**Feature Documentation:**

**NASA EVA Path Phase 4**

**Version 3.0**

**Developers/Testers**

Tristan Arjoon

Mario Curtis

Shane Farmer

Claudel Guembu

Sean Johnson

**SWEN 670 9040 (2188)**

**UMUC Graduate School of Management and Technology**

**Software Engineering Project**

**02 December 2018**

**Fall Semester**

**Table of Contents**

Contents

[Revision History 3](#_Toc531513789)

[1. Introduction 4](#_Toc531513790)

[1.1 Background 4](#_Toc531513791)

[1.2 Intent 4](#_Toc531513792)

[2. Development 5](#_Toc531513793)

[2.1 Code Additions or Modifications 5](#_Toc531513794)

[2.2 Bug fixes 5](#_Toc531513795)

[3. Functional Testing 5](#_Toc531513796)

[3.1 Proposed Functional Test Cases 5](#_Toc531513797)

[3.1.1 Test Case 1: Load correct ISS model on application startup 5](#_Toc531513798)

[3.1.2 Test Case 2: Test all newly configured handrails to produce Route Results 6](#_Toc531513799)

[3.1.3 Test Case 3: Test legend UI is displayed appropriately 7](#_Toc531513800)

[3.2 Assumptions and Constraints 8](#_Toc531513801)

[3.2.1 Assumptions 8](#_Toc531513802)

[3.2.2 Constraints 9](#_Toc531513803)

[References 9](#_Toc531513804)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Author | Date | Description |
| 1.0 | Tristan Arjoon | 10/16/18 | Created & Updated Draft Content |
| 1.1 | Mario Curtis | 10/21/18 | Updated Assumptions and Constraints |
| 1.2 | Shane Farmer | 10/21/18 | Updated sections 1 through 3.12 |
| 2.0 | Shane Farmer | 11/11/18 | Updated all of version 2 |
| 2.1 | Tristan Arjoon | 11/11/18 | Finalize all information / formatting |
| 3.0 | Tristan Arjoon | 12/1/18 | Updated and Finalized Milestone 4 |
|  |  |  |  |

**NOTE: Future phases, please see Design Document, “Phase 4 Milestone 4”, for an update on where phase 4 left off. Section 2 of the Design Document is a great summary of how to get everything up and running (see manual for more detailed steps). Once everything is installed, “run\_nasa.bat” in the ‘nasa-path-finder’ folder will compile and load everything (may need to run it twice, the first time).**

# 1. Introduction

The purpose of this document is to track and provide a concise description of features/functions of the NASA EVA Path Finder system. Each feature is based off high-level requirements as listed in the project backlog. This documentation covers features developed by Phase 4 and is intended to be integrated with prior documentation at the close of Phase 4.

## 1.1 Background

Users of the software are astronauts and the people who train them and write their spacewalk procedures. The software will be used to train astronauts on how to perform spacewalks, plan astronaut spacewalks, and write astronauts procedures.

The user will select originating and destination handrails for one or more astronauts and will be provided (by the software) a step-by-step path to guide simultaneous space walks while avoiding hazards, such as antennas, sharp edges, and astronaut to astronaut collisions. The user will be able to select handrails with a drop-down menu or by clicking specific handrails on a displayed model. As the space station is configurable and modules/exterior components may move, the user will be able to upload updated models to keep up with the most current configuration.

## 1.2 Intent

The intent of this document is to give both development teams and stakeholders a high-level view of features as they are iteratively added. It is intended that this document be integrated with past and future documentation.

# 2. Development

## 2.1 Code Additions or Modifications

* Completely, matched all .str data with .stl data. The entire ISS is now suspended in space with working handrails.
* The entire ISS load flawlessly automatically when “run\_nasa.bat” is executed.
* All handrails load flawlessly automatically when “run\_nasa.bat” is executed.
* Reduce brightness of the ISS
* Designed and implemented a legend for the UI

## 2.2 Bug fixes

Several bug fixes were implemented, majority coming being displayed by UI. This has been corrected via ‘control.js” file. Please reference Design document for more information.

