l

GNG2101

Design Project Progress Update

GROUP 3.1

Submitted by:

Rayyan Faruquee, 300257795 Luka Flanigan, 300375708

Gurkan Emre Uslu, 300350916

Takudzwa Gwavava, 300388232

AbdulWahid Ahmed, 300331802

06 October 2024

University of Ottawa

Table of Contents

[Project Deliverable Report Instructions i](#_Toc176011212)

[Table of Contents ii](#_Toc176011213)

[List of Figures iii](#_Toc176011214)

[List of Tables iv](#_Toc176011215)

[List of Acronyms and Glossary v](#_Toc176011216)

[1 Introduction 1](#_Toc176011217)

[2 Sustainability Report and DFX 2](#_Toc176011218)

[2.1 Sustainability report 2](#_Toc176011219)

[2.2 Design for X 2](#_Toc176011220)

[3 Problem Definition, Concept Development, and Project Plan 3](#_Toc176011221)

[3.1 Problem definition 3](#_Toc176011222)

[3.2 Concept development 3](#_Toc176011223)

[3.3 Project plan 3](#_Toc176011224)

[4 Detailed Design and BOM 4](#_Toc176011225)

[4.1 Detailed design 4](#_Toc176011226)

[4.2 BOM 4](#_Toc176011227)

[4.3 Project plan update 4](#_Toc176011228)

[5 Conclusions 5](#_Toc176011229)

[6 Bibliography 6](#_Toc176011230)

List of Figures

List of Tables

[Table 1. Acronyms vi](#_Toc144997103)

[Table 2. Glossary vi](#_Toc144997104)

List of Acronyms and Glossary

Table 1. Acronyms

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table 2. Glossary

|  |  |  |
| --- | --- | --- |
| **Term** | **Acronym** | **Definition** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 1 Introduction

Bethany Children’s Health Center is actively seeking to enhance its asset management processes by implementing a cheaper modern and efficient inventory system. The goal of this project is to develop a solution that enables staff to check out equipment seamlessly by scanning both a barcode on the equipment and their staff badge. This system not only aims to improve inventory control and accountability but also seeks to streamline workflow, ensuring that staff can focus on providing care without delays caused by inefficient equipment tracking.

This report outlines the development process of an asset inventory system that meets these requirements. It will provide an analysis of the system's potential social, environmental, and economic impacts, in line with the triple bottom line approach to sustainability. The report also details a life cycle assessment (LCA) to ensure the product's development considers its full impact—from production to disposal. Finally, the five most critical factors (or "X factors") in the design process will be identified and justified based on the needs of Bethany Children’s Health Center, ensuring the system is sustainable, efficient, and scalable. 

**2 Sustainability Report and DFX**

## Sustainability report

2.1.1 Objective and Scope:

Objective:

The objective is to develop a user-friendly and efficient inventory system that is cheap to build and maintain. This system will simplify the check in and out process for non-technical users. Also, it would help us to track the assets to see who checked it out and for how long. If an item gets lost, we can also track it down with asking the person that checked it out.

Scope:

Functional Unit: Asset Management (Check in/out, Tracking, Asset detail editing)

System Boundaries: User interface, database management

Stakeholders: Non-technical users (Doctors)

Exclusions: Complicated detailed interface for technicians

Similar product for comparison: SnipeIT

Table 1 Sustainability Report Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Tripple Bottom Line** | **Positive Impacts** | **Negative Impacts** | **Reasoning** |
| **Social impacts** | Provides an efficient method to check assets in and out |  | Due to a faster method to check in and out assets patients will receive more time for their therapy sections |
|  | More training for new system | Since the staff is used to the old system the transition to the new system will take time and rescores to train staff members |
| Makes keeping track of inventory easier |  | Since the old system is not used tracking assets becomes difficult and unreliable however with the new system staff will be more inclined to use the system making tracking inventory much more reliable. |
| **Environmental impacts:** |  | Energy consumption due to servers | The servers will use a significant amount electricity to run. |
|  | The transportation of assets will add to the pollution | Getting the hardware to put together the server takes some sort of transportation. |
|  | Consumption of raw materials for server equipment | To create a server to run the system new hardware needs to be bought |
| **Economic impacts** | Allows for ease of expandability in the company |  | The servers that are used for hosting the software can be expanded for an increased number of programs and storage on them |
| Custom inventory system won't charge per asset | Possible maintenance | An in-house system will not come with a subscription of paying for data, but will require more maintenance due to its tailored design |
| Power consumption |  | The servers will be constantly running, incurring a higher electrical bill. |

### 2.1.2 Inventory Analysis:

Asset Tracking Methods:

Scanning the user’s badge and items barcode to assign the asset to their name. To check out the asset from there name they need to do the same steps. Logistics will have access to adding or removing assets from the system. The user also will add details like approximately how long they will be taking the asset for while checking out the asset.

