**Software Requirements Specification**

**For**

**UAV Swarm**

Team Members:

* Samantha Balistreri
* Tyler Wise
* Joseph Moran
* Michael Fornito
* William Edwards
* Daniela Regueira

Embry-Riddle Aeronautical University

Version 2

Table of Contents

[**1. Introduction**](#_z61y4x1qdolt) **3**

[1.1 System to be Produced](#_h31ybe6jp5pn) 3

[1.2 Applicable Standards](#_xrtgtui3tm6x) 3

[1.3 Definitions, Acronyms, and Abbreviations](#_pzodhv4kc2ex) 3

[**2. Product Overview**](#_xu1rshw3xzjm) **4**

[2.1 Assumptions](#_v2vvcdeed4xe) 4

[2.2 Stakeholders](#_nhwsdzsm97ar) 4

[2.3 Event Table](#_tlvvvvsq9lug) 4

[2.4 Use Case Diagram](#_857mr7saez5j) 5

[2.5 Use Case Descriptions](#_l0kuzpeojvq1) 6

[**3. Specific Requirements**](#_x18wax9tluy5) **6**

[3.1 Functional Requirements](#_hiwtdbcsidte) 6

[3.2 Interface Requirements](#_ojtbmqdezaba) 13

[3.3 User and Human Factors Requirements](#_24xogj971hg5) 14

[3.4 Documentation Requirements](#_csea7zcp85ly) 22

[3.5 Data Requirements](#_jtffgwn70b74) 25

[3.6 Resource Requirements](#_j7ewxmrm8p8o) 27

[**4. Supporting Material**](#_upvy14ljl8si) **28**

# 1. Introduction

## 1.1 System to be Produced

* The product to be produced will be a Swarm Behavior Network that will allow a swarm of no less than 10 drones to fly in swarming formations. The drones will use image processing and swarming algorithms to accomplish this goal. This product will be done strictly in a simulated environment and there is no physical components to this product.

## 1.2 Applicable Standards

* This project should be held to industry standards in both the engineering industry as well as the aviation industry

## 1.3 Definitions, Acronyms, and Abbreviations

* UAV- Unmanned Aerial Vehicle
* UAS - Unmanned Aircraft Systems
* UE4 - Unreal Engine 4
* MVS - Microsoft Visual Studio
* SBF - Swarm Behavior Framework
* AirSim - An open-source, cross platform simulator for drones, ground vehicles such as cars and various other objects, built on Epic Games’ Unreal Engine 4 as a platform for AI research.

# 2. Product Overview

## 2.1 Assumptions

* Assume that the user will have no control over the drones and it will be strictly autonomous
* Assume that the swarm framework will only work in the UE4/AirSim environment
* Assume that the drone swarm is no more than ten drones

