Inventory Locator Assistant for ETG Shop

DESIGN DOCUMENT

Senior Design Team 9
Client: Mr. Leland Harker
Advisers: Mr. Leland Harker

Alejandro Buentello: Lead Hardware Manager
Caleb Gehris: Communications Manager
Jacob Linch: Lead Software Manager
Kurt Markham: Meeting Supervisor
Erin Power: Planning and Time Management Supervisor
Chris Rice: Lead Database Manager

Team Email: sdmay20-09@iastate.edu
Team Website: http://sdmay20-09.sd.ece.iastate.edu

Revised: 11/3/2019 V0.2

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

Summary of Requirements

- Typed search
- Voice Search
- X, Y coordinates for query showing
- Database of items

Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

Table of Contents

List of figures/tables/symbols/definitions

1 Introduction

1.1 Acknowledgement

We acknowledge that Mr. Leland Harker has helped guide us in the implementation of this project.

1.2 Problem and Project Statement

There are thousands of components in the ETG shop in a multitude of cabinets that it can take a long time for an employee to find a single component. In order to reduce the amount of time and confusion involved in finding a component, the ETG shop had requested a locator assistant system that would guide the user to find the correct component with minimal effort.

Our product will implement a solution involving a microphone or tablet that the client can use to request their specific component and a grid system constructed of two LED strips used to identify its location. It will work through accepting voice inputs through the microphone or text inputs on the tablet. It will also have a database that contains all the parts and their locations that can be added, removed, or changed as needed.

1.3 Operational Environment

Our product will operate within the ETG Shop in Coover Hall. Hazards we expect include: people hitting the LED strips, Raspberry Pi, or tablet. We also expect our product to get dirty over its lifetime through spilled liquids or dust.

1.4 Requirements

- The product shall accept commands via voice commands using a microphone or textual commands using a keyboard.
- The product shall allow for the expansion of the system onto multiple cabinets.
- The product shall allow for adjustments of the timing, brightness, and colors of the LEDs.
- The commands for “last search”, “all off”, and “all on” shall be implemented.
- There shall be test and configuration routines for the system.
- The product shall use visual motion to direct the user to the correct LED positions.
- The product shall allow for searches for multiple parts and use different indicators for each individual part.

1.5 Intended Users and Uses

Our intended users for this product are Leland Harker and the current and future employees of the ETG. It is to be used to assist ETG employees locate parts in the shop and for Leland or specific individual update the location of parts or add and remove them from the project’s scope as necessary.

1.6 Assumptions and Limitations

Assumptions

- The product will only be used in the ETG shop and by ETG employees.
- There will only be one user using the product at a single instance.
- The parts database shall be kept up to date.
- The product will have continuous WiFi connection during its operation.

Limitations

- The total budget for the project will not exceed five hundred dollars.
- The product must operate continuously during ETG shop business hours.
- The LED strips must fit onto the front of each of the cabinets without excess.

1.7 Expected End Product and Deliverables

Controller: The controller consists of a Raspberry Pi and its operating software that it uses to receive data and control auxiliary devices and database.

Tablet: The tablet's model is yet to be decided. It will be used to send commands to the controller. It also has its own software developed to communicate with the controller.

Microphone: The microphone is used to receive voice commands from the user and is connected directly to the controller.

LED Strips: There will be two LED strips for each cabinet, one for the width and one for the height. This will result in a total ten strips to be delivered for the five cabinets currently in the ETG shop.

Design Documents: The documents of the design and operation of our project have been requested by our client so that future senior design students may repeat the project and expand upon it.

