CSCE 361

Software Engineering

Software Design Specification Document

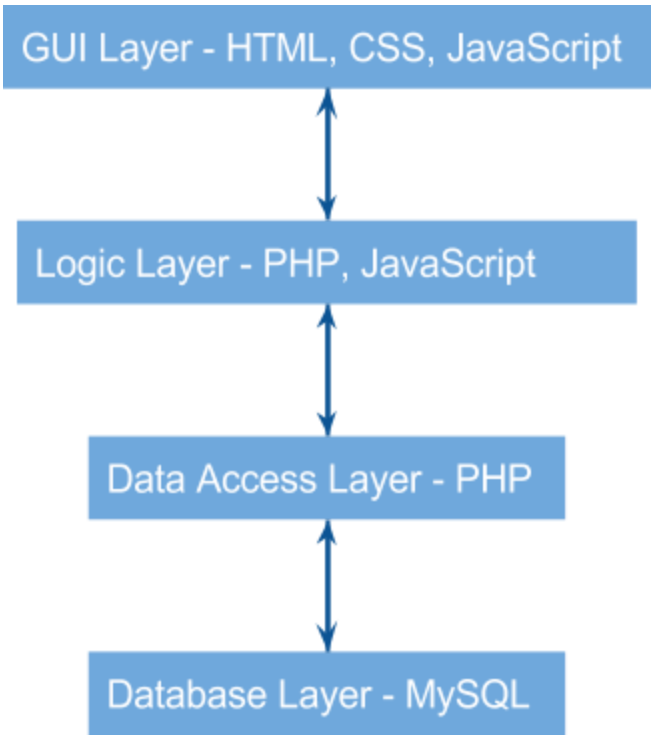
Group 9 - *Keenan Humm, Abdulrahman Asiri, Kevin Henry*

1. Introduction

The purpose of this design document is to describe, at a high level, the project architecture for the Nventory stockroom management system. This document will include entity relation diagrams showing the relationship between different parts of the system, as well as the relationships between tables in the database. The audience of this document is software engineers and system architects who will be implementing and maintaining the described project.

2. Architecture

2.1 Introduction



(Figure 1)

The high level architectural design of the system will be a layered model. The lowest level of this model will be a MySQL database to store data, such as inventory items, users, checkouts, and cost numbers. The data access layer is responsible for creating and maintaining a connection to the database layer. For this, PHP is utilized. The code that is necessary for the basic site functions resides in the logic layer. The GUI layer consists of the code needed to create the interface that the user sees.

2.2 Modules

2.2.1 Database layer

The database layer is responsible for holding all data for the system and defining the relationships between these data. The system utilizes MySQL database, which can be incorporated into the project using PHP in the data access layer.

2.2.2 Data Access Layer

The data access layer should leverage PHP libraries that allow us to connect our GUI and Logic layers to the supporting database. The users will make edits, checkouts, and updates within the user interface that must be reflected in the database. This is why a data access layer is necessary.

2.2.3 Logic Layer

Another module is needed for handling validation and application processes. This layer shall be responsible for validating login credentials, stock checkout, and functionality in user tools.

2.2.4 GUI Layer

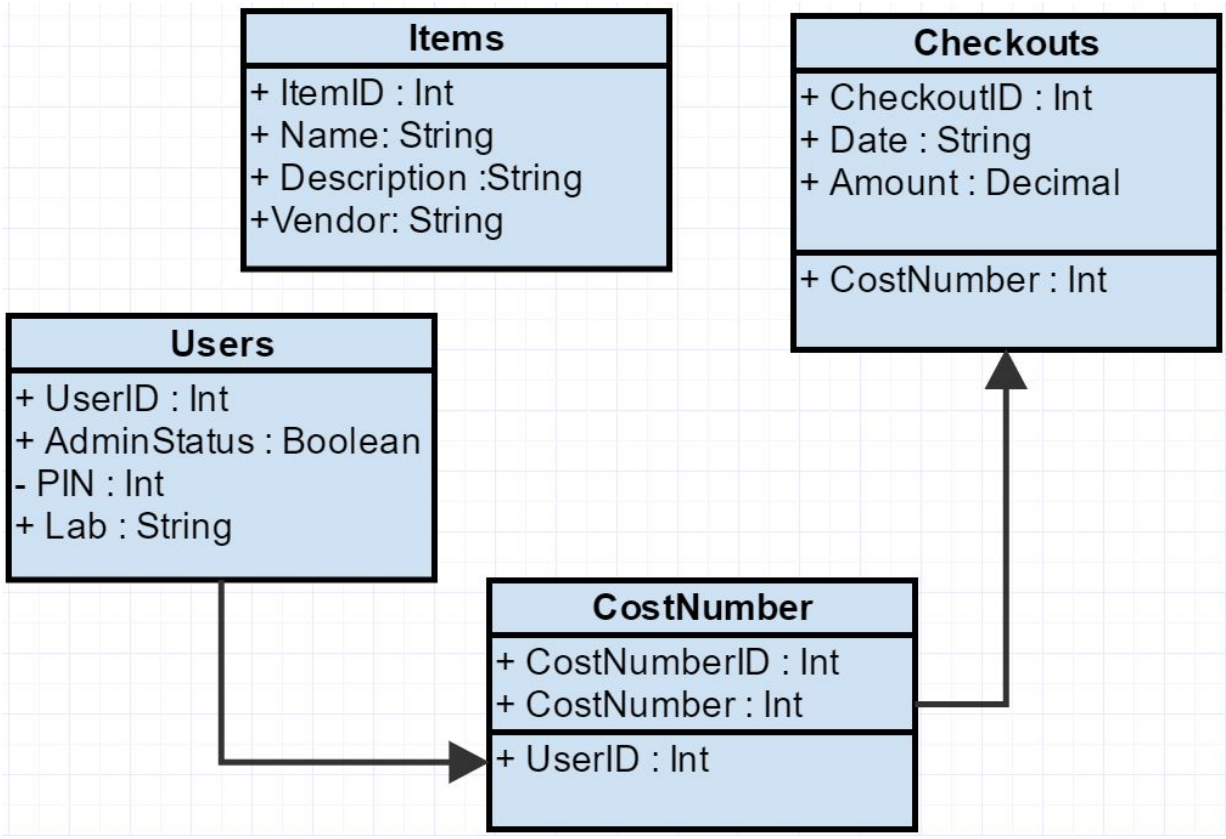
The purpose of the GUI layer is to render what the users will see when launching our application. A combination of HTML, CSS, and Javascript (including the JQuery library) will be used in this layer to generate user-friendly web pages allowing for seamless interaction. Both administrative and lab users should view similar pages with some differences in functionality. User actions will be sent back down the layers and proper changes should take place in the database.