# 3. Functional Testing

## 3.1 Proposed Functional Test Cases

### 3.1.1 Test Case 1: Load correct ISS model on application startup

**Description:** Tests that entire ISS model (correct color, brightness, and handrails) is loaded and rendered on application startup.

**Requirements:** none

**Prerequisites:** none

**Steps:**

**1.** Start application

**2.** Entire ISS is displayed in the model view area (Fig 3.2.1)

**Input:**none

**Expected Output:** The entire ISS model is displayed on startup with correct color, brightness, and all handrails placement.

**Assumptions:** None

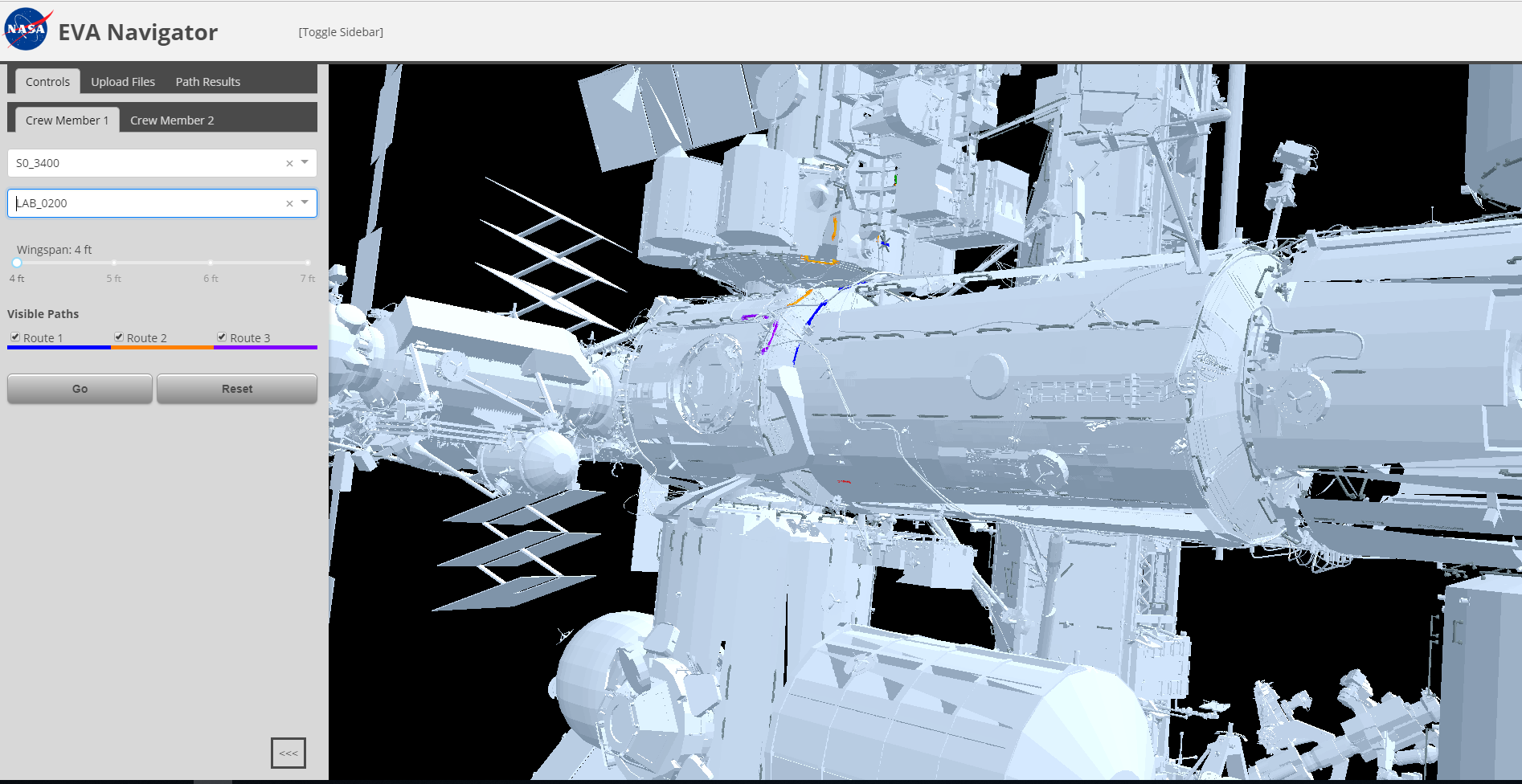


Figure 3.2.1 - Entire ISS shown on startup

### 3.1.2 Test Case 2: Test all newly configured handrails to produce Route Results

**Description:** Tests that all handrails are populated in the start then run through the application to test functionality