2. Specifications and Analysis

2.1 Proposed Design

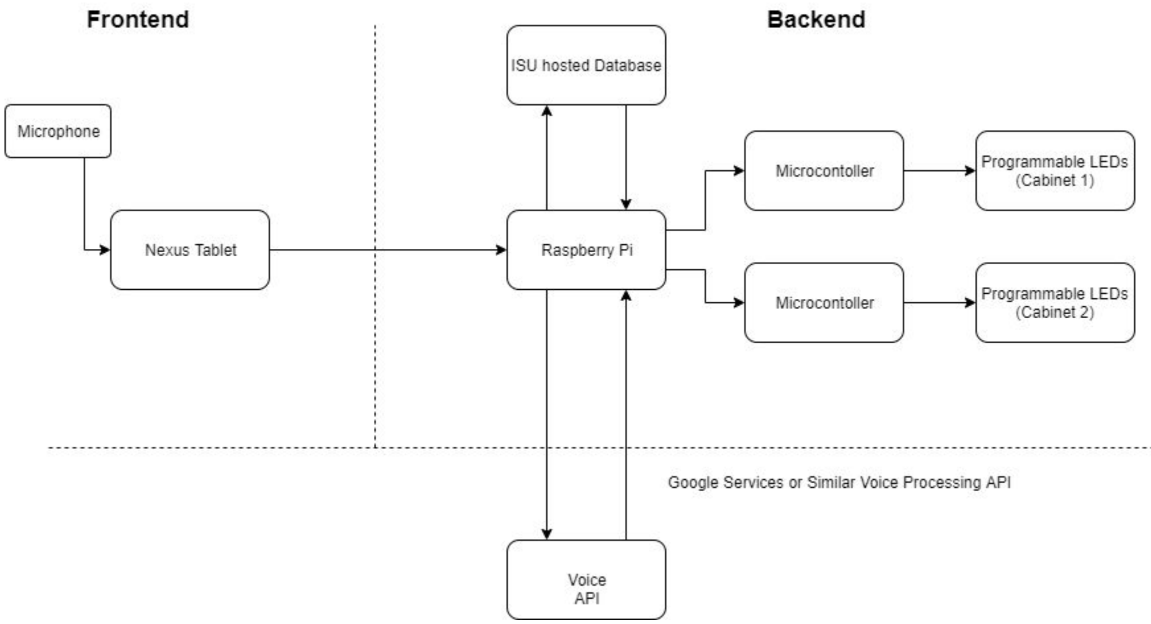


Figure 1: A Block Diagram of our proposed solution

Our proposed design is to use a tablet as a user interface, which would communicate with a raspberry pi backend. This raspberry pi will make the required voice API calls and Database calls to process any user request, then send commands to microcontrollers that directly control each strip of programmable leds. By following this design, we can create not only a functional product, but also one that is easily extendable in the future if needed and that should be easy to learn for our end users.

Some other key design choices we made were the inclusion of a Nexus tablet in the system compared to getting a touch screen for the Raspberry Pi. while the latter would have helped to eliminate some complexity in our design, based on the requirement gathering from our client we decided that the ability to transport the user input system within the ETG was a valuable feature. Another design choice was the use of microcontrollers and strips of programmable LEDs or the use of individual, radio controlled led. While the idea of individual leds is interesting, ultimately the issues of possible interference and greatly reduced extendability of the project pointed us back toward our current design.

2.2 Design Analysis

Currently, we are still in the process of ordering hardware and deploying an initial test environment for our system, so it is still relatively unclear if our design will be wholly successful or not. The biggest strengths of our current design is how easily extended it can be once deployed. By adding another microcontroller and LED strip and making slight code modifications, the system can ideally be extended to help find parts in additional cabinets. Additionally, by basing our design on existing software architectures we should be able to find a large codebase of potential examples and workarounds for any minor issues we run into along the way. The biggest weakness of our design is the Raspberry pi. Because the pi is involved in nearly every process in the system, if it fails the system as a whole will almost certainly fail. While we judged that this was acceptable within the scope of our project, for any larger projects we would want to try to add some backups or fail-safes to help maintain system uptime.

2.3 Development Process

The development process that we will be using is kanban. We wanted to avoid a waterfall design because we have direct communication with our client, so the ability to adjust and change our designs as needed is going to be invaluable as the project progresses. Additionally, we decided against agile because a system of two week sprints would be difficult when our team members can only work on this project ~10 hours per week due to other commitments. By using kanban, we get the ability to work toward our own pace, while also having a good visual representation of the progress we have made compared to what is left to complete.

