System Requirements Specification

for

Spacecraft-Control-Center-Training and Testing Environment (STaTE)

**Version 3.1 approved**

**Prepared by Juliana Altamira, Kyle Garber, Jesse Slager, Jeff Cevallos, AnnaMaria Summer, Carly Bosma**

**2022-23 Senior Design Group**

**Created 10/02/2022**

**Table of Contents**

[**1. Introduction**](#_heading=h.1fob9te) **1**

[**1.1 Purpose**](#_heading=h.3znysh7) **1**

[**1.2 Document Conventions**](#_heading=h.ekt6ul38k6s) **1**

[**1.3 Intended Audience and Reading Suggestions**](#_heading=h.r4k8wie18uep) **1**

[**1.4 Product Scope**](#_heading=h.1i7rvrgn7d1z) **2**

[**1.5 References**](#_heading=h.acq9307wbrnw) **2**

[**2. Overall Description**](#_heading=h.nj6hzkjtuocy) **3**

[**2.1 Product Perspective**](#_heading=h.2s8eyo1) **3**

[**2.2 Product Functions**](#_heading=h.ha95owir1wos) **3**

[**2.3 User Classes and Characteristics**](#_heading=h.n7dmonaqdtb8) **4**

[**2.4 Operating Environment**](#_heading=h.9wz2eeczklne) **5**

[**2.5 Design and Implementation Constraints**](#_heading=h.3d61c9myxi5q) **5**

[**2.6 User Documentation**](#_heading=h.fwh8qsvofthb) **5**

[**2.7 Assumptions and Dependencies**](#_heading=h.xvph614nfpbx) **6**

[**3. External Interface Requirements**](#_heading=h.4vfum5wa7qjs) **6**

[**3.1 User Interfaces**](#_heading=h.2jxsxqh) **6**

[**3.2 Hardware Interfaces**](#_heading=h.2b9fgw153p6m) **8**

[**3.3 Software Interfaces**](#_heading=h.s8l21kasz5ms) **8**

[**3.4 Communications Interfaces**](#_heading=h.tlgr2wi67rh4) **8**

[**4. System Features**](#_heading=h.7h1ctau68zl) **9**

[**4.1 SWA - STaTE Web Application**](#_heading=h.t3vxlh22fdy5) **9**

[**4.2 FOP - Flight Operator Platform**](#_heading=h.4h28k72n01e) **11**

[**4.3 TCP - Test Conductor Platform**](#_heading=h.u7fnc06tf35v) **12**

[**4.4 Control Console**](#_heading=h.qgi5joicki7f) **15**

[**4.5 Simulation Engine**](#_heading=h.7agolykd0xuu) **17**

[**4.6 SimCraft**](#_heading=h.z7hfkhg8jh0p) **18**

[**5. Other Nonfunctional Requirements**](#_heading=h.3whwml4) **19**

[**5.1 Performance Requirements**](#_heading=h.2bn6wsx) **19**

[**5.2 Safety Requirements**](#_heading=h.3y1t2dmmsh2d) **21**

[**5.3 Security Requirements**](#_heading=h.tyajelpauuq7) **21**

[**5.4 Software Quality Attributes**](#_heading=h.6hbpqcj6qti7) **21**

[**5.5 Business Rules**](#_heading=h.7uttcnaoca43) **22**

[**6. Other Requirements**](#_heading=h.lam46emgsgqh) **22**

[**6.1 - Database Requirements**](#_heading=h.2svz3be6s8b6) **22**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Jeff Cevallos | 10/2 | Beginning of document | 1.0 |
| Kyle Garber | 10/29 | Initialize section 4.1-SWA | 1.1 |
| Jesse Slager | 11/1 | Initialize sections 4.4 and 4.5 -Simulation Engine, SimCraft (Object) | 1.2 |
| Jeremy Mog | 11/1 | Initialize sections 4.4 and 4.5 -Simulation Engine, SimCraft (Object) | 1.2 |
| AnnaMaria Summer | 11/1 | Initialize section 4.2-Flight Operator | 1.3 |
| Carly Bosma | 11/1 | Initialize section 4.6-Test Conductor Platform | 1.4 |
| Jeff Cevallos | 11/2 | Initialize section 5.1-Performance Requirements | 1.5 |
| Carly Bosma | 11/2 | Created Documentation Requirements 1-3 in 2.6  Edited Requirement ID's for consistency in 4.3  Edited Requirement ID's for consistency in 4.6  Created Software Quality Attributes 1-10 in 5.4 | 1.6 |
| Juliana Altamira | 11/8 | Revised 3.2 - Hardware Interfaces  Revised 3.1 - User Interfaces  Revised 3.4 - Communication Interfaces  Revised 2.5 - Design Constraints | 2.0 |
| Carly Bosma | 11/8 | Revised 1.2  Revised and added to 1.3  Added to 2.2  Added to 3.3 | 2.1 |
| Jeremy Mog | 11/29 | Fixed formatting  Added description to 1  Added Screenshot to 2.7  Added details to 1.5 | 3.0 |
| Carly Bosma | 12/03 | Ran Spellcheck  Edited 3.4 | 3.1 |
| Kyle Garber | 12/03/2022 | Wording and format modification | 3.2 |

# Introduction

The STaTE product is a web application serving to facilitate real-time management of simulated spacecraft.

## Purpose

The purpose of this product is to create a simulated training and testing environment that trains students to become an effective member on a spacecraft-control-center spaceflight management team. This document contains the requirements that apply to the STaTE system.

## Document Conventions

Priorities for higher-level requirements are assumed to be inherited by detailed requirements.

DC1 - For important notes throughout the document, usually pertaining to incomplete sections, a bolded-underlined note will be present to explain why the section is currently unfinished.

DC2 - Bold larger writing refers to section titles. New Section: 17.

## Intended Audience and Reading Suggestions

The intended audience of this document are software developers either developing or testing the STaTE product. Other readers include the product owner, customers, users, testers and stakeholders of the STaTE product. The following sections of this document serve to create a framework for the development team to reference. Section 1.2 defines the product scope, which is necessary to understand before reviewing system design and functionality requirements. Section 2 focuses on elaborating on the project’s design, functionality, and constraints. Section 3 describes the external interfaces of the project: user, hardware, software, and communication. Due to the web-based nature of the project, there are not many considerations regarding hardware interfaces. Section 4 focuses on functional requirements for each component of the system: SWA, FOP, TCP, Control Console, Simulation Engine, and SimCraft. Section 5 focuses on nonfunctional requirements of the system: performance, safety, security, software quality attributes and business rules. The performance requirements consist of a list of assumptions. Safety and business rules do not have much consideration for the scope of this project. Section 6 contains database requirements. The document concludes with Appendix A: Glossary. Suggestions for how the document should be read based on reader types:

Documentation Writers: Documentation writers refer to those producing documentation for the development team. The development team should refer to and revise this document throughout the development process. Each developer is responsible for reading the document and following established principles.

