

Technology Review: Hyperrail system Applications

Group 25: Yihong Liu

02 November 2018

CS-461, Senior Software Engineering Project, Fall 2018



Abstract

This project aims to develop a system that provide a central server the handle the users in different places without the hardware mic-controller to set some commands for the hyper-rail system for their research. The user can through the GUI to operate the hyper-rail system to do any tasks more convenient and efficient. Our core goal is to allow the sensor package and Hyper-Rail App to interact with existing actuators in the space. Because actuators are machines that perform simple movements on objects, such as valves or switches, this interaction will give the application additional capabilities depending on the application of the actuators.

CONTENTS

1	User interface/GUI	2
1.1	Problem description	2
1.2	Overview	2
1.3	Criteria	2
1.4	Technology Points	2
1.4.1	Wire-frame	2
1.4.2	Web solution	3
1.4.3	GUI testing	4
1.5	User Interface layout	4
1.6	Discussion	5
1.7	Conclusion	5
2	Reference	5

1 USER INTERFACE/GUI

1.1 Problem description

The HyperRail is a small railway where an automated environmental sensor package, which contains a variety of different sensors, can traverse through a space and collect information as it travels along the railway. Using the HyperRail application, the sensor package's settings can be customized and monitored. The application currently allows the user to specify the speed of the package, current length of the rail, and the size of the spool used for the motors. It also monitors the position of the sensor package along the railway and updates its location in real-time. Our proposed solution to this problem is to create a web application that communicates with a central server, which in turn communicates with the sensor system. This web application will be mobile friendly to work on Android, iOS, and personal computers. The web application will let the user configure schedules for the sensors to run, set up intervals for the sensors to collect data, and also display the current status of the sensor system. Once a user saves a configuration, these settings will be uploaded to a central server where it will be deployed to the corresponding sensor system. Scheduling runs will be controlled and initiated by the central server by sending signals to the HyperRail sensor system with locations to go to, rate of travel, and any points of interest to monitor along the way. The data collected by the sensors will be uploaded to the central server and can be viewed by the web application.

1.2 Overview

This article introduces the detailed process about how to make a GUI for our project, there are 3 main parts, Wire-frame design, web solution and GUI testing. each of those technology points has the specific requirements for us to implement them in our project, we also find out some softwares to help us to achieve those functions. At the end, the user can use the GUI interface to operate the sensor package on the hyper-rail system to execute specific researching tasks without the mic-controller through our central web-server, our GUI interface will connect the web-sever to send the corresponding commands to the sensor. Therefore, we can achieve our goal, which is using the hyperrail system at anywhere.

1.3 Criteria

We will be able to use the GUI to let the sensor package on the hyper-rail system do execute the specific task as we command, the sensor package should be able to respond and execute the specific task with a short reaction time and high accuracy.

1.4 Technology Points

1.4.1 Wire-frame

First of all, we need to design our user interface, we need to have a user story and understand what the clients want, then we need a functional map, which means the sort our necessary functions according to our client's demands, then we need to design the UI flow and flow chart well to make sure the user interface can handle the commands efficient. At the end, we should have a prototype for us to test see if something needs to be improved or changed. here are some softwares that we might use for our UI design.

- A. First one is about dynamic interaction, the app is called axure. this application is more focus on the dynamic pages and connect those pages to a dynamic effect
- B. the second one is for static interaction, GUI design studio. this software is focus more on the page to page effect, its advantage is more concise layout and easier to use.
- C. the third one is framer.js, this software is more complex than other two softwares that I mentioned, it provides creativity freedom and it has more platforms available for users. The code in Framer prototypes can be copied and used directly by developers to reproduce hi-fi interactivity patterns. it also have some disadvantages, such as the user needs to learn code, and import issues are also a trouble for some users.

At the end, we should have a hi-fi prototype of UI first, which fits all the requirement from the clients and then we will use the corresponding web language, such as css, html, node.js to handle the logical relationship in the GUI, and we will sent the prototype to our clients first to get some feedback or comments before we really implement it.

.

1.4.2 Web solution

For the specific web stuff that we are going to use with our user interface, I'm going to talk 3 main basic knowledge that we need to be able to design the UI.

