CS Capstone Requirements Document

April 19, 2019

NASA USLI Rocket Team

Prepared for

Dr. Nancy Squires

Prepared by

CS Capstone Team

Trey Elkins
Leif Tsang
Ryan Wallerius

Abstract

This document outlines the client requirements for the computer science portion of the Oregon State University's entry into the NASA University Student Launch Initiative. The systems outlined in this requirements document entail rocket tracking and avionics, rover payload operations, and management of the team website.

CONTENTS

1 Revision Table

Sections have been incremented by 1 from our drafts to our final draft because of the addition of the revision table.

Section	Original	New
1.3,1.6	Overview	Now merged into 2.3 'Scope'. Some portions also removed/rewritten
2.4	Definitions	Added some definitions
2.1.1-2.1.3 now 3.2	System Interfaces	All rewritten and rephrased
3.2	N/A	Added Rover Section
3.3.3	N/A	Added Website functions:
3.1	N/A	Added gps to overall description.
3.1	N/A	Made payload operation more general
4	System features	All written and expanded upon, final requirements outlined and clarified

2 Project Overview

2.1 Introduction

The NASA Student Launch is a competitive experiential exploration and learning activity. It strives to provide relevant, cost-effective, grassroots research and development in the field of rocket propulsion and launch systems. This project offers multiple challenges reaching a broad audience of middle and high schools, colleges, and universities across the United States.

2.2 Purpose

This document outlines the customer and competition requirements for the Computer Science team participating in Oregon State University's entry into the 2019 NASA Student Launch Initiative.

2.3 Scope

This document contains the complete software requirements and specifications for the computer science portion of the NASA University Student Launch Initiative. The software can be organized into three parts:

1) **Launch Vehicle Avionics and Telemetry**, active tracking, logging, filtering, and transmission of coordinate data to a ground station.
2) **Payload Software**, must maneuver 10 feet away from the landed rocket with object avoidance and collect a soil sample specified by the competition requirements.
3) **Website**, the location where NASA will receive design and technical documents required for the competition as well as a place to display other information and facts about the team.

This document has been updated to reflect changes in customer requirements specified by NASA personnel and the team administration over the course of the competition.

2.4 Definitions, acronyms, and abbreviations

AIAA	American Institute of Aeronautics and Astronautics
CS	Computer Science
ECE	Electrical and Computer Engineering
ME	Mechanical Engineering
OSU	Oregon State University
USLI	University Student Launch Initiative
GPS	Global Positioning System

3 Overall description

3.1 Product Perspective

The launch vehicle avionics system will collect and log coordinate data from on-board GPS sensors. Data will be used for analysis of launch vehicle flight profile and drift radius for reporting in NASA documentation, as well as for recovery of the launch vehicle after flight.

The rover payload will use sensors to complete the rover mission along with GPS in order to move at least the specified distance away from the launch vehicle. The program will be written and managed by the CS team. The CS team will be in contact with the ECE team on decisions made for payload management and operations.

The Website will not be graded in the competition itself but is crucial for storage and retrieval of key technical documentation required by NASA for competition purposes. The Website will be managed and created by the CS team and will be published on a domain purchased by the OSU USLI Team.

3.2 System interfaces

The launch vehicle avionics software will interact with the GPS sensors and data logging via a Teensy 3.6 microcontroller running Arduino code.

The rocket avionics will interact with GPS sensors on-board the rocket to collect coordinate data during the flight via a serial interface. The Teensy will manage an XBee Pro S3 900 MHz transceiver (also via serial) that packs filtered data and sends it to a ground station utilizing similar hardware for data output and recovery purposes. The flight units are equipped with SD cards that log the data to text files for later recovery and analysis.

The payload software will control the rover through both a Teensy 3.6 microcontroller and BeagleBone Black microcomputer. The Teensy will run the primary script while the BeagleBone will be continuously processing images through computer vision. The BeagleBone will be connected to the Teensy via UART sending the Teensy instructions based off of processed images. All sensors and components will be directly connected to the Teensy. The Teensy will pull information from these sensors as necessary for the payload mission.

