

# **FLIGHT SIMULATION OVER DESERT AIRPORT IN UNITY 3D**

## **Team Members:**

Hubert Kyeremateng-Boateng  
Veronica Boateng

Dr. Sharad Sharma  
Spring 2020

## OVERVIEW

This project is to create a virtual flight simulation which can be used to train amateur pilots and kids can use to play around on how to fly a plane. The project will show first-hand perspective of the controls of the airplane, the landscape (desert airport) below the airplane when it is the air.

## GOALS AND OBJECTIVES

In this project, we worked as a team of two to develop a flight simulator which is a single first-person controller simulator. The simulator mimics a military base in a desert an amateur pilot is training on how to fly a plane.

This plane take-off simulator is intended, or the target audience are novice/amateur pilots who wants to learn how to pilot a plane. Another intended audience are experienced pilots who are new to this specific plane and wants to fly the plane. For both users, the simulator is a good way to initial way to test the plane using the Virtual Reality plane before flying the physical plane.

The application will be used by the military in training their recruit pilots. Also, the application can also be used in the private sector both commercial airlines and non-commercial or private flight instructors in teaching how to fly a plane. They can use this application to teach their students how a plane take's off, land a plane and maneuver around while in flight.

This application is being done to show that Virtual reality can be used to train pilots before they step foot on the physical plane.



Figure 1. A plane in flight

## MODELLING

The Flight take-off simulator environment is a desert military bases. The pilots use the keyboard key codes to fly the plane from the runway and the landing on the runway. There are two runways in the environment and the pilot always takes off from the larger runway but can land the plane on any of the runways.

Since the virtual environment of the simulation will be done on a military base, there will be military jeeps, transport planes, military helicopter, fighter jets, and aircraft hangers on the base.

**Main Building:** - Military base one-level building structures where training preparation and coaching are done before the flight training sessions.

**Runways:-**

The environment has two north-south parallel runways, one large and small runways separated by 7600 feet.

**Obstacles:**

There will be birds flying over the based environment and a Mast located near the base for which player has to avoid.

**Day and Night flight times:**

The player selects which time of day (i.e. daytime or nighttime) to fly the plane.

The main building is comprised of set of objects including:-

**Three Military Men:** - Trainers who have completed flight training and are walking towards the building.

**Military supervisor:** For Security purpose enforcing safety rules.

**Military jeeps, transport planes:** - conveying trainees to the training base.

**Jet Airplanes:** - One jet airplane for the plane take off and plane landing, the other jet air planes that will move next will swing flaps awaiting signals to proceed to fly in due course

**Bounding Box:** - Assigned to jet airplane so that birds don't hit the plane.

**Birds:** - used to show the intelligent behavior and animation.

**Masts:** player to avoid flying close to the mast or danger zone

**Tower:** server as a guide to player and ease of identification of base on landing the jet airplane. ,

**Level Manager:** Used to smoothen the rocky and sandy ground to level the ground for the airport base.

**Description of features simulated:**

- **Perform a Take Off**

As the jet airplane accelerates on the runway, the pilot maintains the aircraft's attitude and its angle of incidence. At the rotation speed, the pilot uses the elevators to pitch-up the airplane where the nose wheel lifts off the ground. When the nose wheel leaves the ground, it increases the angle of attack or the angle at which the wings are cutting into the wind, which consequently increases the plane's lifting ability. When the rapidly developing lift overpowers the airplane's weight, the main wheels of the aircraft leave the ground.

- **Climb the Airplane**

Airspeeds is used during take-off, since there could be obstacles such as birds and a mast that affects the rate of climb, so the earlier the plane is high above the obstacle the safer

Climbing is performed using outside visual references and flight instruments such as the engine power settings and pitch attitudes that will produce normal and best angle conditions of climb. Best angle of climb airspeed is the airspeed where the most excess thrust is available over that required for level flight and results in a steeper climb path, although the plane will take longer to reach the same altitude than it would at best rate of climb. The best angle of climb is used in clearing obstacles (Masts) after takeoff. Best angle of climb is performed at an airspeed that will produce the most altitude gain in a given distance.



- **Practice Turning Maneuvers**

Basic flight maneuvers are adopted during flying including straight-and-level flight, turns, climbs, and descents.

- **Descend the Airplane**

When the jet airplane enters a descent, it changes its flightpath from level to an inclined plane. The pilot knows the power settings and pitch attitudes that will produce the partial power descent and descent at minimum safe airspeed conditions of descent.

