

# Milestone 5: Warehouse Management System

---

Updated Design and Near-Complete Implementation

Xingze Guo  
xig480

Rongli Han  
roh919

Xianming Luo  
xil430

Spencer Ondrusek  
spo798

Matt Triff  
mdt879

## 1. System Operations

### 1. selectShippingSpecification():

Controller – Shipper System

The user selects a shipping specification from the table in the GUI. They then press “Ship”. A new shipping task is generated within system, and the item number, item name, item quantity, and destination data is assigned to the new shipping task.

### 2. assignShippingTask(int employeeId):

Controller – Shipper System

The user selects a Stock Handler and a shipping company from the tables displayed in the GUI. Once they press confirm, the system will assign the selected Stock Handler and shipping company to the new shipping task for stock gathering activities.

### 3. addEmployee(String name, int ID, String title):

Controller – Manager System

The Manager clicks the “Add” button at the bottom of the Manager UI. A new window is visible and the Manager enters the new employee’s information in the related text fields. The Manager can then click “Confirm” to complete the process, or “Cancel” to abort.

### 4. editEmployee(String name, int ID, String title):

Controller – Manager System

The Manager selects an employee in the current Employee List and clicks the edit button to open a new window. The Manager then updates any text field with their changes and then clicks “Confirm” to complete the process, or “Cancel” to abort.

### 5. deleteEmployee(int ID):

Controller – Manager System

The Manager selects an employee from the GUI and clicks “Delete” to remove their information from the system.

### 6. routeStockTasks(Object[][] itemsList):

Controller – Stock Handler System

A list of items to be gathered (or put away) is passed to the operation. The items are then ordered by the operation to provide the optimal path. The ordered list is then passed back to the stock handler controller.

7. locateProduct(int itemId):

Controller – Map System

An item ID is passed to this operation. The operation then locates the products current location(s) within the warehouse and returns the bin coordinates (x, y) back to the system.

8. drawOnClickBin(int x, int y):

Controller – Map System

When a user double-clicks on a empty bin location on the map, this operation will add the new bin to the bin array and call the repaint() function to draw the bin, which will display on the map as a blue box to the user.

9. drawCoordinate (Graphics g):

Controller – Map System

When a user drags a bin to move it, this operation will draw green lines in the x and y planes to highlight the bin's current location. If the user is over an invalid square (such as a another bin) the coordinates will be drawn with red lines.

10. loadTable(int orderNumber):

Controller – Stock Handler System

Given an ID number for a specific Product Order, the system will call the database, retrieve all of the order's information, format the information, and display it in the table within the GUI.

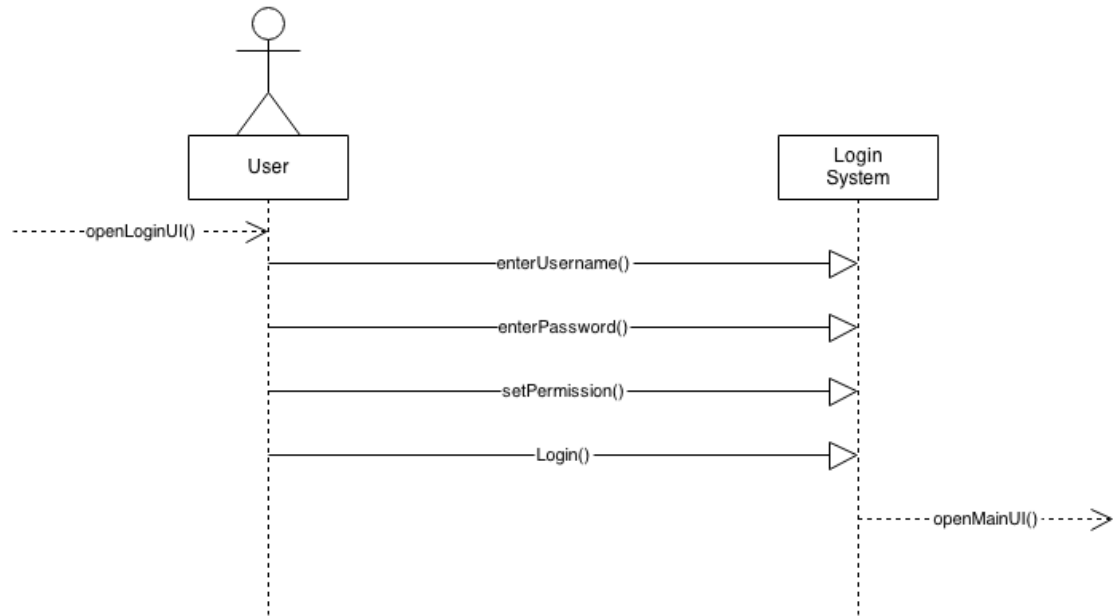
11. printTable(javax.swing.JTable palletTable):