2.4 Design Plan

Our design plan for this semester centers around establishing a base system, from which we can extend and add additional features in the spring semester. Our dependencies all center around securing the required hardware for our project: the raspberry pi, the nexus tablet, microcontrollers and leds. Once we have secured the hardware for the project, each of the other processes is fairly modular. For each of the nexus, the database, microcontrollers and the api we should be able to test that the pi is able to properly communicate in a standalone function, with the use of some tools like postman to help remove dependencies to further help speed the process. From there, we should be able to work around any software dependencies by mocking other modules as needed.

3. Statement of Work

3.1 Previous Work And Literature

This system is based off of an Instructable that was created by Dustin Dobransky. It offers an easy to use locator assistant for a small single bin storage system that helps the user find what they are looking for as well as similar products. However, the system that Dobransky created is not very suitable for larger scale applications or applications that utilize different storage systems like those needed by the ETG shop. We have modified Dobransky's creation to better suit our client by reducing the number of LED's per bin to save cost, creating a variable set up system to allow use with different storage systems, adding a typed search option, adding a system for easy scalability, reducing the types of queries that are not needed, and adding a system to test the functionality of the system so that problems can be troubleshooted.

3.2 Technology Considerations

Our main technology limitations come from the time it would take to implement them and the restraints of the storage systems. We believe that the hardware and software designs that are chosen are currently the best possible choices to balance the cost, ease of use, and ease of integration for the system.

3.3 Task Decomposition

The project is divided into two main parts, hardware and software, with a focus on the former during the first semester and a focus on the latter during the second semester. Furthermore roughly bi-weekly goals are laid out in our timeline that will be the focus during that period. At the beginning of each of these periods the tasks will be further specified and contributions will be divided among the team members.

3.4 Possible Risks And Risk Management

The main risk factors for the completion of this project are major changes in time availability due to other classwork, sickness, or some unforeseen emergency and faulty or broken hardware. The proposed solution for this project will be relatively low cost so the budget is not a concern unless multiple components need replacement.

3.5 Project Proposed Milestones and Evaluation Criteria

Key milestones for the first semester are system connectivity and communication, basic typed and voice search, and tablet integration. Testing in this phase will be less intensive as we focus primarily on getting a functional product, we will focus on testing communication between the hardware, hardware functionality, and basic search functions.

Key milestones for the second semester will be database construction, user interface, and scalability. In this phase there will be a much heavier focus on testing, especially with the user interface and search functionality.

3.6 Project Tracking Procedures

We will track progress on our project through weekly to bi-weekly reports that document work down and met and unmet goals. Current goals will be posted on our Git page and assigned to members. We have weekly meetings to stay on top of the project and to make sure that if aspect of the project is behind schedule the whole team knows about it and can respond accordingly by shifting resources.

3.7 Expected Results and Validation

The desired outcome of this project is an inventory locator assistant for the ETG shop to help individuals find parts within the shop that may be unfamiliar with its organization. To achieve this goal we plan to have a functional prototype of the system completed by the end of the semester (Dec 6th 2019), that exhibits basic functionality: receiving requests through voice or typed search, processing the search, and sending out signals to the appropriate controller. Development of the system will continue in the second semester with a large focus on testing, user interface design, database construction, and scaling. We have set our schedule to allow ample time for testing so that any issues may be found before the products final release.

4. Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

This project will cover two semesters--Fall 2019 and Spring 2020. Over the course of those semesters we project we will hit certain points as follows:

October 6:

- ➢ Parts list
- ➢ Design Document Version 1

October 20:

- ➢ Hardware System set up

November 3:

- ➢ Basic functionality
- ➢ Communication between Raspberry Pi and controllers
- ➢ Ability to send commands to LEDs
- ➢ Design Document Version 2

November 24:

- ➢ Search functionality
- ➢ Basic voice commands
- ➢ Tested queries to with dummy data

December 3:

- ➢ Final presentation
- ➢ Design Document finalized
- ➢ Finished prototype
- ➢ Basic platform, voice commands

Spring 2020:

- ➢ Project poster
- ➢ Finished website
- ➢ Database completion
- ➢ User Interface
- ➢ System fully tested
- ➢ Expansion options
- ➢ Finalized, working project

We plan on sticking to this timeline in order to continuously produce deliverables. This will help keep people on track and allow us to see if we are on time to finish our project before the end of this course. Having a timeline will provide us with the ability to move onto the next section once we finish something.