The Website will be hosted on a domain purchased by the OSU USLI team via the Github.io web service.

3.3 Product Functions

3.3.1 Avionics Functions

- Read coordinate data from GPS sensors.
- Log and filter GPS data.
- During flight, avionics will transmit data to ground station.
- Ground station displays serial data to host PC.
- After flight, ground station transmits trigger signal to payload ejection controller.

3.3.2 Payload Functions

- Read data from individual sensors.
- Be able to properly use object avoidance.
- Drive at least ten feet away from the landed rocket and collect a soil sample.

3.3.3 Website Functions

- Host critical NASA documents.
- Enable downloading of documents.
- Present information on the team and project.

3.4 User Characteristics

The intended users of these systems will be primarily ECE and CS team members - typically college-aged students familiar with software and high level electronics. The primary exception to this is the website, which will need to be navigable and easily usable by any adult, though the real target users will be NASA engineers, members of other teams, and potential sponsors who are curious about our team and documentation.

3.5 Constraints

3.5.1 Regulatory Policies

Competition restrictions dictate that our approach to various problems must accommodate competition requirements. Everything must comply with federal law and with the requirements enumerated in the Student Launch Initiative handbook. Exceptions or changes are at the discretion of the NASA committee orchestrating the competition. We cannot, for instance, transmit flight guidance data to the rocket as that would legally classify it as a guided missile. Another instance of competition policy is the requirement that all transceivers operate under 250 mW of power draw to minimize interference.

3.5.2 Memory and Processing Constraints

Both avionics and payload software must comply with the memory and processing constraints of the Teensy 3.6 as well as buffer and serial constraints of the XBee Pro S3 transceiver. Serial monitors and scripts can be run on Windows machines or any other computer that has support for standard programming languages.

3.5.3 Interfaces to Other Applications

Outside of situations such as radio-frequency interference, our systems must not interrupt or compromise the operation of any other team's systems at competition. In cases where all interaction cannot be eliminated - such as in use of commonplace radio-frequency bands - measures must be taken to minimize impact on the systems of other teams.

3.5.4 Higher-Order Language Requirements

Both the avionics and rover payload software must be written in a high level programming language such as the Arduino programming language. The website must also be implemented using tools or libraries other than just basic HTML, CSS, and JavaScript.

3.5.5 Safety and Security Considerations

The rocket will launch far enough away from spectators not to pose a safety threat. All bystanders must be far enough away from the PLEC receiver that the black powder charges and payload ejection don't cause any personal bodily harm.

3.6 Assumptions and Dependencies

The ECE/ME team will provide the computer hardware necessary to run the Rocket and Payload Programs. The ECE team will provide the CS team with sufficient time and information to test and debug our software using their final version of the flight hardware. Development will be supported by an array of standard libraries, with some non-standard libraries being used where possible and applicable. The team will be operating with a minimal budget - almost all hardware should be provided by AIAA or the USLI team, but some budgetary contributions might be necessary to secure the correct materials, especially if said materials aren't readily available through the CS capstone program.

4 Specific Requirements

4.1 System Features

4.1.1 Avionics

1) **Code Refactor:** Code must be streamlined and documented, with at least a 25% reduction in lines of code and thorough commenting.
2) **Broadcasting:** Avionics system must be capable of receiving GPS data and transmitting to the ground station with enough fidelity that the rocket can be recovered using the data after flight.
3) **Output:** Ground station must print to serial monitor on host PC so that data can be viewed and retrieved remotely.
4) **Data Logging and Storage:** Flight data must be logged, stored, and easily recoverable for data analysis.
5) **Ejection Control:** The ground station must be capable of sending an ejection signal to the payload ejection controller (PLEC) that will trigger black powder charges and eject the rover.
6) **433 MHz Prototype:** The 433 MHz prototype must be capable of transmission of plaintext data wirelessly between two separate units.

