Autonomous Car Racing - Human Machine Interface

Project Specification

Alexis Morales Flores, William Chung

Project Charter

Project Overview

The goal for our project is to develop a human machine interface (HMI) that allows the pit crew of an autonomous race car to view and manage certain parameters of their vehicle, including its health, speed, and distance to objects. The primary customer for this project is the Indy Autonomous Challenge (IAC) team of Triton AI. The IAC is a series of races with full-scale autonomous race cars. The IAC team is in need of a simple, accessible HMI so they can easily monitor and control their race car. The HMI will need to have virtual instruments displaying important data about the race car as well as interactive elements that allow human operators to adjust race car parameters using buttons, sliders, etc.. The HMI will ideally need to be touch-screen enabled in order for an operator to monitor the car and control parameters at the same time from a portable device such as a tablet that can display data in real time. We aim to build a sort of virtual dashboard that not only displays data but also allows control of race car parameters.

Project Approach

Initially we plan on developing the first part of our HMI as a web-based tool. We will use standard web development technologies to create a graphical user interface (GUI) containing the virtual instruments and controls required to monitor car health and modify parameters such as speed. Through the use of a touch screen computer we will verify that the slide controls can be moved with the use of a finger in the same method that a slide volume control button works on a

smartphone or tablet. Our next step will be to connect to a VPN which will allow us to access a network through which we can control the car remotely and securely. Once this can be achieved we will then move our application to a device such as an iPad that can be used by the pit crew of the vehicle.

We plan to complete the project in 3 main phases.

Phase 1: Onboarding (Weeks 1-3)

- Communicate with IAC team and establish what requirements they have for the HMI
- Research how to build an HMI system and see if there are any existing solutions that can be applied

Phase 2: Prototyping (Weeks 4-5)

- Build working prototype of HMI displaying data through virtual instruments and accepting control input through buttons, sliders, etc.
- Test with dummy data (static data read from a file)

Phase 3: Iteration (Weeks 6-10)

- Improve on prototype with feedback from IAC team
- Test with real-time data (data from simulation or from actual hardware on race car)

Minimum Viable Product

Our minimum viable product (MVP) will be a web app that contains a slide control system that adjusts the speed of the vehicle and monitors the car's health. Iteration on the MVP will be guided by feedback from the IAC team who is the primary customer for the project.

Specific objectives of the project are to achieve:

1. Telemetry: receive input data from the race car and display it to the user
2. Teleoperation: send commands to the race car to control parameters

Which specific metrics to display and parameters to control will be determined by the IAC team. The MVP will only contain a very limited set of metrics and parameters compared to a full-fledged HMI.

Longer term goals:

- Expand HMI to include nice-to-have features determined by the IAC team (expand set of metrics and parameters)
- Design software pipeline that will allow for easier development of the HMI in the future

