

**IDEAS and App Development
Internship in Hardware and Software Design**

Rabab D. Alrayes

NASA Kennedy Space Center

Major: Electrical Engineering

Advanced IT Technologies and Software Development Spring Session

Date: April 8, 2016

IDEAS and App Development Internship in Hardware and Software Design

Rabab D. Alrayes¹

University of Michigan-Dearborn, Dearborn, MI, 48128

Abstract

In this report, I will discuss the tasks and projects I have completed while working as an electrical engineering intern during the spring semester of 2016 at NASA Kennedy Space Center. In the field of software development, I completed tasks for the G-O Caching Mobile App and the Asbestos Management Information System (AMIS) Web App. The G-O Caching Mobile App was written in HTML, CSS, and JavaScript on the Cordova framework, while the AMIS Web App is written in HTML, CSS, JavaScript, and C# on the AngularJS framework. My goals and objectives on these two projects were to produce an app with an eye-catching and intuitive User Interface (UI), which will attract more employees to participate; to produce a fully-tested, fully functional app which supports workforce engagement and exploration; to produce a fully-tested, fully functional web app that assists technicians working in asbestos management. I also worked in hardware development on the Integrated Display and Environmental Awareness System (IDEAS) wearable technology project. My tasks on this project were focused in PCB design and camera integration. My goals and objectives for this project were to successfully integrate fully functioning custom hardware extenders on the wearable technology headset to minimize the size of hardware on the smart glasses headset for maximum user comfort; to successfully integrate fully functioning camera onto the headset. By the end of this semester, I was able to successfully develop four extender boards to minimize hardware on the headset, and assisted in integrating a fully-functioning camera into the system.

I. Introduction

This semester I have been working on three projects under the guidance and supervision of my mentor, Jake Hochstadt². The G-O Caching Mobile App was being developed by a team in IT-C2. The main focus of this app is to encourage the employees at Kennedy Space Center (KSC) to discover new things and explore the surrounding work areas. Using this app, we aim to educate employees at Kennedy Space Center about the center's unique places of interest. Doing so will result in a more participative workforce, and allow employees to become more comfortable and engaged at KSC. This app has been successfully completed, and approved for use. I was able to work on the formatting of the app until completion, which assisted in creating a more attractive and user-friendly interface.

The AMIS Web App is currently being developed by another team in IT-C2. They are upgrading the website from an older format, which used the Adobe ColdFusion framework, to a more user-friendly interface using the Apache Cordova framework. The website aims to assist employees working in asbestos management, and allows them to look up all NASA facilities to find the ones that are scheduled to be treated. I completed multiple functions on the website, as well as formatting of the home page.

The IDEAS project is focused on creating a head mounted display system (smart glasses) to be used on Earth and in future space exploration. This technology will provide visual communication to the user while improving situational awareness, which will result in improved safety and efficiency of the user. This project is directly involved in improving technology for future missions, such as NASA's next mission to Mars. The safety and productivity of the user is the goal of this project—to aid the user in reaching their full potential during their mission. The team is currently

¹ NIFS Intern, IT-C2, KSC, University of Michigan-Dearborn

² Computer Engineer, Software Systems, IT-C2, KSC

working towards model 3 and 4 of the headset, with the final prototype planned to be finished by the end of 2016. I was working on the hardware of this project, with concentration on hardware development, PCB design, and camera integration. This task is a major part of this project, as custom accessories are an integral part of the project development. The hardware that has been/is being developed for the headset is essential to allowing the product to function, and connect the display, camera, and computer board. The camera is also an integral part of the system, as it is tied to many new functions the glasses uses.

II. Software Development

During this internship, all code was written using Microsoft Visual Studio. SourceTree and Stash, both Atlassian software, were used as source control and storage for the Git repository, which compiled all of the team members code together and resolved any conflicts between them. This method was very useful and organized, as it allowed a checks and balances system by separating every team member's code, and requiring another team member to review the code before merging it into a final master branch, where all team member's code is merged together. This method decreases errors and conflicts between codes, as they are immediately addressed before being merged into the master branch. In a team setting, using Git has proven to be a more efficient way of completing tasks.

A. G-O Caching Mobile App

I joined the G-O Caching App team during their last sprint of the project. My tasks on this project were formatting in HTML, JavaScript, and CSS to improve the user interface and assist in completing the app for usage on Android and IOS devices. I was able to successfully complete all of my tasks, such as formatting the help page and the waypoint question/answer buttons.