4.1.2 Payload Software

1) **Traversal away from air frame** The rover must move at least ten feet away from the landed rocket using a GPS distance algorithm
2) **Movement** The rover will have robust slow start and slow halt and turn movement features
3) **Object avoidance** The rover must avoid any large objects utilizing 2 front mounted sonar
4) **Soil Collection** The rover must drill for a soil sample utilizing an on board auger with a sealed collection chamber
5) **GPS** The rover will use GPS in order to record locations of collected soil and transmit the location to a ground station
6) **Magnetometer** The magnetometer will be able to output an integer ranging from 0 - 359 based on the heading of the rover
7) **Gyroscope** The rover will be able to use the Gyro in order to detect levelness of the rover at any given time
8) **Routing Algorithms** Routing algorithms will be in place to allow the rover to navigate based on GPS coordinates
9) **Logging** The rover will have the ability to log information to a microSD card on the teensy
10) **Computer Vision** The Beaglebone black on board the rover will be able to capture images and process each image for a target which will have the location transmitted via UART to the Teensy.

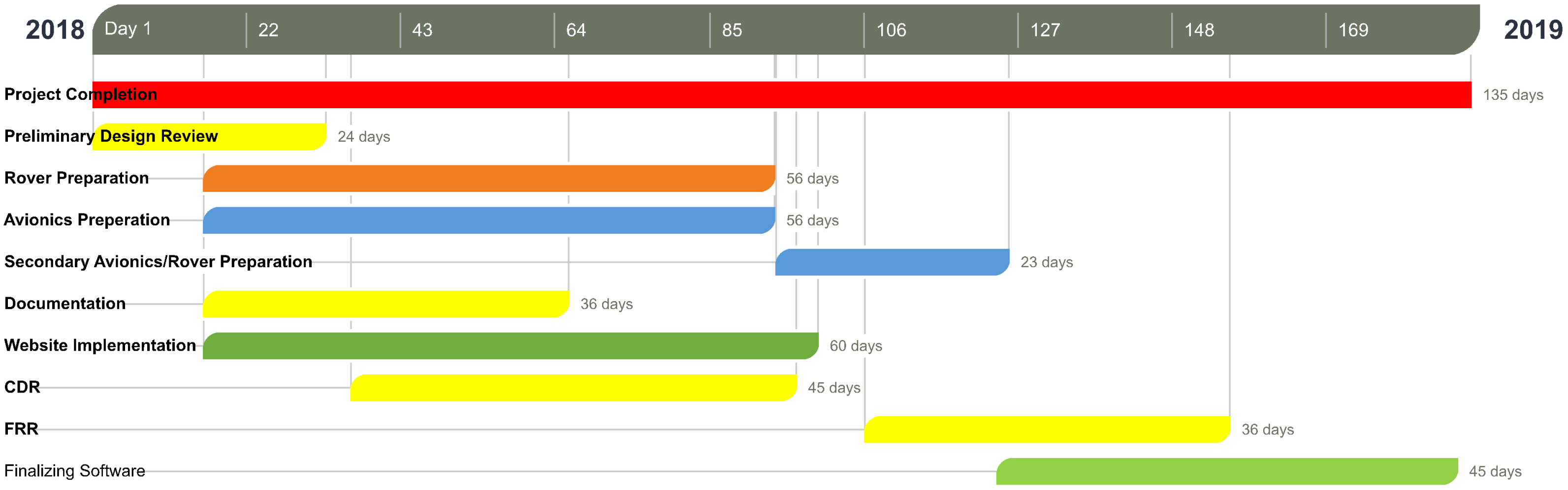
4.1.3 Website

The website will have the following features:

- **Home Page:** The home page will have the OSU USLI team's mission statement and mission picture.
- **About Us:** The about us page will have a list of the capstone seniors that are working on the page. This will include names, email addresses, the program they are in, and the system they are working on.
- **Deliverables:** The deliverables page will have all of the critical NASA documents needed for competition. Anyone will be able to access this page and download any document on the page.

All of this will be done using ReactJS framework.

4.2 Development Chart Timeline


Day 1
Project Completion
135 days
Preliminary Design Review
24 days
Rover Preparation
56 days
Avionics Preperation
56 days
Secondary Avionics/Rover Preparation
23 days
Documentation
36 days
Website Implementation
60 days
CDR
45 days
FRR
36 days
Finalizing Software
45 days