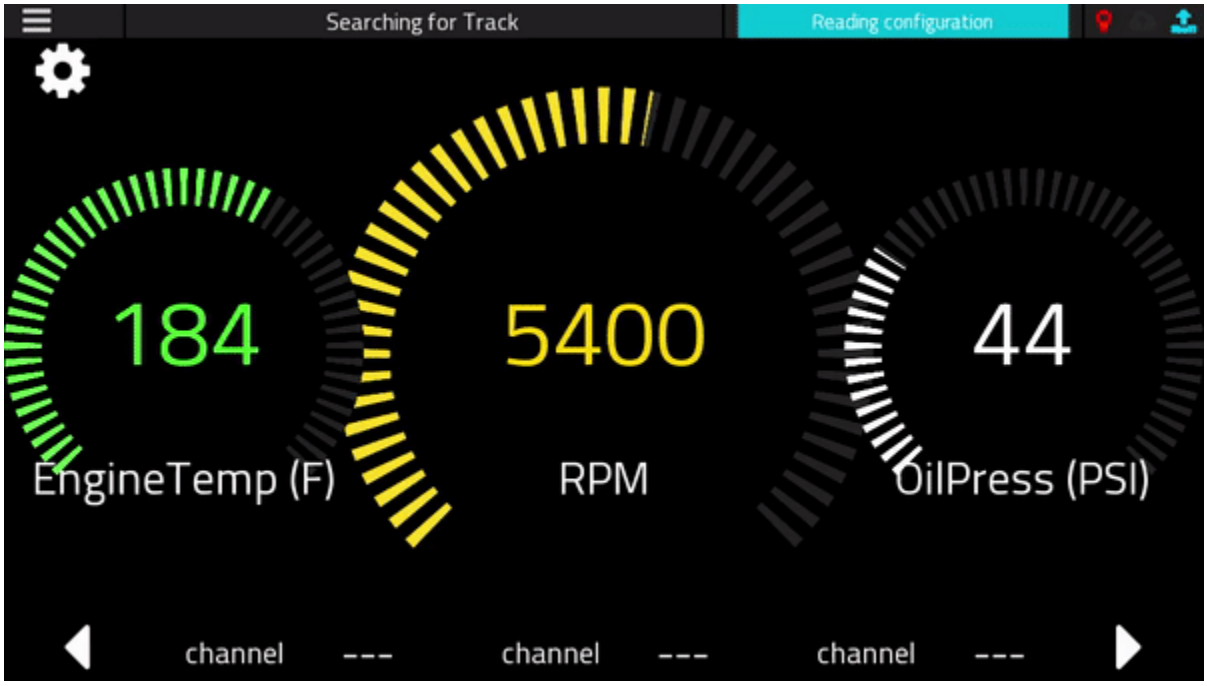

Figure 1. Possible example of displayed data

Constraints, Risk, and Feasibility

The main stumbling blocks we expect to encounter include inexperience with the tools and technologies required to build the HMI and lack of an easy testing method. Even though building the HMI does not require any domain-specific knowledge, achieving effective communication with the race car might require some extra learning that we did not anticipate. Furthermore, there is no robust way for us to test the HMI without access to the actual race car, which is currently located in Hawaii. To ensure feasibility of the project the MVP will be constrained to a very limited set of metrics and parameters. The hope is for this project to serve as a stepping stone for further development of a more advanced HMI.

Risks:

- Time constraint (we may not have enough time to integrate our app onto a mobile device)
    - This is a stretch goal and is not part of the MVP
- HMI might not be fully compatible with actual race car (we do not have access to the car)
    - Careful implementation of the HMI with full knowledge of how to interface with race car will ensure maximum compatibility without need for physical access

Group Management

Management Roles

- Alexis Morales Flores: co-lead
- William Chung: co-lead

Decision Making & Communications

Since the group consists of only 2 people, decisions will be made by consensus and by consulting with the IAC team. Communication will be conducted primarily through Discord and

physical meetings before/after class. We have also scheduled weekly meetings with the IAC team in the Triton AI workspace where we will work on our project for 10 hours per week.

Scheduling

Considering the small team size, a relatively high level of synchronization can be maintained through frequent communication. Any difficulties that lead to falling behind schedule will be brought up immediately via messaging and will be addressed as soon as possible to allow continuation of progress.

Responsibilities

- Alexis Morales Flores
- William Chung

Project Development

Development Roles

- Alexis Morales Flores: telemetry developer
- William Chung: teleoperation developer

We will use our laptops for the development of our HMI app and both of our roles will be to be software engineers to work on the same system as we are the only 2 team members. Initially we will create a web based application and then if time allows we will move the app onto a tablet as the IAC team would like us to have the app on a tablet therefore we may eventually need to request an iPad. As for the software we will use a text editor of our choice such as Microsoft Visual Studio Code which is free software. Documentation will be written on Markdown documents on Github to maintain a centralized repository, and our testing will be completed by following up with the University of Hawaii once we have our MVP to coordinate

on how we can connect our app to the race car. HMI will be a web application that can be accessed through a web browser. Standard web development technologies are sufficient to start (HTML, CSS, Javascript). More advanced frameworks can be integrated as needed (e.g., frameworks for displaying automotive data). The entire project can be built from scratch using standard technologies, but relevant software frameworks will make the process easier and more robust. No special technology is needed for this project.

Testing

Testing will most likely be conducted by using static data (data read from files) or by using real-time data outputted by a simulation. Connecting the HMI to a data source will allow us to see how our system displays data. To test how our system sends commands to the race car we will most likely connect the HMI to a simulation and see how it reacts to various input commands. Ideally the most robust testing would be conducted with the actual race car, but we do not have access to it.

Documentation

Documentation will be recorded on Markdown documents uploaded to the project GitHub repository, which will be a centralized location containing all the code and related documentation for the project.

Project Milestones and Schedule

Milestones

Milestones followed by an asterisk are stretch goals and are not critical to the completion of the project. Both team members are responsible for each deliverable, with each team member focusing on either telemetry or teleoperation as outlined above.

- **Prototype HMI (Week 5)**: complete MVP with must-have virtual instruments and controls
    - Research open-source and/or affordable solutions to implement virtual instrumentation on a web app ***(Week 3)***
    - Define list of virtual instruments and controls that are must-haves and nice-to-have ***(Week 3)***
    - Develop prototype virtual instruments and controls ***(Week 4)***
    - Present project at Jacobs School of Engineering Research Expo* ***(Week 4)***
    - Begin testing (either using static data or real-time data) ***(Week 5)***
    - First version of prototype (V0.1) with must-have virtual instruments and controls the IAC team requested ***(Week 5)***
- **Final HMI (Week 10)**: HMI improved based on feedback from IAC team
    - Improve V0.1 to V0.2 based on customer feedback ***(Week 6)***
    - Connect HMI to simulator showing real-time data on car analytics ***(Week 6)***
    - Improve V0.2 to V0.3 based on customer feedback ***(Week 7)***
    - Connect HMI to hardware for real time display, if feasible (VPN to hardware located in Hawaii) ***(Week 7)***
    - Improve V0.3 to V0.4 based on customer feedback ***(Week 8)***

- Begin to work on the nice-to-have instruments* ***(Week 8)***
- Improve V0.4 to V1.4 based on customer feedback ***(Week 9)***
- Refine repository and train other people ***(Week 10)***