HI-TEC ELECTRONIC SUPPLY LTD

Warehouse Management System

Project Proposal

Prepared for:
Nelson Eng, Instructor
CSIS 3275 002 – Software Engineering

Development Team
Matthew Lai
Manjot Sangha
Manveer Sidhu
Curtis Windsor
Sam Zhou

SUMMARY & SCOPE

This document serves as a software requirements specification for HI-TEC Electronic Supply Ltd., a fictional business-to-business supplier that operates in the Lower Mainland; dealing primarily in individual computer components as well as complete pre-built machines. The software system in question is a management system for the company's warehouse; tracking current stock, incoming orders, and outgoing shipments. This system is comprised primarily of: 1) A mySQL Database, 2) A website for clients, and 3) A program for employees.

The ultimate **objective** of this project is to create a comprehensive system for managing the movement of products in the warehouse.

Deliverables of this project will be:

1. A database complete with data
2. A fully-functional website with read-only capability from the database
3. A fully-functional application written in C# with different GUIs for specific departments
4. A comprehensive technical manual for the end user detailing the use of each module of the system
5. A detailed documentation log for programmers designed to assist in the system's upkeep

Technical requirements of this system include:

1. A server running a LAMP stack
2. All machines running the application must have:
    a. Windows 7 and above
    b. 1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor
    c. 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit)
    d. 16 GB available hard disk space (32-bit) or 20 GB (64-bit)
    e. DirectX 9 graphics device with WDDM 1.0 or higher driver
    f. Internet Access

Limitations of this software system include:

1. The system assumes there are already separate systems in place for Sales, Accounting, and so forth to handle issues such as payment processing.
2. Due to the warehouse focus of the system, the components that deal with orders and shipments exists solely to track inventory inputs and outputs.
3. Inventory inputs and outputs are entered into the system via the application when the appropriate waybill is received.

The development team for this system project consists of the following people with the following primary areas of responsibility. Despite this, it is projected that team members will assist each other in differing areas as required.

Project Management: Matthew Lai
Website Design: Manjot Sangha and Manveer Sidhu
Software Programming: Sam Zhou
Server Administration: Curtis Windsor

COMPONENTS & TECHNOLOGIES

Component A – MySQL Database

This will store the information needed to track our products in the warehouse. An ERD is available in the appendix of this document. (Fig. I)

CA - Technologies Involved:

- MySQL
- phpMyAdmin

CA - Development Tools:

- AWS EC2
    - Linux – Ubuntu 15.04
    - Apache
    - Cloud software specifications:
        - 1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor
        - 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit)
        - 16 GB available hard disk space (32-bit) or 20 GB (64-bit)
        - DirectX 9 graphics device with WDDM 1.0 or higher driver
- MySQL Workbench 6.3 CE
- Mockaroo

Component B – Client Website

This displays information about our warehouse inventory for use by clients. This is necessary for our system as it assumes clients order products by phone only.

CB - Features and Functions:

- The user can view our stock, with the following product information displayed:
    - SKU
    - Name
    - Price
    - Manufacturer
    - Category
    - Image
- The user will be able to filter the displayed products by:
    - Product Category (Fig. II)
    - Product Manufacturer
- The user will be able to search for a specific product by:
    - Product SKU
    - Product Name
- All the above display results can be ordered by price ascending/descending
- A contact form for clients who want more information will be provided

CB - Technologies Involved:

- HTML
- JavaScript
- AJAX
- PHP
- CSS
- Bootstrap

CB - Development Tools:

- Adobe Dreamweaver
- Codelobster
- FileZilla
- AWS EC2
    - Linux – Ubuntu 15.04
    - Apache
    - Cloud software specifications:
        - 1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor
        - 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit)
        - 16 GB available hard disk space (32-bit) or 20 GB (64-bit)
        - DirectX 9 graphics device with WDDM 1.0 or higher driver

Component C – Staff Application

This program allows warehouse staff to manipulate the information stored in the database as products are moved around. For security reasons, different functions of the program are locked to an employee's permission group.