Database Structure:

The database will be in a server room as locally. This will add to the security and maintainability of the database. There will be computer stations in each inventory room that will be connected to main server to check in/out assets.

User Interaction: The user will interact with the system with their work on the computer and computer stations that are in the inventory rooms. You can find the checking in/out process in figure 1. This process will take approximately 1-2 minutes.

System Integration: The local servers and networking will be checked if there compatible to add a system like this. If not new server and networking equipment will be installed.

Life Cycle Inventory:

Graph 1: Lifetime CO2 emission impact by percentage

Materials For Server and Computer Stations: Putting servers, networking and computer stations together will require 150 kg of new hardware.

Transportation and Assembly: For the systems to be build and be operational approximately 500 kWh of energy will be used this is approximately 0.2 metric tons of CO2.

Usage: In this phase the servers, networking and computer stations will consume around 400kWh each year which is equivalent to 0.16 metric tons of CO2 emissions a year.

End of Life: End-of-life disposal of servers is expected to generate 100 kg of e-waste, of which 60-100% could be recycled.

* + 1. Impact Assessment:

Social Impacts:

The inventory system will be much faster and easier to check out or check in an asset from the inventory. This process should make the checking in process about 30-60 seconds. 3 times faster than the current system. This time reduction could save 50 labor hours a year if the business handles 100 assets every week. As a disadvantage the system is going to change and with this the users needs to be trained to how to use the system.

Environmental Impact: The main servers for the asset inventory system will run locally. So, this will decrease energy consumption by not having to communicate with the outside servers. This will reduce carbon footprint by %7. On the other hand, new servers and networking system is going to be put together and at the end of its life the system will be e-waste. Over the course of their ten-year lifespan, the new servers are anticipated to produce 10 kg of e-waste. This will consume raw material and create waste at the end. The system is planned to run for at least 10 years and maintained every 6 months. To reduce the e-waste out of commission materials should be sold or given away to people that will use them.

Economic Impact: The inventory system is going to be one time purchase and won’t require subscriptions or any other charges. This will be more economical for the client. But as a disadvantage to run the system there needs to be server and networking equipment bought. The upfront cost of $4000 for servers and networking equipment will be offset by not using the per asset charging software, resulting in a net savings of $20000 over 5 years.

* + 1. Interpretation:

Reduce Database Maintenance: Select a design that includes integrated troubleshooting wizards and automated backups if impact evaluations show that non-technical users find database complexity difficult to understand.

Quick Asset Check In/Out: The system should allow for a rapid check in/out. Also the system should update any changes made to the available assets in the system quickly.

Intuitive Interface Features: The interface that the users are going to use should be provide more than enough detail to them so they can choose the right asset easily and quickly. This can be done with big pictures of the asset good descriptions and more details.

## Design for X

**Important Factors:**

1. An efficient method to check in and out assets (designed for speed)
2. Search feature (design for efficiency)
3. Tracking system which displays who was what asset and when they checked it out (design for safety)
4. Asset information (name of equipment, manufacture, serial # and notes tab) (design for usability
5. Feature to add and remove assets (Desing for expandability)

**Explanation of Ranking**

1. During the first client interview, she said that the main purpose of this new asset inventory system was to reduce the time it takes to check in and out assets (put a quote here). Saving even a few minutes with this new inventory system would greatly aid the staff since with such sort sessions with the kids every minute becomes so important.
2. The warehouse holds over 2000 individual assets at the children's center; thus, finding individual assets can become rather difficult and time-consuming. However, with a search feature, these problems are solved, which also aids with the previously mentioned important factor. Furthermore, the client said that their current inventory system has a search feature, and they would like the new system to have a similar feature.
3. Tracking assets is a key part of staying organized in a big warehouse. With the client's current system, a majority of the staff do not sign out the assets, meaning there is no way of tracking them; the client stated that this is a problem because if an asset is not checked out, another nurse would think it's still available, causing major disorganization in the workplace.
4. With such a diverse inventory of assets, knowing the specific details about each asset will make it easier to pick an asset for a specific situation, provides convenience with maintenance, provides ease with the expansion of assets, and provides an effective way to communicate with other staff through the notes tab.
5. The client stated that she adds and removes assets weekly, if not more regularly, and having a system to do this would save the client and other nurses a significant amount of time each week. However, the client also said that this system is not the main focus of the new inventory system, which is why this is the least important feature of the system.

