

**DATE:** November 29, 2022

**TO:** Dr. Hu Tao

**FROM:** Joshua Pacis, Nick Daarol, Joshua Ruiz, Eduardo Ochoa Aragon, Casey Kaufman, Mkrtich Taymizyan

**SUBJECT:** Project Progress Report - Milestone #2

## **Introduction**

When machines are not available for use when customers are at a Round1 2 location, a potential source of revenue is lost, especially if there is only one machine of its kind. The Internet System we proposed allows customers to report issues with machines to technicians with the capacity to carry out repairs on the arcade cabinets by scanning a QR code on any arcade machine, or interacting with a repair request kiosk, or simply interacting with the appropriate page on the Round1 website. This report will discuss sample designs for the user interface of the repair request reporting system, as well as detailing the back-end infrastructure of the proposed Internet System.

## **Project Analysis Objectives, Activities, and Deliverables**

### **IS Requirement Gathering Methods**

To gather the Information System requirements, we will be employing a broad combination of the following gathering methods:

- Conducting Interviews
- Direct Observation
- Interface Analysis
- Workshops
- Role-Play

Interviews and periods of observation will be conducted on both employees, technicians, and customers alike to gather information such as the scope of the project (who it will affect and how) and/or interest in the project. Workshops will be held to iron out and refine what information and data, out of all gathered, will be necessary for the development and implementation of the project. Finally, Interface Analysis and Role-Play within our project is to test and refine the development of our system from a non-developer perspective, whether that be from the perspective of a customer or a non-technical employee at Round1.

## **Data Entities, Logic Flow, Boundaries of Project**

### **Entities:**

- Front desk staff
- Customer
- Engineers/technicians
- Arcade machines

### **Input**

- Arcade Issue Confirmation
- Arcade Fix Request

### **Output**

- Arcade Request Receipt
- Arcade Fix Reports
- Deployed Technicians Team

### **Data Stores**

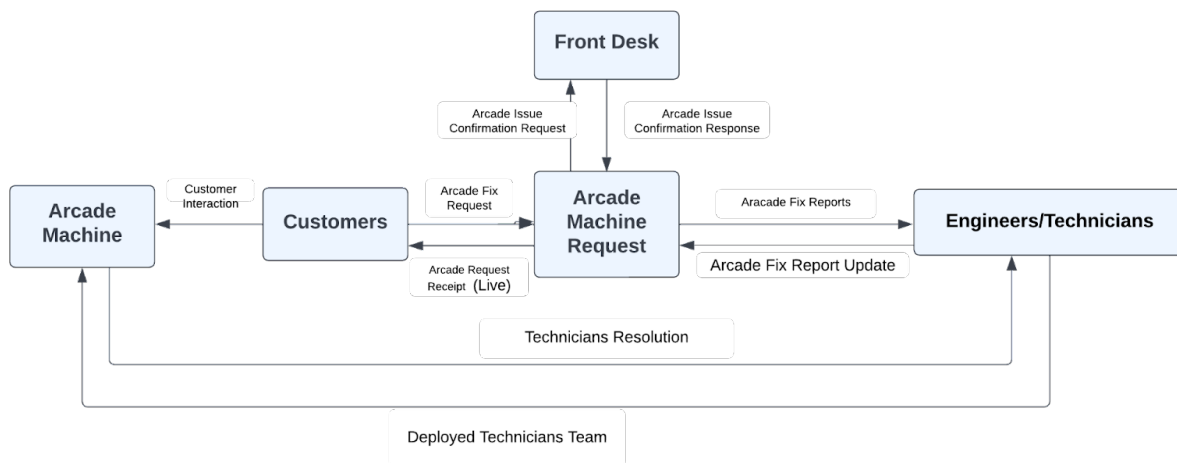
- D1: Front Desk Confirmation File
- D2: Customer Arcade Fix Request File

