INVENTORY MANAGEMENT SYSTEM

Olaf Carlson

INFO-451 May 1 2023

Table of Contents

Design of Tests

Customer Problem Statement

Technicians are arriving at sites to perform maintenance on equipment with an inventory system that is not accurate. This extends restoration of customers service by hours or days when spare inventory equipment is not available onsite for the repair. The feedback from technicians in the field is that the current inventory system is cumbersome to use and takes up an incredible amount of employee hours to validate inventory at each site yearly and keep it maintained.

The system application developed needs to be user friendly, reducing labor hours to inventory each building location yearly and keep inventory updated throughout the year. Login will require employees to enter their employee username and employee password to log into application as a security feature. Once logged into the application technician will be able to add, delete, and change status equipment by building location number. The equipment status options in inventory will include in-service, lost, pending return, spare, broken, and staging. The ability to search equipment by building location with queried options, HECI, Status, and Bay Location enables technician to validate equipment quickly by clicking a box. The equipment database includes the HECI, Status, Quantity, Cost, Description, Bay Location and Building Location number for each piece of equipment.

In summarizing everything that has been stated so far, technicians need an application that allows them to quickly pull up equipment by building location number with additional queries to validate the equipment as in-service, lost, pending return, spare, broken, installed, or staging. The application will allow technicians to reduce the amount of time validating equipment and maintaining inventory in building locations to free up more time meeting the needs of customers.

Glossary of Terms

- **HECI –** A set of letters and numbers associated with a piece of equipment by the manufacturer. Located on the front of the equipment.
- **Status –** in-service, lost, pending return, spare, broken, or staging. Equipment that is installed and working is **in-service**. Equipment that is not in building but is in inventory will be **lost**. Equipment to be returned to warehouse will be **pending return**. The equipment available will be **spare**. Equipment that are defective or not working will be **broken** status. Equipment that is part of future installation project at location have a status of **staging**.
- **Quantity –** count of specific HECI equipment in building location.
- **Building Location-** A two letter number followed by four digits specific to building location. State abbreviation such as IN for Indiana followed by a four-digit number.
- **Bay location –** Is an eight-digit number in which first two are the floor location followed by second four numbers as frame identification followed by two-digit number as bay location. Example 020004.03 is located of second floor, aisle 4 frame identification and bay 3.
- **Cost –** equipment in dollars.
- **Description-** Short description of pack less than forty characters.
- **Username –** Technicians' internal company username
- **Password –** Provided password giving technician access to inventory system.

System Requirements

Functional Requirements

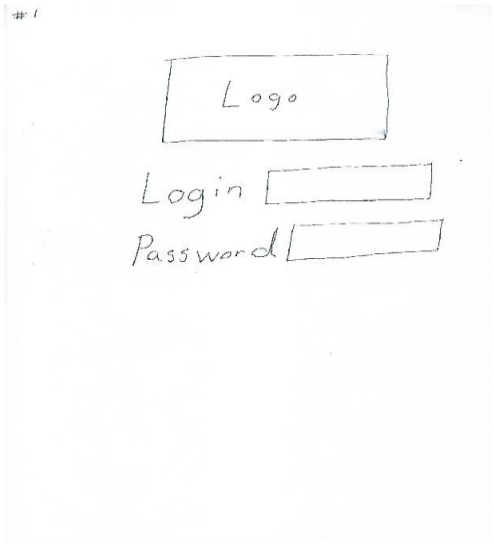
No.	Priority Weight	Description
REQ-1: Search Inventory	High	Allow technicians to search by building location number.
REQ-2: Validate equipment	High	Allow technicians to mark equipment as validated.
REQ-3: Change Status	High	Allow technicians to change status of equipment.
REQ-4: Shipping Document	Low	Allow technician to create shipping documents.
REQ-5: Log In	High	Allow technicians to enter AT&T username and AT&T password to log into application.
REQ-6: Add Equipment	High	Allow technician to add equipment to building location.
REQ-7: Print out Inventory	Medium	Allow technicians to print out inventory by building location.

Nonfunctional Requirements

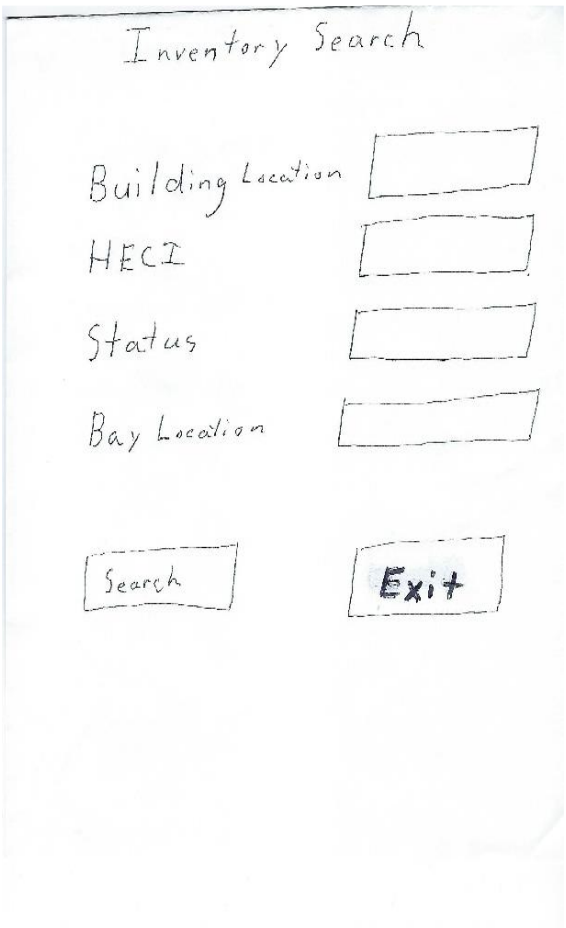
1. Functionality – application will be able to handle the volume and requirements of the system.
2. Usability – the application will be user friendly.
3. Reliability – the application will meet performance requirements throughout the day.
4. Performance – the application will maintain a rapid response time and be scalable.
5. Supportability – support will be accessible from six am to six pm Monday through Friday with updates being done on Sundays between ten pm to three am when the application has the least amount of traffic.

User Interface Requirements:

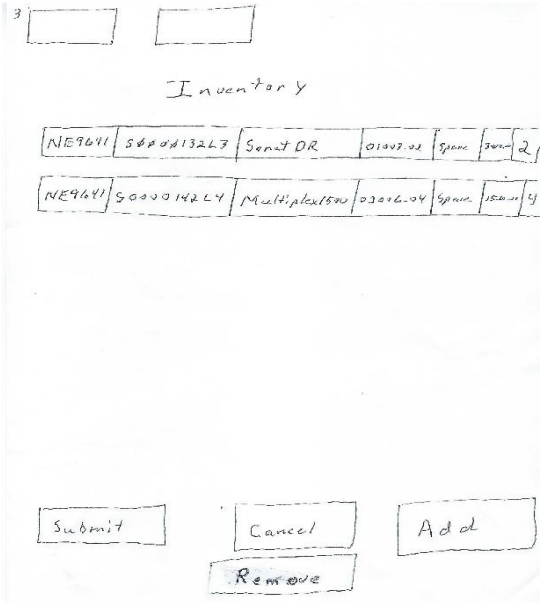
Application will run on a Windows operating system. To start the application user will click on a desktop icon to open login screen. Users will put in their login name and corporate password. After Login the users will have menu buttons at the top of the screen to navigate the application. Below are sketches of all screens along with a description of the screens that will be included along with locations of buttons, text boxes and menus.



1. Login Screen – A logo will appear at top of text box login UserName and Password. The user will type in their login name for username and then enter password.

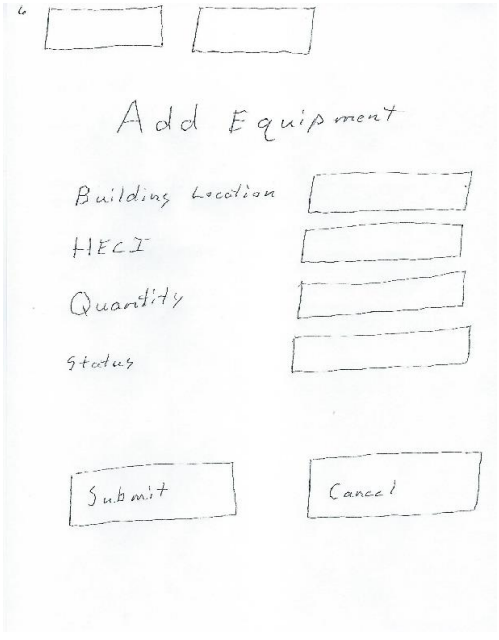


2. Inventory Page – The inventory page appears when the user logins to application. Users can input the building location number with queried options HECI, Status, and Bay Location. The building location number is a requirement to perform search and additional queries can be included to narrow down the equipment. Search inventory button along with cancel button appear at the bottom of screen. Clicking the search button will send the user to inventory list page. If inputs are not valid or no building location is inputted, the application will notify the user. Exit button will exit out of application.