- A. Html is Hypertext Markup Language, a standardized system for tagging text files to achieve font, color, graphic, and hyperlink effects on World Wide Web pages. we can control the UI flow with Html, we can received the user's input to navigate the users to the page that they want, and we can upload images with html, what's more, we can do compute things through Html.
- B. Css describes how HTML elements are to be displayed on screen, paper, or in other media. it saves a lot of work. It can control the layout of multiple web pages all at once. Compare to html, css is more focus on the layout of the interface page, such as color, font, template, spacing, etc.
- C. Javascript is a scripting or programming language that allows you to implement complex things on web pages every time a web page does more than just sit there and display static information for you to look at displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. The common use for Javascript are image manipulation, form validation, and dynamic changes of content.

Besides that, we probably will use a lot of IDEs to write the API for connecting our back end port, such as Webstorm, WebStorm is a powerful IDE for modern JavaScript development. Besides client-side applications, WebStorm helps you develop server-side applications with Node.js, mobile apps with React Native or Cordova, and desktop apps with Electron. Also, this ide can be used to write node.js, css, html, and handle a lot of logic problems between front end and back end. .

1.4.3 GUI testing

After we have finished the UI prototype, we need to test the UI that we designed, I'm going to talk 3 main tools that can help us to do the test work in the future project.

A. Abbot is a java-based test framework, it can automatically to generate and verify java GUI, it can let the user to explore, operate the system easily. developer can through the script or compiling the code to use the Abbot framework.

B. EggPlant doesn't interact with the basic code, eggplant is more focus on those user interfaces with problems, those problems might be caused by FLash, java, HTML, net, etc. And the application can play back to the place that the interface has problems, so we can take a look at the specific crux of the problem, then fix it.

C. HP winRunner is a automatic GUI testing application, it tests the script and go back to check the interaction process in the UI, to find the disadvantages, the user can know which part in the UI flow didn't run as they expected. And we can do the corresponding change to improved.

The reason why we want to the GUI testing is that there are many unpredictable bugs happen when we test the GUI, there are several bugs when we test the GUI, the first one is Overlap, this means the context is overlapped some other contexts, the second one is text wrapping, one line sentence is getting split by two lines, the third one is lost shortcut, usually, every UI can be operated by keyboard, but if the user has no mouse, we definitely want to make sure those shortcut works fine on the UI. And those testing tools will help us to find the bug in our code.

1.5 User Interface layout

The web application is a web interface for users to operate the sensor on the hyperrail system to collect corresponding data for their research. The current implementation of the web application UI will be made by JavaScript and NodeJS .

The current list of planning views are as follows:

- Account: Provides views for account management, including login, registration, password recovery, login failure, email confirmation, other related account information.
- Facility: Provides views for facility staff to manage residents of their facility who own the Hyperrail system, and some related information about term of service.
- Home: A home page with general information about the Hyperrail system, such as documentation and contact information on the top bar.
- Setting: Provides additional views for account management beyond login, including password changes, binding other information to the account, logout, etc.
- Project: A place to save an unfinished research or create a new research, for example, the user can get the research process and the data from last time they use the Hyperrail system. Also, the user can create a search with different command parameters.
- Operating page : A page shows the correspond information that request the user to input, such as "rail length" , "spool radius" , "velocity" , "whatever user wants to test", and user can change the input.
- Data information : the page shows the collected data from the sensor on the hyperrail system after the user entered the command parameters.

1.6 Discussion

according to the client's requirement, we need to create the user interface that allows user in different place to be able to operate the hyper-rail system remotely. First we need to design the high usability UI base on the research that we are going to do, then we need to make sure the html,javascript web stuff can work well with the central server that we are going to make. At the end, we need to use those test application to test whether our GUI is working correctly or not.

1.7 Conclusion

In this document, we introduced some basic processes about how to create a good UI for our project, and the specific softwares that we are going to use to help us to design, editing the user interface. Also the web stuff needs to cooperate with our back-end well to handle the commands that user's input and give the corresponding the action. At the end, we talked about the testing tools that we might going to use in our project, through those testing applications we can find out the ways to improve our UI's efficiency and user experience. Also, we camp up with a draft layout for our UI design, which we want to make sure our clients will accept our UI design.

2 REFERENCE

[1]MDN web Docs. (2018). "What is Javascript."

[online] Available at: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript].

[2]blog.framer.com (2017). "Prototyping in Framer: Pros and Cons." Available at: <https://blog.framer.com/framer-pros-and-cons-346778e091f8>.