Table 1: Revision History

Date	**Developer(s)**	**Change**
Sept 25	Muhanad Sada	Workflow Plan, POC Demo Plan
Date2	Name(s)	Description of changes
...	...	...

Development Test
ProgName

Team #, Team Name
Student 1 name and macid
Student 2 name and macid
Student 3 name and macid
Student 4 name and macid

[Put your introductory blurb here. —SS]

1 Team Meeting Plan

2 Team Communication Plan

3 Team Member Roles

4 Workflow Plan

Team members will use the GitHub repository dedicated for the capstone project. The feature branch workflow will be used whenever there are any code changes except for simple fixes such as syntax errors, comments, variable renaming, etc. Branches will also be utilized for any significant documentation changes such as section additions/modifications and diagram insertions. Pull requests will be used in conjunction with branches to review/verify code and document changes. Branches will follow the following naming structure: scope/(description), ex. feat/adding new function

4.1 Issues

The issues feature in GitHub is used to track all of the tasks for the project. Once the team or individual members identify a task, an issue will be created. When creating an issue, a team member will select one of the issue templates based on the scope of the task. There are a total of five templates:

- Bug report - any tasks used to report a bug and fix it
- Feature request – any tasks that involve requesting and implementing a feature

- Enhancement - any tasks that require updating code for enhancement purposes
- Documentation – tasks that involve adding or editing documentation
- Miscellaneous – any tasks that are not covered under the scope of the other templates

4.2 Project Board

The project board is used to organize and identify the status of each task. The project contains five columns each describing the current status of the issue:

- To-do - When tasks are first created, they are placed in this list
- In-progress – The issue has been assigned to a team member and is currently being worked on
- In-Review – The work has been completed and now to needs to be reviewed
- Done – Once team member(s) review and approve the changes, the issue will be moved to this stage
- Outdated/Ignored – issues that were created but later determined to be unnecessary

5 Proof of Concept Demonstration Plan

The proof of concept demonstration should prove four essential functionalities of the product. The first is the ability of sensors to measure desired data and send that information to the hardware. The second is having the capability to receive/send data at three different levels, which includes hardware, desktop application, and database. The POC should be able to show that hardware can receive information from a sensor and send that information to a simple desktop application. The application should then be able to receive that data and display it on the GUI. At this point, the application would be able to send that information to a database, which is populated accordingly. The third ability, is to show live data on the application's GUI, however implementation difficulties are expected. This is due to the tediousness of creating a connection that provides both smooth and continuous data transfer between the hardware and the application. In addition to these functionalities, there is the risk of being constrained in testing as we might not have access to a formula E car or it will be difficult to duplicate. Therefore, the POC should also have a testing environment that mirrors the conditions/setup of the mechanical parts of a Formula E car upon taking measurements. If the implementation of the above essential abilities and testing environment are verified then the level of confidence of creating a successful data automation product.

6 Technology

- Specific programming language
- Specific linter tool (if appropriate)
- Specific unit testing framework
- Investigation of code coverage measuring tools
- Specific plans for Continuous Integration (CI), or an explanation that CI is not being done
- Specific performance measuring tools (like Valgrind), if appropriate
- Libraries you will likely be using?
- Tools you will likely be using?

7 Coding Standard

8 Project Scheduling

[How will the project be scheduled? —SS]