CC - Features and Functions:

- A **login screen** (Fig. III) that will prompt the user for their credentials. Depending on the permissions the user has access to; the relevant UI will then appear.
    - The current permission groups are:
        - Warehouse Employee
        - Receiving Clerk
        - Shipping Employee
        - Operations Supervisor
        - Manager
        - Administrator
    - This is the UI that appears when the program is first run. It will first attempt a connection to the database. If that connection fails, the user will not be able to access any functions and will be prompted to check with their network administrator.
    - User inputs are checked against the EMPLOYEE and EMPLOYEE_CREDS tables in the database.
- An **inventory checker UI** (Fig. IV) with read-only capability to the DB. For regular warehouse employees, this is their primary screen that appears after they log in.
    - For lower-tier users (any non-managerial or supervising staff), the information displayed about the inventory will be limited to basic information. Higher-tier users will have full access.
    - This UI mainly draws data from the PRODUCT table, along with that tables associated with it through its foreign keys.
- A **receiving UI** (Fig. V) with write access to the DB. This interface is for employees who deal with incoming products to the warehouse. As such, it has the functionality to update and add new records to the DB.
    - This UI has the same data limitation scheme mentioned previously. Due to this UI having the ability to write to the DB, the function to add a new product and its associated supplier and manufacturer will be restricted to higher-tier employees.
    - This UI mainly draws data from the ORDER and ORDER_PRODUCT tables, with more detailed information about the orders being drawn from the PRODUCT and SUPPLIER tables.
- A **shipping UI** (Fig. VI) with write access to the DB. This interface is for employees who deal with outgoing products to clients. As such, it will have the functionality to update records to the DB.
    - Employees can create a new shipment whose information, including the associated products, are stored to the DB. Until the shipment is actually shipped however, the products are not yet deducted from the inventory stock.

    - Shipping employees can then go through the list of shipments that have yet to ship and mark them as 'Shipped', and only then are the products removed from the inventory. If the inventory does not contain enough of a product the user will be notified.
    - This UI mainly draws its data from the SHIPMENT and SHIPMENT_PRODUCT tables. Supporting details for the shipment data is drawn from the LOCATION table and its associations.
    - This UI is data limited with the same scheme as the other UIs.
- **A warehouse history log** (Fig. VII), which will show all the shipment and order information that has occurred.
    - This UI mainly draws its data from both the SHIPMENT and ORDER tables and their associations. However, this UI differs from the dedicated Receiving and Shipping UIs in that it serves only as a log, and thus is read-only.
    - This UI is data limited with the same scheme as the other UIs. Higher-tier users will be able to access detailed information about the clients and suppliers associated with each shipment and order.

CC – Technologies Involved:

- C#

CC – Development Tools:

- Microsoft Visual Studio 2015
- Microsoft Visual Studio 2012 Express

<u>Non-Component Tools</u>

Additionally, the following services and tools will be used in order to provide version control and documentation throughout the entire development process, regardless of module.

- Git through Github
- SourceTree

TIMELINE

WEEK	TASK(S)
8	▪ Application GUI design ▪ Application login function ▪ Server set-up ▪ DB construction ▪ Website page layouts
9	▪ Application UI construction ▪ Website display functions ▪ DB construction ▪ DB data tweaking
10	▪ Application UI functions ▪ Application UI linkage ▪ Website display functions ▪ Website contact form
11	▪ Application testing ▪ Website CSS ▪ Website function testing
12	▪ Final systems testing ▪ Technical manual and documentation check ▪ Presentation

