capas: Se refiere a un estilo de programación que tiene como objetivo separar la lógica de diseño de la lógica de negocios. Una de las ventajas que podemos destacar sobre este estilo es que el desarrollo del software se puede llevar a cabo en varios tipos de niveles, así, cuando suceda algún cambio solo nos iremos sobre el nivel requerido.

* Capa de presentación: Se refiere a la presentación del programa frente al usuario, esta presentación debe cumplir su propósito con el usuario final, una presentación fácil de usar y amigable. También las interfaces deben ser consistentes con la información dentro del software (Por ejemplo; en los formularios no debe haber más que lo necesario), tomar en cuenta los requerimientos del usuario, la capa de presentación va de la mano con capa de la lógica de negocio.
* Capa de lógica de negocio: En esta capa es donde se encuentran los programas que son ejecutados, recibe las peticiones del usuario y posteriormente envía las respuestas tras el proceso. Esta capa es muy importantes pues es donde se establecen todas aquellas reglas que se tendrán que cumplir, decía anteriormente que la capa de presentación tiene comunicación con la capa de lógica de negocio ya que se tienen que comunicar para recibir las solicitudes y presentar los resultados.
* Capa de datos: Esta capa es la que se encarga de hacer las transacciones con la base de datos y con otros sistemas para descargar o insertar información al sistema. La consistencia en los datos es sumamente importante, es decir, los datos que se ingresan o insertan deben ser precisos y consientes. Aquí definimos las consultas que vamos a realizar en la base de datos, o consultas para reporteo. La comunicación de esta capa con la capa de lógica de negocio se refiere a que la capa de datos es la que le enviara información a la capa de negocio para que sea procesada e ingresada en objetos según sea necesario (encapsulamiento).

Java: lenguaje de programación orientado a objetos y en su última versión java 8 permite además la posibilidad de programar de manera funcional.

Netbeans: Es un entorno de desarrollo muy popular para el desarrollo de aplicaciones en Java y otros lenguajes de programación.

API de Google Maps: API son un conjunto de comandos, funciones y protocolos informáticos que permiten a los desarrolladores crear programas específicos para ciertos sistemas operativos. Las API simplifican en gran medida el trabajo de un creador de programas, ya que no tiene que escribir códigos desde cero. Estas permiten al informático usar funciones predefinidas para interactuar con el sistema operativo o con otro programa. En este caso la API de google maps no permitirá el uso de funciones como la de crear un mapa, trazar rutas etc.

MAGIC ROUTE

¿Qué es Magic Route?

Magic Route es una aplicación para computadoras el cual fue diseñado con el fin de ayudar al turista en una mejor selección de lugares a visitar. Brindándole la facilidad de visualizar rutas turísticas y sus respectivos lugares junto con imágenes, videos e información sobre los mismos.

Desarrollo de Magic Route.

Para la creación del programa se usó el lenguaje de programación java junto con el ID de netbeans.

Se aplicó la programación por capas debido a que brinda una mejor manera de crear el programa y también porque el programa usa en parte la programación orientada a objetos.

Las capas usadas fueron: **ver Fig1**.

Capa Lógica:

Se crearon clases para la creación de algunos objetos a usar.

StaticMaps: Nos sirve para crear objetos de tipo Image para implementar el mapa de google maps.

Archivo: Esta clase lee los archivos txt que contiene todas las latitudes y longitudes de las diferentes rutas.

Ruta: Devolverá una matriz con el contenido del archivo leído.

Capa de Presentación:

Interfaz1: Interfaz gráfica para el ingreso al programa.

Prueba1. Interfaz que contiene todos los servicios de la aplicación.

Capa de Datos:

Datos: Contiene los .txt con las rutas.

Imágenes: Contiene las imágenes para la interfaz principal.

ImagenesLugares: Contiene todas las imágenes de los diferentes lugares turísticos.

Uso de la API de Google Maps.

Para Street View.

StreerView Image API nos permite insertar una imagen panorámica estática de una parte de la ciudad en específico. Hay que tener en cuenta que es una imagen estática en la cual no podemos interactuar (directamente).

METODOS Y ATRIBUTOS DE LA CLASE STREETVIEW

Atributo String URLRoot La solicitud de la Imagen Street View se realiza mediante una URL de HTTP. La imagen se especifica usando parámetros de solicitud la cual se colocarán a continuación del URL separados por “&”. Para el correcto manejo del API se debe colocar dos parámetros obligatorios, el cual es, la localización en coordenadas del sitio especifico, el cual queremos acceder y el tamaño de salida de la imagen en píxeles, el tamaño se especifica como [width]\*[height]. **Ver Fig2.**

Latitud y Longitud: Del lugar que se quiere proyectar.