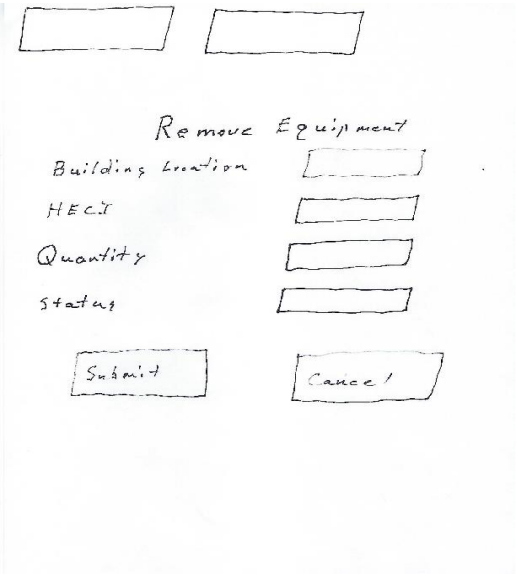
3. Inventory List Page – The inventory list page displays the list of inventoried packs searched from the inventory page. BuildingLocation, HECI, Description, Cost, Bay Location, Status, and Cost of the equipment for each inventoried equipment at building location is listed. Drop down menu will allow change of status to in-service, lost, pending return, spare, broken, or staging. Drop down menu will be provided for Quantity to change HECI equipment count. Bottom of the page displays a submit button to change inventory or cancel button to cancel changes returning to Inventory page. The Add-Equipment button at bottom directs the user to Add-Equipment page. Remove button at bottom will direct user to Remove-Equipment page. Button at top of page will direct user to Display Inventory page. Exit button at top of page will exit application.

4. Display Inventory Page – displays all the inventory in a building location. Users can print the page via the browser. User will exit by closing browser.



5.

Add Equipment Page – user is directed when Add-Equipment Button is pressed on Inventory List Page. Users input Building Location, HECI, Quantity and Status to submit new equipment. If invalid or not entered the user will be notified. Cancel will return user to View Inventory List Page. Exit button on top of page will exit the application.



6. **Remove Equipment Page-** users are directed to once Remove-Equipment Button is pressed on Inventory List Page. User must input Building Location, HECI, Quantity and Status to remove equipment. If is invalid or not entered users will be notified. Cancel will return user to View Inventory List Page. Exit button on top of page will exit the application.

Functional Requirement Specification

Stakeholders: Below is a list of Stakeholders that would be interested in this application.

- Technicians / Managers

Actors and Goals: Below is a list which will demonstrate the roles if people that will interact with the system. All users have the same view and application ability.

Primary Actors:

-Technicians and Managers: Technicians and Managers can Log into the system, Logout of the system, View inventory for a specific site, Validate inventory, Search inventory, Print inventory for a specific site, Add inventory equipment and Delete Inventory.

Secondary Actors:

System: Responsible for giving details of inventory equipment and making changes to that equipment based on user selection. Database is part of the system.

Use Cases:

Estimate how long needed to implement use case:

Technician (total: 21)

-Add Log into: To add login to system(2)

-Add Logout: To add logout of system (1)

-Add View Inventory: To display inventory for specific site (3)

-Add Validate inventory: To be able to Check off packs verified, make changes to status (5)

-Add Search inventory: To display prompt search criteria to search database (2)

-Add Print Inventory: To display inventory to be printed in printer format for a specific site (2)

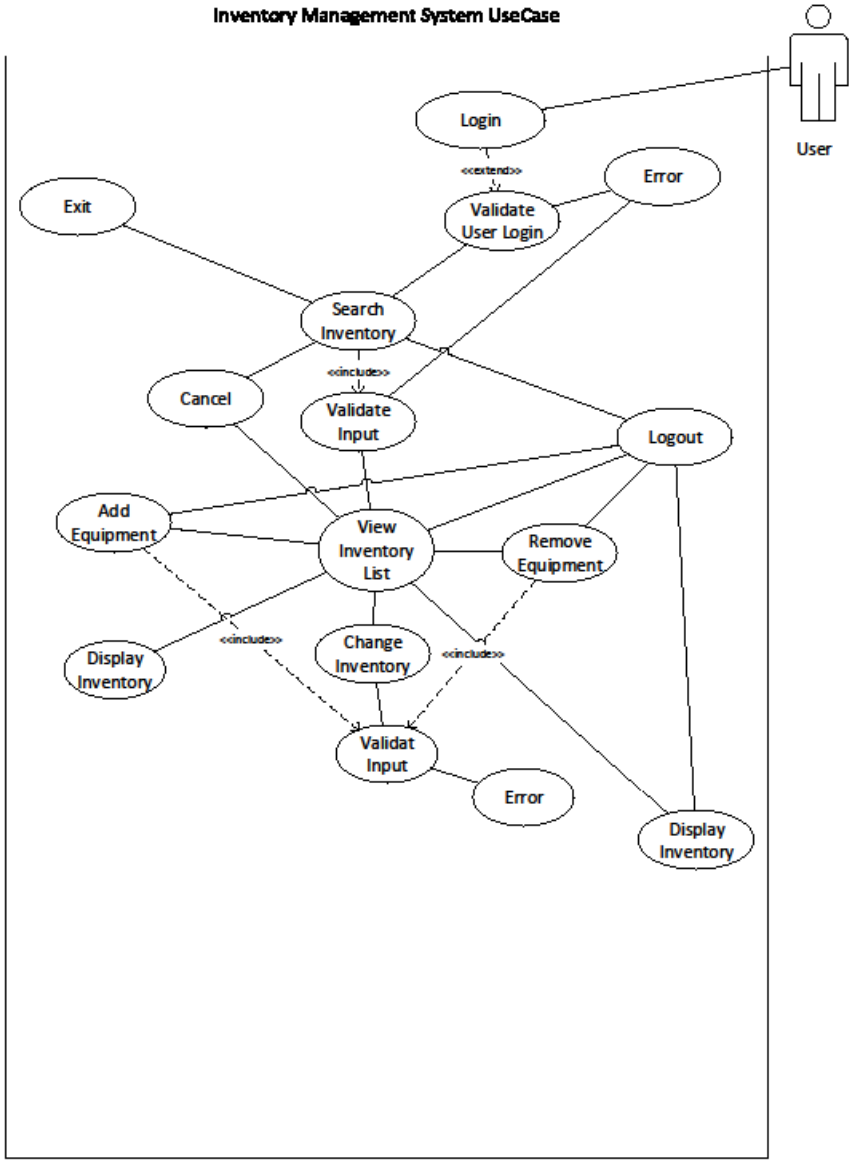
-Add Inventory: To be able to add equipment to database (3)

-Add Delete: To be able to delete inventory equipment in database (3)

System (total 23)

- Add backend: Develop MySQL database (7)
- Add front end: Develop GUI (10)
- Merge front end and back: Develop integration with MySQL database, GUI and Java application code (6)

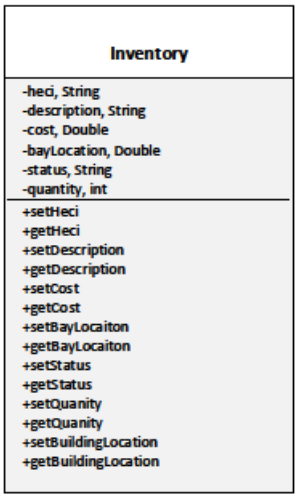
Use Case:



Class Diagram

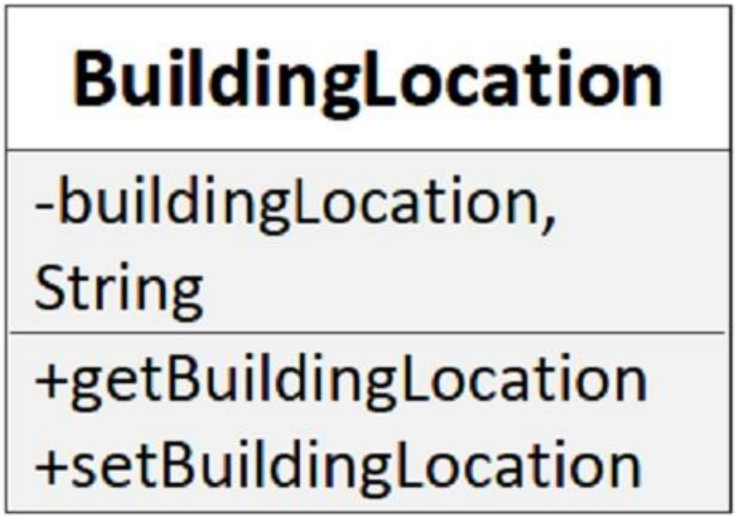
Inventory

We have six inventory fields: heci, description, cost, bayLocation, status, and quantity. Inventory Class will be associated with MySQL database.