Product Owner: The Product Owner is the member of the team responsible for maximizing team value to the customer. The Product owner should refer to this document to help the development team prioritize the product backlog and ensure system design and requirements meet the needs of the customer. Some focus should be given to section 2.2, 2.4, 2.7, 3.2 and all of sections 4 and 5. Appendix A: Glossary contains definitions of specialized terms utilized throughout the document.

Customer: The customer of this product should review this document with the product owner, focusing on sections that explain the product and product functionality. These are primarily sections 1 and 2. Appendix A: Glossary contains definitions of specialized terms utilized throughout the document.

System Testers: A System Tester is anyone testing the functionality of the product against its requirements. Depending on the system being tested, a System Tester should refer to both the section defining the system’s design and its requirements. A System Tester can review the document against their testing to ensure product design and functionality is as defined in the document.

Capstone Course Related Readers: Capstone Course Related Readers should read the document in its entirety and provide feedback to the document writers in order to enhance the quality of the document.

## Product Scope

This product shall be a platform for creating and maintaining continuously running spacecraft simulations. These simulations are designed to enable Flight Operators to solve complex issues that may arise during spaceflight or from Test Conductor manipulation. The training environment shall also demonstrate the importance of responsibility delegation to Flight Operators and train their leadership abilities.

## References

References for Spacecraft-Control-Center Consoles:

* *How the Mission is Controlled: Inside NASA and Boeing Joint Operations.* (Gary Jordan, Dec. 20, 2019*)*
  + <https://www.nasa.gov/feature/how-the-mission-is-controlled-inside-nasa-and-boeing-joint-operations>
* *Major Tom: Mission Ops for the 21st century.* (Marshall Culpepper, Aug. 4, 2017)
  + <https://medium.com/kubos-tech/major-tom-mission-ops-for-the-21st-century-329905913911>
* *Xplore’s Major Tom® software delivers satellite operations testing for NOAA with Microsoft Azure Orbital.* (PRWeb, Jun. 3, 2022)
  + <https://www.prweb.com/releases/xplores_major_tom_software_delivers_satellite_operations_testing_for_noaa_with_microsoft_azure_orbital/prweb18732185.htm>
* *Major Tom* (Xplore, N.D.)
  + <https://www.xplore.com/services/operations-as-a-service/major-tom.html>

# Overall Description

## Product Perspective

This product is a simulation environment for spacecraft-control-center spaceflight management team member training and testing.

This product is designed and developed for educational purposes. This product is designed to be easily integrated into a learning environment. The intended use case of this product is for students acting as Flight Operators to be alerted via email about an anomaly occurring on assigned SimCraft in real-time. The Flight Operators on the SimCraft’s flight operation team must then respond to correct the effects of the anomaly.

## Product Functions

2.2.1 User management system hosts both Flight Operator and Test Conductor users.

2.2.2 Simulation Engine facilitates creation of SimCraft with Test Conductor defined subsystems.

2.2.3 SimCraft continuously updates attributes to simulate spaceflight as defined by Test Conductor during creation

2.2.4 Test Conductor Platform facilitates assignment of Flight Operators to manage simulated spacecraft.

2.2.5 Flight Operator Platform facilitates control of SimCraft via widgets defined by the SimCraft’s subsystems.

2.2.6 Test Conductor Platform maintains a log of Flight Operator performance for each SimCraft accessible by the Test Conductor.

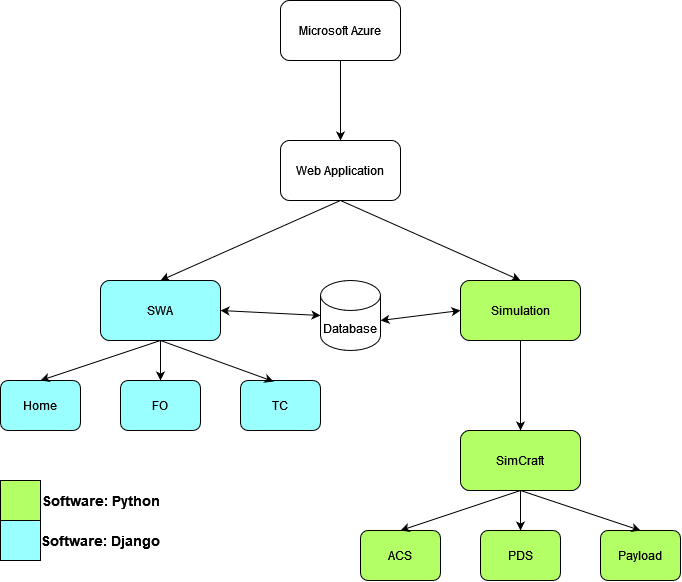
****

Figure 1: Top level Data Flow diagram.

This diagram shows the main functions of the web application is to handle the SWA and simulations. The SWA and Simulations communicate with the database to perform their intended jobs which are represented by the sub-boxes underneath them.

## User Classes and Characteristics

For this product, a team of 4-6 Flight Operators are expected to manage SimCraft. These flight operators are expected to be a part of a class which will typically consist of around 20 students. Therefore, about 5 simulations per class are expected to be running simultaneously.

A team consists of a flight conductor, a test conductor, and some flight operators.

Test Conductor:

* A user who can create a SimCraft, assign Flight Operators, and administer SimCraft anomalies.
* Typically fulfilled by the professor instructing the class.
* Each SimCraft only has one Test Conductor.
* A Test Conductor can be a Test Conductor for multiple SimCraft.
* A Test Conductor will most likely never be a student for its intended use case, but the option to open this role to a student user should be considered for alternative deployments of the STaTE product.

Flight Operator:

* A user who can manage SimCraft that they are assigned to via SimCraft’s defined controls.
* Typically fulfilled by students in a flight operations class.
* Each SimCraft has 4-6 assigned Flight Operators.
* Flight Operators can be assigned to multiple SimCraft.