- Perform A Landing

Throttle is used to control and adjust altitude since the jet is properly trimmed with flaps, player can manage the approach with throttle and with very little use of elevator though the elevator is available to adjust pitch with authority before the touchdown.

## PROGRAMMING

The simulator was built using Unity3D 2019.3.11f development engine and platform with the programming language used is C# programming language in Microsoft Visual Studio.

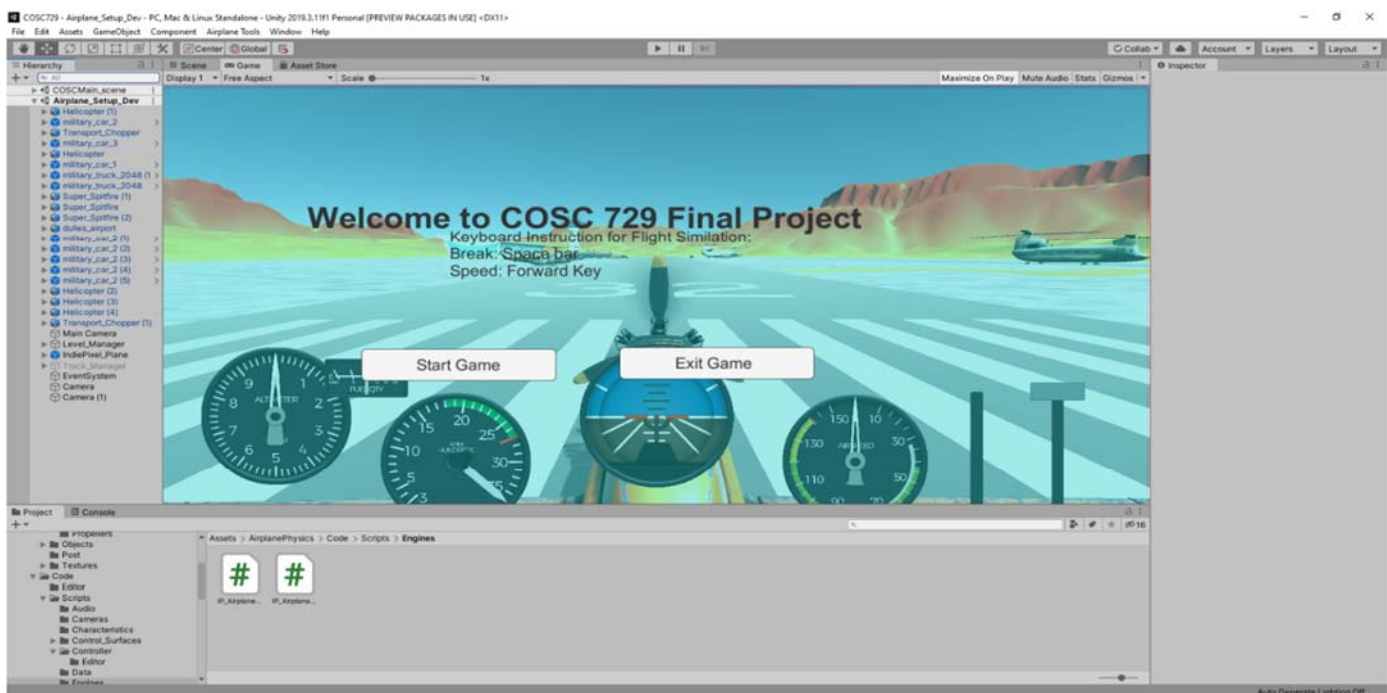


Figure 2: Unity 3D development environment

Also a third-party framework (Indie framework) was used to facilitate the development of the simulator.