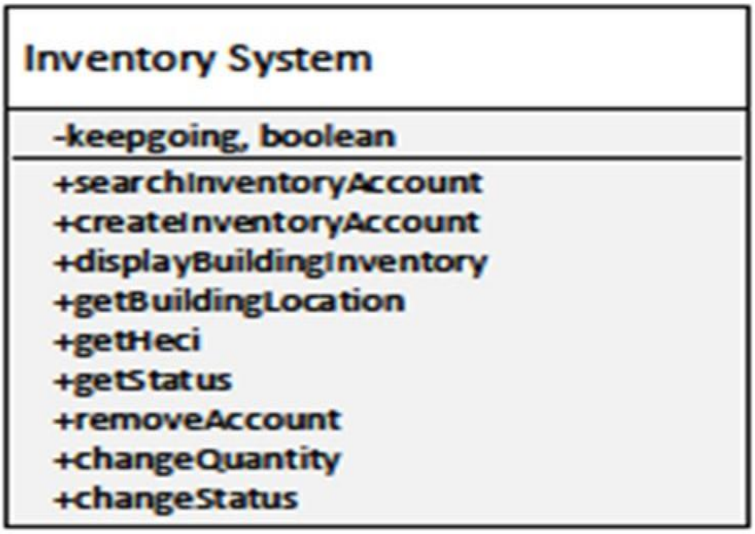
Building Location

This is an attribute for the class building location. It will be associated with a secondary mySQL database.

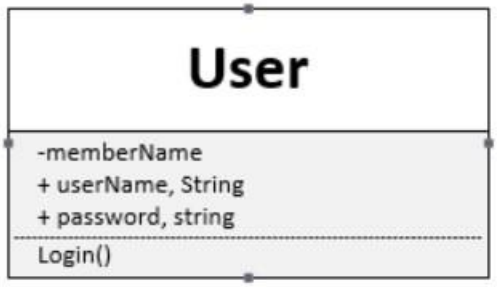


Inventory System

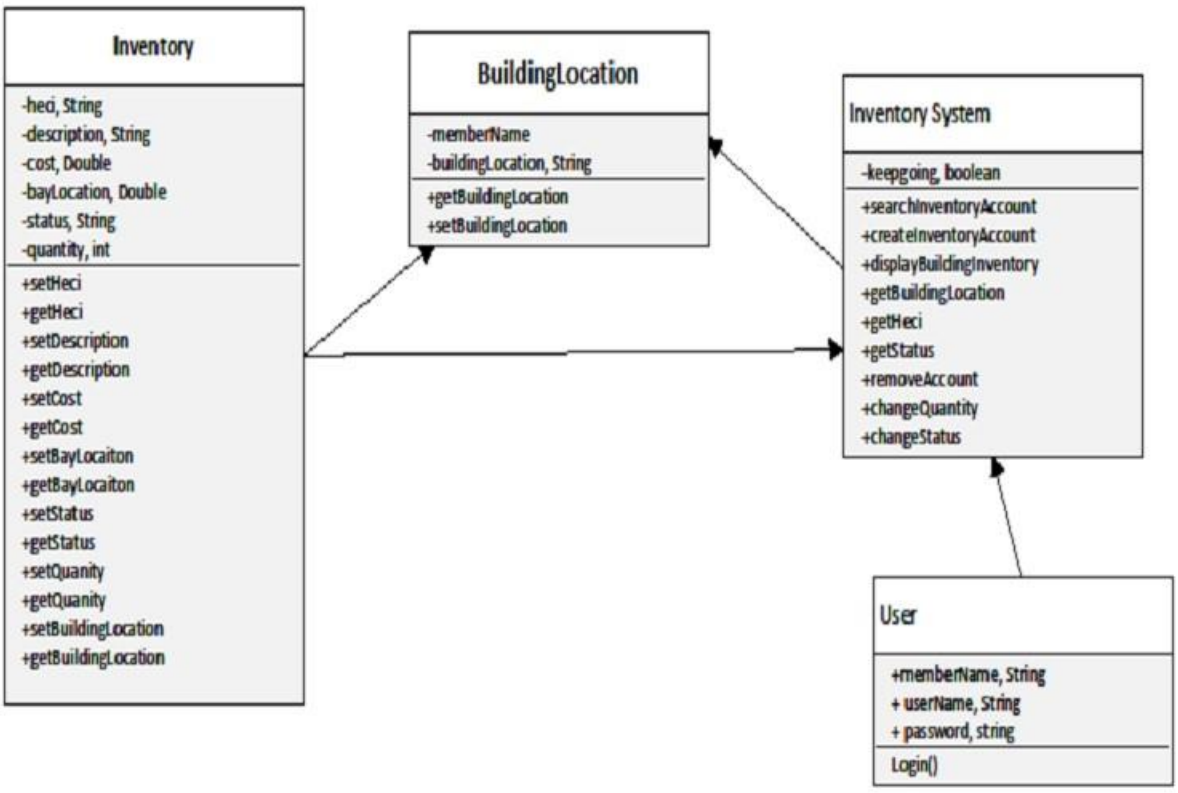
The Class allows user to input queries required Building Location with Heci or Status to search inventory or create inventory. User will be able to display building inventory by inputting some queries with the class. User will be able to remove account, change quantity and change status. This class will manipulate the mySQL databases using BuildingLocation and Inventory databases.



User Class allows the user to login to the application utilize a mySQL database for username and passwords.



Class diagram of Inventory Management System:
We can summarize the class diagram of the system as follows:

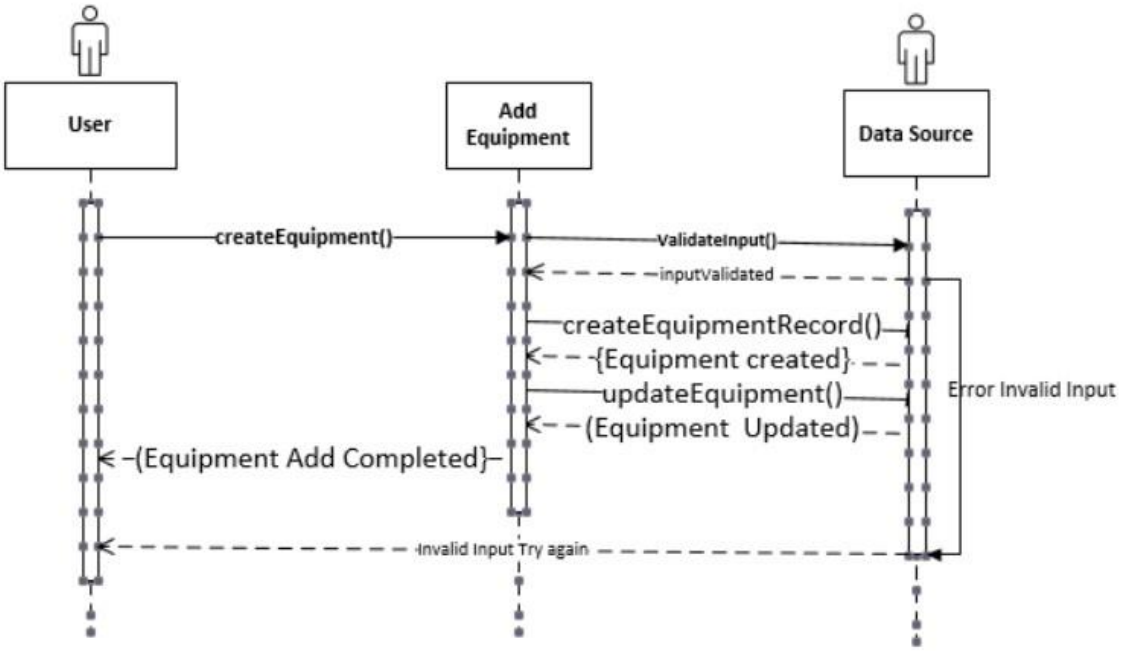


System Sequence Diagrams

Steps of Add Equipment

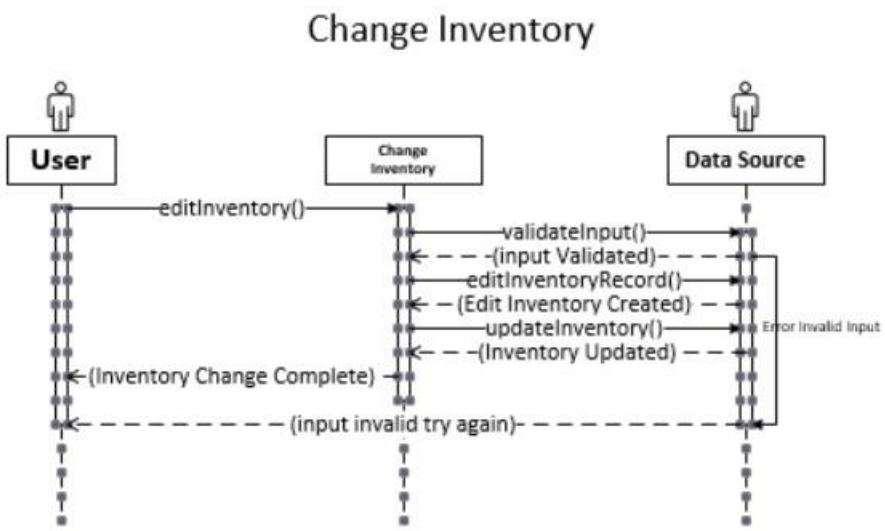
1. User inputs Building Location, HECI, Quantity, Status into form.
2. User clicks on submit.
3. User creates add equipment.
4. Add equipment data source validates if input valid.
5. Data Source creates equipment record.
6. Equipment is created.
7. Updates equipment record
8. Equipment is updated.
9. Equipment adds complete.

Add Equipment



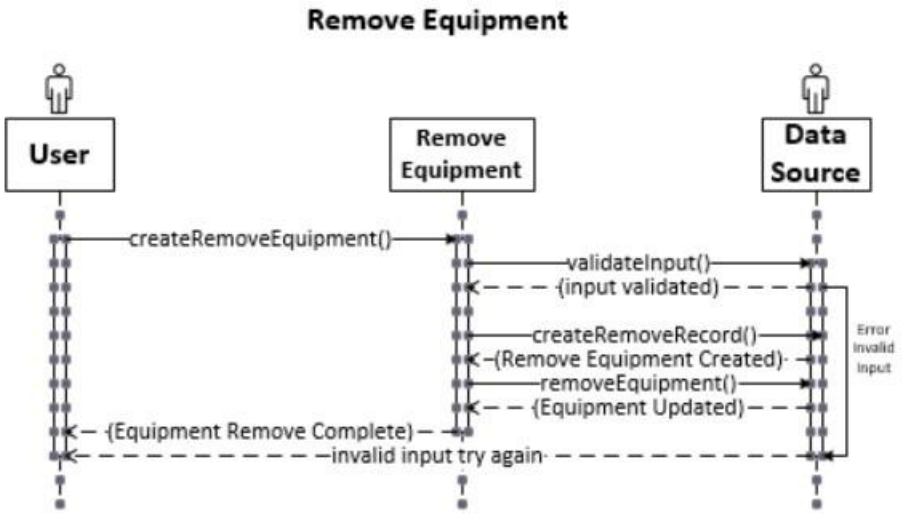
Steps of Change Inventory

1. User selects drop down boxes under Inventory List page to change status, and/or quantity.
2. User hits submit.
3. User creates change inventory.
4. Change equipment data source validation if input valid
5. Edit inventory record is sent to data source.
6. Edit inventory record is created.
7. Update equipment record.
8. Change inventory is updated.
9. Equipment change is complete.

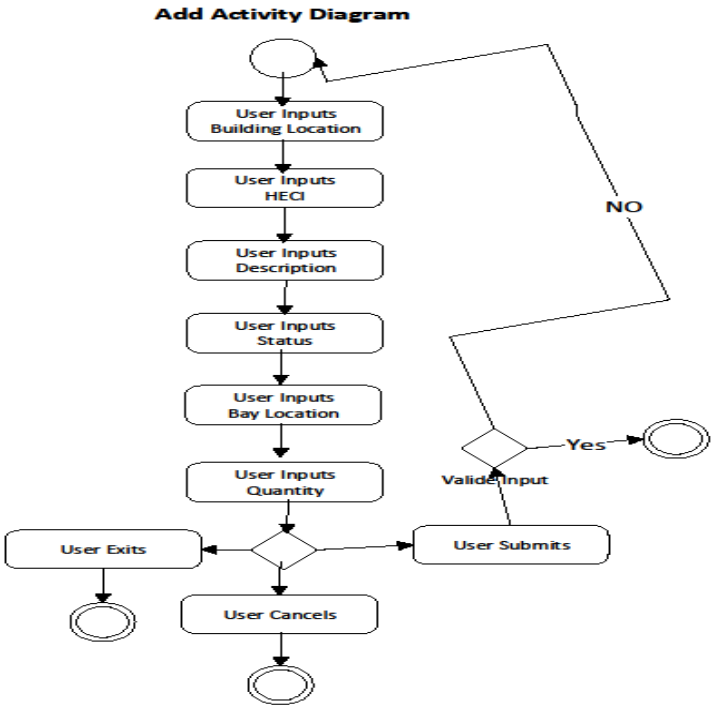


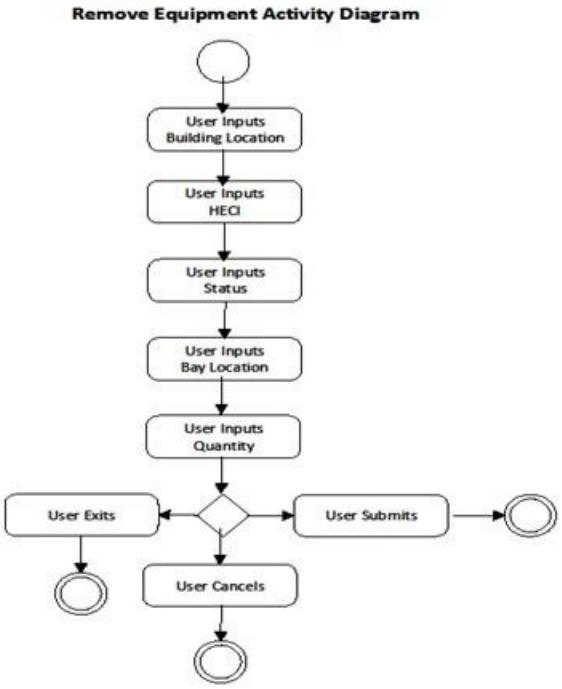
Steps of Remove Equipment

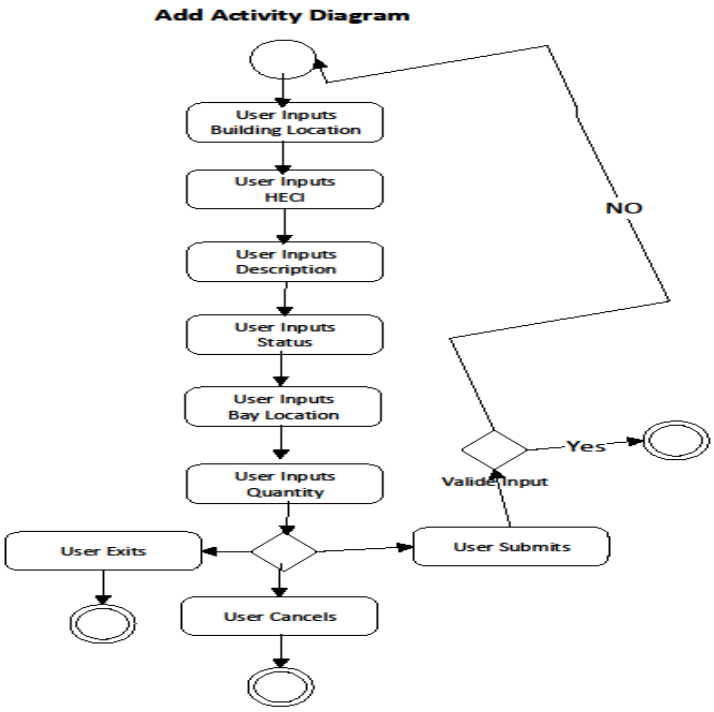
1. User inputs Building Location, HECI, Quantity, Status into form.
2. User clicks submit.
3. User creates remove equipment.
4. Remove Equipment data source validates if input valid.
5. Data Source creates remove record.
6. Remove record created.
7. Update equipment record.
8. Equipment is updated.
9. Equipment removes complete.