### **Processes**

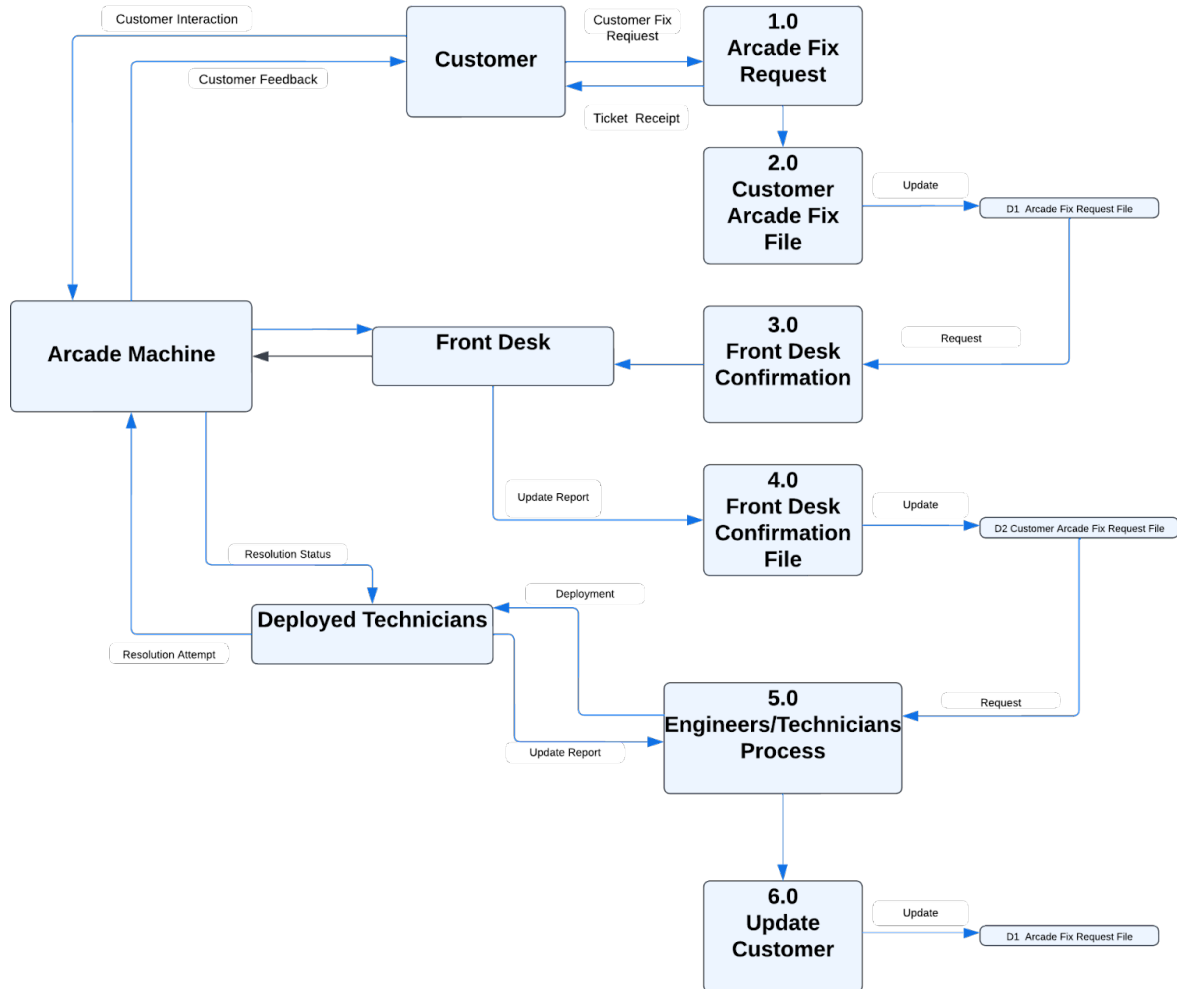
- 1.0 - Receives Arcade Fix Request
  - 1.1 - Receive Fix Request
  - 1.2 - Create Receipt
  - 1.3 - Create Arcade Live Request Receipt/Report
- 2.0 - Customer Arcade Fix Request File
  - 2.1 -Update Arcade Fix Request File