**Examples of need statement/ metric constraint**

1. Time to check out an asset (> 1 min)
2. The system needs a feature to filter assets by search
3. The system needs to keep track of when each asset was checked out
4. The system needs to store asset information
5. The system needs to be modifiable to add or remove assets

# Problem Definition, Concept Development, and Project Plan

## Problem definition

The staff at Bethany's Children's Health Center seek an invitation to their asset inventory system to address the limitations of their current system, focusing on usability and efficiency aspects. the problem involves developing an inventory system that provides a fast way to check in and out assets while being simple to use and incorporates a diverse set of sub-features, including a search feature, tracking feature, and an information display, while being cheap to run.

**Performance Metrics:**

1-Time taking for each checking in or out should take less than 10 seconds.

2- Fixing errors can be completed in 2-3 steps

3- Asset Update takes less than 5 minutes to add or edit items.

4- Keeping expenses low with no fees per assets.

5- Employees can become proficient in using the system within half an hour.

6-Reliability is 99,9% uptime.

This results in a system that is easy, fast, and cost-effective.

Table 2 User’s Needs and Design Criteria

|  |  |  |
| --- | --- | --- |
| **Need Statement** | **Design Criteria** | **Metric Constraint** |
| **Simple UI** | Ui, which is easy, can be used by non-technical people | NA |
| **Fast Way To Check In And Out Assets** | Takes less that 10 seconds to process (check in or out the asset) RAM must be at least 4gb | Time to check out <20 seconds |
| **Can Be Run Affordably** | Does not charge per item. Charge for the entire system, not per number of items | Cost< 8000$ |
| **Display Asset Information** | Have a screen to display asset information | Boolean |
| **It Needs To Be Able To Store A Large Amount Of Assets.** | System is able to store a large amount of assets without effecting performance | Number of assets>=2000 |
| **Needs A Method To Fastly Find Assets** | Include search function | Boolean |
| **System To Track When Assets Were Checked In And Out** | Tracking system of the location and date when item was checked in or out | Boolean |
| **Needs To Be Customizable** | System to add and remove assets from the system manually | Boolean |

**Benchmarking:**

Table 3 Benchmarking for Different Products

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **SNIPE-IT** | **Fresh service** | **Spiceworks IT Asset Management** | **Asset Panda** | **OfficeInventory** |
| Simple User Interface | 4 | 3 | 1 | 4 | 5 |
| Fast way to check in and out assets | 3 | 1 | 2 | 4 | 5 |
| Affordability | 2 | 1 | 5 | 4 | 3 |
| Displaying Asset Information | 5 | 5 | 5 | 5 | 5 |
| Storing Large Amount of Assets | 5 | 5 | 5 | 2 | 2 |
| Asset Search feature | 2 | 5 | 2 | 4 | 4 |
| Tracking System for Assets | 3 | 5 | 3 | 4 | 5 |
| Adding and Removing Assets | 5 | 5 | 5 | 5 | 5 |
| **TOTAL** | **29** | **30** | **28** | **32** | **34** |

metin, yazılım, bilgisayar simgesi, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 1 UI of OFFICEINVENTORY

metin, yazılım, bilgisayar simgesi, web sayfası içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 2 UI for FRESH SERVICE

metin, yazılım, sayı, numara, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 3 UI for Asset Panda

metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 4 UI for Spiceworks

metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 5 UI for Snipe-IT

**Target specifications:**

1. Easy to use: Simple steps for staff to check in or check out equipment with a barcode scanner or fingerprint.
2. Fast checking in and out: Transaction processed in less than 5 minutes.
3. Low cost: There will be no additional charges per asset.
4. Simple management: simple to edit information and correct mistakes without needing a technical supporter.

**Relation to constraints**: In this design, having target specifications with exact numbers is problematic since most of the client’s needs are either yes, it can incorporate the subsystem, or no, it can’t. The global design allows for integrating all the subsystems, which are described in more detail under prototype c; therefore, for needs 2,4, 6-8, the design fits the target specifications. As for the other target specifications, the system allows for a method to check out assets simply by scanning them, which will be less than 20 seconds fitting the constant. Finally, since the system will mostly rely on QR codes to scan assets as well as a few data chips for the staff, the future cost won’t be very significant, and after completing benchmarking, it was clear that the cost of using even high-end inventory systems is less than 8000$, fitting into the target specifications.