Remove Equipment
User
Remove Equipment
Data Source
createRemoveEquipment()
validateInput()
{input validated)
createRemoveRecord()
Error invalid input
(Remove Equipment Created)
removeEquipment()
{Equipment Updated)
{Equipment Remove Complete)
invalid input try again

Activity Diagrams

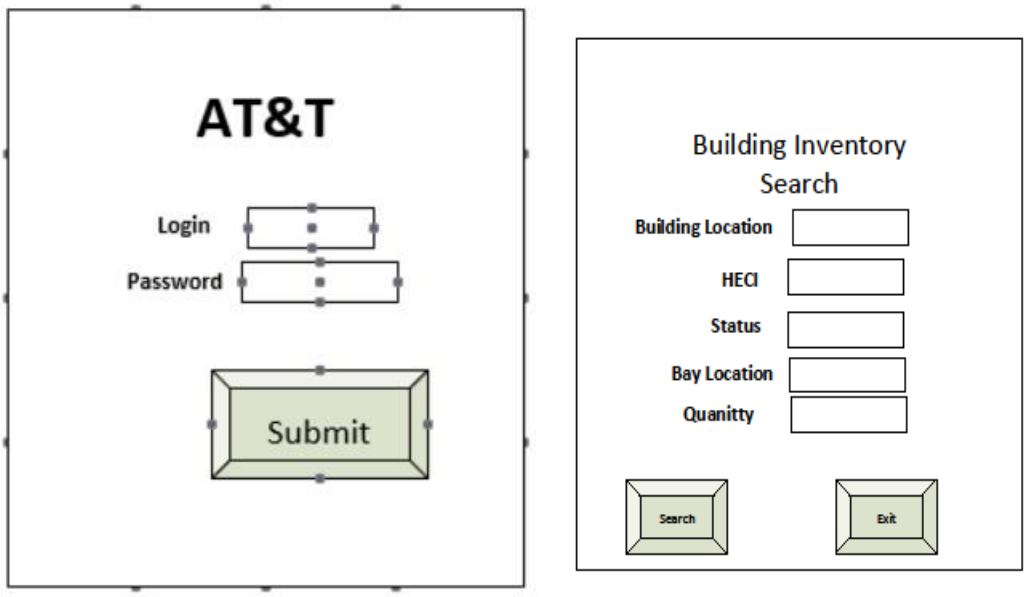




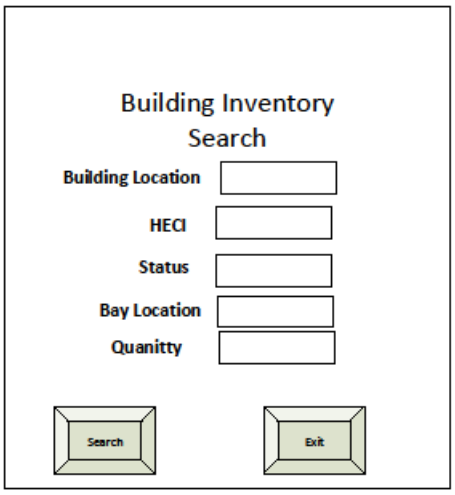

Add Activity Diagram
User Inputs Building Location
User Inputs HECI
User Inputs Description
User Inputs Status
User Inputs Bay Location
User Inputs Quantity
User Exits
User Cancels
User Submits
Valide Input
NO
Yes

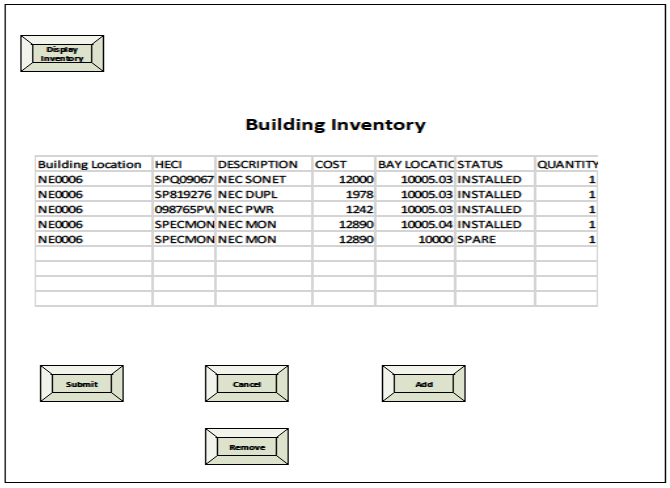
User Interface Specification

Preliminary Design: Shows what happens when user enters information and/or presses a button.

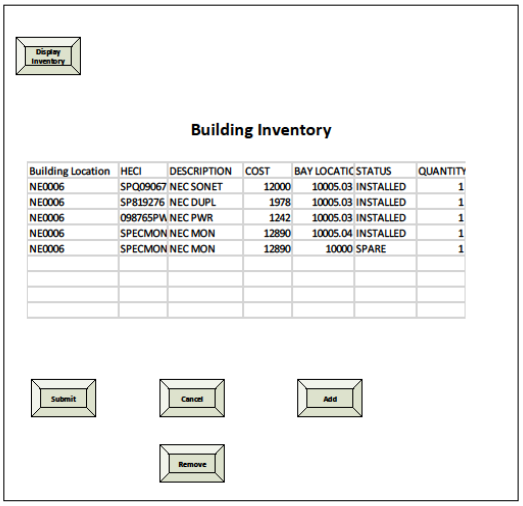


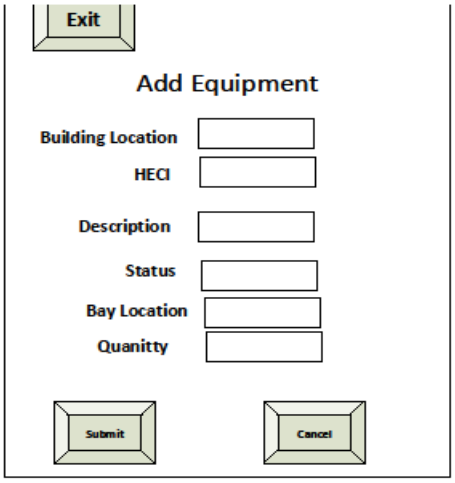
User inputs Login and Password upon pressing submit be taken to Inventory Page (Building Inventory Search page.



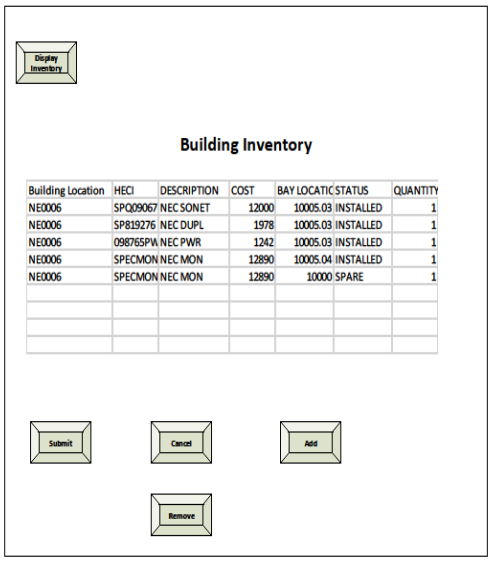


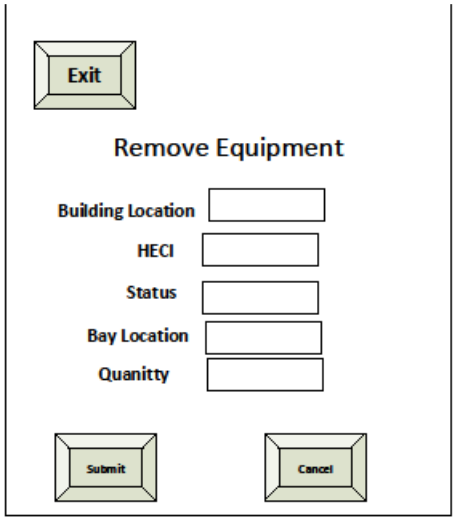
User inputs Building Location and any queries into Inventory Page taking them to a list of the Building Inventory.



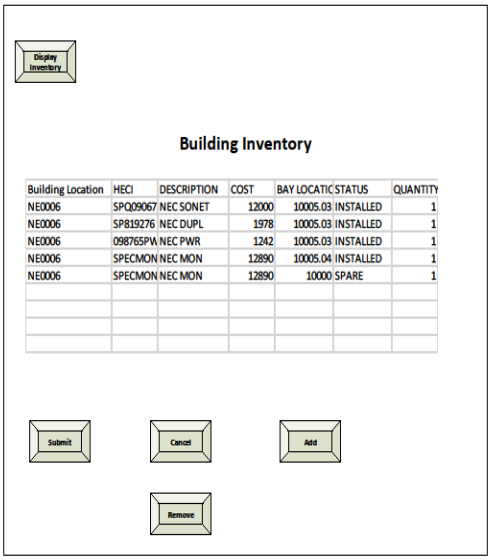


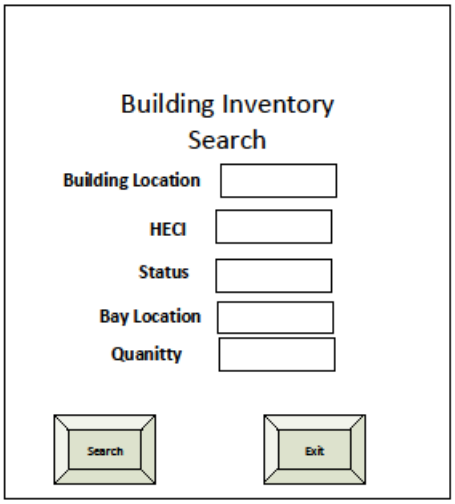
User presses add button is sent to Add equipment page.