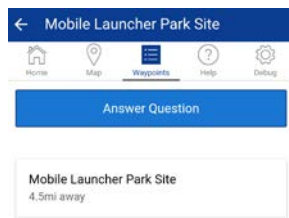


Figure 1. Waypoint Answer Button



Figure 3. Waypoint Submit Answer Button

I used my knowledge of HTML and CSS languages to develop the format of the app. I received guidance from my mentor, Jake Hochstadt, as well as the other two software developers on our team, as needed. I conducted research on HTML and CSS to gain more knowledge on formatting for the app, using the website Pluralsite. This task was fairly short, as the app was near completion and our last sprint ended in the following two weeks.

The tasks I completed required me to understand the many folders of code in the project, and how each one is connected. I learned how the CSS code, HTML code, and JavaScript code all work together and build off of the other in this app. For example, to format parts of the Help page it was necessary to write parts of the code in CSS, and call that code in the HTML file.

Using the Cordova framework, we were able to write the code in HTML and JavaScript, and adjust the style using CSS to create a pleasing user-interface that looks professional, and easy for the user to navigate. I began this project with very little experience in these languages, but studied them during my training, and learned from my mentor and the rest of my team as we continued to develop the app. The Cordova framework was a great tool used during this project, as it automatically compiles HTML, CSS, and JavaScript into

native IOS and Android code, thus eliminating the need to write the code for Android and IOS apps individually. It also uses Bootstrap built in functions, such as the buttons, animations, tabs, etc. to organize the layout of the app and assist in styling and formatting to create a modern user interface.

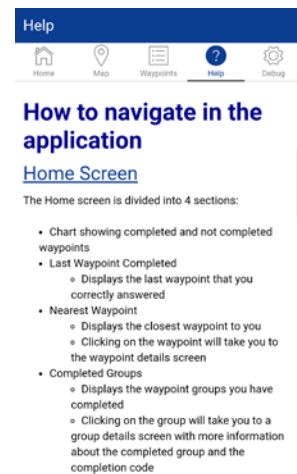


Figure 2. Help Page

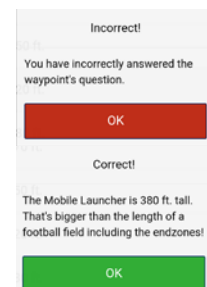


Figure 4. Waypoint Answer Popup Button

B. Asbestos Management Information System (AMIS) Web App

Similar to the G-O Caching app, the AMIS web app uses HTML, CSS, and JavaScript code, as well as C#. It is built on the AngularJS framework which, much like the Cordova framework, allows developers to use all four of these codes together, using built-in functions to create a pleasing user-interface.

On this project, I completed various formatting tasks for the home page, as well as a type-ahead function, which is an autocomplete function allowing the user to type in the first few letters of a word, then provides a drop down menu with suggestions of words you could be typing. This function is helpful as a user may not know the entire facility number, but by scrolling through the options of the drop down menu, they will be able to find the facility they are looking for. This function was written in HTML and JavaScript, styled in CSS, and connected to a database in C#, which controls all of the facilities at NASA.

The formatting I completed for this project provided easier readability as well as an improved layout of the home page. This allows users to quickly and efficiently find what they are looking for while using the web app.

After completing the G-O Caching app, and familiarizing myself with Cordova and AngularJS frameworks, I was able to jump into web development. I received assistance from my mentor and the other developers on our project, as needed. I was able to become increasingly familiar with the project as we continued to work on it.

Name	Number	Address	State	Zip	Installation_Id
NEIL ARMSTRONG OPERATIONS and CHECKOUT BUILDING	M7-0355				

Primary	Name	Mail Code	Email	Work
<input checked="" type="checkbox"/>	Todd W Baker	ISC-6400	todd.w.baker@nasa.gov	(321)867-7142
<input type="checkbox"/>	Kathryn Ann Nelson	ISC-1300	kathryn.a.nelson@nasa.gov	(321)861-3384

Figure 5. AMIS home page *Displaying use of the type-ahead function and style formatting of the page*

III. Hardware Development

While working with the IDEAS team on the wearable technology project, one of the tasks I was assigned was camera integration. I was encouraged to research plausible options of cameras that may be integrated into the headset. I was able to use the internet as a resource to find the camera with the correct specifications for this project, under the supervision of my mentor, Jake, and Ondrej Doule, the user hardware designer on our IDEAS team. I studied smart glasses products similar to ours to find a camera best suited towards our needs. We selected multiple cameras to inspect and test before selecting a final camera to integrate onto the headset. Once the inspection was completed and the best camera was selected, we began integration onto the prototype. I used my previous knowledge of hardware design and the Inforce circuit board we are using the power the system to connect the camera with guidance and supervision from Jake and Ondrej.

