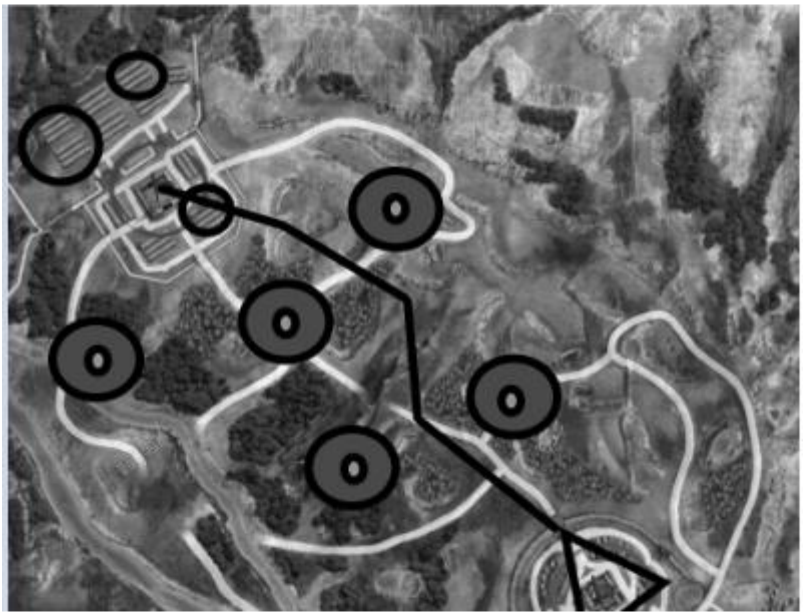
Project

Introduction: Military Simulations

The military air force has contacted your group to create an application for their ground teams to run simulations routes for their planes to scout an enemy position. The client insisted that the teams working on these projects be no bigger than 3 people in a group using GIT to collaborate.

These simulations will help in ensuring the survival of their pilots on these missions. The application must allow the user to add obstacles such as enemy emplacements or RPG squads (see these as obstacles to avoid) to the simulation with a simple drag and drop technique. The application must be reactive to the locations of these obstacles, this means that the route created must be dynamic in nature, if the position of the obstacles changes, then your aircraft should adapt accordingly after making the necessary calculations. Find below an example of how the flight may look, however on your application the plane must be moving on a route. This is just a demonstration of a map that will be discussed in class, if you use this map in your application, you will receive zero for the user interface.



Requirements from client

When designing the algorithm there are a few constraints to keep in mind:

The air force will have one main base camp that will house all the different jet aircrafts. When an aircraft is initially deployed, it will serve as a scout, seeing the different territories inside the enemy camps (Armory, Barracks, Officers Quarters, mess hall, tank depo, hospital etc…). This information should only be unveiled as soon as the aircraft flies over the enemy camp. On route to the enemy camp, the aircraft needs to avoid certain obstacles. These obstacles represent danger for the pilots of the aircrafts.

While these aircraft are on route to the enemy camp or on their way back to base camp, your application needs to consider numerous factors:

- Fuel tank of aircraft. This should be calculated by checking amount of fuel the aircraft had on departure, the speed at which it is travelling and the weight of all the items in inventory that affects the amount of fuel that burns to reach certain speeds and altitudes. The fuel tank should be visually displayed on the application and as the aircraft flies, the visual should adapt to represent it accurately.
- Altitude that the specific aircraft is flying at.
- The speed at which the aircraft travels, from lift off until it arrives back at base camp.
- The specific items in inventory that the aircraft is carrying. If any of the inventory items are used while flying, the inventory should be updated to reflect it, this data should also be live.
- While the aircraft is on route to the enemy camp, it should avoid all obstacles. These obstacles should be seen as life threatening to the pilot flying the aircraft. These obstacles can shoot at the aircraft.

Concerning the enemy camp:

- It needs to be located on the map of the application.
- It should have different sections inside the camp (Armory, Barracks, Officers Quarters, mess hall, tank depo, hospital etc…).
- The aircraft should be able to identify the target that it should aim for, that will deal the most damage.

Concerning the obstacles:

- Obstacles should be able to be placed at any coordinates on the map by the person using your application, before running the simulation.
- After these obstacles have been placed, the aircraft should calculate the shortest route to the enemy camp while still avoiding the obstacles at a safe distance. This should also be considered when the aircraft flies back to base camp.
- These obstacles can shoot up at the aircraft, all obstacles can only shoot a certain height above them, thus if your aircraft can reach an altitude quickly enough to safely fly above the obstacles, then it should, if it shortens the route.

Additional requirements:

- There can only be jet aircrafts for the simulation, however there can be different types of jet aircrafts, differing in top speeds, fuel tank size inventory capacity, types of weapons. You can do further research into this topic if needed.
- The obstacles can fire their ammunition at different heights, this can be avoided by the aircrafts if they have a flight plan higher than the limits that the obstacles can fire to. This means that an aircraft does not have to avoid the obstacle if it cannot be hit by it.
- The encampment can have targets placed within it that the aircraft must try and hit. Some of these targets have a higher priority than others, the aircraft should try to target infrastructure of the enemy camp rather than cause maximum number of casualties.
- If the plane gets diverted by an obstacle, it decreases its chances on hitting its targets directly. This means that the aircraft must try and do the maximum amount of damage to the infrastructure of the encampment without sacrificing itself.

Keep in mind that you should include the following in your application:

- Data Access (DB management)
- Business Logic
- Windows forms
- File Handling
- Custom exceptions
- Demonstrate working with GIT amongst group members.

Your application should also be able to perform the following functionality:

- When a simulation finishes, a window that will display a statistical report of the simulation results must appear.
  - This report must include but is not limited to:
    - Damage analysis (of the enemy camp) and success rating.
    - Description on the actual strike. Time of strike, the location that was targeted and location that was hit.
    - Targets identified, and targets chosen.
    - Obstacles identified on coordinates.
    - Obstacles avoided and reasons why.
    - Aircraft specifications before and after strike. Consider inventory status and fuel status.

Additional Information

- All work must be done on your own. Belgium Campus have software that can **scan for plagiarism** and a student caught doing this will get 0 for this assignment.
- Late assignments will not be accepted; missing the deadline is an automatic 0.