



Universidad:

Instituto tecnológico de Santo Domingo (INTEC)

Asignatura:

Programacion Avanzada

Profesor:

HAROLD LAWRENCE MARZAN MERCADO

Sección:

01

Título:

Documentacion del proyecto

Estudiante:

Luis José Domínguez

Matrícula:

12-0263

Fecha de entrega:

22/01/15

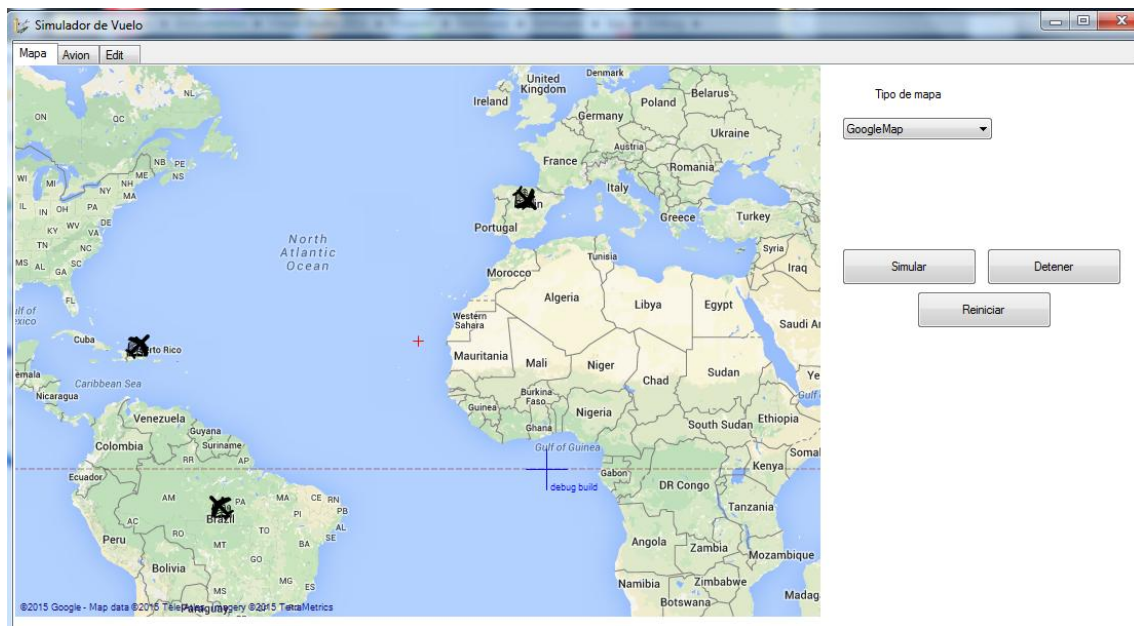


Introduccion

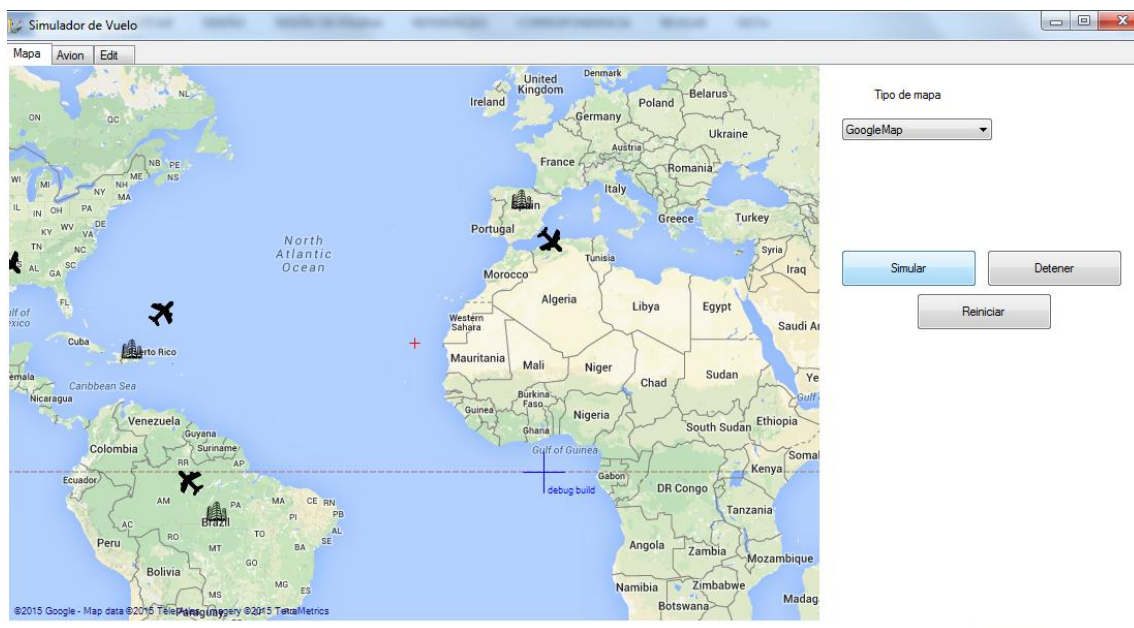
El objetivo de la aplicacion SimVuelo(Simulador de Vuelo) es generar varios aviones en diferentes aeropuertos y ver como eligen un destino y luego van hacia el en el momento que llegue el avión a su destino desde su posición inicial se creara un avión que valla a otro destino(dado que es random se puede repetir el mismo pero si con datos diferentes).