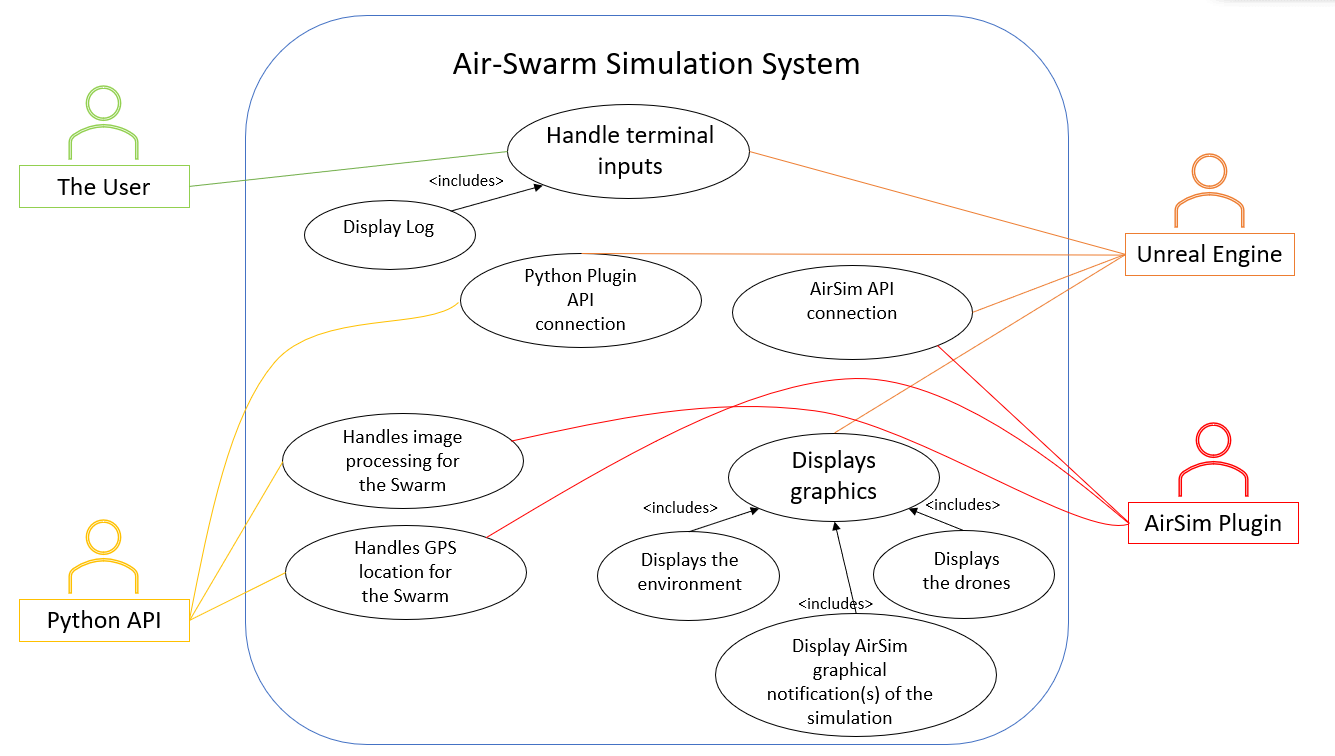
## 2.2 Stakeholders

* Customer - Dr. Akbas
  + Identifies requirements and is invested in the final product
* Product Owner - Juan
  + Assisting the scrum team in prioritization and evaluating project progress
* Scrum Team Members - Samantha, Tyler, Joseph, Michael, Will, and Daniela
  + Completion of the project

## 2.3 Event Table

|  |  |  |  |
| --- | --- | --- | --- |
| Event Name | External Stimuli | External Responses | Internal data and state |
| Start | User clicks button to start the simulation | Simulation displays the running simulation | Algorithm begins running |
| Pause | User clicks the button to pause the simulation | Simulation stops moving but simulation is not reset | Algorithm pauses and can continue to run when prompted |
| Stop | User clicks button to stop the simulation | Simulation is no longer running and UI resets | Algorithm stops running and resets |
| Image | User calls for the python function for the image-based simulation mode | Terminal displays the swarm utilizing the image-based algorithm | Runs algorithm that is appropriate for image-based swarming |
| Add Drone | User calls for the python function to add the drone in the swarm | Terminal displays if it was successful | Modifies settings file |
| Remove Drone | User calls for the python function to remove the drone from the swarm | Terminal displays if it was successful | Modifies settings file |

## 2.4 Use Case Diagram



## 2.5 Use Case Descriptions

1. Display graphics – This System (Unreal Engine) shall display the necessary graphical information to the user
2. Display the drones – The System shall display the drones via Unreal Engine and the player object from AirSim
3. Display the Environment – The System shall (Unreal Engine) shall display the environment from 3rd party marketplace
4. Display the AirSim notifications – The System shall display graphical notifications automatically provided by the AirSim API
5. Handles Image Processing of the Swarm – Python shall be used to handle the computation of the swarm so that it may be processed for AirSim to the Unreal Engine
6. Handles GPS Location of the Swarm – Python shall be used to handle the computation of the swarm so that it may be processed for AirSim to the Unreal Engine
7. Handle Terminal Inputs – The user shall call their desired python functions
8. Display Log**-** The terminal will output to the log file
9. Python Plugin API Connection – Unreal shall connect the python script pythons so that it may be accessed from the terminal
10. AirSim API Connection – AirSim shall connect to Unreal so that the AirSim can use the Unreal Functions

# 3. Specific Requirements

## 3.1 Functional Requirements

|  |
| --- |
| No: F1 |
| Statement: The StartUp python script must connect to Unreal Engine |
| Source: Basic functions |
| Dependency: F2 |
| Conflicts: None |
| Evaluation Method:   1. Click the start button in Unreal Engine 2. Click the start button in the user’s choice of IDE to execute the StartUp python script 3. If successful it will say “connected”, if not it will say “not connected” in the python console. |
| Revision History:   1. Daniela Regueira - 09/25/2020 - Version 1 2. Michael Fornito - 11/30/2020 - Version 2 |

|  |
| --- |
| No: F2 |
| Statement: The program shall allow the user to start the simulation. |
| Source: Basic functions |
| Dependency: F1 |
| Conflicts: None |
| Evaluation Method:   1. The simulation with the at least 1 drone in the settings file 2. The simulation with the mode in the settings file set as Multirotor 3. Click the run button in the Unreal Engine 4. Execute the python script called StartUp |
| Revision History:   1. Daniela Regueira - 09/25/2020 - Version 1 2. Michael Fornito - 11/30/2020 - Version 2 |