- 2.2 - Format Arcade Fix Request File
- 2.3 - Send Arcade Fix Request File Front Desk
- 3.0 - Send Front Desk Confirmation
  - 3.1 - Create front Desk confirmation request
  - 3.2 - Send Front Desk Confirmation request
  - 3.3 - Receive Front Desk Confirmation request
- 4.0 - Front Desk Confirmation File
  - 4.1 - Update Front Desk Confirmation File
  - 4.2 - Format Front Desk Confirmation File
  - 4.3 - Report to Engineers/Technicians
- 5.0 - Engineers/Technicians Resolution
  - 5.1 - Receive Desk Confirmation File
  - 5.2 - Deploy Technician/ Team
  - 5.3 - Technician Arcade Solution Report
  - 5.4 - Send Arcade Solution Report
- 6.0 - Technician Arcade Fix Request File
  - 6.1 -Update Arcade Fix Request File
  - 6.2 - Format Arcade Fix Request File
  - 6.3 - Report Customer

## Context Data Flow Diagram



## Level 0:



## Entities:

- Front desk staff
- Customer
- Engineers/technicians
- Arcade machines

## Input

- Machine Issue Confirmation
- Machine Fix Request

## **Output**

- Arcade Request Receipt
- Arcade Fix Reports
- Deployed Technicians Team

## **Data Stores**

- D1: Front Desk Confirmation File
- D2: Customer Arcade Fix Request File

## **Processes**

- 1.0 - Arcade Fix Request
  - 1.1 - Receive Fix Request
  - 1.2 - Create Live Receipt
- 2.0 - Customer Arcade Fix File
  - 2.1 -Update Arcade Fix Request File
  - 2.2 - Format Arcade Fix Request File
  - 2.3 - Send Arcade Fix Request File Font Desk
- 3.0 - Front Desk Confirmation
  - 3.1 - Create front Desk confirmation request
  - 3.2 - Send Front Desk Confirmation request
  - 3.3 - Receive Front Desk Confirmation request
- 4.0 - Front Desk Confirmation File
  - 4.1 - Update Front Desk Confirmation File
  - 4.2 - Format Front Desk Confirmation File
  - 4.3 - Report to Engineers/Technicians
- 5.0 - Engineers/ Technicians
  - 5.1 - Receive Desk Confirmation File
  - 5.2 - Deploy Technician/ Team
  - 5.3 - Technician Arcade Solution Report
  - 5.4 - Send Arcade Solution Report
- 6.0 - Update Customer
  - 6.1 - Update Customer Live Receipt

## **Project Design Objectives, Activities, and Deliverables**

## **Design Deliverables and Specifications**

Round1's web portal kiosk stations will be integrated with the current website to allow a seamless landing page for all repair requests. QR codes on each individual machine will automatically bring customers to the respective location's page when scanned via their device. Alternatively, the web portal can be accessed directly via the current website and, upon access of location data whether by manual entry or automatic detection, users can initiate a repair request all the same. Repair numbers are auto generated but require user input to select the appropriate machine in need of repair from a provided drop down menu. Users will then select a type of issue that the machine is experiencing from a predefined list. Completed requests will be stored on a central database in a remote designated location and will be made accessible to only the repair technicians working at that location. One kiosk is designated to each Round1 storefront, and all requests are sorted by location in the central database. Within each location, the database automatically sorts at regular intervals, prioritizing machines with the most repair requests.

### **Interface for repair request web form will include at minimum:**

- Machine Name & Number
- Type of Issue
- Text Box (to add additional info if needed)

### **Mockup #1**

The image is a screenshot of a web browser displaying the Round1 website. The browser's address bar shows the URL <https://www.round1usa.com/fix>. The website's header features the Round1 logo on the left, a navigation menu with links for 'What's New', 'Activities', 'Locations', 'Membership', 'Food Menu', and 'Careers' in the center, and a red 'BOOK A PARTY' button on the right. The main content area has a red background and is titled 'Submit a Repair Ticket'. It contains a form with two dropdown menus: 'Machine #' and 'Type of Issue'. Below the 'Machine #' dropdown is a text box with the placeholder 'Drop down bar with lists of arcade machines'. Below the 'Type of Issue' dropdown is a text box with the placeholder 'Drop down bar with lists of possible issues'. Below these two text boxes is a larger text box with the placeholder 'Text Box for User to add more info on issue'. At the bottom right of the form is a black 'Submit' button. Below the 'Submit' button, the text 'Thanks for submitting!' is displayed.