Controller – Stock Handler System

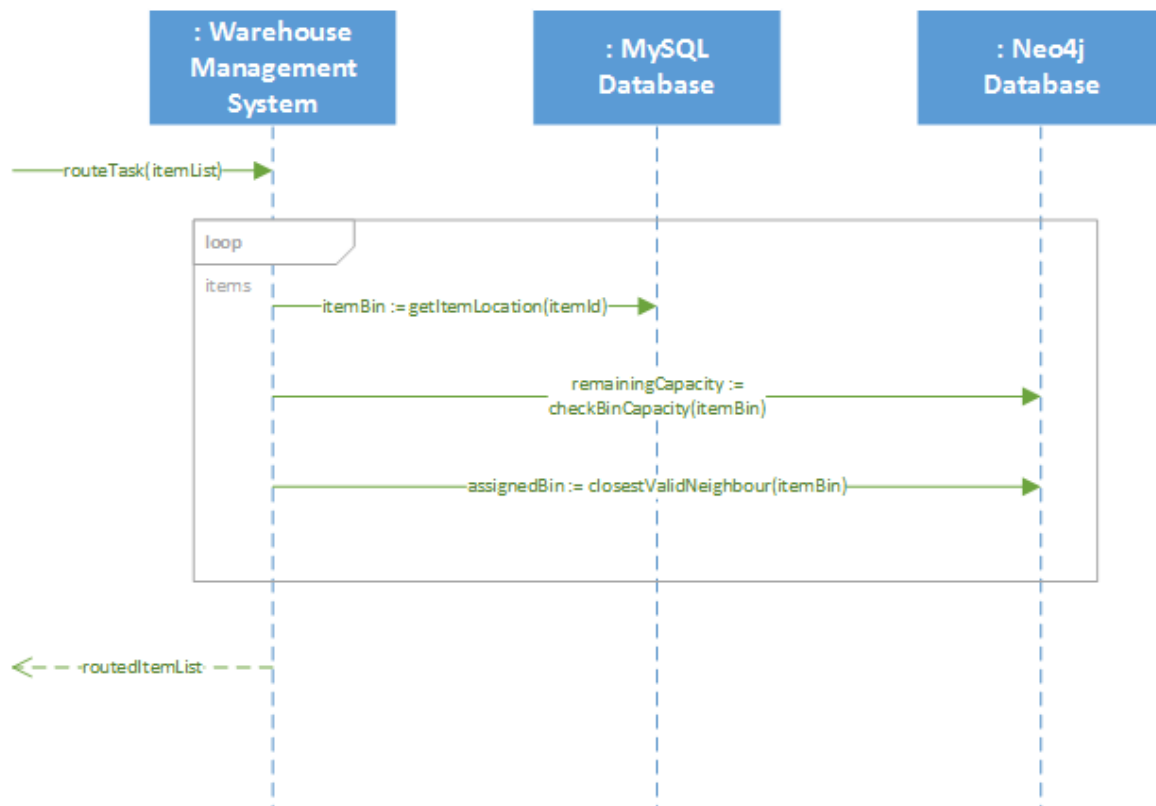
Format's the currently selected table and sends it to the user's printer. The user (a Stock Handler) can then take the paper with them as a guide for their daily tasks.