Dimensión: tendrá dos atributos enteros que serán las dimensiones del objeto. El constructor nos recibirá como parámetros el ancho y el largo para realizar su instanciación. Por lo tanto, lo que se debe colocar en el parámetro del método mencionado es una instancia de la clase Dimensión, colocando el ancho y el largo del objeto.

Grados: Nos mostrara la imagen en 360 dependiendo del valor entre (0 y 360).

STATICMAPS

El GOOGLE STATIC MAPS API nos permite agregar una imagen estática del servidor de GOOGLE MAPS en nuestra aplicación. El servicio que nos provee GOOGLE STATIC MAPS API nos permite crear un mapa en función de los parámetros de la dirección URL que hayamos enviado a través de una solicitud HTTP.

METODOS Y ATRIBUTOS DE LA CLASE STATICMAPS

Atributo String URLRoot Como hemos mencionado anteriormente en el apartado STREETVIEW, debemos realizar la solicitud de la imagen mediante una URL de HTTP. Así también deberemos especificar ciertos parámetros obligatorios, teniendo en cuenta las mismas consideraciones de separación de cada parámetro con un “&”.

**Ver Fig3.**

String centerAddress: Deberemos colocar la ubicación central del mapa. Podremos insertar cualquiera de la siguiente información para su correcto funcionamiento:

Dirección (Nombre, Ciudad , País) o Longitud y Latitud.

Int zoom: Nos permite elegir el nivel de acercamiento que tengamos en nuestro mapa. Como máximo valor entre (0 y 21)

Dimensión size: Tamaño de la Imagen del Mapa.

Int scale: Este valor nos indica el número de píxeles que se muestra. Tiene un valor muy parecido al booleano ya que solo admite dos valores (1 y 2). El valor predeterminado es “1”.

Format format: GOOGLE STATIC MAPS API nos permite retorna una imagen estática de diferentes formatos (GIF, PNG, JPG) en respuesta a la solicitud HTTP que enviemos.

Maptype maptype: Este parámetro definiremos el tipo de mapa a mostrar. En la aplicación se ha usado el tipo RoadMap.

* Roadmap: Mapa de carreteras, la que muestra GOOGLE MAPS por defecto (La más sencilla).
* Satellite: Mapa del lugar obtenida mediante satélite.
* Hybrid: Nos devolverá un mapa de relieve físico en la cual nos mostrará relieve y vegetación.
* Terrain: Combina en una misma imagen el mapa satelital y la de carreteras

Servicios y Uso de Magic Route.

Magic Route cuenta con varios servicios en esta sección se detallara cada uno de ellos.

Al Abrir el programa se abre la primera ventana. **Ver Fig4**. Esta venatana cuenta con dos botones, uno para permitir el acceso al programa principal y otro para cerrar el programa.

Una vez ingresado a la ventana con el programa principal el usuario podrá ver en si toda la interfaz con lo que ofrece la aplicación. **Ver Fig5**. En los que encontrará:

* Un panel de Rutas
* Un panel de lugares turísticos
* Los diferentes botones de interacción (Ver Video Youtube, Simular Recorrido, Anterior y Siguiente Punto, Botones de Zoom, Ver Ruta, Anterior y Siguiente para el visor de Imágenes).
* Un visor de Imágenes
* Un mapa con los marcadores de los lugares
* Un label (Etiqueta) con Street View.

Panel de Rutas: Este panel contiene las rutas disponibles de los cuales el turista podrá elegir uno. Las rutas son cuatro (Ruta de Catedrales eh iglesias, Ruta de Parques, Ruta de Museos y una Ruta especial con los lugares más sobresalientes de Cuenca).

Una vez que el usuario elija una ruta podrá dar clic sobre el Label donde esta el mapa y de esta manera se generará un mapa con los diferentes lugares que contiene la ruta.

Panel de Lugares: Este panel se llenará de acuerdo a la elección de la ruta en el panel antes mencionado, es decir si se selecciona la ruta de Catedrales- Iglesias este pane mostrará cuales con las catedrales e iglesias disponibles.

Label de Mapa: En este Label se mostrará el mapa con los marcadores de lugares además de servir para mostrar como se debe mover el turista para recorer todos los lugares de la ruta.

Botón Siguiente punto: Este botón sirve para indicar por donde se debe mover el turista para llegar al siguiente lugar turístico.

Botón Anterior punto: Este botón sirve para que el usuario retroceda una ubicación y pueda regresar hacia el lugar anterior.

Label Street View:

Label de Street View