Round1

What's New Activities Locations Membership Food Menu Careers

BOOK A PARTY

Submit a Repair Ticket

Machine #

Type of Issue

Drop down bar with lists of arcade machines

Drop down bar with lists of possible issues

Text Box for User to add more info on issue

Submit

Thanks for submitting!

## Mockup #2

The second Mockup #2 involves a location specifier to redirect the user to a repair request form of a certain storefront.



Selected State:

California ▾



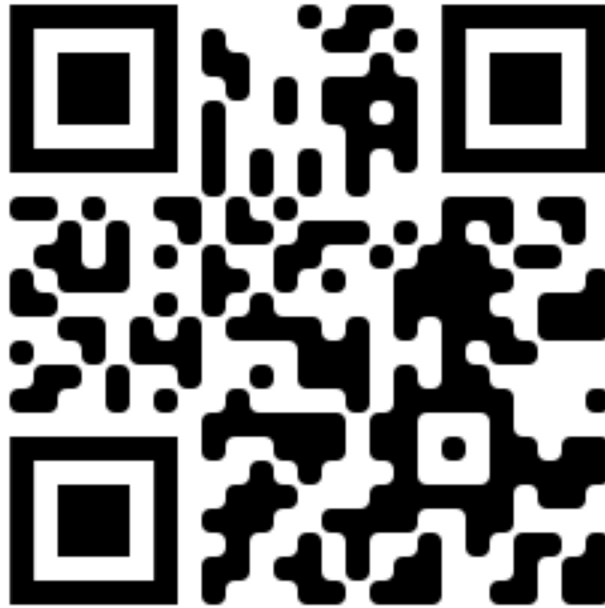
Select a machine ▾

What is the issue with the machine? ▾

Please describe the issue in detail.  
(optional)

**QR Code Example**





## **Usability Assessment**

Both designs for the interface are perfectly usable. Mockup #1 values simplicity and functionality, while Mockup #2 pursues a visually pleasing design on top of that. It is worth noting that Mockup #2 is more time, money, and work to implement, especially with Round1's future growth. Graphics would have to be made for each location, and these graphics would have to be sorted by state to achieve the desired functionality. It seems that Mockup #1 would be more suited to be accessed by QR code, while Mockup #2 would be more suited if accessed directly via the Round1 website. In light of these factors, we have decided to tentatively opt for Mockup #1 for the lower cost of implementation.

A mock website, which has the exact functionality of Round1's website, will be created and connected to our repair portal system using our integrated design. Many test repair orders will be placed first manually by our employees using the kiosk, while we test to make sure the service requests of our interface are updating dynamically through its connection to the website's integrated database. Next, QR based repair orders will be created in large quantities at a time to test our system's capabilities of handling and storing many service requests at once. Functionality of all of our new systems (Database, Kiosks, and web portal page) will all be rigorously tested.

## **Repair Request Database Sample Design**

<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <div style="border: 1px solid black; padding: 2px; text-align: center;">Search</div> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <b>Advanced filters</b>   - Repair # ()  - Machine # ()  - Date ()  - Type of issue ()  - Notes ()  - Repair Status () </div> </div>	<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;"><b>Database of all orders</b> (Example #'s)</p> <div> 6002 -- Order Date - Machine #  6001 -- Order Date - Machine #  6000 -- Order Date - Machine #  5999 -- Order Date - Machine #  5998 -- Order Date - Machine #  5997 -- Order Date - Machine #  5996 -- Order Date - Machine #  5995 -- Order Date - Machine #  5994 -- Order Date - Machine #  5993 -- Order Date - Machine #  5992 -- Order Date - Machine #  5991 -- Order Date - Machine #  5990 -- Order Date - Machine #  5989 -- Order Date - Machine # </div> </div>
---	---