## 2. Interaction Diagrams/Sequence Diagrams

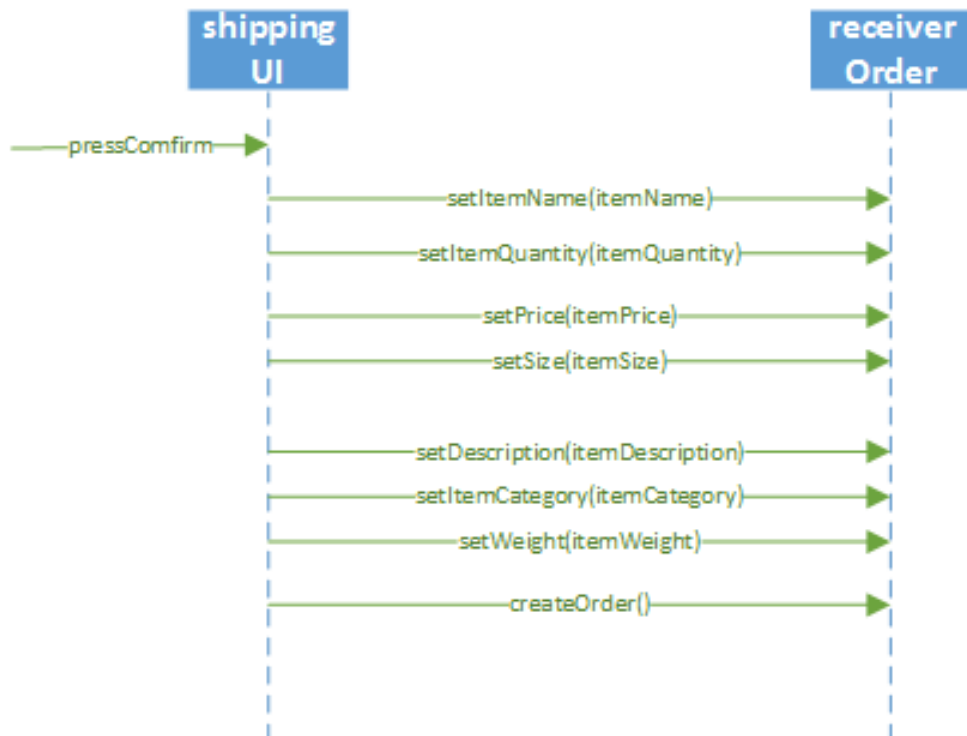
*Login – openLoginUI()*



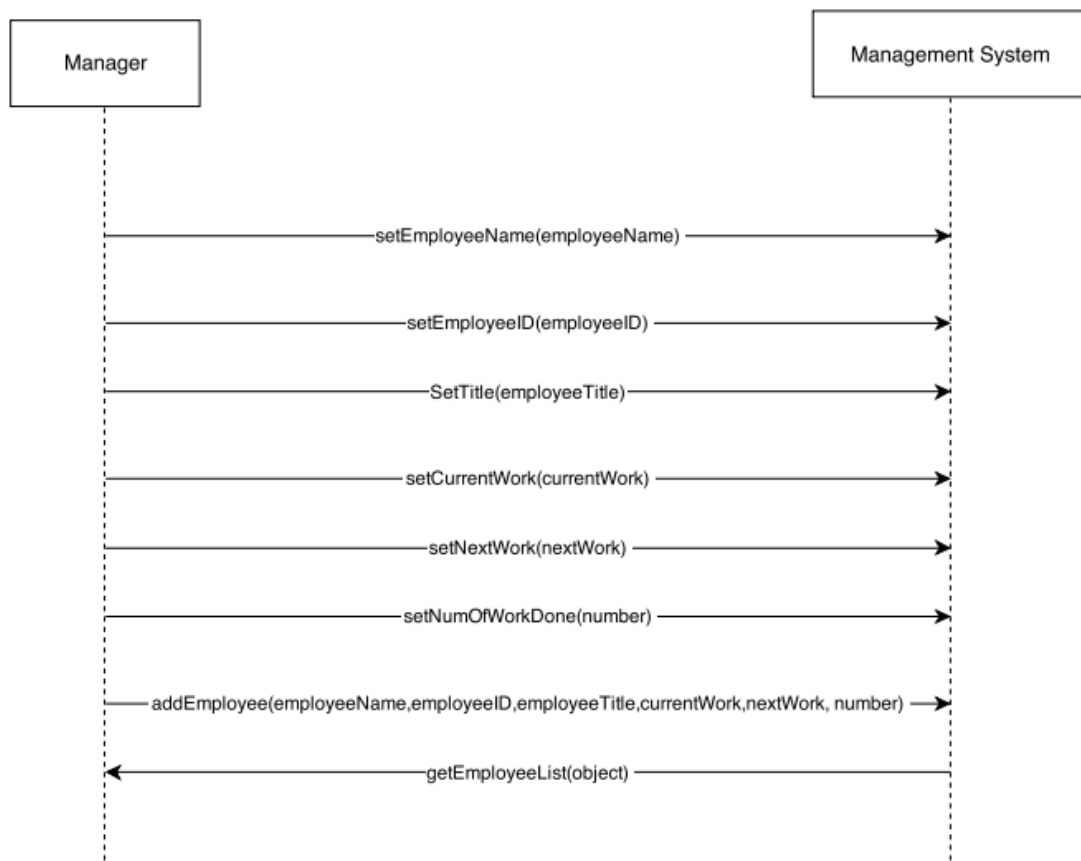
*Routing Stocking Tasks – routeTask(itemList)*