## Operating Environment

The product is designed to be a simulation tool that users can remotely access via a website. The website is handled by Django which is a high-level Python web framework.

For this product to operate, the following is considered:

* A server that can be remotely accessed
* The most common web browsers
  + Google Chrome
  + Apple’s Safari Browser
  + Microsoft Edge

§ Other browsers may be considered, but within those four web browsers, roughly 93% of all internet users are contained within those browsers.

§ Anyone not using those browsers are most likely to be using an outdated web browser.

## Design and Implementation Constraints

Using Microsoft Azure through a student license has posed some design/implementation constraints. These constraints include:

* Inability to pull & edit GitHub repository files from VS codes azure extension without IT admin permissions
* RAM limited to 1.5GB

## User Documentation

Documentation for users will include:

Start-Up Guide:

* A guide for Test Conductors covering:
  + How to set up a mission
  + How to place students in missions
  + How to cause an anomaly
  + How to set up jobs for students
  + Create a SimCraft

Flight Operator Guide:

* A guide for Flight Operators covering:
  + How to join a mission
  + How to manage a SimCraft

UD-STANDARD-1: The documentation will be provided online through the web application.

UD-STANDARD-2: The documentation can be downloaded, sent, or given in physical form to users.

UD-STANDARD-3: The intended audience of the documentation are the teachers conducting spacecraft control center training and the students in training.

## Assumptions and Dependencies

The primary assumption for this program is that the student is knowledgeable of the function of the options they are presented with when adjusting the simulated spacecraft.