APPENDIX

FIGURE I: Database ERD

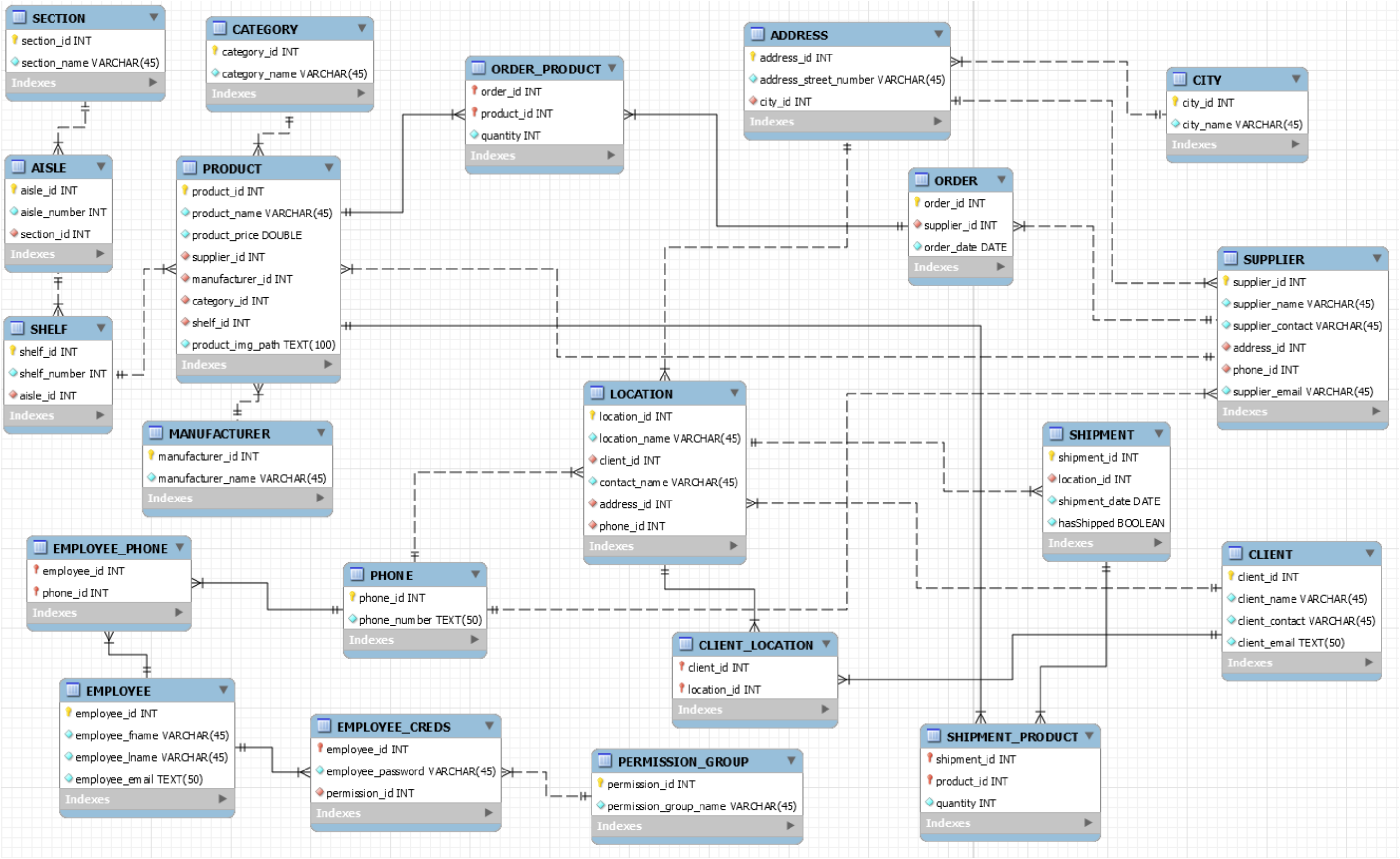
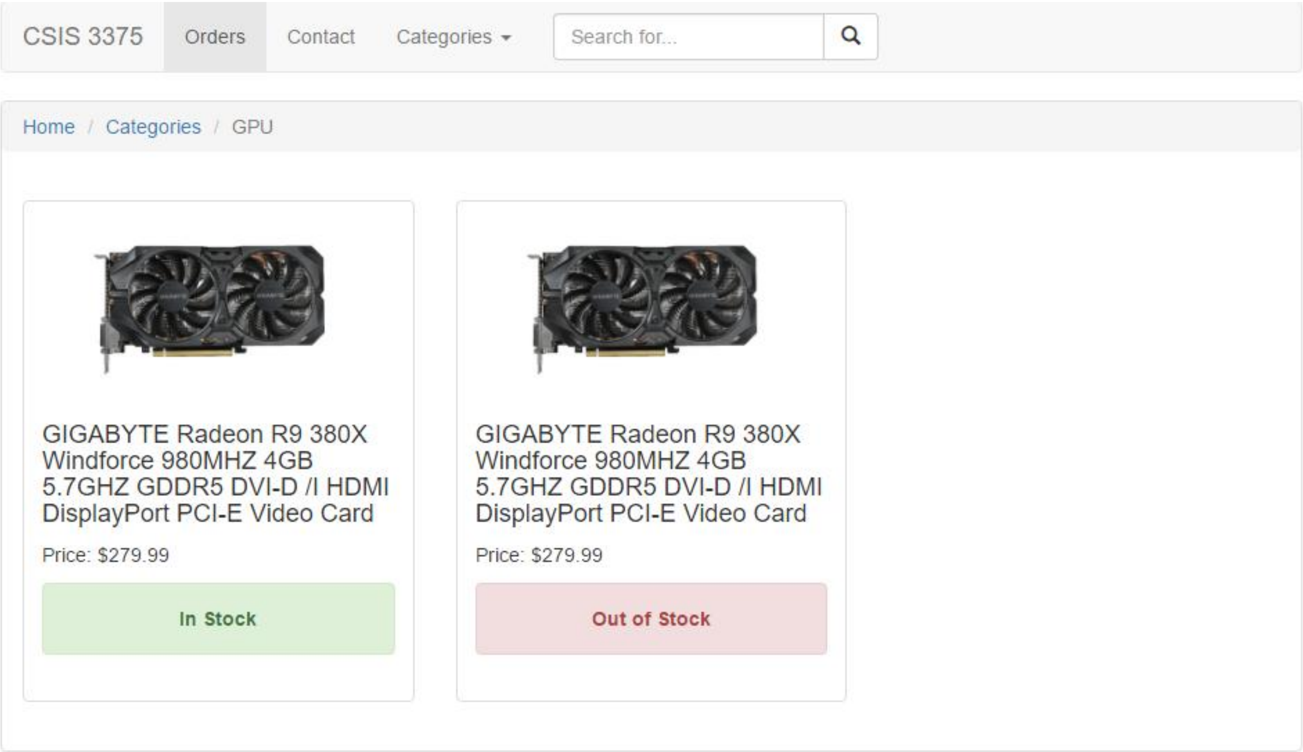