**Limitations:** In our team, only a few of the members have had little experience with coding and making UIs, so it is unknown if the time given will be enough to incorporate all the subsystems. As a result, our team will focus on the main needs before incorporating the UI subsystems (search system, a method to add and remove assets…) since the problem at hand involves an inefficient system. Secondly, it is unclear if incorporating a scanning gate in the warehouse is feasible since we don’t know what the doors look like, and it is unknown if the owners would allow such a change. If such is the case, we would pivot to a separate idea that doesn’t require modifications, such as prototype b or c.

## Concept development

## Prototype Ideas

**Prototype A**: Automatic check-in and out system

**Intro**: It was clear that the clients number one priority is to reduce the time it takes to check in and out assets thus completely eliminating the manual labour to check in and out the assets would result in one of the most efficient asset inventory systems.

**Idea**: Each asset and staff member would have its personal chip attached to it and the doors to the Wearhouse would scan this chip whenever it passes through it. To sing into the system you would simply walk through the door and to sign out you would walk through it again. Furthermore, whenever an assets it brought through this door it will automatically be singed out to the nurses name then when its brought back through the door it is singed back in automatically

**Contribution towards solution**: Prototype a mainly focuses on the design for speed and design for simplicity aspects, which are ranked number 1 and 4 respectively on the DFX part on DB b.

**Unknowns**: Our team still doesn’t have a good picture of what the warehouse at the center looks like so it is possible that the doors would be too big for the system and the same can be said about the size of assets

**Limitations:** Since the number of assets excides 2000 placing a individual tag on each asset would be super time consuming and would cost a significant amount of money.

**Prototype B**: Customizable UI

**Intro:** This prototype focuses on the main usability aspects of the inventory system rather then the ability to sing in and out assets

**Idea**: Since the company that we’re designing the inventory system to has such a diverse staff designing one system would not satisfy all the users; to help solve this problem the idea is to create multiple interfaces which the staff can switch between there include: having a recently searched assets tab, A frequently searched assets tab and a pinned assets tab. Other customization would be the size of the font, page layout and background colours.

**Contribution towards solution:** Prototype c mainly focuses on the design for usability and expandability aspects which rank number 4 and 5 respectively on the DFX part on DB b.

**Prototype C**: Voice-activated sign in

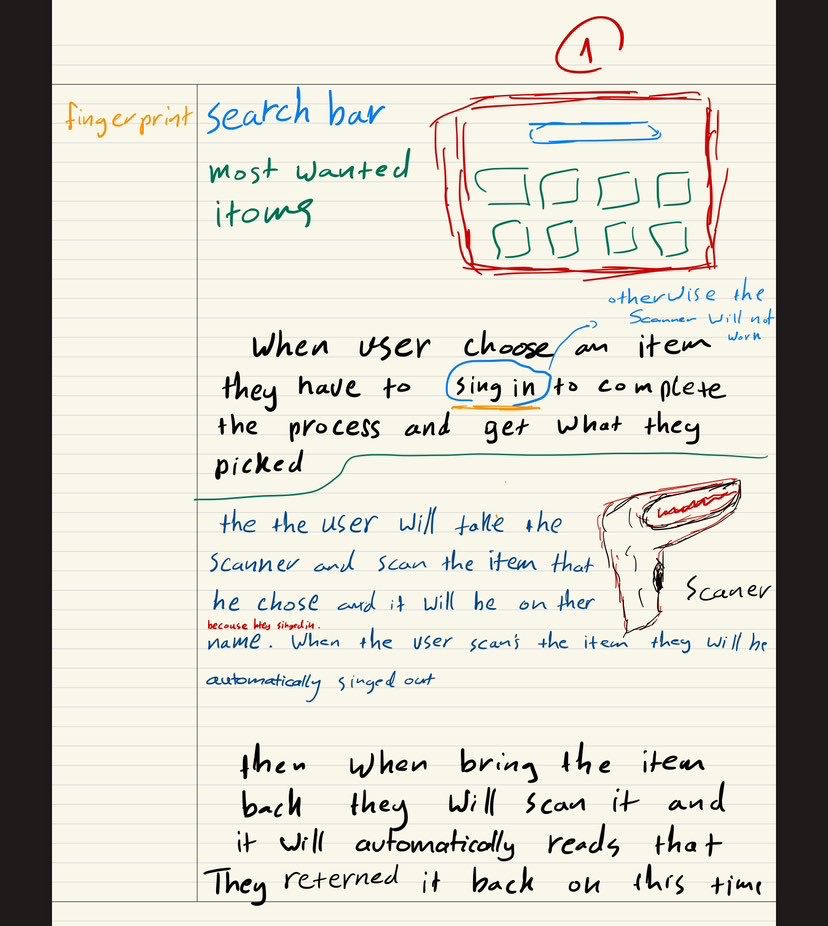
**Intro**: Similar to prototype the main idea is to reduce the time to sing in and out assets aligning with the clients’ main needs.