# External Interface Requirements

## User Interfaces

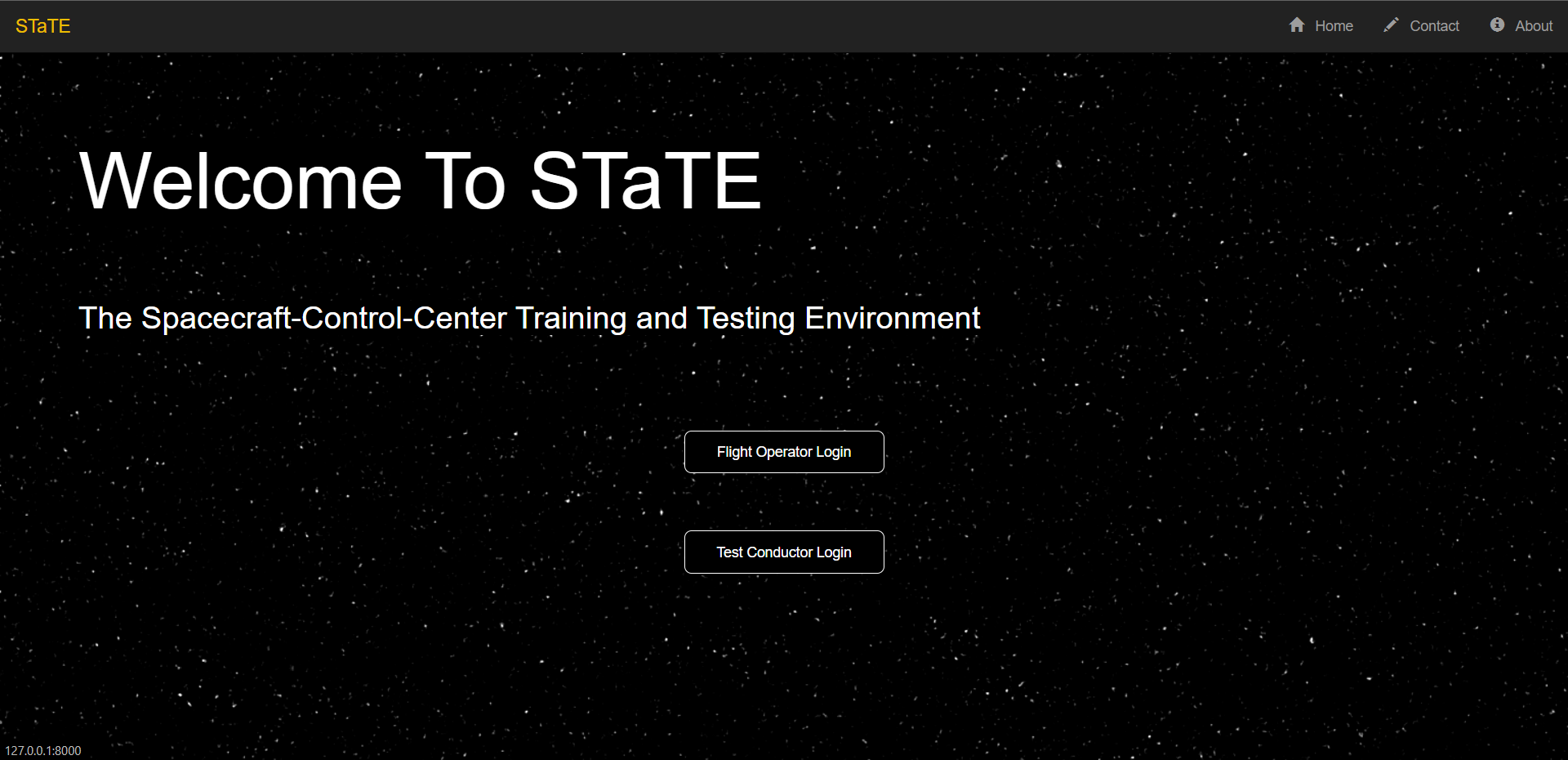


Figure 2: SWA Home screen

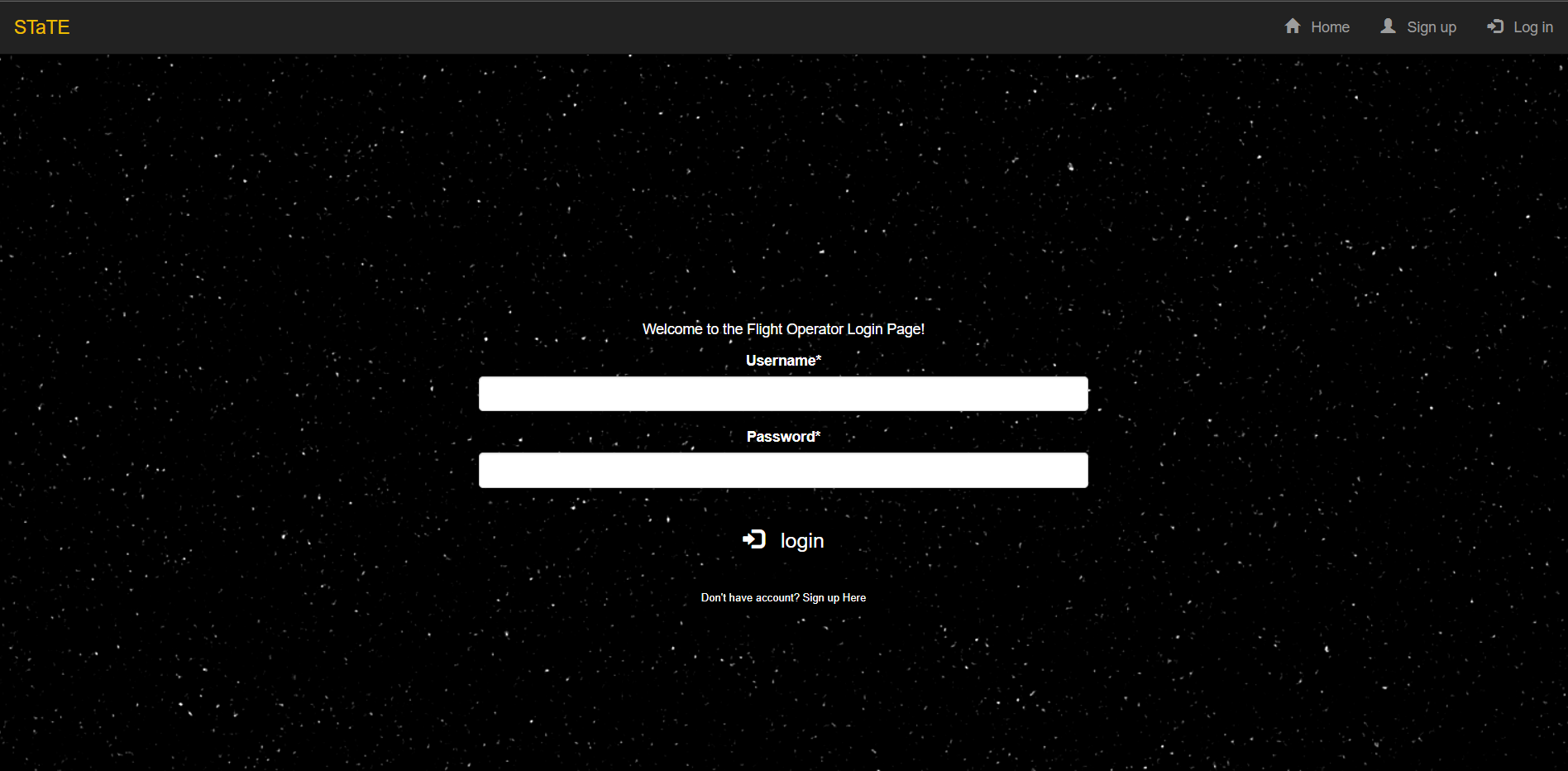
**

Figure 3: FO Login Page

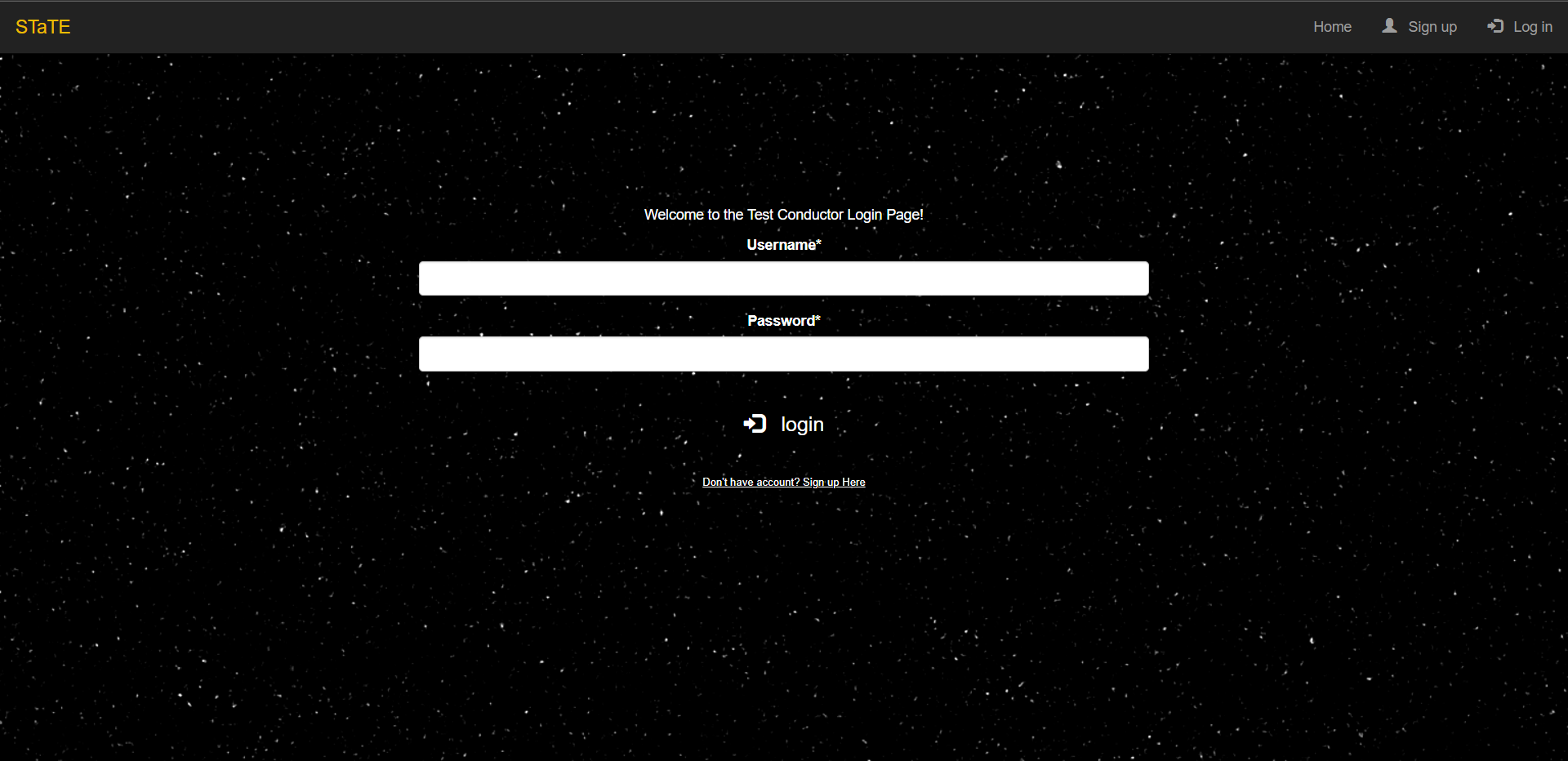


Figure 4: TC Login Page

There are two distinct user interfaces projected for this project: the Flight Operator user interface and the Test Conductor user interface.