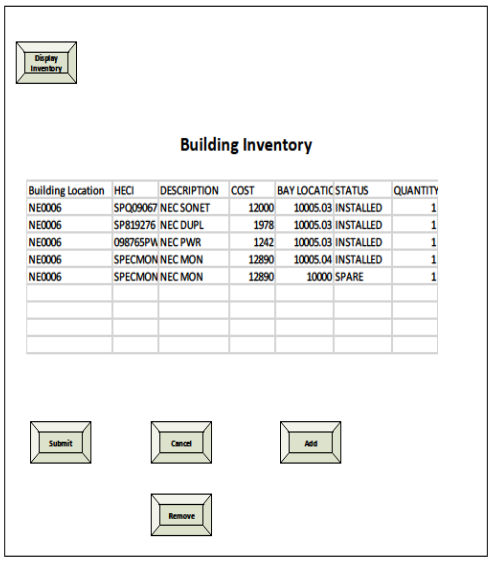


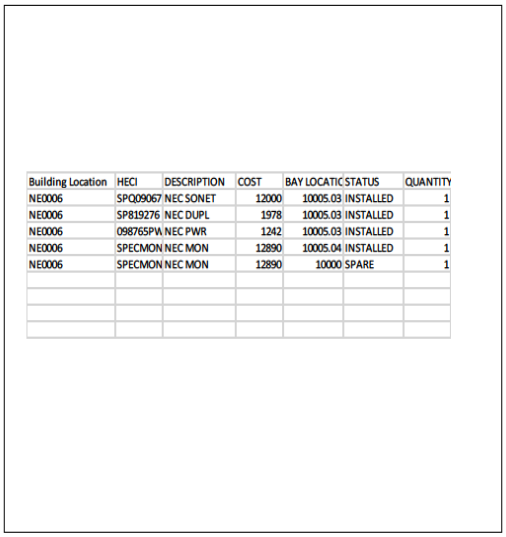
User presses Remove button is sent to Remove Equipment page.



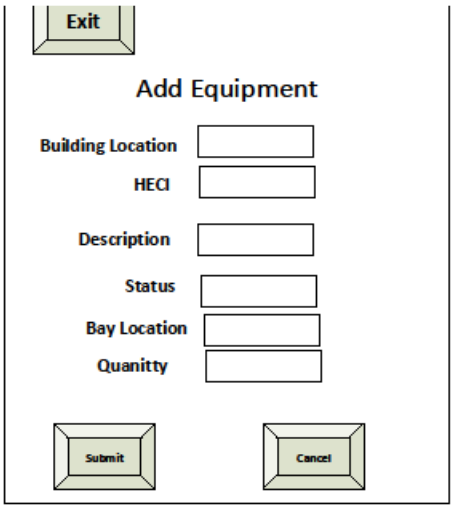


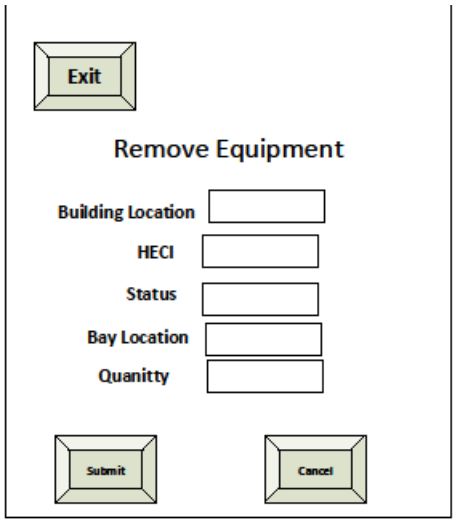
User presses cancel on Building Inventory page sent to Building Inventory Search page.

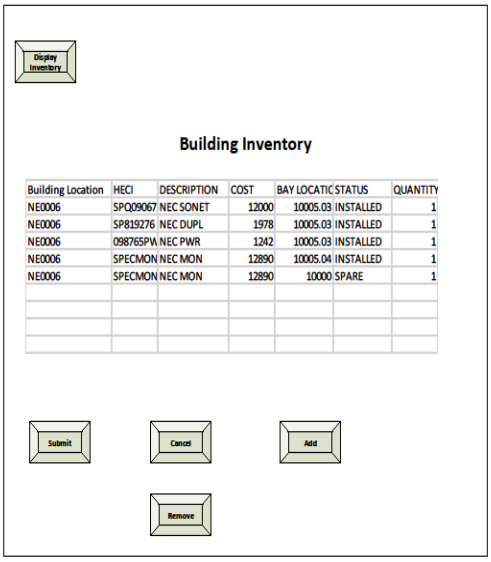




User presses Display Inventory button sent to Display Inventory page where they can print via browser.







User presses cancel in Add Inventory or Remove Inventory sent directly to Building Inventory page.

User Effort Estimator

Use cases and the number of estimated clicks to complete each task.

Use Case	Estimated Minimum Clicks
Login Page	3
Search Inventory	**5**
Edit/Update Inventory	4
Display Inventory	**1**
Add Equipment	5
Remove Equipment	5

Traceability Matrix

System Requirements with Test Cases Description

Req-ID	Requirement	Priority Weight	Test Case ID	Test Case Description	Status
1	Login	HIGH	TC01	Login with invalid username and password	Passed
			TC02	Login without username and password	Passed
			TC03	Login without valid username or password	Passed
2	Logout	High	TC04	Click Exit on Inventory Page	Passed
			TC05	Click Exit on Inventory List Page	Passed
			TC06	Closing Browser on Display Inventory Page	Passed
			TC07	Click Exit on Add Inventory Page	Passed
			TC08	Click Exit on Remove Inventory Page.	Passed
3	Search Inventory	High	TC09	User can search inventory by Building Location	Passed
			TC10	User can search inventory by Building Location & HECI	Passed
			TC11	User can search inventory by Building Location & Status	Passed
			TC12	User can search inventory by Building Location & Bay Location	Passed
			TC13	User can search inventory by Building Location, HECI, Status, and Bay Location	Passed
			TC14	User can search inventory by Building Location, HECI & Status	Passed
			TC15	User can search inventory by Building Location, Bay Location & Status	Passed
			TC16	User can search inventory by Building Location, HECI & Bay Location.	Passed

			TC17	**User does not put in Building Location.**	**Passed**
4	**Change Inventory**	**High**	**TC18**	**User can change inventory status on Inventory List Page**	**Passed**
			TC19	**User can change inventory quantity**	**Passed**
5	**Add Inventory**	**High**	**TC20**	**User can press Add Equipment Button on Inventory List Page to bring up Add Equipment Page**	**Passed**
			TC21	**User can input Building Location, HECI, Quantity and Status to Add Equipment**	**Passed**
			TC22	**User is missing any required Location, HECI, Quantity or Status user will be notified.**	**Passed**
6	**Cancel**	**High**	**TC23**	**User can cancel out of Add Equipment page**	**Passed**
			TC24	**User can cancel out of Remove Equipment page**	**Passed**
			TC25	**User can cancel out of Inventory List page.**	
7	**Remove Inventory**	**High**	**TC26**	**User is missing any required Location, HECI, Quantity or Status user will be notified.**	**Passed**
			TC27	**User can input Building Location, HECI, Quantity and Status to Remove Equipment.**	**Passed**
8	**Display Inventory**	**High**	**TC28**	**User can display inventory page**	**Passed**
9	**Print Shipping Document for equipment return**	**Medium**	**TC29**	**User can print shipping document to return equipment**	**Passed**

Use Case with Description:

No.	Description
UC1	Login with AT&T id and password
UC2	Logout
UC3	Search Inventory
UC4	Change Inventory
UC5	Add Inventory
UC6	Cancel
UC7	Remove Inventory
UC8	Display Inventory
UC9	Print Return shipping document