Forma de usar la aplicación

Luego de que se inicie el programa puede hacer varias cosas entre ellas:



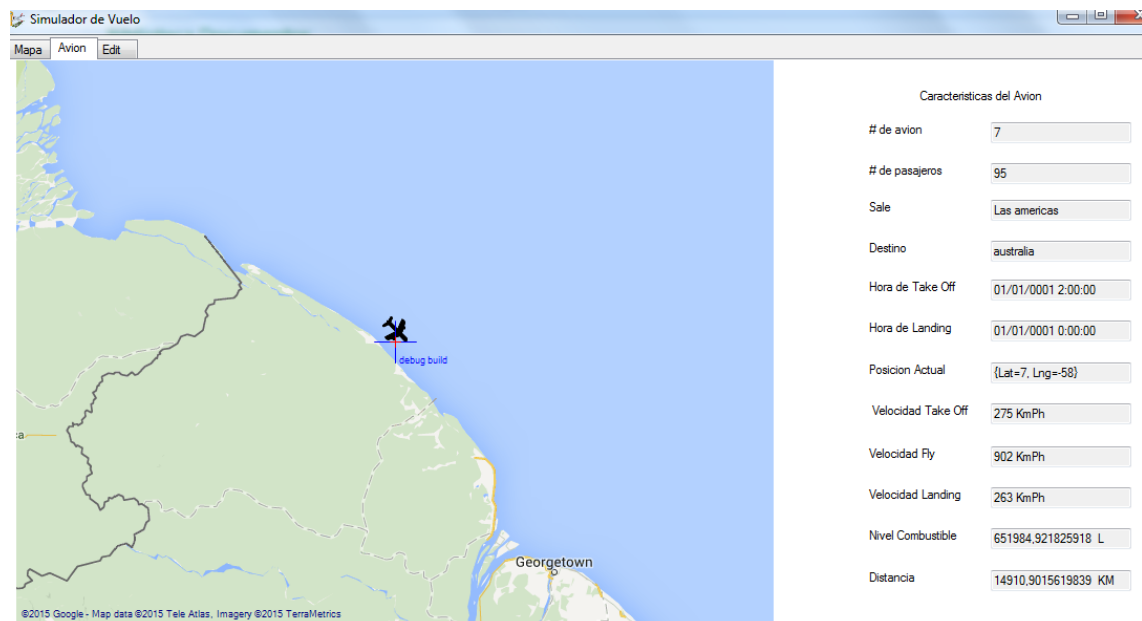
Iniciar la simulación, donde empezaran los aviones a moverse hasta que se presione el botón de detener o el botón de reiniciar la aplicación.



Nota: si en esta vista los aviones parecen que van chocar y no aparece el mensaje requerde que no es la distancia verdadera y para ver la real haga zoom en el mapa