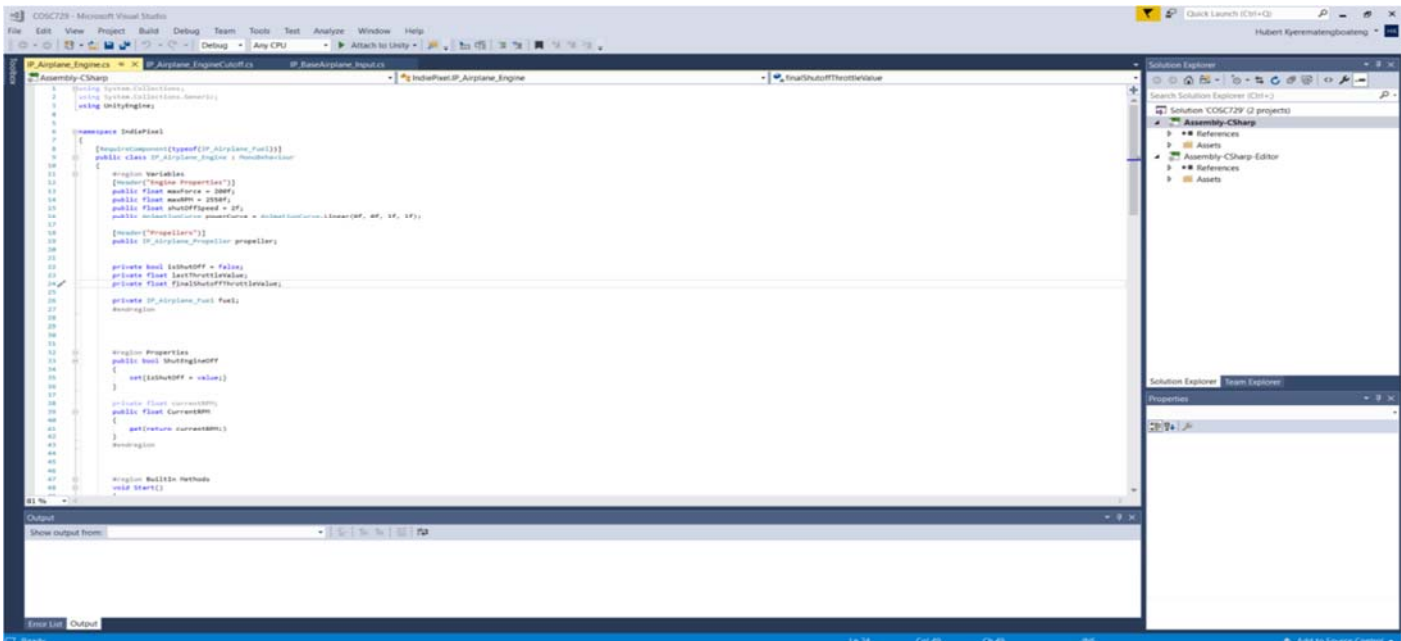


Figure 3: Visual Studio C# development environment

## FUNCTIONALITY

- **Sound** – There is a sound of the plane’s engine. There is also the increase/decrease in the sound pitch of the plane when the engine throttle is increase/decrease respectively.
- **Interactively** – The user selects whether to fly the plane in the daytime or night to me environment. The user also interacts with the simulator by taking off the plane, maneuvering in the air to prevent the plane from hitting a bird and also landing the plane.
- **Animation** – Animations in the virtual environment include transport helicopter blades rotating.
- **Vision** – Models from Google Sketchup and 3D Max where included in the environment. Also models from third-party Indie was used especially the plane used in the simulation was from the Indie framework.
- **Avatars** – 10 avatars, two Security Men, 8 Military trainees
- **Sensors** - Sensors: Implemented is Proximity, Time, Touch
- 

## WHY FLIGHT SIMULATOR

The Flight take-off simulator is very important for airplane instruction because it offers trainers and trainees to experience the plane they will be training on/with firsthand before flying the plane. Due to the high cost of flying the actual plane in terms of monetary and human life. Using the flight simulator allows trainees to make any mistake and learning from their mistake which would be costly (in terms

loss of life, cost of the airplane, etc.) if that same mistake or accident had occurred while flying the physical plane.



Figure 4: Night time cast over Military base.



Figure 5: Day time flight Simulation

*Flight take-off Simulation Over Desert Airport – By Hubert Kyeremateng- Boateng and Veronica Boateng – BSU Spring 2020*



Figure 6: Day time closeup look of Military base.

## PROBLEMS ENCOUNTERED

There were numerous problems encounters during the project, initial problems were modelling the avatars and also implanting the unity physics. But with the help of Indie Pixel, we were able to over this problem.

## USER'S MANUAL

- Vertical
  - **Increase/positive:** key code **Up** ↑
  - **Decrease/negative:** key code **Down** ↓
- Horizontal
  - **Increase:** Key code A
  - **Decrease:** Key code D
- Yaw
  - Left: Arrow Key Left ←
  - Right: Arrow Key Left →
- Throttle
  - **Increase:** key code **W**
  - **Decrease:** key code **S**
- Restart Simulator:
  - Key code: **R**
- Lights (Day/Night time):
  - Key code: **L**
- Engine CutOff



- Keycode: **O**
- Flaps:
  - Down: Key code **F**
  - Up: Key code **G**

## REFERENCES

- Indie Pixel - <https://www.indie-pixel.com/unityCourses/Intro-to-Airplane-Physics-in-Unity-2018>
- Unity 3D Scripting Reference - <https://docs.unity3d.com/ScriptReference/>