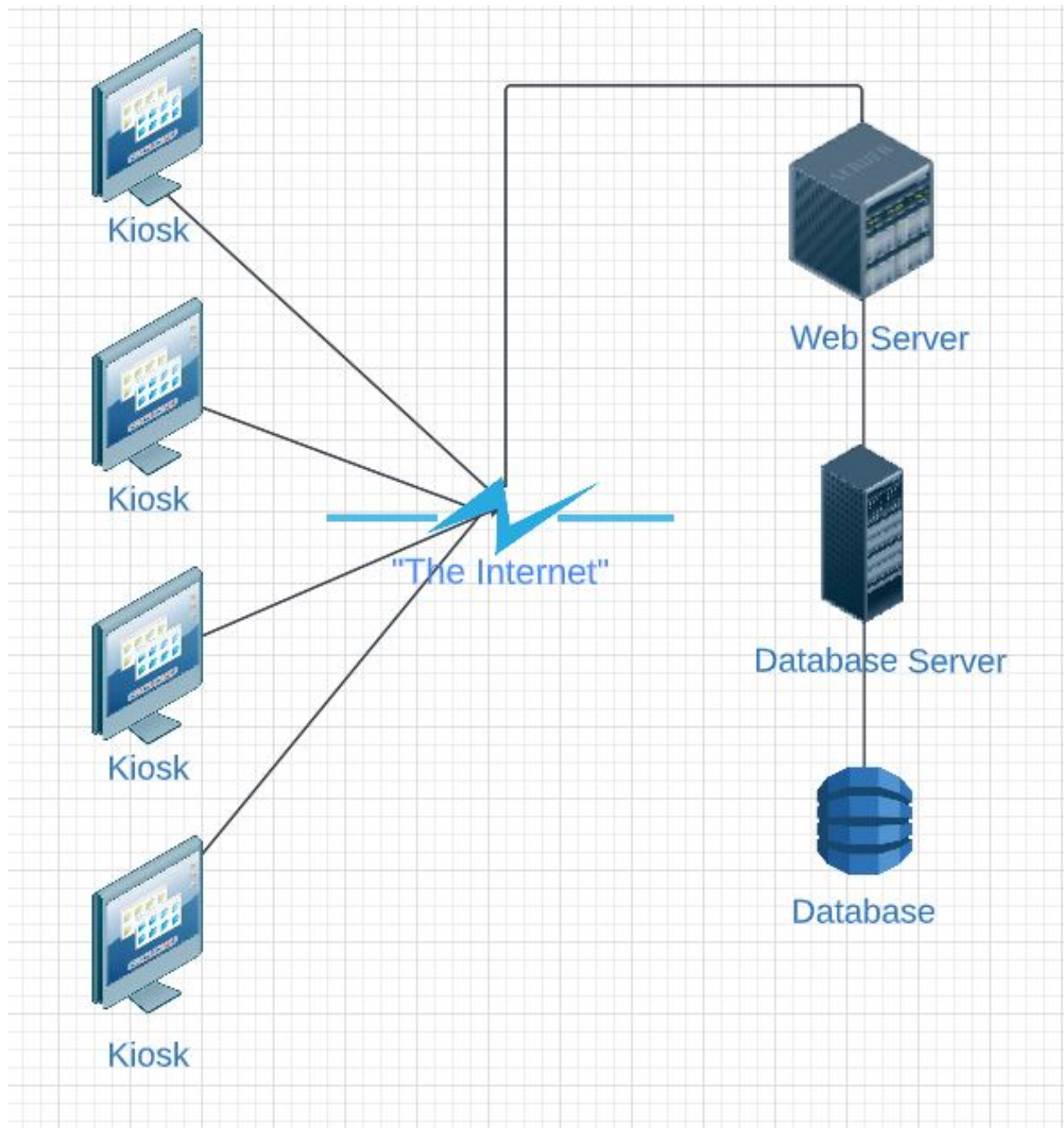
**Database with all repair orders is included (list starting from most recent)**

- Repair request ID numbers are generated when a customer places a repair request (Ascending)

**Search functionality by:**

- Repair Request ID Number
- Machine number
- Date
- Type of Issue
- Notes
- Repair status (Repaired/Not Repaired)

**Network Diagram**



The network diagram consists of individual kiosks that reside in each U.S. based location. Each location is designated just one Kiosk for the entire store. Each kiosk hosts a webpage that contains the landing page of the repair portal. The Web Server is in communication with the Database Server, processing all incoming repair requests. Requests are stored in the database and sorted appropriately by storefront. Each storefront has access to just their own repair requests. Repair technicians are able to sort through repair requests based on date created and can also filter requests based on the type of issue, allowing the prioritization of repair

requests based on machines that require parts that need to be ordered, or according to the severity of repair needed.