The Flight Operator user interface includes a login screen, where the user will be directed to log in to or create their Flight Operator account. These inputs will be interpreted as text box entries or button clicks. The output of a successful Flight Operator login leads to a page change into the simulation(s).

The Test Conductor user interface shares the same login screen as the Flight Operator interface. Though, there is a link that will bring up a login page that is solely for the Test Conductor. Upon successful login, there is a page change that will lead to a dashboard of accessible information to the Test Conductor.

## Hardware Interfaces

The primary hardware component expected for the user to use with the system is an internet-connected device that has the capability of inputting commands through touchscreen or by keyboard and can run one of the following compatible web browsers:

* Google Chrome
* Microsoft Edge
* Safari

Devices that the users will utilize include devices such as desktops, laptops, mobile devices, and tablets.

## Software Interfaces

The Django web application and simulation will connect to a MySQL lite database. The web application needs the database to get and verify login data, and the web application saves new users to the database. Once logged in the web application uses the database to create and save simulation data. The simulation needs the database to pull simulation data and set up the simulation based on the TC specifications. The simulation saves simulation results back to the database. The database is required for a simulation to end because there is no communication between the web application and simulation, so the database has a flag that when the simulation reads it as off, the simulation ends.

## Communications Interfaces

Software Communication: Being a webpage, the web browsers (also stated above) that this website will be compatible for are:

* Google chrome
* Safari
* Microsoft Edge
* Chromium based web browsers

Note: Firefox and possibly some other unlisted browsers have proved to have unwanted defects and they should be avoided.

HTTPS will be the communication standard that the webpage will use. There will be a login system for the webpage, so encryption of a password may be needed, however, information like the student’s school email address may serve as enough credentials for the project.

Internal communications architecture includes interactions between the simulation and the database, and communications between Django and SWA.

Hardware Communication: All hardware components will communicate with each other through browser interaction.

# System Features

## SWA - STaTE Web Application

4.1.1 Description and Priority

Priority: High

SWA is a hosted web application that serves as the platform for the project’s sub-applications and features. Top-level site traffic is routed through the STaTE Web Application. SWA also maintains a Home Page, Contact Page, and About Page.

4.1.2 Stimulus/Response Sequences

§ User enters (URL TBD) into web browser. SWA responds by displaying the site Home Page in the browser window.

§ User enters (URL TBD)/Home into web browser. SWA responds by displaying the site Home Page in the browser window.

§ User enters (URL TBD)/About into web browser. SWA responds by displaying the About Page in the browser window.

§ User enters (URL TBD)/Contact into web browser. SWA responds by displaying Contact Page in the browser window.

§ Test Conductor enters (URL TBD)/TC into web browser. SWA responds by displaying Test Conductor Login Page defined by the Test Conductor User Interface feature in the browser window.

§ Flight Operator enters (URL TBD)/fo into web browser. SWA responds by displaying the Flight Operator Login Page defined by Flight Operator User Interface feature in the browser window.

§ User selects “Login” button on Home Page. SWA responds by displaying the Flight Operator Login Page defined by Flight Operator User Interface feature in the browser window.

§ Flight Operator selects “Create an Account” button on Home Page. SWA responds by displaying the Flight Operator Login Page defined by Flight Operator User Interface feature in the browser window.

§ User selects “About” button on Home Page. SWA responds by displaying the About Page in the browser window.

§ User selects “Contact” button on Home Page. SWA responds by displaying Contact Page in the browser window.

§ User selects “Home” button on About Page. SWA responds by displaying Home Page in the browser window.

§ User selects “Home” button on Contact Page. SWA responds by displaying Home Page in the browser window.

4.1.3 Functional Requirements

SWA-FR-Section-I: Maintained URLs

SWA-FR-1: SWA shall maintain a hosted web application at (URL TBD).

SWA-FR-2: SWA shall maintain a Home Page at (URL TBD)/Home.

SWA-FR-3: SWA shall maintain an About Page at (URL TBD)/About.

SWA-FR-4: SWA shall maintain a Contact Page at (URL TBD)/Contact.

SWA-FR-5: SWA shall maintain a Flight Operator django app at (URL TBD)/fo.

SWA-FR-6: SWA shall maintain a Test Conductor django app at (URL TBD)/tc.

SWA-FR-Section-II: Page URLs

SWA-FR-7: SWA shall navigate to (URL TBD)/Home when a user enters (URL TBD) in their browser.

SWA-FR-8: SWA shall display the Home Page when a user enters (URL TBD)/Home in their browser.

SWA-FR-9: SWA shall display the About Page when a user enters (URL TBD)/About in their browser.

SWA-FR-10: SWA shall display the Contact Page when a user enters (URL TBD)/Contact in their browser.

SWA-FR-Section-III: Home Page Definition

SWA-FR-11: Home Page shall display a “Login” button.

SWA-FR-12: The “Login” button displayed on Home Page shall navigate the user’s browser to (URL TBD)/fo when selected.

SWA-FR-13: Home page shall display an “About” button.

SWA-FR-14: The “About” button displayed on Home Page shall navigate the user’s browser to (URL TBD)/About when selected.

SWA-FR-15: Home Page shall display a “Contact” button.

SWA-FR-16: The “Contact” button displayed on Home Page shall navigate the user’s browser to (URL TBD)/Contact when selected.

SWA-FR-17: Home Page shall display a welcome message and short description of the STaTE project.

SWA-FR-Section-IIII: About Page Definition

SWA-FR-18: About Page shall display a “Home” button.