**Idea**: Each staff member will input their voice/name into the system then whenever the staff member says their name, they will be singed in. Moreover, each asset will be assigned a code preferably a 4-digit number. To sign out an asset the staff will say “sing out asset #XXXX” then the asset will be singed out and to sign an asset back in they would say sing in instead of sing out.

**Contribution towards solution**: Prototype b mainly focuses on the design for speed, design for simplicity and design for expandability aspects, which are ranked number 1 and 4 respectively on the DFX part on DB b. Furthermore, since this prototype provides an inexpensive way to track the assets the cost of use would be lower than other prototypes.

**Unknowns**: Again, similar to prototype we don’t know what goes on inside the warehouse and there is a chance that there are loud vents that could interfere with the system.

Prototype D:



Weighted Desing Matrix

Table 4 Analyzing and evaluating Prototypes Against Target Specifications

|  |  |  |  |
| --- | --- | --- | --- |
| **Desing criteria** | **Prototype A** | **Prototype B** | **Prototype C** |
| Ui, which is easy, can be used by non-technical people  (x1.5) | 3x1.5 | 3x1.5 | 5x1.5 |
| Takes less that 10 seconds to process (check in or out the asset) RAM must be at least 4gb  (x2) | 5x2 | 4x2 | 3x2 |
| Does not charge per item. Charge for the entire system, not per number of items  (x1.25) | 2x1.25 | 4x1.25 | 5x1.25 |
| Have a screen to display asset information  (x1) | 3 | 3 | 4 |
| System is able to store a large amount of assets without effecting performance  (x1) | 3 | 2 | 2 |
| Include search function  (x1) | 5 | 5 | 5 |
| Tracking system of the date when item was checked in or out  (x1) | 5 | 5 | 5 |
| System to add and remove assets from the system manually  (x1) | 5 | 5 | 5 |
| **Total** | **38** | **37.5** | **37** |

After analyzing each prototype idea based on the design matrix our team has agreed upon the following list of 3 main prototypes.

1. Prototype A (automatic check-in and out system)
2. Prototype B (barcode badge sign-in and out)
3. Prototype C (voice activation)

**Prototype A:**

In this protype the idea is to check out/in assets by using a RFID tag on the assets. With this check in and out will be fast and simple. You just need to grab the asset and walk out. The RFID scanners will read the tags and write the asset to your name.

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 6 RFID Scanning Working Diagram

tekerlek, giyim, oğlan, kişi, şahıs içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 7 Basic Showing on How The System Will Work Prototype A

Our database and code will run on a local server located in the server room with networking to each user and computer stations on the inventory rooms. There will be three different User Interfaces. First on is for the normal users to search assets. The second interface is for computer stations on the inventory rooms. This interface is just for checking in and out items. The last interface is for administrative personnel. In this interface you can control everything for example add and remove assets from the system, add users remove users and all sorts of stuff.

metin, yazı tipi, ekran görüntüsü, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Prototype B:**

Our database and code will run on a local server located in the server room with networking to each user and computer stations on the inventory rooms. There will be three different User Interfaces. First on is for the normal users to search assets. The second interface is for computer stations on the inventory rooms. This interface is just for checking in and out items. The last interface is for administrative personnel. In this interface you can control everything for example add and remove assets from the system, add users remove users and all sorts of stuff.

metin, yazı tipi, ekran görüntüsü, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

In this prototype the idea is to check in and out assets by scanning a barcode on the asset and the users ID. This idea is simple and easy to operate. The proses is like this:

elektronik donanım, ekran görüntüsü, klavye, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu

tekerlek, giyim, taslak, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 8 Basic Showing on How The System Will Work Prototype B