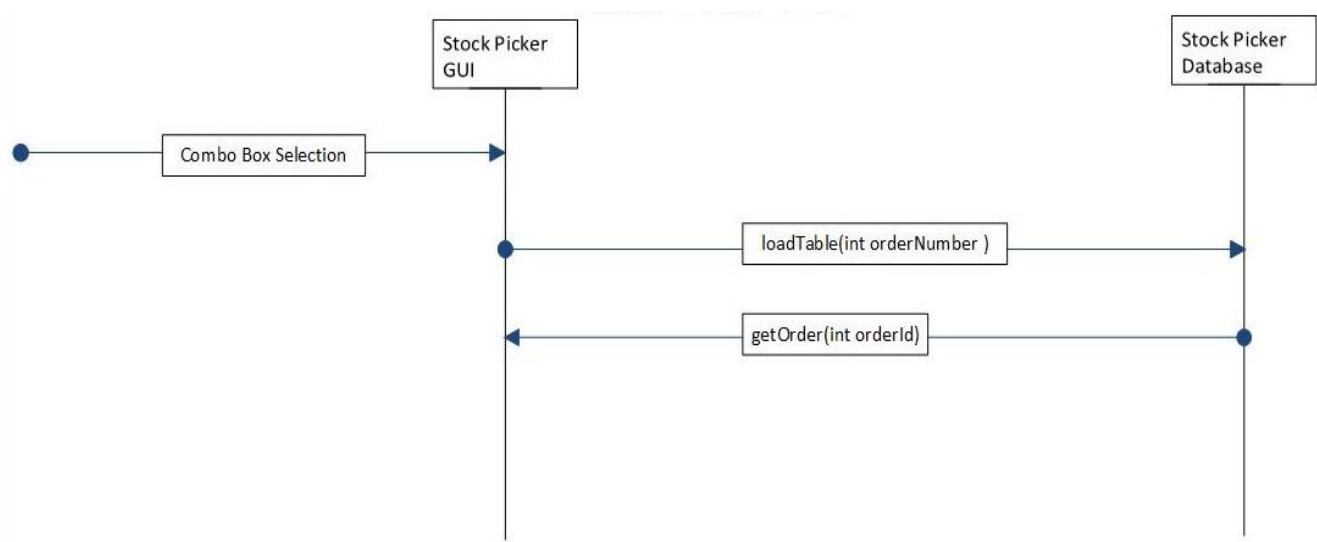
*Confirm Products Received – confirmReceive(item)*