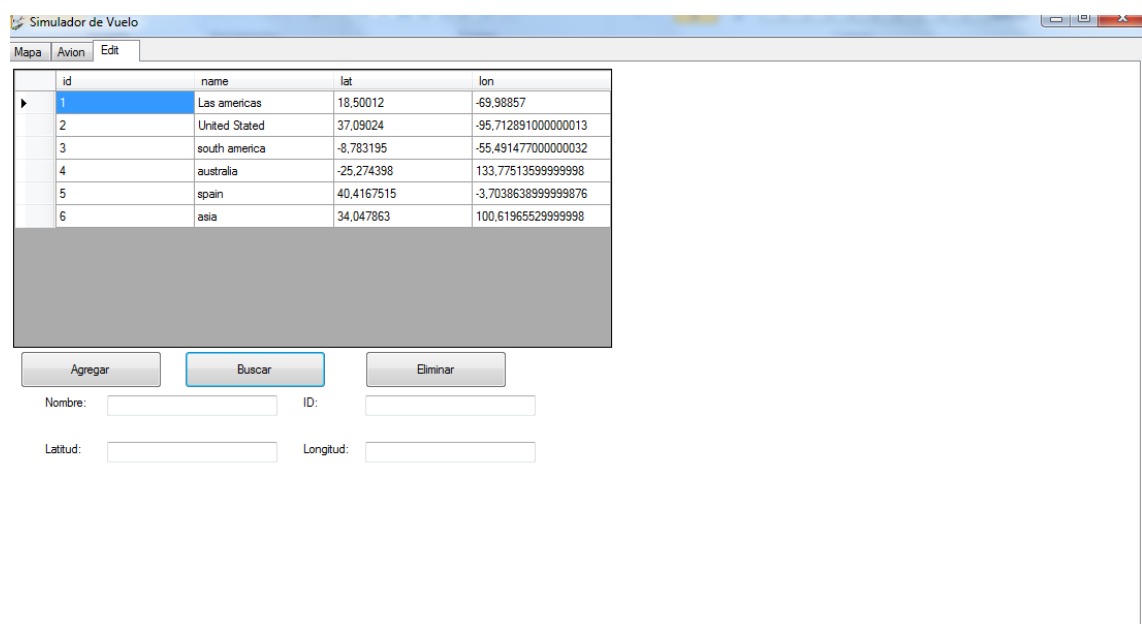
Luego de que se inicia la aplicación para ver la posición de uno de los aviones puede colocar el mouse encima de uno de ellos y por medio de un tooltip se le indicara la posición actual del avión deseado.

Para ver toda la información de uno de los aviones debe de dar click encima de uno de ellos y luego se le enviara a una pestaña donde apreciara el avión enfocado en un mapa y toda la información disponible a su derecha.



Antes de iniciar la aplicación puede cambiar la vista de mapa a su preferida.

También puede ir a la pestaña edit para agregar, ver o eliminar un aereopuerto.



Calculos

para calcular la ruta primero se valida si la cantidad de pasos de irse por adelante es mayor o menor que irse por atrás y luego se calcula la ruta dependiendo cual se mas corta;

```
f (Math.Abs(lon1) + lat2 < (180 - lon1) + (180 - lon2))  
if (lon1 + Math.Abs(lon2) < (180 - lon1) + (180 - Math.Abs(lon2)))
```

Para calcular la ruta hacia adelante se valida si la inicia es mayor o menor y luego se le suma o se le resta para llevarla a la de destino. Estas se guardan en una lista solo para latitud y otra para la longitud y luego usando el método del merge se unen en una sola lista de PointLatLng.

```
private void crear_ruta(PointLatLng fin)
```

La mayoría de los datos de los vuelos se generan aleatoriamente pero de en un rango realista y los otros se calculan

```
Random r = new Random();  
num_pasajeros = r.Next(150);  
num_vuelo = r.Next(20);  
hora_takeoff = new DateTime();  
hora_landing = hora_takeoff.AddHours(2);  
VI = r.Next(257, 291);  
VF = r.Next(240, 268);  
VA = r.Next(885, 933);  
altitud = r.Next(28000, 39000);  
combustible = r.Next(200000, 216840);
```

para calcular la distancia de un punto a otra se utiliza una formula que consiste en multiplicar el radio de la tierra por la multiplicación de los senos y cosenos de la latitud y la longitud

```
public double Distance(PointLatLng start, PointLatLng end)  
{  
    double deg2radMultiplier = Math.PI / 180;  
    double lat1 = start.Lat * deg2radMultiplier;  
    double lon1 = start.Lng * deg2radMultiplier;  
    double lat2 = end.Lat * deg2radMultiplier;  
    double lon2 = end.Lng * deg2radMultiplier;  
  
    double radius = 6378.137;  
    double dlon = lon2 - lon1;  
    distancia = Math.Acos(Math.Sin(lat1) * Math.Sin(lat2) +  
Math.Cos(lat1) * Math.Cos(lat2) * Math.Cos(dlon)) * radius;  
    return distancia;  
}
```

Para determinar si existe una colision entre dos aviones se calcula en el timer por cada avión la distancia entre ellos y si la diferencia de la altitud es mayor de un rango se lanza un mensaje y se ajusta las altitudes.