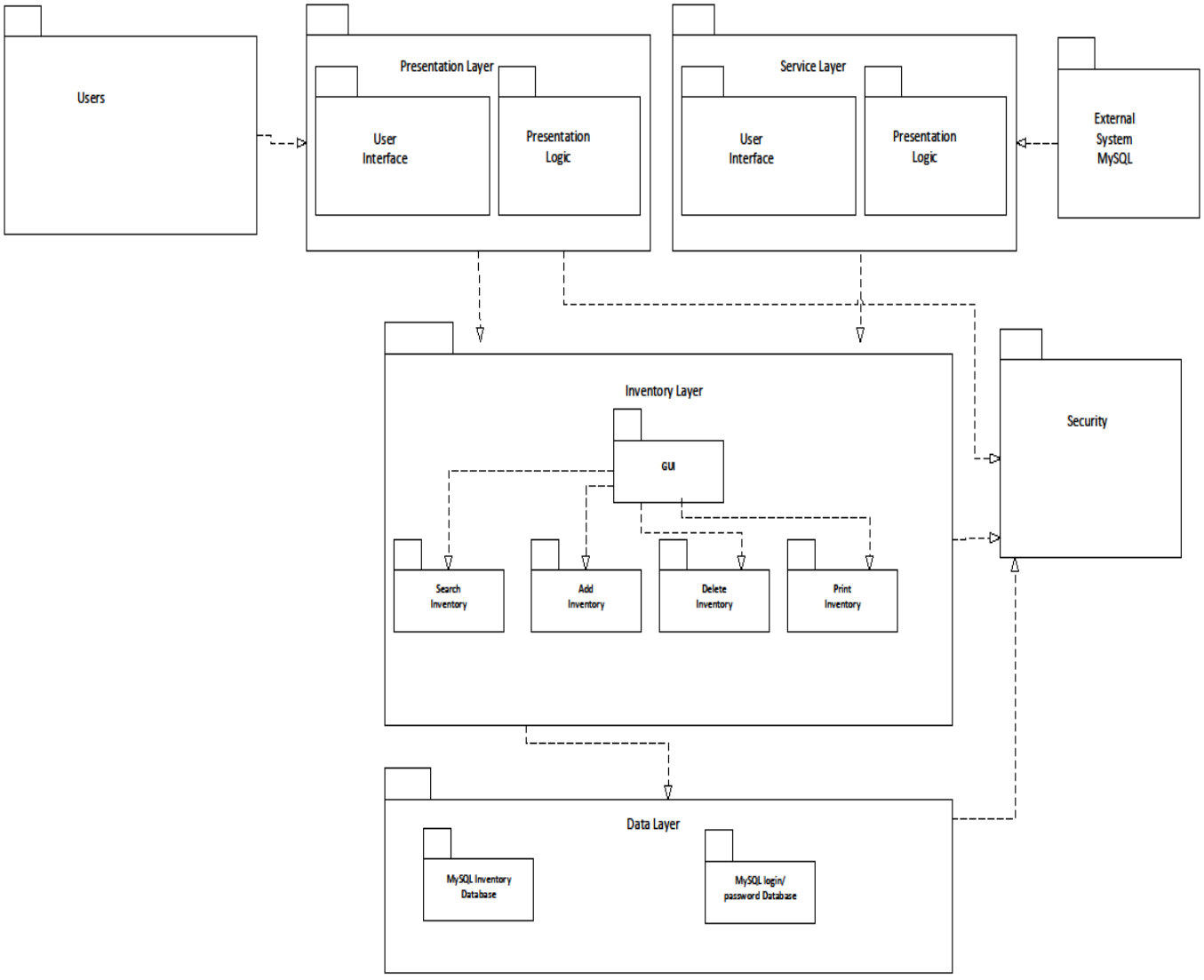
REQ	PW	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9
1	5	X								
2	5		X							
3	5			X						
4	5				X					
5	5					X				
6	5						X			
7	5							X		
8	5								X	
9	2									X
MAX PW		5	5	5	5	5	5	5	5	2
TOTAL PW		5	5	5	5	5	5	5	5	2

System Architecture and System Design

Architectural Design:

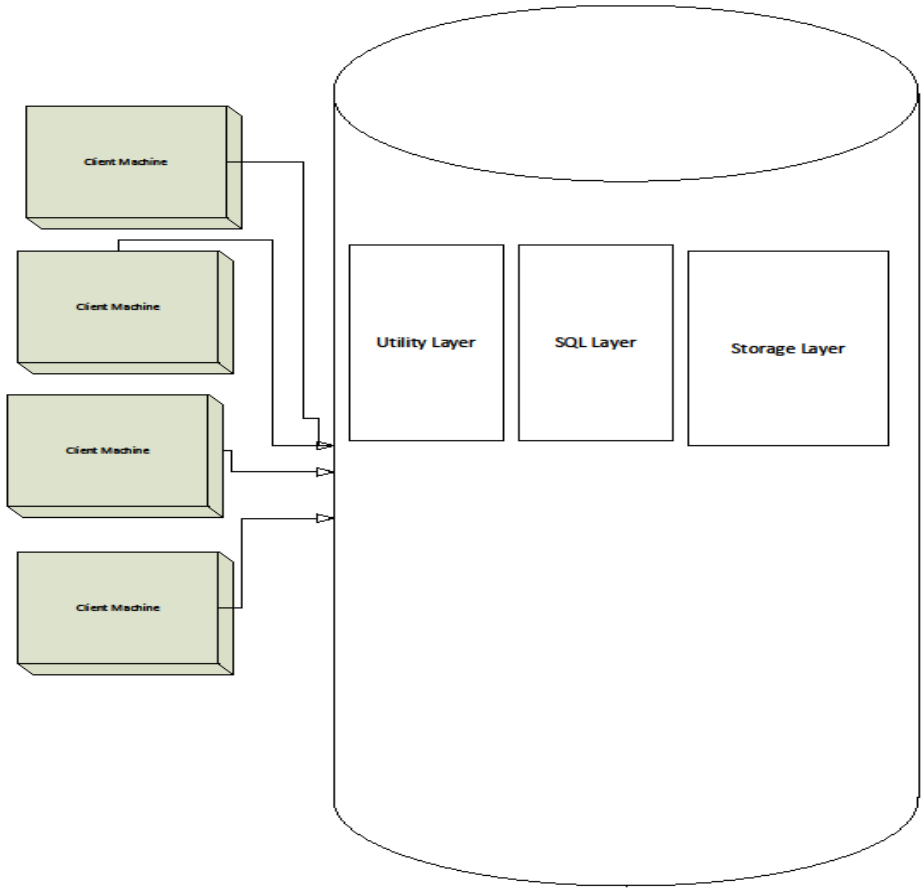
The architectural design of my design is a Client/Server design because the client handles the GUI, entering data, querying the data, and screen displaying output. The server utilizing my sql saves the data for passwords and inventory.

UML Package Diagram



UML Package Diagram Description: The GUI along with Java application sites on Client and then communicates utilizing Java JDBC protocol to MySQL database. Security is built in with Username and Password login making it part of GUI presentation layer, Inventory layer and data layer.

Mapping Subsystem to hardware



Utility Layer includes Connection Pool, Parser and check permissions.

SQL Layer includes Executor, Optimizer and Rewriter

Storage Layer includes Inner DB

Network Protocol

Will be using Java JDBC protocol because of its ability to read any database with no content being converted. It supports query and stored procedures. It enables Synchronous and Asynchronous processing.

Global Control Flow

Execution Orders: The system is procedure-driven in which all users go through the same procedures.

Time Dependency: No timers are built into the system. It is a system response time with no dependency on real time.

Concurrency: Does not have multiple threads.

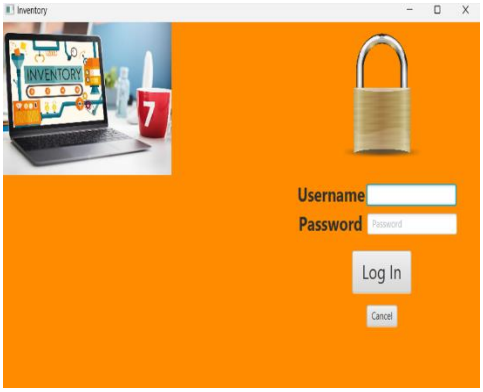
Hardware Requirements: Requires a desktop or laptop would recommend 8 GB of ram, 100 MB free on hard drive along with a color display 800p by 600p display Recommending a minimum network bandwidth of 3 Mbps.