*Manage Employees – updateEmployee(name, ID, title, work, nextWork, number)*

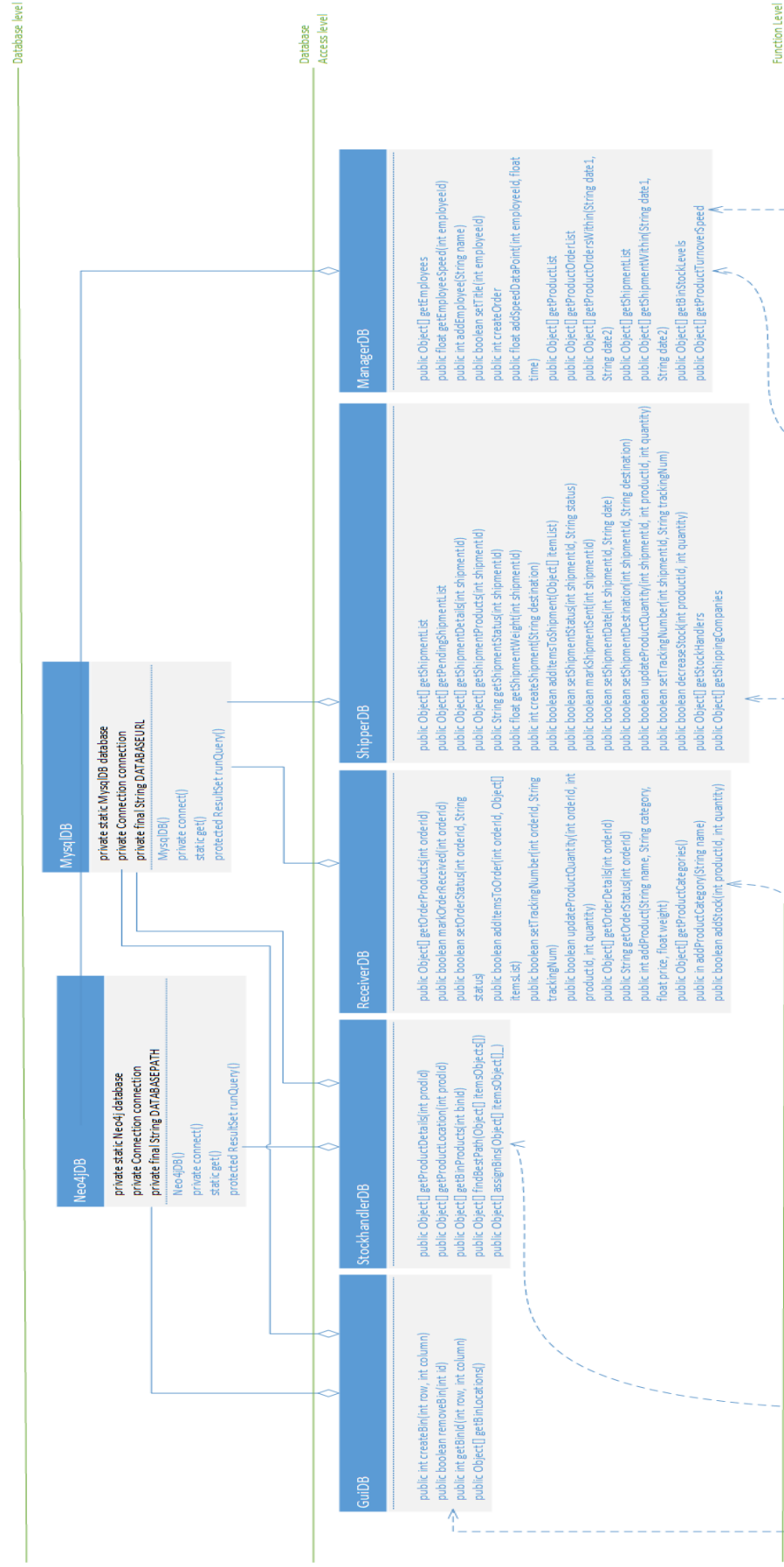


*Load Table Item – loadTable(int orderNumber)*



### **3. Class Diagram**

Due to the size of our class diagram, the diagram below has been split over two pages. To view the diagram in its entirety, please open `group02_classDiagram.pdf`, also contained in our submission.







## 4. Implementation

Please see the included ReadMe.txt file for instructions on running the project.

*Note:* Although each use case has had a success scenario fully implemented, these use cases are not feature complete, and special cases, failure paths, etc. may not be handled correctly.

### f. User Manual

#### ***Login User Manual***

Currently, the system only has one user, to log in enter the credentials as follows and click the “Login” button:

User ID #: 1

Password: password

Permission: Manager

Any other credentials will deny your access to the system. An employee with “Manager” permissions has access to every part of the system.

#### ***Database User Manual***

The Warehouse Management System utilizes two different databases, a MySQL Database, and a Neo4j Graph Database.

The MySQL database is located on the University of Saskatchewan servers, and is connected to remotely by the user’s machine upon the first query the system makes to the database.

The Neo4j database is stored locally within the application, in the graphDatabase/warehouse-db folder. No set up is required by the user to install or use Neo4j.

*Important:* Only one instance of the program may be run at a time. Multiple instances will cause the application to fail, due to issues connecting to the Neo4j database.

#### ***Map User Manual***

The map system allows users to create and view a digital floor plan of their website, to aide them in planning efficiencies and navigating their warehouse.

The bins and locations on the map will be saved on each change (via the graph database) and reloaded on each subsequent runs of the program. A simple floor plan has been initialized with the files submitted.

To completely clear the map, simply delete the graphDatabase/warehouse-db folder (a new, empty version will be created the next time the system starts).

The user can interact with the map in the following way:

1. Add Bin: Double click an empty (grey) cell.
2. Delete Bin: Right click a non-empty cell and select the “Delete” option in the popup menu.
3. Move Bin: Click and drag the bin around the map. If the coordinate lines appear red, you are not allowed to move the bin to that location. Moving the bin too fast at the start of the drag motion may cause the bin not to be fully deleted.
4. Locate Bins: From the menu bar, choose Search -> Locate and enter the item ID, then press the “Locate” button.
5. Check Details: Double click a non-empty cell.

*Note:* Do not move the bins out of the map

### ***Stock Handler User Manual***

The Stock Handler system allows Stock Handler employees to select from the list of tasks assigned to them, and print a step-by-step plan of their tasks.

From the map screen, click on the “Stock” button on the right side of the screen.

Next, select the order you wish to complete the stock gathering tasks for by clicking on the drop down list in the top left corner of the new window.