FIGURE II: Website Display Filtering Mock-up



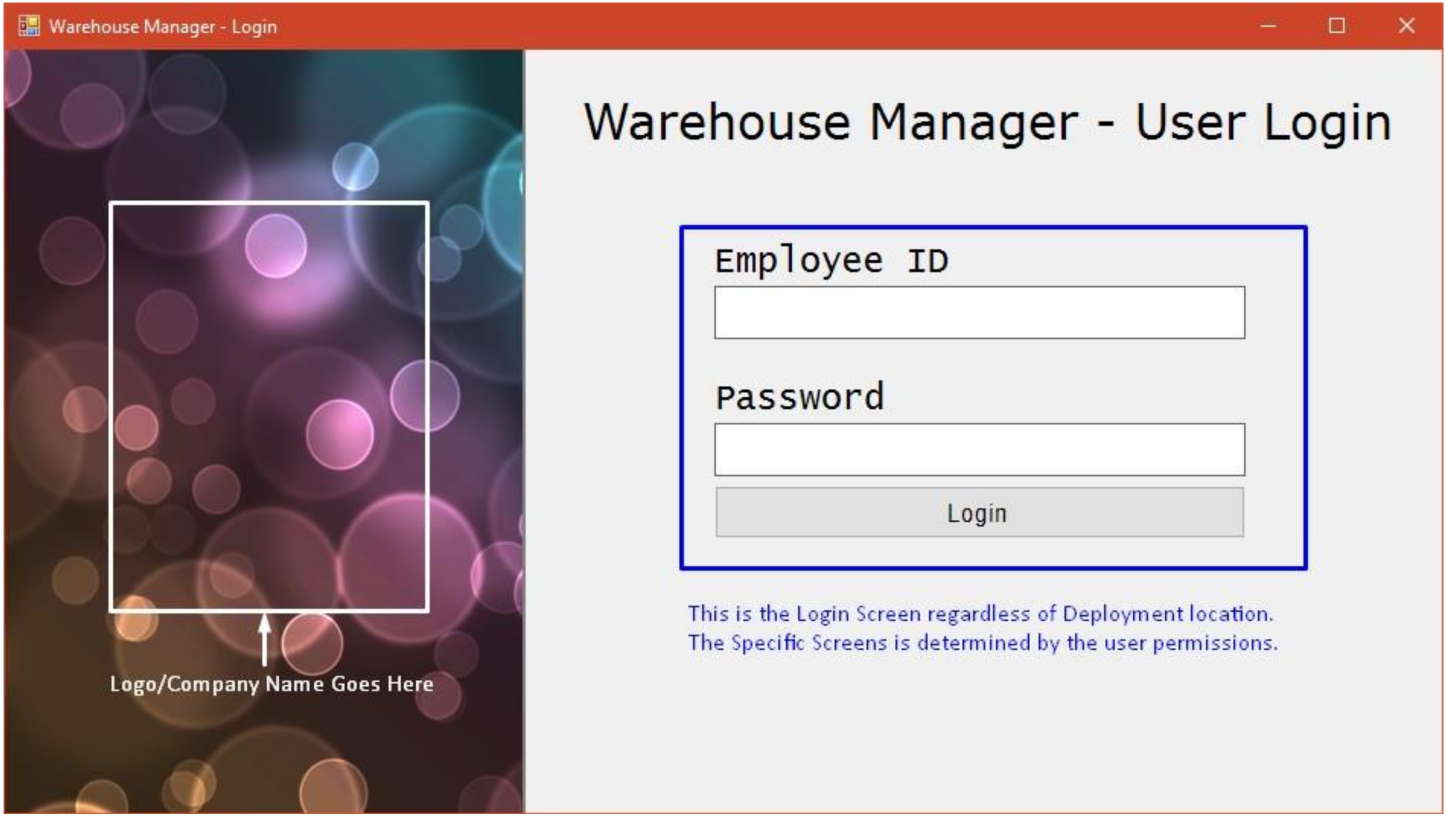
GIGABYTE Radeon R9 380X Windforce 980MHZ 4GB 5.7GHZ GDDR5 DVI-D /I HDMI DisplayPort PCI-E Video Card