**Prototype B:**

Our database and code will run on a local server located in the server room with networking to each user and computer stations on the inventory rooms. There will be three different User Interfaces. First on is for the normal users to search assets. The second interface is for computer stations on the inventory rooms. This interface is just for checking in and out items. The last interface is for administrative personnel. In this interface you can control everything for example add and remove assets from the system, add users remove users and all sorts of stuff.

metin, yazı tipi, ekran görüntüsü, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

The idea in this prototype is that the check in and out is made by voice activation. User will say “Check Out” and the system will check out the asset to there name also this works for checking in the user will have the say “Check In”.

bilgisayar, dizüstü, çıkış cihazı, masa içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 9 Basic Showing on How The System Will Work Prototype C

Our group will be presenting these ideas to the client as well as addressing any concerns we have regarding the idea. Moreover, the feedback we obtain on these ideas will be essential towards the rest of the project since it will determine which design we start building prototypes on. Furthermore, we decided to incorporate prototype c into all our designs since it obtained the highest score in the weighted design matrix, and it doesn’t directly contribute towards main idea in the problem statement and will automatically incorporate design for expandability and usability into our main prototypes meaning they all satisfy at least 4/5 of the DFX determined in PD B.

**Global design**

To design the best system for the client, our team has decided to incorporate prototypes a and d into one system; this new prototype will automatically sign into staff when they walk in the door. However, staff will use a barcode scanner to sign in and out assets. Moreover, the concepts from prototype c will also be incorporated into the UI. By eliminating the cost problem in the prototype, the design aligned better with the client’s budget needs since there would be no need to purchase tags and place them on all the assets. Furthermore, this design fits most of the DFX criteria. It provides a fast way to sign in and out assets (design for speed), allows for an inexpensive way to track future assets, allows a simple method to add and remove assets from the system (design for expandability), incorporates a tracking system for when assets were taken out, and the system is safe to operate (design for safety), customizable UI  and simple method to check in and out assets (design for usability) and incorporates time-saving systems such as a system to add and remove assets and a search function (design for efficiency).

## metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim Açıklama otomatik olarak oluşturulduProject plan

# Detailed Design and BOM

## Detailed design

## BOM

## Project plan update

Add a screenshot of your gantt chart.

# Figure 10 Updated Gnatt Chart from Deliverable A to Deliverable D4. Detailed Design and BOM

**Introduction**: This outlines the design details of an inventory system for our prototype, along with a bill of materials (BOM) for the parts required. This report's objective is to provide a detailed conceptual framework for our inventory system, reflect on its feasibility, and present the components necessary for its implementation.

From the client meeting, we learned that the client had tried our idea with the gate scanners in the past, to little effect. Our other options would require new badges and barcodes for the equipment, which the client was willing to pursue. Other details the group learned included the client’s current system automatically generating serial numbers for assets, including individual files if there were duplicates of a certain type of asset. We had also gotten the client’s priority for our design being, the customizability from the search bar and recently searched function, the tracking ability, the ability to display asset information, and the speed of the system.

## 4.1 Detailed design

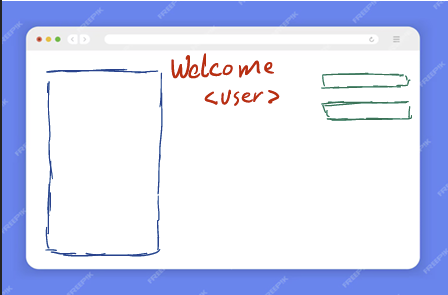


Figure 11: UI Before Logging in

Before opening the full UI, the software will prompt the user to enter their ID, which will be entered into the system by scanning a barcode. Once this is done, the user will have the ability to interact with the UI (Figure 11). Functions available through this UI include an asset search through a catalog, checking in or out an asset, or in the case of a system administrator using the software, adding or removing an asset from the database.

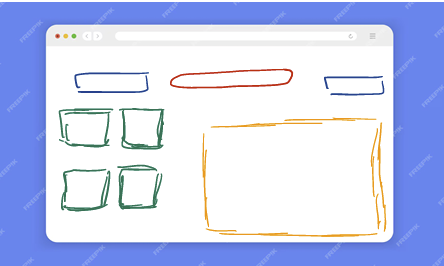


Figure 12: UI After Logging in

Individual assets will be treated as classes in the code, storing data such as names, serial numbers, availability, check-out times, check-out users, and notes on the asset. These classes will then be stored in a Python list, which the code will interact with.