**Requirements:** none

**Prerequisites:** none

**Steps:**

**1.** Start application

**2.** Choose a starting point

**3.** Choose an ending point

**4.** Click on “Go” button

**5.** Click on Path Results tab to see results (Figure 3.2.2)

**Input:**none

**Expected Output:** All handrails show functionality. Path results tab shows information when ran.

**Assumptions:** None

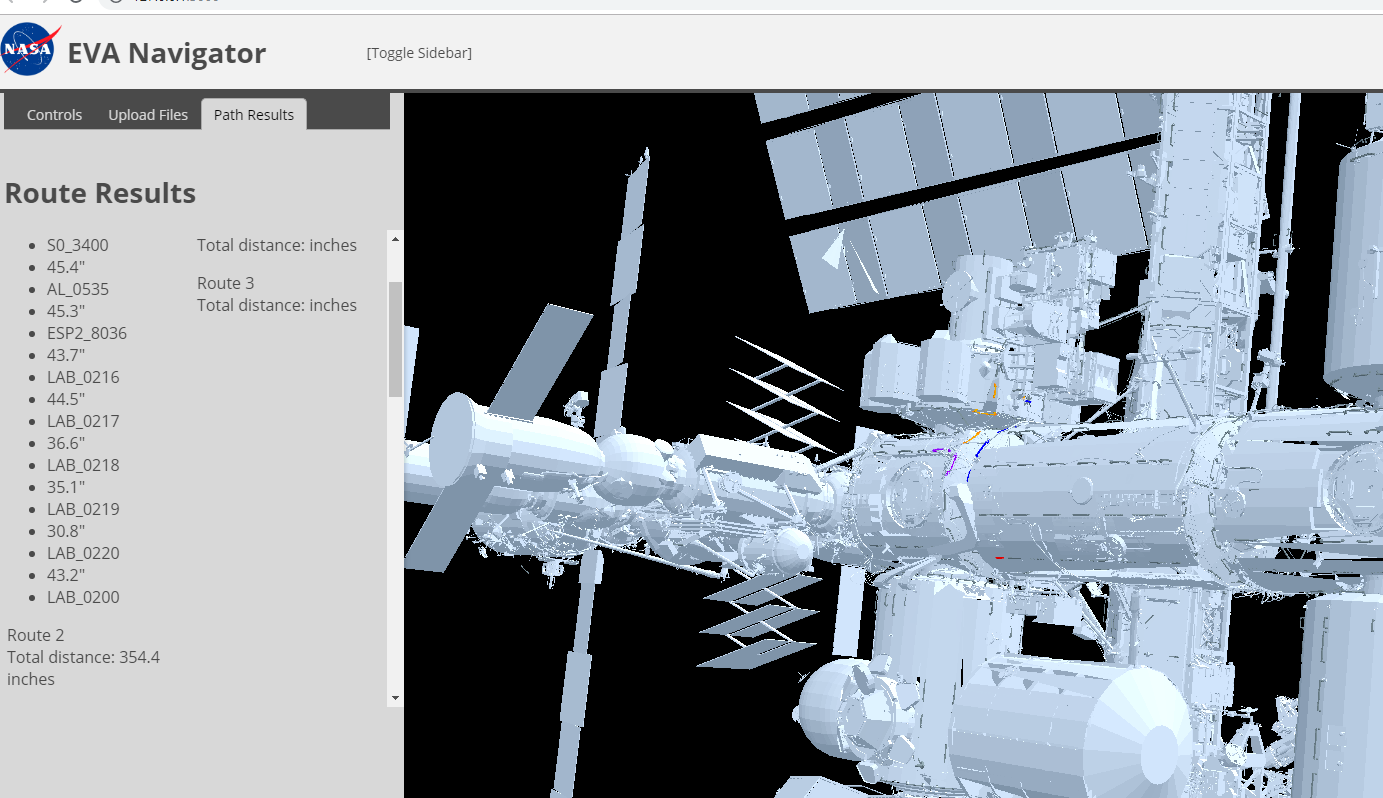


Figure 3.2.2 – Handrail pathing results

### 3.1.3 Test Case 3: Test legend UI is displayed appropriately

**Description:** Tests that the legend UI is present upon application execution in both tabs

**Requirements:** none

**Prerequisites:** none

**Steps:**

**1.** Start application

**2.** Look for legend in “Crew Member 1” tab

**3.** Look for legend in “Crew Member 2” tab

**Input:**none

**Expected Output:** A UI displayed.The user can use the UI legend to understand how to maneuver the ISS.