Price: $279.99

GIGABYTE Radeon R9 380X Windforce 980MHZ 4GB 5.7GHZ GDDR5 DVI-D /I HDMI DisplayPort PCI-E Video Card

Price: $279.99

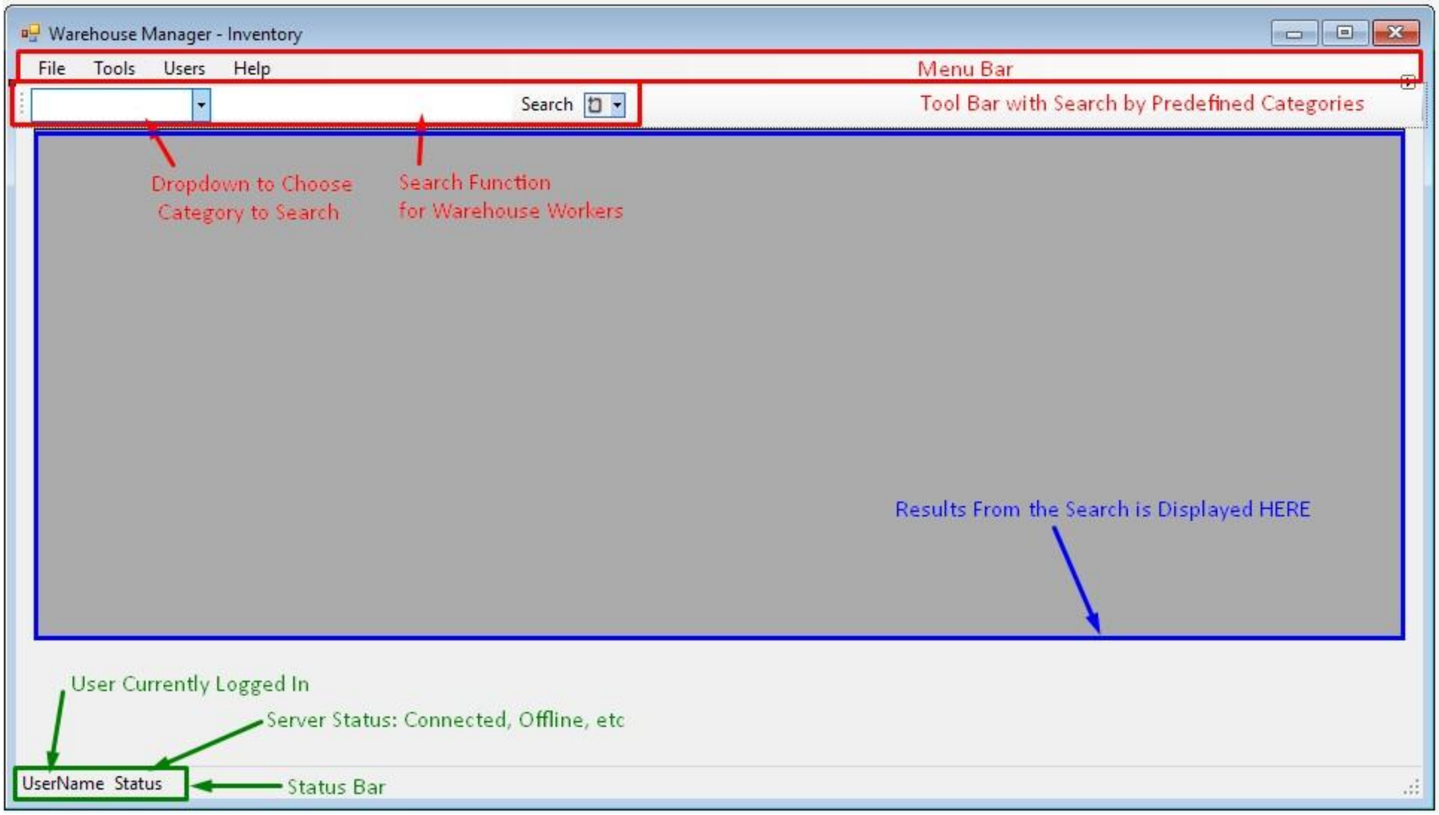
FIGURE III: Application Login UI



Logo/Company Name Goes Here

This is the Login Screen regardless of Deployment location.
The Specific Screens is determined by the user permissions.