4.2 Feasibility Assessment

At the end of this course, this project should be able to input both voice and typed commands to find parts in the cabinets of the ETG. It will light up the designated area to alert the user of the position of the requested part. It will be able to add onto the database and look up the last search. We might have some issues when creating the database because there are a lot of parts that aren't regularly stocked and therefore might not ever be there.

4.3 Personnel Effort Requirements

Task	**Estimated Time Requirement**
Design Document	14 hours
Hardware Setup	20 hours
Database Creation	20 hours
Raspberry Pi Implementation	25 hours
LED connection	8 hours
Testing	50 hours

Table 1: Estimated hours needed

4.4 Other Resource Requirements

This project will require LEDs, a Raspberry Pi, multiple controllers, a power system, and a user interface.

4.5 Financial Requirements

We have been given a budget of $500 for the entirety of the project. This will go to buying hardware such as LEDs and controllers as well as other physical parts such as adhesives. All of our purchases will be made through Mr. Leland Harker.

5. Testing and Implementation

5.1 Interface Specifications

Currently we don't have any interface defined for testing the project. By the end we need to have a hardware interface for testing the lights to see if they're working.

5.2 Hardware and software

To test the hardware we will be using a lot of manual testing. The software testing will be done using Android framework such as JUnit and Espresso UI testing. The JUnit testing will make sure that we have all of our internal functions working. The Espresso will make sure that all of the UI behaves in the proper way with different changes to the system.

5.3 Functional Testing

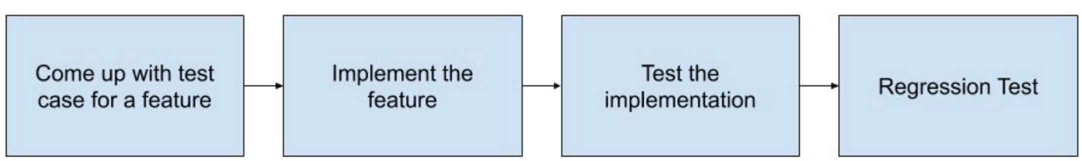
For our testing we will be using unit testing to verify all the internals of the software work correctly. To test the completeness of the system we will use integration testing for the interaction between the raspberry pi we will use and the controller boards for the lights.

5.4 Non-Functional Testing

For the non-functional testing we will do a lot of manual and regression testing on the software and hardware. As we develop we hope to run regression testing on all of our changes. For the security we will use a program such as wireshark to check to make sure all of the connections are secure and sending non confidential information. For compatibility, we don't need to test as much because the software we will be using will only run on one tablet.

5.5 Process

We haven't had the chance to implement any of the methods in Section 2 yet. The process we would like to use during the project is as follows:



5.6 Results

No results as of this time

6. Closing Material

6.1 Conclusion

To conclude, we are currently in the testing phase for different hardware options to best achieve our goal to design a system that has a central Raspberry pi as the backend for our project. We will be testing different ways of transmitting data, such as Bluetooth and WiFi, so as to create a more efficient system for our client. We plan to create a hardware test bench as a starting point, with the plan to make the system easily expandable to cover all the storage cabinets. While testing, we will need to keep in mind key variables such as cost, so that our solution does not become more complicated and/or expensive than the current approach to the problem.

6.2 References

- Voice controlled organizer
  - https://www.instructables.com/id/FindyBot3000-a-Voice-Controlled-Organizer/)
- AIY Voice Kit Instructions
  - https://aiyprojects.withgoogle.com/voice
- Initial scoping of microprocessors
  - https://randomnerdtutorials.com/esp32-esp8266-rgb-led-strip-web-server/
- Creating a LAN for our microprocessors to talk to the central pi
  - https://thepi.io/how-to-use-your-raspberry-pi-as-a-wireless-access-point/