**Assumptions:** None

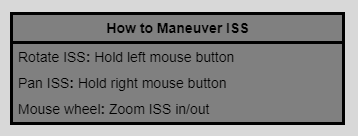
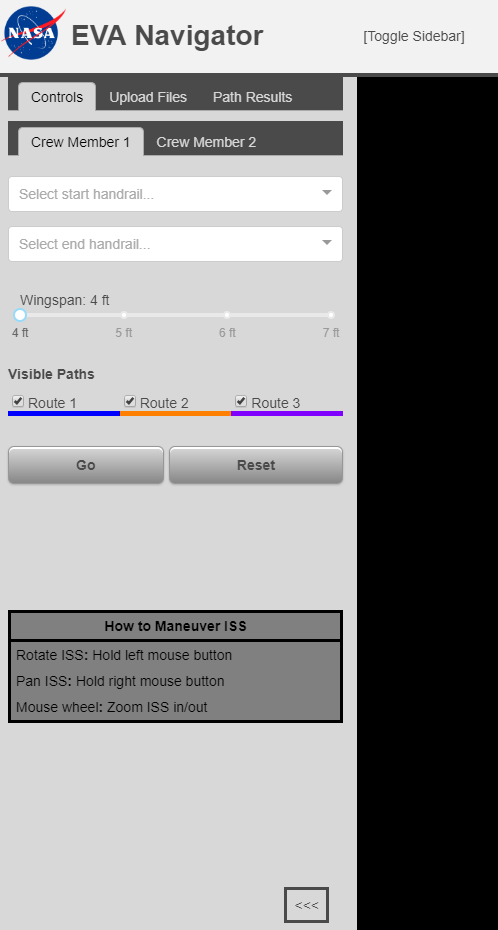


Figure 3.2.3 – UI Legend for ISS



Placement of legend in UI

## 3.2 Assumptions and Constraints

### 3.2.1 Assumptions

This test plan assumes that the installation document has been followed in the described order and on a system compatible with the application as outlined in the system and software requirements.

### 3.2.2 Constraints

This test plan only performs function tests on requirements outlined in the Design Document. Any requirements that were incomplete or inadequately captured will not have coverage here as a result.

# References

* Guembu, C. (2018). Navigation for space walks. Retrieved from <https://github.com/claudelguembu/nasa-path-finder>