From the list, select the order ID you wish to be displayed in a sorted order by optimal route. The table in the main window will be populated with all of the items for the specified order. Click the “Print” button in the top right corner to open the “Print Options Menu”.

Specify your printer options, or simply click print to print the document to output the list of items, ordered in the fastest possible route.

### ***Shipping User Manual***

The Shipping system allows Shipper employees to view current and future orders from customers. The Shipper employees can then assign the stock handling tasks for each current order to Stock Handlers, and specify the shipping company who will transport the products.

From the map screen, click on the “Shipping” button on the right side of the screen.

Within the Shipping interface, the user can view all of the shipments ready to be shipped today.

Users can view future shipments (shipments that are currently pending) by clicking “Future Plan”.

To make a shipment, select a shipment listed in the table within the UI and click the “Ship” button.

After clicking the “Ship” button, the user can assign the shipping task to Stock Handler and shipping company, as well as a Tracking Number, by clicking an option from each of the two menus. The tracking number is in fact an alpha-numeric String.

Select the Stock Handler and Shipping Company, and input Tracking Number, then click the “Confirm” button to create the new shipping task.

To cancel input at any time, click “Cancel”.

### ***Receiver User Manual***

The Receiver system allows Receiver employees to enter any newly received products into the warehouse system.

From the map screen, click on the “Receiver” button on the right side of the screen.

Completely enter all the information for the incoming item and press “Confirm”, the new item will then be recorded to the database.

To cancel input at any time, click “Cancel”.

*Note:* The expected input format for this menu is as follows:

Name: String

Item Number: int

Quantity: int

Price: float

Size: float

Description: String

Weight: float

## ***Order Stock User Manual***

The Order Stock menu allows the user to create purchase orders for more stock for the warehouse.

From the map screen, click on the “Manager” button on the right side of the screen and click on the “Order Stock” tab on top of the Manager interface.

Users can search the details of the item by following setup:

1. Input item name or item number.
2. Click the “Search” button.
3. If system can find the name or number in database, other information will fill in the text field automatically.  
(If user input both name and number, the system will only use the name to search)
4. Input Quantity

If the same item is not already in the database, the user will need to input all of the information for the product. If the system cannot find the item, none of the fields will update.

When the user clicks the “Confirm” button, the new order will be created.

*Note:* This functionality is not complete. To test for now, use the following input:  
Input "apple" or "banana" in the Name field, or  
Input "1" or "2" in number text field

## ***Modify Inventory***

The Modify Inventory menu allows the user to edit the attributes for a specific inventory item in the warehouse.

From the map screen, click on the “Manager” button on the right side of the screen and click on the “Modify Inventory” tab on top of the Manager interface.

Users can search the details of the item by following setup:

1. Input item name or item number.
2. Click the “Search” button.

3. If system can find the name or number in database, other information will fill in the text field automatically.

(If user input both name and number, the system will only use the name to search)

If the same item is not already in the database, the user will need to input all of the information for the product. If the system cannot find the item, none of the fields will update.

After the search, the user can modify any text field and click the “Confirm” button to write the new data to the database.

*Note:* This functionality is not complete. To test for now, use the following input:

Input "apple" or "banana" in the Name field, or

Input "1" or "2" in number text field

### ***Management System User Manual***

The core management system allows the user to view, add, update and delete both employees and products from the system.

From the map screen, click on the “Manager” button on the right side of the screen and click on the “Employee List” or “Product List” tabs on top of the Manager interface.

Choose Employee List in the top menu bar:

1. Add New Employee Info

Click “Add” to create a new employee in the database.

2. Edit Employee Info

Select an employee and click “Edit” to update their information and save it in the database.

3. Delete Employee

Select an employee and click “Delete” to remove them from the system.

*Note:* Delete functionality is not fully implemented, and the users will not be removed from the system.

Choose Product List in the top menu bar:

1. Add New Product Info

Click “Add” to create a new product in the database.

2. Edit Product Info

Select a product and click “Edit” to update their information and save it in the database.

3. Delete Product

Select a product and click “Delete” to remove them from the system.