SWA-FR-19: The “Home” button displayed on About Page shall navigate the user’s browser to (URL TBD)/Home when selected.

SWA-FR-20: About Page shall display information about the STaTE project including intended usage and copyrights.

SWA-FR-Section-IIIII: Contact Page Definition

SWA-FR-21: Contact Page shall display a “Home” button.

SWA-FR-22: The “Home” button displayed on Contact Page shall navigate the user’s browser to (URL TBD)/Home when selected.

SWA-FR-23: Contact Page shall display contact information for the developers and the administrators of STaTE.

## FOP - Flight Operator Platform

4.2.1 Functional Requirements

Priority: High

Each student in a group is a Flight Operator (FO). The Flight Operator Platform gives the FO the ability to interact with the control console to change the simulated spacecraft data of a particular subsystem.

4.2.2 Stimulus/Response Sequence

§ The Flight Operator can edit subsystem data while on the Flight Operator Platform. The Flight Operator Platform responds by passing the Control Console the changes.

§ The Flight Operator Platform passes changes made to a subsystem by the Flight Operator. The Control Console responds by changing the subsystem data to the parameters provided by the Flight Operator Platform.

4.2.3 Functional Requirements

FOP-FR-1: The FOP shall enact a visual change of a subsystem when a student changes a value.

FOP-FR-2: The FOP shall accurately pass user inputs to the control console.

## TCP - Test Conductor Platform

4.3.1 Description and Priority

Priority: High

TC is one of the apps routed to by the SWA. This app serves as the login, verification, and interface for a TC user, assuming login credentials are passed. TCA maintains a Login Page (AKA TC Page), TCHome Page, Simulation Settings Page, Simulation Display Page, Simulation Record Page.

* TC Page: The login page for TC user
* TCHome Page: The home screen for a TC user after login
* Simulation Management Page: The page a TC would use to create, edit, start, and delete upcoming simulations.
* Simulation Display Page: The page a TC would use during an active simulation. The TC should be able to pause and end simulations on the page. This page would also handle real-time anomalies.
* Simulation Record Page: The page a TC would use to access all simulation records, including student reports and student comms.

4.3.2 Stimulus/Response Sequences

§ User enters (URL TBD) into web browser. SWA responds by displaying the site’s Home Page in the browser window.

§ User enters (URL TBD)/Home into web browser. SWA responds by displaying the site’s Home Page in the browser window.

§ User enters (URL TBD)/TC into web browser. SWA responds by displaying Test Conductor Login Page defined by Test Conductor User Interface feature in the browser window.

§ User clicks button (example: “To Login as TC, click here”) on (URL TBD)/Home. SWA responds by routing user to (URL TBD)/TC.

§ User enters login credentials into the respective text boxes. User selects “Login” button on TC Login Page. TCA responds by validating entered data with data in database. TCA passes or fails the data. On pass, user is routed to (URL TBD)/TC/TCHome.

§ Test Conductor selects “Simulation Management” button on TCHome Page. TCA responds by displaying Simulation Management Page in the browser window.

§ Test Conductor selects a simulation to start from Simulation Management Page. TCA responds by displaying the Simulation Display Page in the browser window.

§ Test Conductor selects “ Simulation Records” button on TCHome Page. TCA responds by displaying Simulation Records Page in the browser window.

4.3.3 Functional Requirements

TCP-FR-Section-I: Maintained URLs

TCP-FR-1: TCP shall maintain a hosted web application at (URL TBD).

TCP-FR-2: TCP shall maintain a TC Home Page at (URL TBD)/TCHome.

TCP-FR-3: TCP shall maintain a Simulation Management Page at (URL TBD)/TCSimManage.

TCP-FR-4: TCP shall maintain a Simulation Display Page at (URL TBD)/TCSimDisplay.

TCP-FR-5: TCP shall maintain a Simulation Records Page at (URL TBD)/TCSimRecords.

TCP-FR-Section-II: URL Navigation

TCP-FR-6: SWA shall navigate to (URL TBD)/TC when a user clicks button to login as TC.

TCP-FR-7: TCP shall navigate to (URL TBD)/TCHome when a user enters the correct login credentials.

TCP-FR-8: TCP shall navigate to (URL TBD)/TCSimManage when a TC clicks button to manage simulations.

TCP-FR-9: TCP shall navigate to (URL TBD)/TCSimDisplay when a TC clicks button to start a simulation on the TC Simulation Management Page.

TCP-FR-10: TCP shall navigate to (URL TBD)/TCSimRecords when a TC clicks button to view Simulation records.

TCP-FR-11: TCP shall navigate to (URL TBD)/TCHome from any TC Page “Home” button.

TCP-FR-Section-III: TC Login Page Definition

TCP-FR-12: Login Page shall display 2 text boxes: 1 for username, 1 for password.

TCP-FR-13: Login Page shall display a “Login” button.

TCP-FR-14: Login Page shall display a “Login as student” button that navigates user back to (URL TBD)/FO/Login.

TCP-FR-Section-IV: TC Home Page Definition

TCP-FR-15: TC Home Page shall display a “Manage Simulations” button.

TCP-FR-16: The “Manage Simulations” button displayed on TC Home Page shall navigate the TCP user’s browser to (URL TBD)/TCSimManage when selected.

TCP-FR-17: TC Home Page shall display a “Simulation Records” button.

TCP-FR-18: The “Simulation Records” button displayed on TC Home Page shall navigate the TCP user’s browser to (URL TBD)/TCSimRecords when selected.

TCP-FR-Section-V: Simulation Management Page Definition

TCP-FR-19: TC Simulation Management Page shall display an “Add new Simulation” button.

TCP-FR-20: TC Simulation Manage Page shall display an “Edit Simulation” button.

TCP-FR-21: TC Simulation Manage Page shall display a “Start Simulation” button.

TCP-FR-22: The “start simulation” button displayed on Simulation Management Page shall navigate the TC User’s browser to (URL TBD)/TCDisplay when selected.

TCP-FR-23: The “Home” button displayed on the Simulation Management Page shall navigate the TC user’s browser to (URL TBD)/TCHome when selected.

TCP-FR-Section-VI: Simulation Display Page Definition

TCP-FR-24: The Simulation Display Page shall display information about the active simulations.

TCP-FR-25: The Simulation Display Page shall allow the TC to enter anomalies.

TCP-FR-26: The Simulation Display Page shall allow the TC to pause/stop an active simulation.