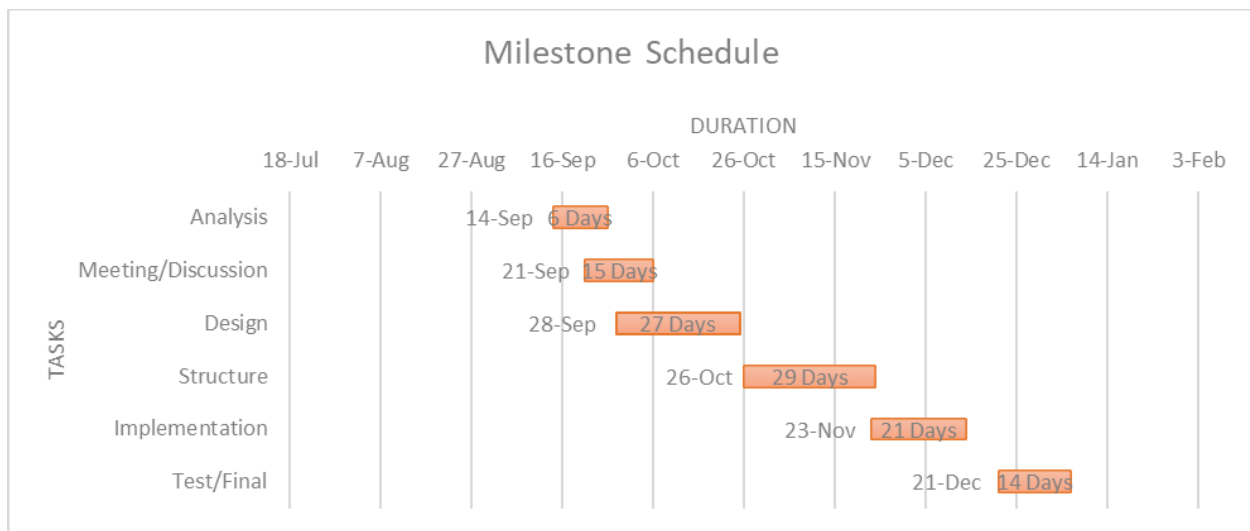
A few strengths of the Internet System have been identified:

- A customer can act as a second pair of eyes, keeping track of which machines may need maintenance or repair.
- More operating machines means more potential for revenue.

A few weaknesses with the Internet System have been identified:

- It's possible a customer might not report any problems with the machine and it will be down for an unexpected amount of time.
- There is no failsafe against "false" requests by users with malicious intentions.
- The system somewhat expects a user to have some level of technical knowledge about the machine they are reporting (the customer may not be able to differentiate between a gimmick or feature of a machine or a defect). Otherwise, it may end in a "false" repair request.
- There's no guarantee a customer will report a malfunctioning machine, leaving it in disrepair, losing revenue for the company.

### Milestone Schedule



### Project Status

Task	Scheduled Time	Projected Completion	Actual Completion	Status	Issues
Analysis	Week 2	8 Days	6 Days	Completed	N/A
Meeting/ Discussion	Week 2-3	14 Days	15 Days	Completed	N/A
Design	Week 4	28 Days	27 Days	Completed	Re-designed/
Structure	Week 5	29 Days	29 Days	Completed	N/A
Implementation	Week 12	31 Days	21 Days	Not Completed	N/A
Test/Final	Week 16-24	21 Days	14 Days	Not Completed	N/A

### Work Completed, Remaining and Current

- ~~Proposal and Planning~~
- ~~System Analysis/Feasibility Studies~~
- ~~Development~~
- Implementation
- Testing

The above is a checklist of tasks that have yet to be implemented. The next stage of work to be completed is Implementation of the Internet System.

### Conclusion

In conclusion, the internal infrastructure of our Internet System has been redesigned. Additionally, a full stack software that gives customers a front-end method of communication to back-end entities has been developed. This software facilitates the initiation of front-end repair requests to be collated and organized into back-end infrastructure, allowing for efficient and organized execution of operations which include machine repairs, machine parts procurement, and/or machine maintenance.