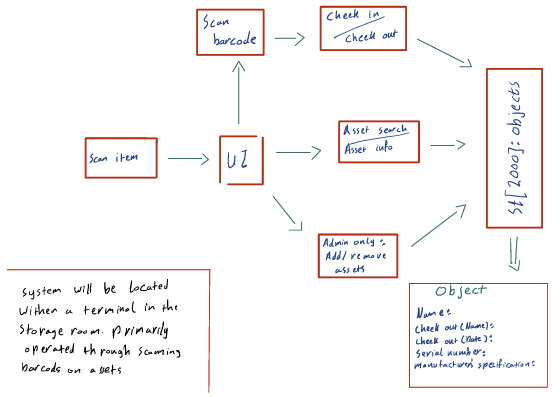


Figure 13: Flow Chat of System

The search function will work by having the user type terms relevant to the asset they are searching for, either part of the name or the serial number, and the UI will display the relevant information associated with the asset. This will function by comparing the asset names in the list to what the user is looking for. In this function, the user will have ability to edit the notes associated with the asset.

The check out function will work by having the user scan a barcode on the physcial asset, which will retrieve the serial number and retrieve the appropriate asset from the data list. The program will automatically input the user information into the file from their log-in credentials, as well as the timestamp from the computer. To complete the transaction, the user will scan their ID badge again, authorising the request. To check the asset back in, the user will need to scan the item barcode, which will retrieve the asset file, clear the check-out data for that asset, and mark the availability as “Available”.

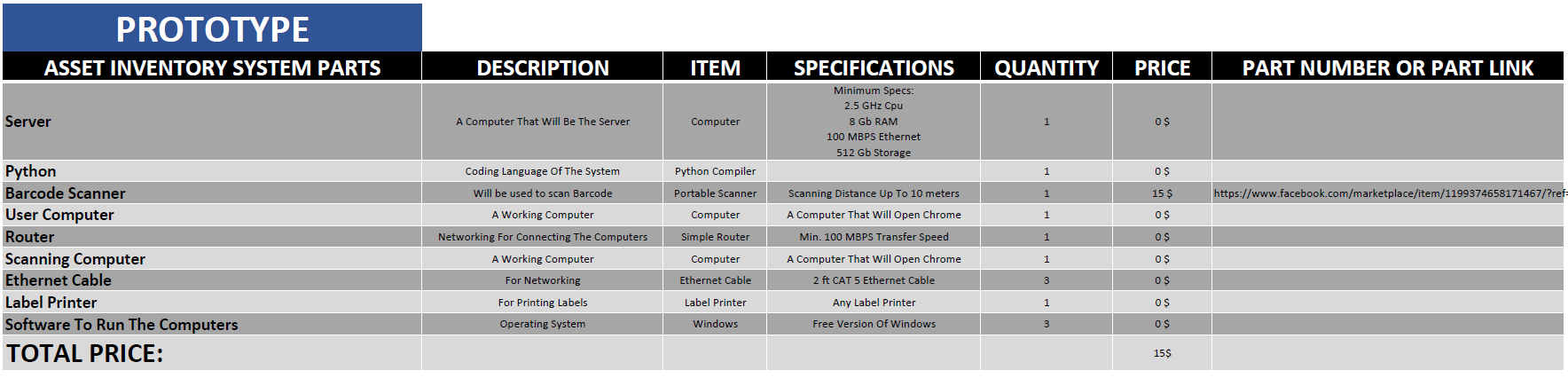
The addition function, which will be accessible to system administrators only, will create an empty asset file and automatically generate a new serial number. The administrator will then fill in the asset information and confirm the addition by scanning their badge. To delete an asset from the system, the administrator will retrieve the asset file through the search function and authorise the deletion by scanning their badge.

Significant care should be given to the design of the UI and the efficiency of the system to meet the requirements of designing for efficiency and speed, since both these requirements were brought up in both client meetings.

Available to the group are the Python programming suite and at least one member with basic experience with the language. To help the group with programming the software, there are YouTube videos available to fill knowledge gaps.