FIGURE IV: Application Inventory UI



Menu Bar

Tool Bar with Search by Predefined Categories

Dropdown to Choose Category to Search

Search Function for Warehouse Workers

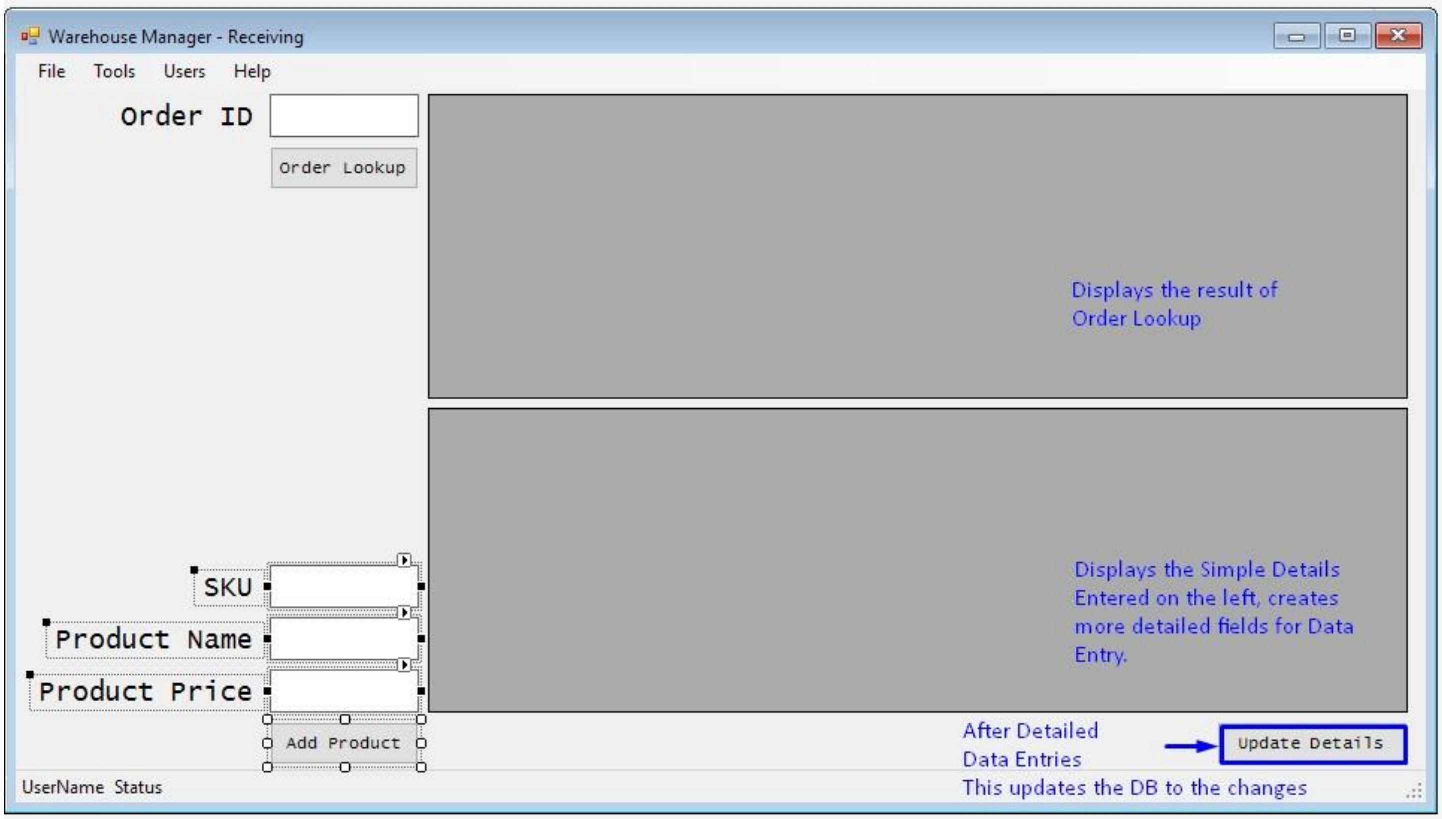
Results From the Search is Displayed HERE