TCP-FR-27: The “Home” button displayed on the Simulation Display Page shall navigate the TC user’s browser to (URL TBD)/TCHome when selected.

TCP-FR-Section-VII: Simulation Records Page Definition

TCP-FR-28: The Simulation Records Page shall display information about the previous simulations, including student reports and simulation comms.

TCP-FR-29: The Simulation Records Page shall allow the TC to download student records.

TCP-FR-30: The “Home” button displayed on the Simulation Records Page shall navigate the TC user’s browser to (URL TBD)/TCHome when selected.

## Control Console

4.4.1 Description and Priority

Priority: High

The simulated control consoles each provide monitoring, command, and control capabilities of a specific system aboard SimCraft. The console simulations approximate how a real world control team performs space flight operations procedures. Each console interfaces with the appropriate simulated systems in SimCraft to allow for monitoring and control of those systems.

4.4.2 Stimulus/Response Sequence

§ The Flight Operator Platform passes the Control Console the user inputted changes. The Control Console responds by passing the changes to SimCraft.

§ The SimCraft passes the Control Console updated subsystem data. The Control Console responds by updating the interface for the user to see.

§ The SimCraft passes the Control Console what data to flag. The Control Console updates the interface with the correct flag.

4.4.3 Functional Requirements

Note: Each section is one iteration of a Control Console.

Control Console-FR-Section I: The ACS console

Control Console-FR-1: The Control Console shall present the indicators given by SimCraft.

Control Console-FR-2: The Control Console shall allow commands for Angle of Incidence between 0 and 90 degrees.

Control Console-FR-3: The Control Console shall allow commands for Rotational Drift from 0.0 to 2.0 degrees/hour.

Control Console-FR-4: The Control Console shall allow commands for the Fuel Level Sensor #1 from 0 to 100%.

Control Console-FR-5: The Control Console shall allow commands for the Fuel Level Sensor #2 from 0 to 100%.

Control Console-FR-6: The Control Console shall allow commands for the Fuel Pressure Sensor #1 from 0 to 30 psi.

Control Console-FR-I7: The Control Console shall allow commands for the Fuel Pressure Sensor #2 from 0 to 30 psi.

Control Console-FR-Section II: The PDS console

Control Console-FR-8: The Control Console shall provide monitoring capabilities for the Power Distribution Subsystem.

Control Console-FR-Section III: The Payload Console

Control Console-FR-9: The Control Console shall provide monitoring capabilities for the payloads.

Control Console-FR-10: The Control Console shall provide command capabilities for the payloads.

## Simulation Engine

4.5.1 Description and Priority

Priority: High

Simulation environment for a spacecraft. Anomalies (either random or controlled by test conductor) will be simulated in this program where students will have to come up with responses to the issues presented and those solutions should reflect the results of their decisions. This subsystem will also be the class that actually runs all the math behind the trajectory, batteries, etc. using initial conditions given to it by the Test Conductor using it to update Simcraft objects created and managed by the Simulation Engine.

4.5.2 Stimulus/Response Sequences

§ The Test Conductor interacts with simulation engine attributes. The Simulation Engine responds by altering an event of a deployed object.

§ The student will interact with the simulation through the web page as a Test Conductor or Flight Director/Operator.

4.5.3 Functional Requirements

Simulation Engine-FR-Section-I: Attitude and Control Subsystem Console

Simulation Engine-FR-1: The Simulation Engine Attitude and Control console shall control SimCraft in a Low Earth Orbit (LEO), with a period of 90 minutes.

Simulation Engine-FR-2: The Simulation Engine Attitude and Control console shall control the SimCraft’s exposure to sunlight for 45 minutes and the Earth's shadow for 45 minutes.

Simulation Engine-FR-3: The Simulation Engine Attitude and Control console shall control sensors related to the Attitude and Control Subsystem, with nominal and off-nominal values.

Simulation Engine-FR-Section-II: Power Distribution Subsystem Console

Simulation Engine-FR-4: The Simulation Engine Power Distribution console shall provide power to SimCraft via power stored in the system’s batteries.

Simulation Engine-FR-5: The Simulation Engine Power Distribution console shall distribute power as needed to the payload.

Simulation Engine-FR-6: The Simulation Engine Power Distribution console shall have the ability to distribute power to the payload as needed, with excess power stored in the system’s batteries.

Simulation Engine-FR-7: The Simulation Engine Power Distribution console shall have the ability to distribute power in the batteries to the SimCraft when the object is within Earth’s shadow.

Simulation Engine-FR-8: The Simulation Engine Power Distribution console shall monitor solar panel power production, related to the angle of incidence with the sun, where the angle of incidence is defined as the angle between a line normal to the surface of the solar panel and the line pointing to the SimCraft to the sun.

Simulation Engine-FR-9: The Simulation Engine Power Distribution console shall have the ability to control the angle of incidence of the SimCraft only to one degree of rotational freedom.

Simulation Engine-FR-10: The Simulation Engine Power Distribution console shall monitor the ACS, presumed to maintain an incidence angle of +/- 5 Degrees when operating normally, allowing for maximal energy capture during the daylight portion of the SimCraft orbit.

## SimCraft

4.6.1 Description and Priority

Priority: Medium

Class or classes holding all information relating to the spacecraft/satellite. Will be created and managed by Simulation Engine.

4.6.2 Stimulus/Response Sequences

§ Simulation Engine creates SimCraft giving required parameters

§ SimCraft returns information when requested to Simulation Engine

4.6.3 Functional Requirements

SimCraft-FR-Section-I: SimCraft System

SimCraft-FR-1:The SimCraft shall monitor the spacecraft's angle of incidence with 0-5 Degrees as nominal indicator (green), 5-30 Degrees as off-nominal warning (yellow), and 30-90 Degrees as off-nominal error (red).

SimCraft-FR-2: The SimCraft shall monitor the spacecrafts rotational drift with 0.0-0.1 Deg/hr as nominal indicator (green), 0.1-2.0 Deg/hr as off-nominal warning (yellow), and greater than 2 Deg/hr as off-nominal error (red).

SimCraft-FR-3: The SimCraft shall monitor the spacecrafts Fuel Level Sensor #1 with 30-100% as nominal indicator (green), 10-30% as off-nominal warning (yellow), and 0-10% as off-nominal error (red).