*Note:* Delete functionality is not fully implemented, and the users will not be removed from the system.

g. Meeting Minutes

<b>Meetings</b>			
<b>Date</b>	<b>Purpose</b>	<b>Attendance</b>	<b>Summary</b>
9/16/2013	Decide on Project, Assign Work Units	All	Decided on Warehouse Management System
4pm-5pm			Decided on Everyones Tasks
			- Han - Management System
			- Guo - Shipper/Receiver
			- Sam - GUI
			- Spencer - Stock Picker
			- Matt - Database/Back end
			Decided to use GitHub for version control
9/24/2013	Decide on GUI for Toy Prototype	All	Decided on general GUI for toy prototype
5pm-5:45pm	Assign Tasks for Milestone 3		Decided who will do what for Milestone 3
			- Han - 1.2
			- Xingze Guo - 1.3
			- Spencer - 1.4, 1.7
			- Sam - 1.11
			- Matt - 1.1,1.5
			- Everyone - 1.6, 1.8
10/7/2013	Divide Tasks for Milestone 4	All	Read and divided tasks for Milestone 4
10:30am-11:20am			- Everyone: 1 Fully Dressed, 1 SSD, 2 OC, 1 Skeleton
			- Han: 3 Summaries
			- Matt: 2 Summaries, Setup up DB and Ant
			- Spencer: 1 Full Implementation
			- Xingze: Use case diagram
			- Sam: Domain model
10/22/2013	Status Update on Milestone 4	All	Reviewed status of everyone's tasks
4:50pm-5:20pm			Assigned each person 1 Summary Specification
10/25/2013	Clarify Remaining Tasks for Milestone 4	Xingze, Sam, Spencer, Matt	Discussed any remaining
10:30am-11:20am			Clarified ETA for completion
			Discussed GUI changes required for fully implemented use cases
			Clarified Milestone requirements
10/30/2013	Discuss UI, Goals for Friday	Sam, Spencer, Han, Matt	Discussed UI changes for all screens (add a global menu)

11:00am-11:20am			Discussed UI of Stock Picker
			Discussed Login UI Setup
11/5/2013	Discuss plan for Milestone 5	All	Assign and divide tasks for Milestone 5
5pm-5:45pm			- Heavy implementation milestone
			- Set deadlines at intervals between now and due date (sprints)
11/12/2013	Clarify Remaining Tasks for Milestone 5	All	Discussed any remaining
4:15pm-5:00pm			Clarified Class Diagram for completion
			Clarified Sequence Diagram for completion
			Clarified System Operations for completion
11/15/2013	Finalize the code for Milestone 5	Matt, Xingze, Han, Sam	Discussed some modification and updating in the code for Milestone 5
10:30am-11:20am			

#### h. Git Log

See the included file group02\_gitLog.txt.

### 5. Project Plan

#### Completed Tasks

9/16/2013	All	1.5	Meeting to decide on project and tasks
9/17/2013	Matt Triff	0.25	Set Up GitHub Repo, added everyone
9/18/2013	Rongli Han	1.5	Write management part for early project proposal
9/18/2013	Spencer Ondrusek	1.5	Design and write early proposal of function Stock Picker
9/18/2013	Xingze Guo	1.5	Design and write early proposal of function Shipper and Receiver
9/19/2013	Matt Triff	1.5	Combined and edited Milestone 1: Early Project Definition
9/24/2013	All	1.5	Discussed the Project and divided sections again
9/24/2013	Rongli Han	1.5	Design the basic GUI for management and reporting
9/24/2013	Spencer Ondrusek	1.5	Design the GUI for the stock picker
9/24/2013	Xingze Guo	1.5	Design the GUI for Shipper and Receiver
9/26/2013	Spencer Ondrusek	1.5	Milestone 3 part 1.4
9/26/2013	Spencer	1.5	Milestone 3 part 1.7