User Currently Logged In

Server Status: Connected, Offline, etc

Status Bar

FIGURE V: Application Receiving UI



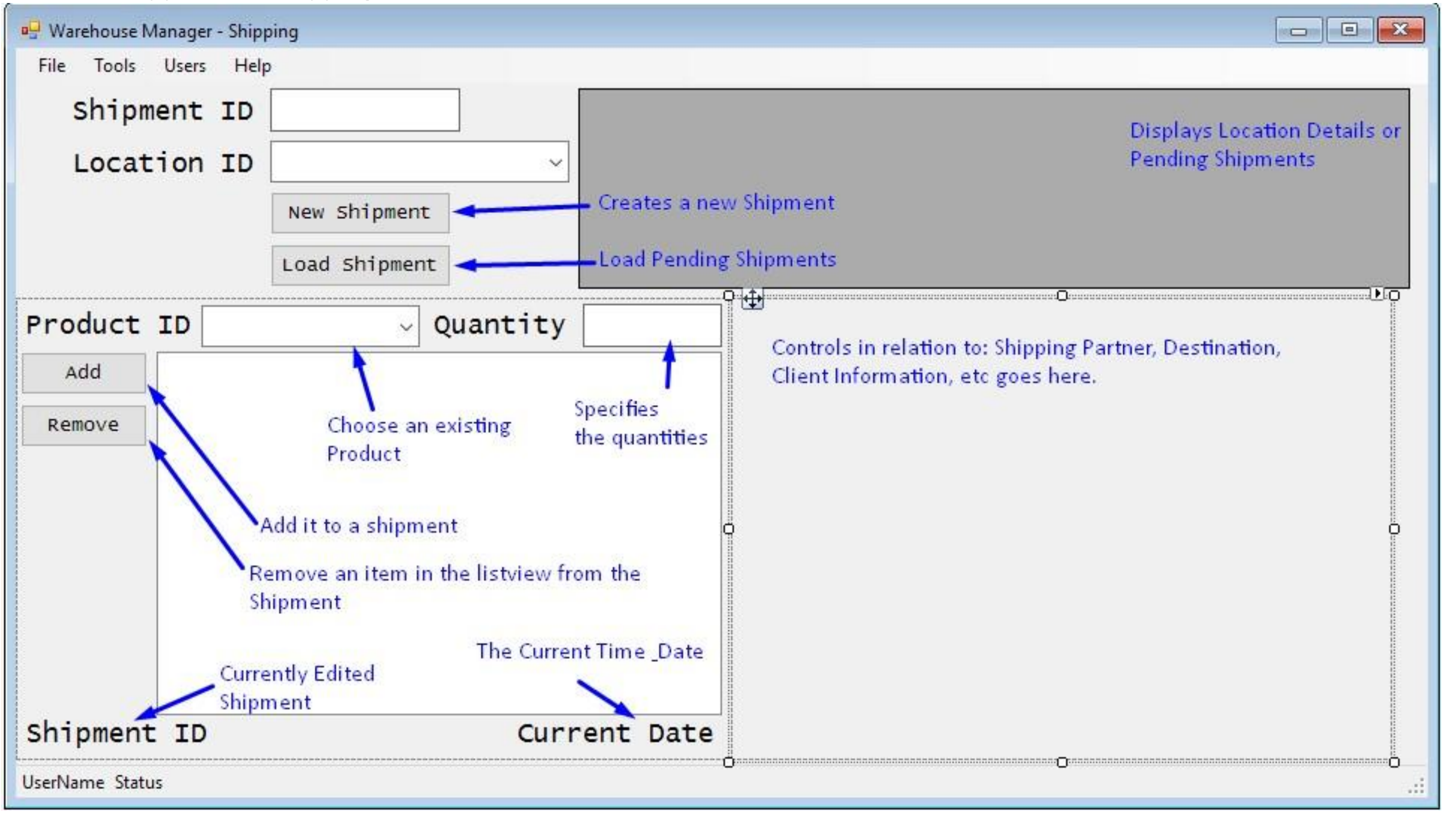
Displays the result of Order Lookup

Displays the Simple Details Entered on the left, creates more detailed fields for Data Entry.