## 

|  |
| --- |
| No: F3 |
| Statement: The program shall allow the user to stop the simulation. |
| Source: Basic functions |
| Dependency: F2 |
| Conflicts: None |
| Evaluation Method: The simulation will end. |
| Revision History: Daniela Regueira - 09/25/2020 - Version 1 |

## 

|  |
| --- |
| No: F4 |
| Statement: The program should allow the user to control the swarming algorithm through distribution size and number of drones |
| Source: Extra functions |
| Dependency: F2 |
| Conflicts: Only one formation is allowed to be selected. Default formation is “Air Space Optimization”. |
| Evaluation Method: The simulation will run the formation selected. |
| Revision History: Daniela Regueira - 09/25/2020 - Version 1  Joseph Moran - 11/30/2020 - Addressed comments and specified how users would control the algorithm. |

|  |
| --- |
| No: F5 |
| Statement: The program shall display the user with an error message if the user does not put in a number (int or float) as one of the parameters for the drones position when adding a drone. |
| Source: Extra functions |
| Dependency: None |
| Conflicts: None |
| Evaluation Method:   1. Type in addDrone(X coordinate, Y coordinate, Z coordinate) 2. Drones only accepts numbers as inputs |
| Created on 11/29/2020 - Michael Fornito |

## 

|  |
| --- |
| No: F6 |
| Statement: The program shall display the user with an error message if the user removes a drone but does not use a number (int) |
| Source: Extra functions |
| Dependency: None |
| Conflicts: None |
| Evaluation Method:   1. Type in removeDrone(index) 2. Only accepts numbers as inputs |
| Created on 11/29/2020 - Michael Fornito |

## 

|  |
| --- |
| No: F7 |
| Statement: The program shall display the user with an error message if the user tries to remove a drone where there is only one drone in the settings file |
| Source: Extra functions |
| Dependency: If F6 doesn’t throw an error |
| Conflicts: None |
| Evaluation Method:   1. Type in removeDrones(index number) 2. Cannot have less than 1 drone available |
| Created on 11/29/2020 - Michael Fornito |

## 

## 

|  |
| --- |
| No: F8 |
| Statement: The program shall display the user with an error message if the user tries to add more than 10 drones |
| Source: Extra functions |
| Dependency: If F5 doesn’t throw an error |
| Conflicts: None |
| Evaluation Method:   1. Type in addDrone(X coordinate, Y coordinate, Z coordinate) 2. Cannot have more than 10 drones available |
| Created on 11/29/2020 - Michael Fornito |

## 

|  |
| --- |
| No: F9 |
| Statement: The program shall display the user with an error message if the user tries input a non-number (int or double) the distribution size. |
| Source: Extra functions |
| Dependency: None |
| Conflicts: None |
| Evaluation Method:   1. addDistributionSize( number) 2. Must be a number only |
| Created on 11/29/2020 - Michael Fornito |

## 

|  |
| --- |
| No: F10 |
| Statement: The program shall display the user with an error message if the user tries input a number that is out-of-bounds when removing a drone based on index |
| Source: Extra functions |
| Dependency: None |
| Conflicts: None |
| Evaluation Method:   1. removeDrone( index number) 2. Out of bound inputted value |
| Created on 11/29/2020 - Michael Fornito |

## 3.2 Interface Requirements

|  |
| --- |
| No: #I1 |
| Statement: The camera on each drone shall broadcast its current image to the sink drones no less than every 5 seconds. This data shall be encoded as a byte array. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The message with the raw data is received within a 5 second interval. |
| Revision History:   1. Created by Tyler Wise, 9/24/20 |

|  |
| --- |
| No: #I2 |
| Statement: Each sink drone shall distribute the relative location of each drone of itself to the master drone within a 5 second interval. This data shall be encoded as a dictionary of guides to vectors. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The message with the data is received within a 5 second interval. 2. The number of elements in the list is equal to the number of drones in the simulation. |
| Revision History:   1. Created by Tyler Wise, 9/24/20 2. Updated data types, TJW, 10/29/20 |