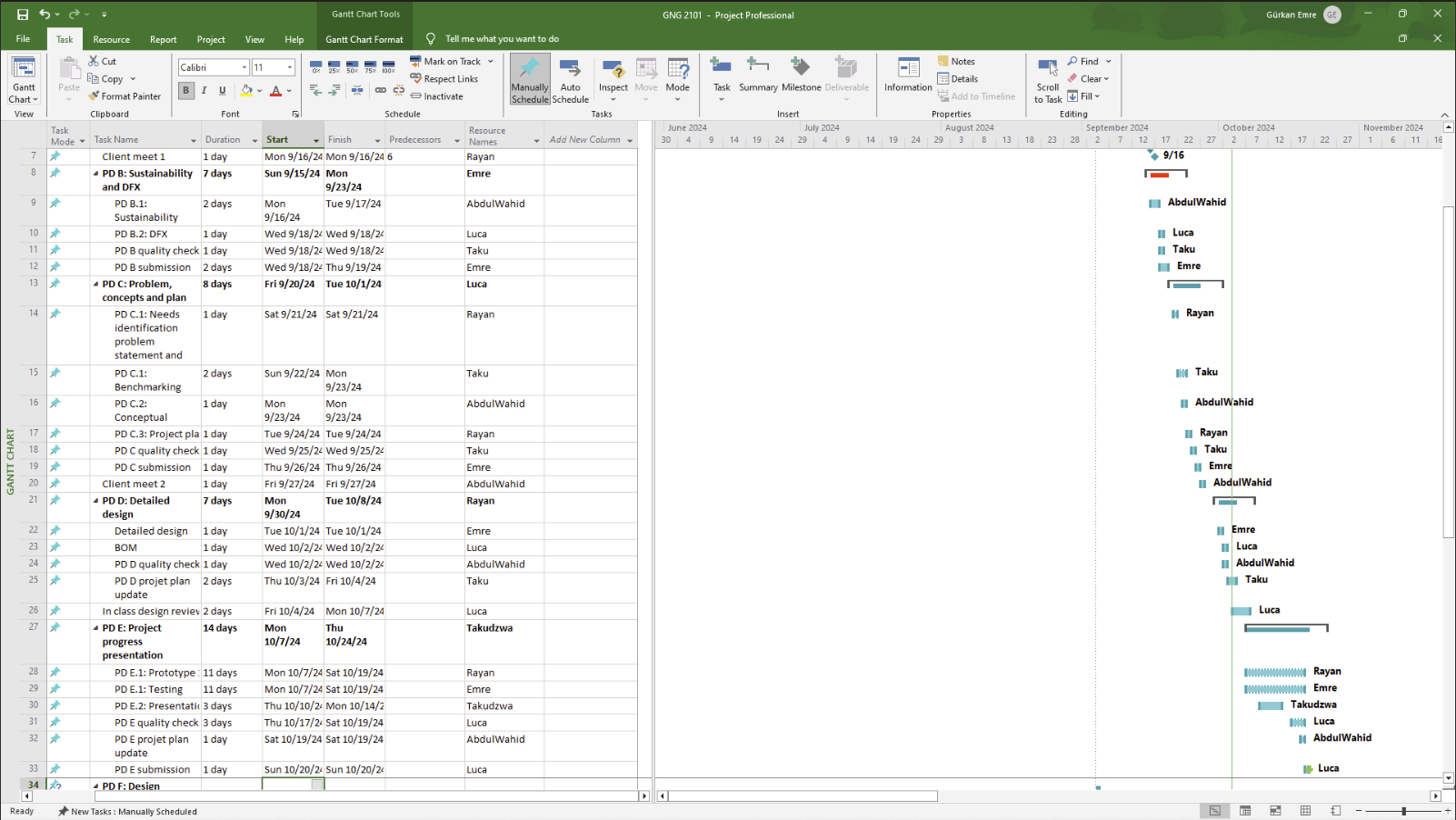
Based on the overall experience level of the team, the final prototype is expected to take roughly 5 weeks to complete, while the group has 7 weeks before Design Day.

## 4.2 BOM

Figure 14: Bill of Materials for Us

Due to this project being software-oriented, costs associated with prototyping and development will be low. To interact with asset barcodes, a barcode scanner will be required, which will be purchased for $15. If others wish to replicate our prototype, the bill of materials will be as follows.

## Figure 15: Bill of Material For Others4.3 Project Plan Update



# Conclusions

In conclusion, the asset inventory system for Bethany Children’s Health Center will provide significant benefits, such as improved asset management, and greater operational efficiency. While there are challenges related to production and sustainability, a life cycle assessment will help minimize environmental impacts. By focusing on five key design factors, the system will meet current needs and remain adaptable to future demands, positioning the health center as a leader in innovation and sustainability in the healthcare sector.

# Bibliography

United States Environmental Protection Agency. (2014, October). *Printed Circuit Board Recycling Methods*. EPA. <https://www.epa.gov/sites/default/files/2014-05/documents/handout-10-circuitboards.pdf>