After Detailed Data Entries

This updates the DB to the changes

FIGURE VI: Application Shipping UI



Displays Location Details or Pending Shipments

Creates a new Shipment

Load Pending Shipments

Controls in relation to: Shipping Partner, Destination, Client Information, etc goes here.

Choose an existing Product

Specifies the quantities

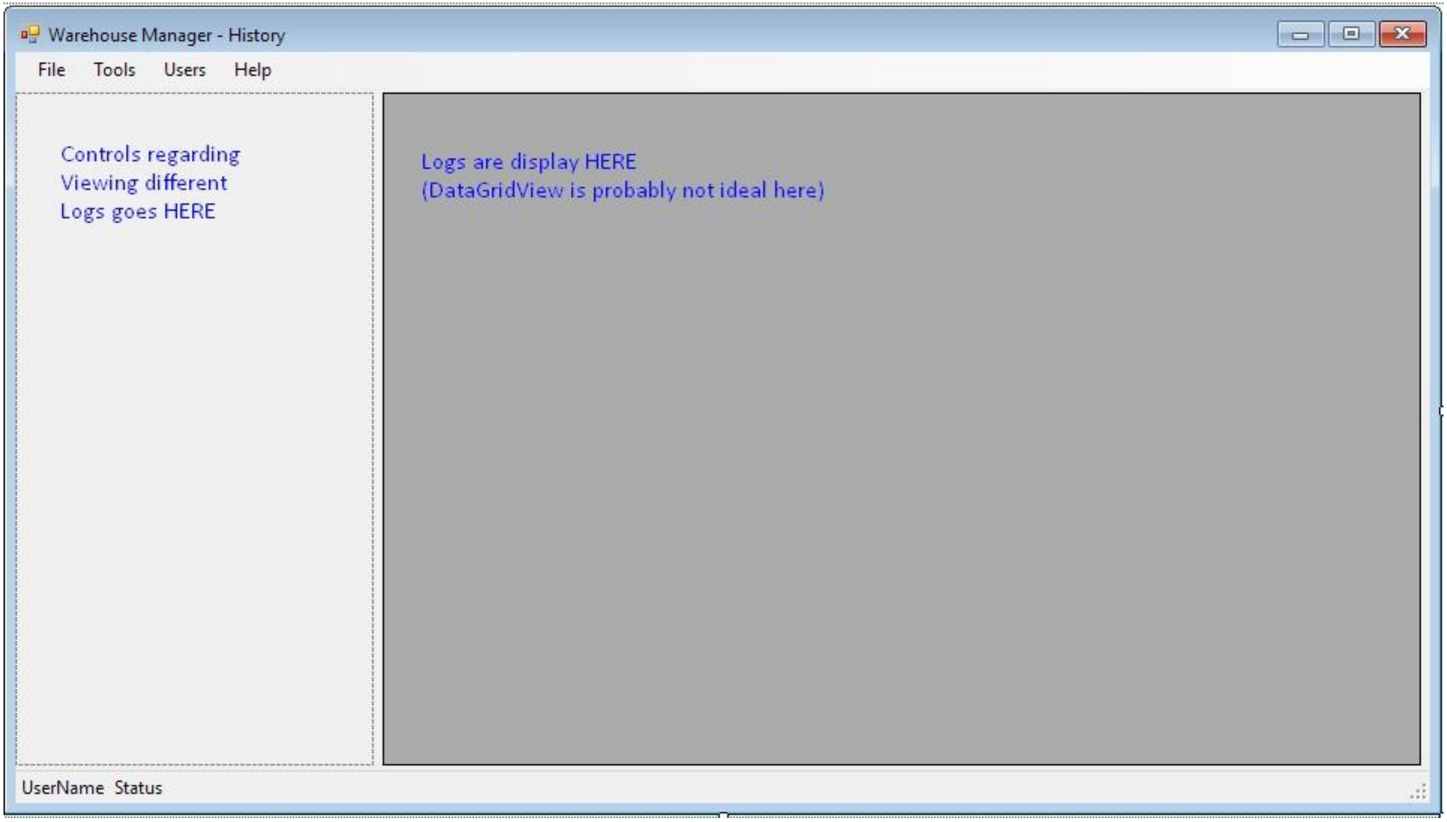
Add it to a shipment

Remove an item in the listview from the Shipment

The Current Time _Date

Currently Edited Shipment

FIGURE VII: Application Transaction UI



Controls regarding
Viewing different
Logs goes HERE

Logs are display HERE
(DataGridView is probably not ideal here)

UserName Status