## 3.3 User and Human Factors Requirements

* The following are user interface requirements and system interface requirements that will allow the user to control how they want to operate the UAV drones in the simulation.

|  |
| --- |
| No: #HF1 |
| Statement: User shall be able to have the python scripts connect to the Unreal Engine |
| Source: Basic Functions |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. User shall start the python file called StartUp outside of Unreal in their own python IDE 2. User clicks the start button on the Unreal Engine |
| Revision History:   1. Created by Michael Fornito, 9/22/200 2. Modified on 9/24/2020, Evaluation Method 3. Modified on 10/29/20, Eval Method & Statement 4. Michael Fornito - 11/29/2020 - Modified Evaluation Method and Statement |

|  |
| --- |
| No: #HF2 |
| Statement: User shall be able to start the simulation |
| Dependency:   1. The number of drones must exist in the settings file, at least 1 2. The user must start the modify\_settings python file |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The dependencies stated in HF2 have been inputted    1. If not the simulation will not start 2. The system will display a visual representation of the swarm |
| Revision History:   1. Created by Michael Fornito, 9/22/200 2. Modified on 9/24/2020, Dependency 3. Modified on 10/29/2020, Evaluation Method and Dependency 4. Michael Fornito - 11/29/2020 - Modified on 10/29/2020, Evaluation Method and Dependency |

|  |
| --- |
| No: #HF3 |
| Statement: User shall be able to pause the simulation |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method: The drones will stop their simulation |
| Revision History:   1. Created by Michael Fornito, 9/22/20 2. Modified on 9/24/2020, Evaluation Method |

|  |
| --- |
| No: #HF4 |
| Statement: User shall be able to end the simulation |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The drones stop swarming and the execution of the program has ended |
| Revision History:   1. Created by Michael Fornito, 9/22/20 2. Modified on 9/24/2020, Evaluation Method |

|  |
| --- |
| No: #HF5 |
| Statement: User shall be notified which drone(s) have failed during the simulation |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The user has started the simulation, ensure #HF2 is followed 2. One or more of the drones crash 3. The user is notified during the simulation that the drone(s) have crashed through the terminal |
| Revision History:   1. Created by Michael Fornito, 9/22/20 2. Modified on 9/24/2020, Evaluation Method 3. Modified on 10/29/2020, Evaluation Method |

|  |
| --- |
| No: #HF6 |
| Statement: User shall be notified which drone(s) have successfully finished the simulation |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The user has started the simulation, ensure #HF2 is followed 2. The drones do not crash during the entire simulation run 3. The user is notified at the end of the simulation that it was successful through the terminal |
| Revision History:   1. Created by Michael Fornito, 9/22/20 2. Modified on 9/24/2020, Evaluation Method 3. Modified on 10/29/2020, Evaluation Method |

|  |
| --- |
| No: #HF7 |
| Statement: User shall be able to see the perspective of the selected drone flying one at a time |
| Dependency: None |
| Conflicts: At least one or more drones must be inputted |
| Supporting Materials: None |
| Evaluation Method:   1. The user has started the simulation, ensure #HF2 is followed 2. The user can select the drone they want to see 3. The selected drone’s POV is displayed to the user |
| Revision History:   1. Created by Michael Fornito, 9/22/20 2. Modified on 9/24/2020, Evaluation Method 3. Modified on 9/25/2020, Conflicts |

|  |
| --- |
| No: #HF8 |
| Statement: User shall input the distribution shape of the drones |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The user shall call the python function to change the distribution size, modify the settings files 2. The user is notified that it was successfully changed or not 3. The user clicks the start button 4. The simulation shall execute with the given size |
| Revision History:   1. Created by Michael Fornito, 9/28/20 2. Modified on 10/29/2020, Evaluation Method 3. Michael Fornito - Modified on 10/29/2020, Evaluation Method |

|  |
| --- |
| No: #HF9 |
| Statement: User shall call the python function to add a drone |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The user imports the modify settings package 2. The user calls the add drone function, modifies the settings file 3. The terminal shall tell the user if it was successful or not 4. The settings were changed based on the input 5. The user has started the simulation, ensure #HF2 is followed 6. The simulation shall display the correct inputted # of drones |
| Revision History:  Created by Michael Fornito, 10/29/2020 |

|  |
| --- |
| No: #HF10 |
| Statement: User shall call the python function to remove a drone |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The user imports the modify settings package 2. The user calls the remove drone function 3. The terminal shall tell the user if it was successful or not 4. The settings were changed based on the input 5. The user has started the simulation, ensure #HF2 is followed 6. The simulation shall display the correct # of drones |
| Revision History:   1. Created by Michael Fornito, 10/29/2020 2. Michael Fornito - Modified on 10/29/2020, Evaluation Method |

|  |
| --- |
| No: #HF11 |
| Statement: User shall call the python function to display the drone data in the terminal |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The user imports the modify settings package 2. The user calls the display drone function, modifies the settings file 3. The terminal shall tell the user the drone’s data that is within the settings file |
| Revision History:   1. Created by Michael Fornito, 10/29/2020 2. Michael Fornito Modified on 10/29/2020, Evaluation Method |