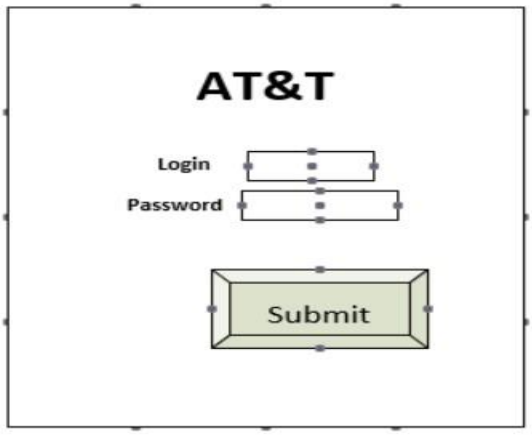
Algorithms and Data Structure

No Algorithms in the inventory project

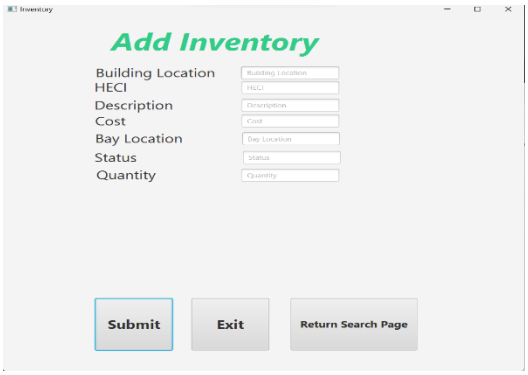
Data Structures - Inventory project utilizes MySQL which is not considered a complex data structure.

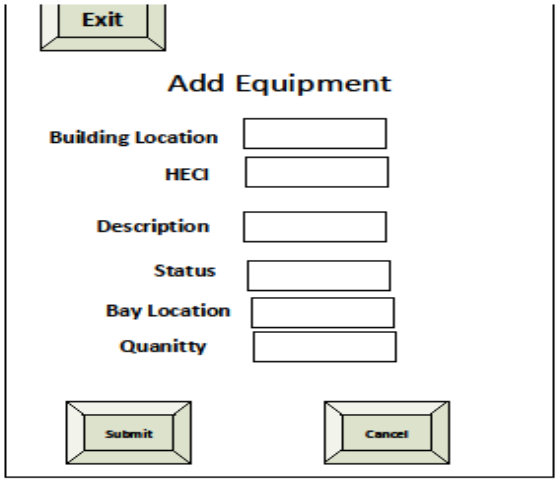
User Interface Design and Implementation:



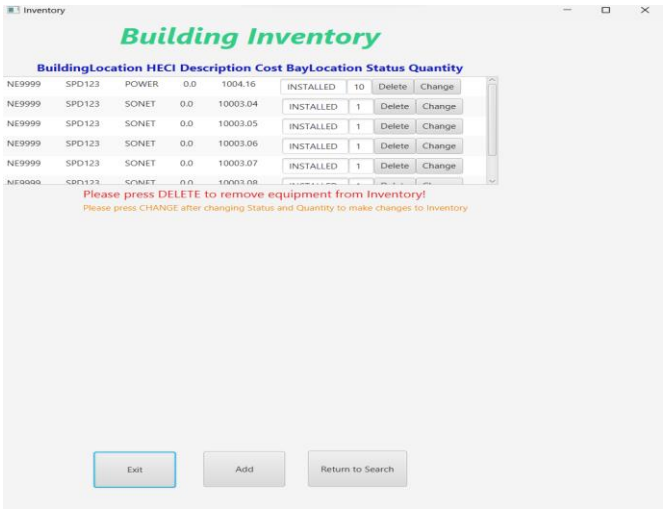


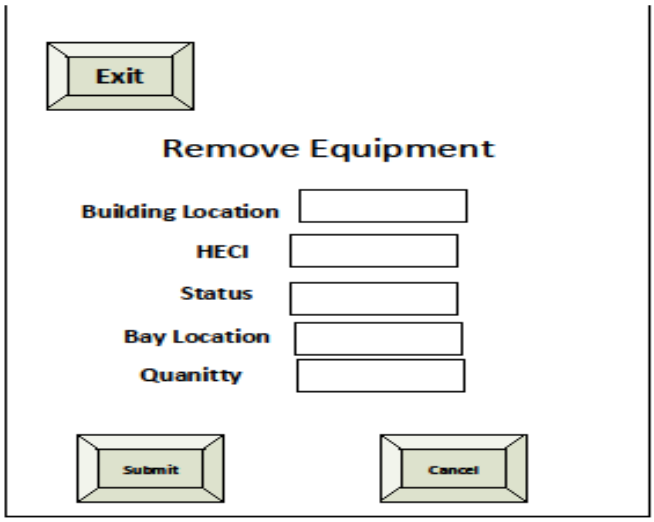
Utilized Scene Builder in IntelliJ and found out it was easy to develop a clean design utilizing graphics. It is cleaner and more attractive for the user. Really happy with the way the login GUI turned out. Changed submit button to a Log In button and utilized a cancel to exit application.



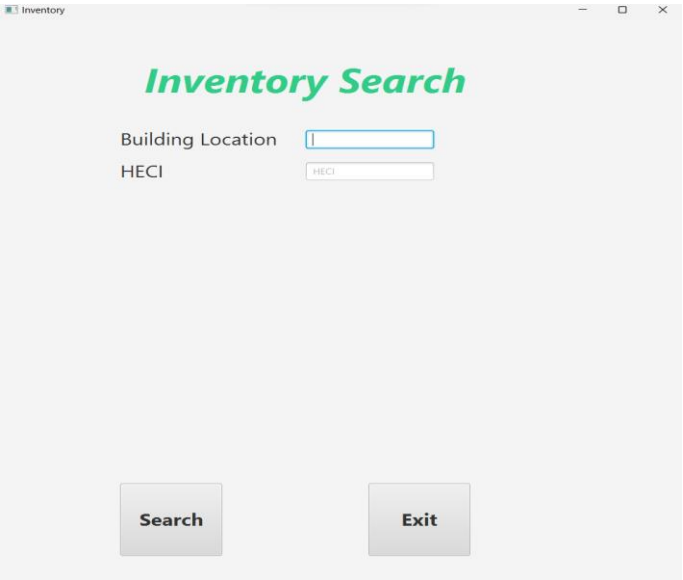


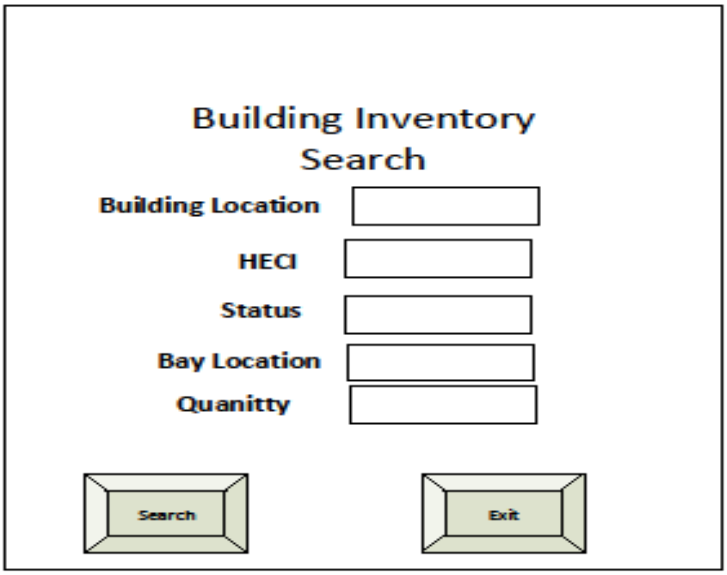
The Add Inventory GUI changed it from Add Equipment put in a little color and put all the buttons on the bottom. Scene Builder made it so easy to create a clean design.





Significant change in GUI put Delete button in Building Inventory List to remove Inventory by user. Also added on this pain Change button to make Text Fields for Status and Quantity changes to inventory.





Changed wording to Inventory Search and queries are only Building Location and HECI.

Several Changes encompassing Delete and Change features in Display Inventory Window GUI and several minor modifications to remainder GUIs in Application. Really pleased with how it turned out.

Design of Tests:

The use cases tested will include:

- **Login**
- **Add Inventory**
- **Delete Inventory**
- **Search Inventory**
- **Change Inventory**
- **Exit**

Perform Statement Coverage and Condition Coverage:

The method for testing these will be to log into the program and demonstrate each use case being performed. The login demonstration will show what happens when you enter an incorrect username or password and how the program reacts to each.

Add Inventory will show what happens when user inputs correct information into the applications and what errors the application presents when information is not inputted correctly.

Delete Inventory will show the user how it works as designed when correct information is inputted and how software handles errors when incorrect information is inputted.

Search Inventory will show the user how it works as designed and how it handles information not found in the query search.

Change Inventory will show the user how it works as designed and what happens if incorrect information is inputted into Change Inventory GUI.

The exit button will show user that it exits the software without affecting backend of software.

Integrity Testing:

Verify the database is manipulated per software design after performing each statement coverage and condition coverage software testing. Process to verify the backend of MySQL database is updated correctly per case study. This will test whether the backend is connected to the front end of the software.

User Interface Testing:

Through statement coverage, conditional coverage, and integrity the GUI will be able to demonstrate to the user that it performs as intended utilizing.