One issue that we were battling with is camera size, as we were originally interested in USB cameras, which are physically large and difficult to integrate with our custom hardware. Our second option was MIPI-C cameras, which are smaller and required additional custom hardware.

To fully understand and solve this issue, I studied other spy/smart glasses products to see if it is possible to integrate an ultra-compact USB compatible camera that may be small enough to integrate into the headset. This task was more difficult and time-consuming, as finding the right camera that is small enough to be used on the headset while maintaining minimal hardware took a lot of time and research.

In the end, we decided to integrate a MIPI-C camera that was small enough to build in the glasses, yet was still compatible with our Inforce board, thus eliminating any need for software drivers. We were able to satisfy the size requirements of the hardware designer by minimizing the

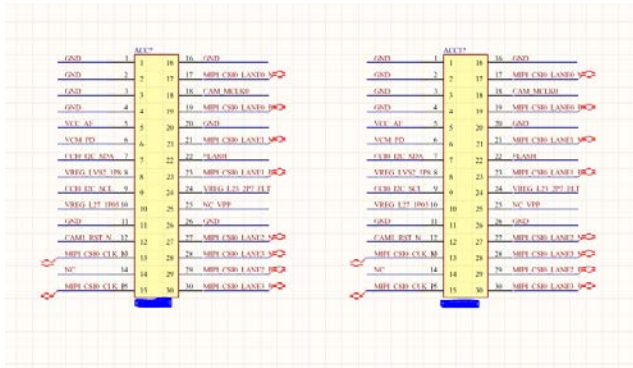


Figure 7. Camera Extender Schematic

footprint of the Connector to attach to the camera cables, creating the Schematic component, creating the schematic of the board, and finally wiring the routes of the board together. I was able to complete this with the guidance and supervision of Jake, and Andrzej Jackowski, another intern who is also involved in the IDEAS project.

Another task I was assigned on the IDEAS project was to minimize the hardware on the headset as much as possible, as well as debug any hardware issues to complete a fully operational prototype. Using Altium, I was able to design three more boards to extend cables and move hardware from the headset to the utility belt, and from the front of the headset to the back.

I designed a 51-pin extender board, which connects a 51-pin FFC (flat flexible cable), a 41-pin extender board, which connects a 41-pin FFC, and a 57-pin extender board, which connects a 57-pin FPC (flat printed circuit). The 41-pin extender board connects the camera to the Inforce power board. The 51-pin extender connects the converter board to the Inforce board, which is connected to the display using the 57-pin extender. The converter board is necessary due to the Inforce board only outputting MIPI-DSI, while the display uses LVDS. After implementing these extender boards, we were able to move most of the large and bulky hardware off of the headset.

To develop these boards, I worked with Andrzej to create the footprint for each component, then created the schematic, then created the PCB board. These boards all included differential pairs, which made routing the PCB more difficult—the routes for each pair must be the same length, and they should be in parallel with one another. The 57-pin board was my greatest struggle, as the pins of the component have very minimal spacing between them, which did not allow room for the pairs to be routed. With assistance from Jake and Andrzej, I was able to complete the board, and we successfully printed it to integrate into the headset.

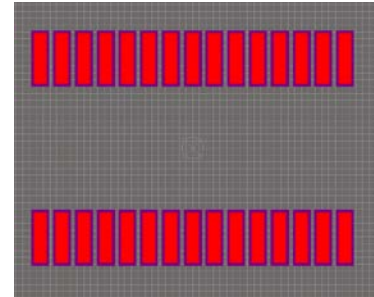


Figure 6. Camera Extender Footprint

amount of hardware on the MIPI-C camera.

To complete this task, we decided to separate the camera module from its hardware, and extend the cable between them to move the hardware away from the front of the headset. To do this, I used PCB design to develop an extension board to extend the cable between the camera and the circuit board. I completed this using the software Altium Designer—first creating a

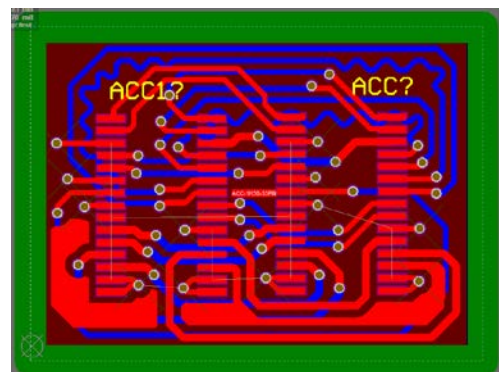


Figure 8. Camera Extender PCB Board

Figure 9. 41-pin Extender Schematic

Figure 10. 41-pin Extender PCB Board

Figure 11. 51-pin Extender Schematic

Figure 12. 51-pin Extender PCB Board

