System Requirements Specification

for

Modeling Unmanned Aerial Swarms Using Unreal Engine and AirSim Simulator

Version 3.0 approved

Prepared by Dillon Mead

Embry-Riddle Aeronautical University EECS Department

30 Nov 2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Dillon Mead	13/09/21	Initial document creation.	1.0
Dillon Mead	27/09/21	First version edit.	1.1
Dillon Mead	26/10/21	Second version edit.	2.0
John Mueller	26/10/21	Added Stimulus/Response in section 4	2.1
Dillon Mead	29/11/21	Third version edit.	3.0

1. Introduction

1.1 Purpose

The purpose of this SRS is to define the requirements for a simulated unmanned aerial swarm in Microsoft's AirSim simulator. This SRS encompasses all requirements for the simulated unmanned aerial swarm system.

1.2 Document Conventions

This document does not currently have any conventions.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, testers, and document writers.

1.4 Product Scope

This system simulates an unmanned aerial swarm. The aerial swarm will be autonomous with unified, cohesive behavior. The general mission for the aerial swarm will be to collect data from a three-dimensional environment (e.g. the volume of a plume of smoke).

1.5 References

[1] Microsoft. "AirSim." www.github.com. https://github.com/Microsoft/AirSim
[2] Aerial Swarm Simulators. "Modeling Unmanned Aerial Swarms Using Unreal Engine and AirSim Simulator System Design Document." Version 3.0. November 30, 2021. www.github.com. https://github.com/mead-d/Modeling-Unmanned-Aerial-Swarms-Using-Unreal-Game-Engine-and-AirSim-Simulator/tree/main/Deliverables.

2. Overall Description

2.1 Product Perspective

This system is a new product inspired by earlier research from Professor Ilhan Akbas. The original idea was to use aerial wireless sensor and actor networks (aerial swarm) to perform observation tasks while mitigating risk to the user. For example, the user could be a geologist and the observation task is the smoke plume from a volcano. It is easier to gather data with an aerial swarm and minimize potential injury or loss of life.

2.2 Product Functions

- Multiple Unmanned Aerial Vehicles operate as unified, cohesive swarm.
- Each UAV maintains communication with the swarm.
- The aerial swarm maintains communication with the ground station.
- The ground station may transmit mission orders to the swarm.
- The ground station may transmit changed mission orders to the swarm.

2.3 User Classes and Characteristics

There is only one user class for this system. The user must have knowledge of computers and programming. Specifically, the user must know how to use Microsoft's AirSim Simulator.

2.4 Operating Environment

This system must use Microsoft's AirSim Simulator. There is no explicit hardware requirement. Microsoft's AirSim may be used on any computer using any of multiple operating systems. The current minimum software necessary for Microsoft's AirSim is Visual Studio 2019 and Unreal Engine 4.26.

2.5 Design and Implementation Constraints

Although there are no hardware requirements, Unreal Engine is graphics intensive and may require a powerful graphics processor.

2.6 User Documentation

The most recent version of the System Design Document, System Requirements Specification, and Test Plan may be found within the Aerial Swarm Simulator team's GitHub repository within the deliverables folder: https://github.com/mead-d/Modeling-Unmanned-Aerial-Swarms-Using-Unreal-Game-Engine-and-AirSim-Simulator/tree/main/Deliverables.
The only exception is the AirSim Simulator documentation that may be found at: https://microsoft.github.io/AirSim/.

2.7 Assumptions and Dependencies

Assumptions:
We assume that Unreal Engine 4 and Microsoft's AirSim Simulator will remain open source.

Dependencies:
This system is currently only usable within Microsoft's AirSim environment. AirSim is itself dependent on Unreal Engine 4 and Visual Studio 2019. Any changes to Unreal Engine 4, Visual Studio 2019, or Microsoft's AirSim have the potential to cause errors and irregularities within the Aerial Swarm Simulator system.

3. External Interface Requirements

3.1 User Interfaces

[Req 1] The Aerial Swarm Simulator system shall be modelled and simulated in Microsoft's AirSim Simulator.
[Req 2] Visual Studio 2019 shall be used to edit files and environment variables.
[Req 3] The User shall implement mission scenarios by executing the appropriate script in the AirSim Simulator.

3.2 Hardware Interfaces

This system has no hardware interface requirements.

3.3 Software Interfaces

Microsoft's AirSim Simulator interfaces with Unreal Engine 4 by declaring which graphical environment to generate. The Unreal Engine 4 physics system is also utilized for AirSim object attributes and interactions. AirSim also uses OpenCV for object detection/ image recognition in the Unreal Engine 4 environment.