## 3.4 Documentation Requirements

|  |
| --- |
| No: #DOC1 |
| Statement: The user manual shall explain how to operate the simulation. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The instructions in the manual shall be followed and allow for the simulation to be started. 2. The instructions in the manual shall be followed and allow for the simulation to be stopped. |
| Revision History:   1. Created by Sam Balistreri, 10/29/2020 2. Tyler Wise Modified on 11/29/2020, Formatting |

|  |
| --- |
| No: #DOC2 |
| Statement: The user manual shall explain how to configure the simulation. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The instructions in the manual shall be followed to add drones. 2. The instructions in the manual shall be followed to remove drones. 3. The instructions in the manual shall be followed to modify swarm distribution size. |
| Revision History:   1. Created by Sam Balistreri, 10/29/2020 2. Tyler Wise Modified on 11/29/2020, Formatting |

|  |
| --- |
| No: #DOC3 |
| Statement: The developer documentation shall explain how to extend the simulation. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The document shall include a current data flow diagram. 2. The document shall include a current UML class diagram. |
| Revision History:   1. Created by Sam Balistreri, 10/29/2020 2. Tyler Wise Modified on 11/29/2020, Formatting |

|  |
| --- |
| No: #DOC4 |
| Statement: The user manual shall explain how to install the simulation. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The instructions in the manual can be followed to initialize the simulation environment. |
| Revision History:   1. Created by Sam Balistreri, 10/29/2020 2. Tyler Wise Modified on 11/29/2020, Formatting |

## 3.5 Data Requirements

|  |
| --- |
| No: #D1 |
| Statement: A collinear attractive force will be calculated for each actor drone to each sink drone based on the masses of the drones, the current radius of the actor from the sink, and a configurable parameter to control the strength of the force. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. A force collinear with the direction between a pair of drones is computed with increasing strength as the distance between the drones decreases. |
| Revision History:   1. Created by Tyler Wise, 9/24/20 |

|  |
| --- |
| No: #D2 |
| Statement: A collinear repulsive force will be calculated for each actor drone to every other actor drone based on the masses of the drones, the distance between each pair of drones, and a configurable parameter to control the strength of the force. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. A force collinear with the direction between a pair of drones is computed with decreasing strength as the distance between the drones increases. |
| Revision History:   1. Created by Tyler Wise, 9/24/20 |

|  |
| --- |
| No: #D3 |
| Statement: The relative position of a given drone to all other drones shall be available at all times. This position shall be updated no less than once every ten seconds. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The perceived relative position of each drone in a formation is within 10% of true relative position of the drones. |
| Revision History:   1. Created by Tyler Wise, 9/24/20 |

|  |
| --- |
| No: #D4 |
| Statement: The data to control the simulation will be available in the settings.json file. |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: None |
| Evaluation Method:   1. The number of drones in the simulation is reflective of the value in the file. 2. The location of the drones in the simulation is reflective of the values in the file. 3. The distribution of the drones in the simulation is reflective of the value in the file. |
| Revision History:   1. Created by Tyler Wise, 10/28/20 |

## 3.6 Resource Requirements

* Windows 10 platform, 64-bit
* Epic Games Launcher
* Unreal Engine 4.24 or newer
* Visual Studio 2019
* AirSim
* Three sprints
  + Sprint 1 ends 10/1/2020
  + Sprint 2 ends 10/29/2020
  + Sprint 3 ends 12/09/2020

# 4. Supporting Material

* Reference Projects and Papers
  + APAWSAN: Actor Positioning for Aerial Wireless Sensor and Actor Networks by Mustafa Ilhan Akbas and Damla Turgut, University Of Central Florida
  + Vision Based Formation Control in Unreal Engine -Air Sim by Kaveh Fatian
  + Actor Positioning Based on Molecular Geometry in Aerial Sensor Networks by Mustafa ˙Ilhan Akbas¸ Gurkan Solmaz and Damla Turgut, University of Central Florida