3. Class Diagrams

3.1 Data Table Classes

The system shall use a MySQL database to hold all the persistent data necessary. This includes information on users, cost numbers, inventory items, and checkouts. The figure below shows each table in the database with its respective columns as well as the relationships between the different tables

3.1.1 Schema



(Figure 2)

3.1.2 Schema Information

The data in the database shall be organized in the following tables and columns

Users: Holds the data for the users of the system. This table will include data such as the userID, admin status, the PIN they login with, and name of the lab they belong to.

Items: This table holds the data for all the items held within the university. This includes the items name, ID, a short description, the quantity in stock , and the vendor that provides the item.

CostNumber: This table will contain all cost numbers associated to users. It will contain the ID for the cost number, the cost number itself, and the user (lab) it is associated with.

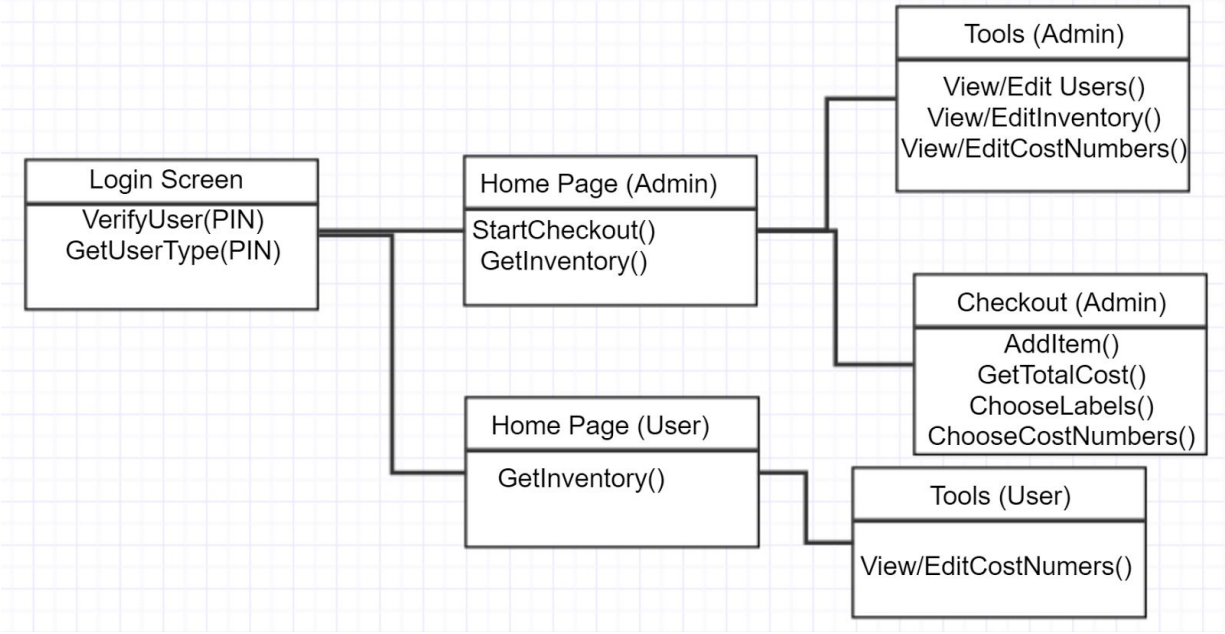
Checkouts: The checkouts table shall hold a record of every checkout be a lab and details associated with it. Information needed is simply the checkout total, cost number assigned, and date of checkout.

3.2 Class Information

Classes will be implemented in PHP and JavaScript. Class structure, for the most part, will reflect the different tables in the database. For example, a Item class will be implemented with attributes for ItemID, ItemName, Description, etc. These classes, residing in the logic layer, can be utilized to populate the web page with the proper information.

3.3 GUI Layer

The first layer introduced to the user upon entrance to the web application is the login page. The login page contains an input field for the user PIN and a submit button. Upon entering a valid user PIN, the user will be redirected to the appropriate home page. Home pages are nearly identical for both users and admins. The admin home page will have the option to start the checkout process. Both user and admin home pages will display a list of available items in the inventory. Both should have a button that will redirect them to a tools page. The tools page for both admin and general users will be similar as well. The admin tools page contains the ability to view and edit users, inventory, and cost numbers. The lab user tools section will contain only the ability to view and edit the cost numbers associated with their lab. As well as a tools page, the admin will have the ability to start a checkout process. Initializing the checkout process will bring the user to another page where they will be able to add items to a list generate total cost and assign cost numbers to the transaction.



(Figure 3)