	Ondrusek		
9/26/2013	Xingze Guo	2	Milestone 3 part 1.3, 1.6 for Shipper and Receiver and 1.7
9/27/2013	Matt Triff	2	Milestone 3, parts 1.1, 1.5
9/27/2013	Rongli Han	2	Milestone 3 part 1.2(Business Case), 1.4and 1.8 for management and reporting
9/27/2013	Xianming Luo (Sam)	2	Design the GUI for the prototype
9/28/2013	Matt Triff	5	Prepared Milestone 3 document to be submitted
9/28/2013	Xianming Luo (Sam)	3	Implement the GUI toy prototype
9/30/2013	Matt Triff	2	Completing final edits and submission of Milestone 3 document
10/4/2013	All	1	Prepare for Presentation
10/7/2013	Matt Triff	0.5	Update Tasks List for Milestone 4
10/8/2013	Matt Triff	4	Created database schema, created all tables in database, completed 2 use case summaries, set up branches and GitHub repository
10/11/2013	Rongli Han	1	Completed Summary Use Case for order stock, login and manage employee, Milestone4
10/11/2013	Rongli Han	1.5	Completed Fully Dressed Use Case for manage employee, Milestone 4
10/11/2013	Rongli Han	1	Completed Sequence Diagrams of manage employee, Milestone 4
10/11/2013	Xingze Guo	1.5	Completed fully dressed use case of make shipment
10/11/2013	Xingze Guo	0.5	Completed Operation Contracts of Make Shipment
10/11/2013	Xingze Guo	1	Completed Use Case Diagram
10/11/2013	Xingze Guo	0.5	Completed Sequence Diagrams of Make Shipment
10/16/2013	Matt Triff	1.25	Completed fully dressed use case, tallied status of milestone4 tasks and emailed team
10/17/2013	Rongli Han	0.5	Work on Operation Contracts for checkEmployeeStatus, Milestone 4
10/18/2013	Matt Triff	3	Completed skeleton for database package
10/18/2013	Spencer Ondrusek	2	completed fully dressed use case, operations and SSD
10/20/2013	Spencer Ondrusek	3	worked on implementing the print function and skeleton, communication on revising GUI
10/21/2013	All	0.5	Meeting to discuss status of Milestone 4
10/24/2013	Rongli Han	3	Created Skeleton for manage employee and manage product, Milestone 4
10/24/2013	Xingze Guo	2	Design skeleton for Shipper
10/24/2013	Xingze Guo	1	Design skeleton for Receiver
10/26/2013	Rongli Han	0.5	Work on Supplementary Specification Usability, Milestone 4
10/26/2013	Xingze Guo	3	Working on code of use case: Make shipment
10/26/2013	Rongli Han	0.5	Created Skeleton for Report, Milestone 4
10/26/2013	Spencer Ondrusek	0.5	Supplementary case - Supportability
10/26/2013	Xianming Luo (Sam)	2	Worked on Domain Model Diagram
10/26/2013	Xianming Luo (Sam)	0.5	Worked on Supplementary Specification for Interfaces
10/26/2013	Xingze Guo	0.5	Completed Supplementary Specification -- Reliability
10/26/2013	Xingze Guo	2.5	Completed fully implement of make shipment, modify code of GUI, run the code with GUI successfully
10/26/2013	Matt Triff	3	Editing draft of Milestone 4 Report
10/27/2013	Xianming Luo (Sam)	0.5	Modified UI for stock picker and shipper

11/2/2013	Xianming Luo (Sam)	7	Implement the Map System (GUI)
11/3/2013	Xianming Luo (Sam)	3	Implement the Map System (GUI)
11/5/2013	Xingze Guo	4	Implement Receiver System
11/6/2013	Matt Triff	1.5	Implemented MySQL database connection
11/7/2013	Spencer Ondrusek	3.5	Attempted to fix corrupt github
11/7/2013	Rongli Han	1	Create productInfo (GUI) and employeeInfo (GUI)
11/8/2013	Xingze Guo	3	Implement Shipping System
11/12/2013	Matt Triff	3	Updated database functions
11/12/2013	Rongli Han	0.5	Completed Sequence Diagram
11/13/2013	Xianming Luo (Sam)	5	Modified UI
11/13/2013	Xingze Guo	4.5	Implemented Modify Inventory in Manager System
11/13/2013	Xingze Guo	1	Completed Sequence Diagram
11/13/2013	Xingze Guo	0.5	Completed System Operations
11/13/2013	Matt Triff	1.5	Implemented Neo4j database connection
11/13/2013	Spencer Ondrusek	4	Implemented table printing and loading a specific table from a combo box
11/14/2013	Matt Triff	2	Documentation and Project Management
11/14/2013	Xingze Guo	3	Implemented Order Stock in Manager System
11/14/2013	Xingze Guo	1	Completed User Manual
11/14/2013	Rongli Han	1	User manual, System operations
11/14/2013	Rongli Han	3	Modify management system
11/14/2013	Matt Triff	4	Graph Database Implementation
11/15/2013	Xingze Guo	3	Implemented Product List in Manager System
11/15/2013	Xingze Guo	2.5	Implemented Employee List in Manager System
11/15/2013	Xianming Luo (Sam)	3	Modify User Interface
11/15/2013	Spencer Ondrusek	5.5	User manual, operations, and class diagram
11/15/2013	Matt Triff	8	Database Implementation, Report Preparation

### Upcoming Tasks

Who	Task	Estimated Hours	Due Date
All	JUnit testing integration	10/each	11/30/2013
Matt	Complete MySQL Database integration for all functions	3	11/20/2013
Sam	Complete Locate Stock	5	11/23/2013
Han	Complete Deletion of Products and Employees	5	11/23/2013
Xingze	Refine Receiver and	6	11/22/2013

	Shipper functionality		
--	-----------------------	--	--