SimCraft-FR-4: The SimCraft shall monitor the spacecrafts Fuel Level Sensor #2 with 30-100% as nominal indicator (green), 10-30% as off-nominal warning (yellow), and 0-10% as off-nominal error (red).

SimCraft-FR-5: SimCraft shall monitor the spacecrafts Fuel Pressure Sensor #1 with 10-30 psi as nominal indicator (green), 5-10 psi as off-nominal warning (yellow), and 0-5 psi as off-nominal error (red).

SimCraft-FR-6: The SimCraft shall monitor the spacecrafts Fuel Pressure Sensor #2 with 10-30 psi as nominal indicator (green), 5-10 psi as off-nominal warning (yellow), and 0-5 psi as off-nominal error (red).

SimCraft-FR-Section-II: Solar Panel Subsystem

SimCraft-FR-7: The Solar Panel Subsystem shall monitor the SimCraft orbit, and solar arrays alignment showing energy gathering efficiency.

Simulation Engine-FR-1: The Solar Panel Subsystem shall monitor power intake when the SimCraft’s solar panels are pointed in the sun’s direction.

SimCraft-FR-Section-III: Payload Subsystem

SimCraft-FR-8: SimCraft shall have 3 payloads to manage independent of each other.

SimCraft-FR-9: The payload subsystem shall manage required power for each independent payload.

SimCraft-FR-10:The payload subsystem shall show power and general status indicators.

# Other Nonfunctional Requirements

## Performance Requirements

The performance of the system will be based on key assumptions that are based on the requirements of the customer. To reiterate, this product is to be used as a student learning tool for a high-level class that is taught primarily in-person. With this in mind, we can expect a relatively low number of local users.

The assumptions of the users going forward will be the following:

* The maximum number of users is expected to be 50 individual users, where each user is connecting from a different device.
* The maximum number of users will not increase during the development of this product, nor in the years following the delivered final product.
* The location of the user is going to be in the same region as the location of the hosting of this product.
* Because of the two campuses of ERAU, and due to the uncertainty if this product will be used in the Prescott Campus, this product will take into consideration the possible student body connecting from Prescott in conjunction with the students connecting from Daytona Beach.

Efficiency

PR-1: The program shall not exceed a response time of 10 ms from connections made in the United States, unless the current number of users exceeds the maximum number of users.

PR-2: The program shall not exceed a response time of 20 ms from the connections made in the United States, unless the current number of users exceeds double the maximum number of users.

PR-3: The program shall be capable of running <n> simulated missions in tandem while connected with the maximum number of users.

PR-4: Local spacecraft data written regarding a simulated spacecraft shall not exceed 2 gb in size per simulated mission.

Reliability

PR-5: The program shall update data transmitted to the connected user no less than every <n> seconds.

PR-6: The program shall save the simulated spacecraft data every <n> hours.

PR-7: The program shall save all flight conductor and flight operator inputs.

Survivability

PR-8: If the program suffers an error that ceases runtime, the program shall not lose any local data regarding conductor inputs.

PR-9: If the program suffers an error that ceases runtime, the program shall not lose any local data regarding a simulated spacecraft.

PR-10: If the program suffers an error that ceases runtime, the program shall attempt to restart the corresponding simulation using the latest version of the saved data.

PR-11: If the program suffers an error that ceases runtime, the program shall alert all Test Conductor users.

## Safety Requirements

No safety requirements as there are no physical components to the application.

## Security Requirements

For security requirements, the only people who shall access this webpage are the intended users and the developers.

Intended users are defined as the teachers, teacher assistants, and students of ERAU.

Teachers and teacher assistants shall be the only ones in the intended users group that shall be allowed to read and alter the simulation, as well as any saved data. Teachers and teacher assistants shall also be able to access Flight Operator, Flight Conductor and Test Conductor controls.

Students can access the webpage, but only as a Flight Operator or a Flight Conductor.

The team will be aware that improper awareness of the safety/security risks could possibly lead to consequences not only for the project, but also the school.

Possible harm that can come from using the simulation/webpage will come from any unintended vulnerabilities that may arise when using a server. Currently as it stands, only developers have access to the simulation code and webpage code, but without proper care, an ERAU IT vulnerability may be present.

The team will be aware that improper awareness of the safety/security risks could possibly lead to consequences not only for the project, but also the school.

## Software Quality Attributes

**5.4.1 - Usability**

SQA-1: The program shall be intuitive for the user to interact with, requiring little to no explanation for interacting with the program.

SQA-2: After login, both user interfaces shall have a locatable navigation pane on each page.

SQA-3: After login, both user interfaces shall have a locatable “Logout” button on each page.

SQA-4: On both login pages, the username vs password text box must be clearly marked and locatable.

SQA-5: On both login pages, there must be a marked and locatable “Login” button.

SQA-6: The program shall be intuitive for teachers to add anomalies in <n> clicks.

SQA-7: The program shall be intuitive for teachers to set up scenarios in <n> clicks.

**5.4.2 - Interoperability**

SQA-8: The program shall be cross-platform compatible with all major browsers as of writing this document (Oct 2022).

**5.4.3 - Reusability**

SQA-9: The program shall be reusable, so that it may act as a teaching tool long after the project’s development.

**5.4.4 - Maintainability**

SQA-10: The program shall be reliable, so that the department using this program will not have to (a) go through the code, or (b) employ further senior design groups to attempt to fix the program.

## Business Rules

* Only professors/instructors should have the role of Test Conductor unless stated otherwise
* Flight Director is assigned by Test Conductor

# Other Requirements

## 6.1 - Database Requirements

DR-1: The Database shall store simulation states, which consist of all information of each simulation at a given time.

DR-2: The database shall be able to send and receive data sets from the program.

DR-3: The database shall be able to store text communications from users after a simulation has concluded.

**Appendix A: Glossary**

DR - Database Requirement.

FOP - Flight Operator Platform.

FR – Functional requirement. Describes an atomic piece of system functionality that is testable.

HTTPS - Hypertext Transfer Protocol Secure

SimCraft - Simulated spacecraft. Within the STaTE system, these are created by a test conductor and managed by a flight operator.

STaTE – Spacecraft-Control-Center Training and Training Environment.

SWA – STaTE Web Application.

TCP - Test Conductor Platform.

URL – Uniform Resource Locator.

**Appendix B: To Be Determined List**

Documentation specified in section 2.6: User Documentation is slated to be completed during sprint 5.