3.4 Communications Interfaces

The system does not currently use any communications interfaces.

4. System Features

4.1 Ground Station/ Control

4.1.1 Description and Priority

Ground Station is the user/ computer that is used. Ground Station is what transmits the mission task input and what receives the sensor data and swarm status.

4.1.2 Stimulus/Response Sequences

Action: User issues mission task for aerial swarm.
Response: Aerial swarm begins mission and reports status of aerial swarm.
Action: Aerial swarm transmits sensor data.
Response: Ground station records sensor data.
Action: User modifies, or re-assigns mission task.
Response: Aerial swarm changes status if necessary and adjusts mission task.

4.1.3 Functional Requirements

[Req 4] Ground station shall have the ability to assign missions to the aerial swarm.
[Req 5] Ground station shall have the ability to modify missions.
[Req 6] Ground station shall have the ability to re-assign missions to the aerial swarm.
[Req 7] Ground station shall receive reports on the aerial swarm status including status of all individual UAV.
[Req 8] Ground station shall receive sensor data from aerial swarm.
[Req 9] Ground station shall record sensor data.

[Req 10] Ground station shall display aerial swarm status including status of all individual UAV.

4.2 UAS Swarm

4.2.1 Description and Priority

The aerial swarm is a unified, cohesive grouping of multiple individual UAV. There is a designated lead UAV that leads the swarm and organizes the swarm while operating, collects all the individual UAV statuses, and reports the overall swarm status and individual UAV statuses to the ground station.

4.2.2 Stimulus/Response Sequences

Action: User spawns in an aerial swarm
Response: First UAV spawned is designated as the lead UAV
Action: User assigns mission task to aerial swarm
Response: Lead UAV organizes and coordinates with other UAVs to complete mission
Action: User requests UAVs' status
Response: Lead UAV gathers each individual UAV's status and relays that information back to ground control

4.2.3 Functional Requirements

[Req 11] The aerial swarm shall designate a lead UAV for swarm organization and communication.
[Req 12] The aerial swarm shall reassign the lead UAV when the current lead becomes non-functional.
[Req 13] The lead UAV shall receive status data from all individual UAV.
[Req 14] The lead UAV shall transmit status data of the aerial swarm and all individual UAV.
[Req 15] The aerial swarm shall transmit sensor data to a repository in the ground station.
[Req 16] The aerial swarm shall determine the positioning of individual UAV and transmit the data to individual UAV.
[Req 17] The aerial swarm shall adjust and continue the mission task when an individual UAV becomes inactive.
[Req 18] The aerial swarm shall acknowledge receipt, or modification, of mission task.
[Req 19] The aerial swarm shall return to ground station when mission task is complete.

4.3 Individual UAV

4.3.1 Description and Priority

An atomic vehicle of the aerial swarm. Individual UAV will have a position and status that is communicated with the swarm. Each UAV will also have some payload that includes sensors such as a camera.

4.3.2 Stimulus/Response Sequences

Action: User spawns in an aerial swarm with a mission
Response: Each individual UAV relays its position and status to the lead UAV throughout the mission
Action: User assigns payloads to certain UAVs
Response: Lead UAV is relayed this information and adjusts the formation as needed
Action: User requests UAVs' status
Response: Lead UAV gathers each individual UAV's status and relays that information back to ground control

4.3.3 Functional Requirements

[Req 20] Individual UAV shall communicate position with the aerial swarm.

[Req 21] Individual UAV shall communicate functional status with the aerial swarm.
[Req 22] Individual UAV shall avoid collisions with objects including other UAV.
[Req 23] Individual UAV shall carry a payload that will house sensors.
[Req 24] Sensor data shall be routed through the aerial swarm via the lead UAV.

4.4 Performance Requirements

[Req 25] The aerial swarm shall continue task and attempt to finish the mission queue when any individual UAV becomes inactive.

4.5 Safety Requirements

We do not have any safety requirements.

4.6 Security Requirements

We do not have any security requirements.

4.7 Software Quality Attributes

We have not considered software quality attributes yet.
Adaptability, availability, correctness, flexibility, maintainability, reusability, testability, usability

4.8 Business Rules

There are no business rules for this system.

5. Other Requirements

There are no other requirements.

Appendix A: Glossary

Active: UAV is operational and in flight.
Inactive: UAV is not operational or is not in flight.
UAV: Unmanned Aerial Vehicle.
UAS: Unmanned Aerial System – “aerial swarm”.

Appendix B: Analysis Models

Analysis Models may be found in the System Design Document [2].