WEB BASED CUSTOMER ORDERS AND ACCOUNTS MANAGEMENT SYSTEM

FOR

CAPITAL HARDWARE

A W M AZAD

BIT Registration Number - R227020

BIT Index Number - 2270201

Name of the supervisor – Mr. Suren Sarathkumara.

MARCH 2023

BIT

UCSC

This dissertation is submitted in partial fulfillment of the requirement of the Degree of Bachelor of Information Technology (External) of the University of Colombo School of Computing

Abstract

Capital Hardware is a well-established hardware store located in the Colombo wholesale hardware market with a large customer base throughout the island. Currently, customers place their orders via telephone, and the store staff prepares the invoice and delivers the ordered items through transport service providers or Capital Hardware's delivery vehicles. Payments are made either by bank cheques or cash, depending on the customer's payment status.

However, the manual system used for conducting business has resulted in several difficulties for Capital Hardware, including challenges in handling sales, payment collections, and maintaining ledgers, as well as unnecessary paper and labor costs. Moreover, management and staff have limited opportunities for face-to-face business activities with customers.

To address these issues, the management has decided to develop an online customer orders and payments management system. Currently, sales and payments are manually maintained, including customers' cheques. Therefore, automating the system will provide efficient customer orders and payment management, reducing the hassle of manual ordering systems for customers.

For my final year BIT project, I have chosen to develop this system using a client-server architecture and based on object-oriented principles. The development methodology and design languages used are Rational Unified Process (RUP) and Unified Modeling Language (UML), respectively. The development tools used include Typescript (Angular 10 framework), HTML, CSS, Node.js (server-side), and MySQL (database support).

The developed system features two views: one for customers and another for administration, both of which are single-page applications developed using the Angular framework.

Table of Contents

Chapter 01: Introduction

1.1 Capital hardware background

Capital Hardware is a well-known store in the wholesale hardware market located in Colombo. They have been successfully running their business for the last 10 years. Capital Hardware specializes in selling hardware items in large quantities to the local market. Due to the quality of their products and reliable customer service, they have a potentially large market share in the western province.

Capital Hardware has a large customer base throughout the island, and they currently manage their sales, payment transactions, and account transactions manually using files and ledgers.

1.1.2 Sales process of Capital hardware

The customers of Capital Hardware are mainly delivery dealers or retail hardware stores who purchase merchandise items from them. Capital Hardware offers credit sales to their customers and collects payments from them periodically. Customers from across the island place their orders through phone or fax with Capital Hardware. Once an order is received, the staff prepares the items and sends them through transport services to the customer's destination. Payments from customers are collected by Capital Hardware's sales representatives or customers send payments through postal services in the form of dated cheques. The invoice is then prepared and sent along with the ordered items to the customer.

1.2 Motivation for the project

Although Capital Hardware has a large customer base throughout the island, they currently lack an effective system for managing their sales and payment transactions. The business currently relies on a manual file handling system, which is inefficient and time-consuming. Other competing hardware stores have implemented computer-based solutions to manage their transactions and stay competitive. Without such a system, Capital Hardware may struggle to survive in the modern marketplace and gain a

competitive advantage. The proposed system aims to address these issues and provide an easier and more efficient way of handling transaction information in their daily business processes, while also reducing conflicts and supporting smoother operations. Furthermore, the proposed system can generate important management reports to assist in making informed decisions.

1.3 Project Scope

The scope of this project includes developing a computer-based system for Capital Hardware that can efficiently manage their sales and payment transactions, as well as customer orders. The system should be able to handle customer orders placed by phone, fax, or online and process payments made by bank cheques or cash. The project aims to reduce the inefficiencies and time-consuming nature of the current manual system by introducing automation and streamlining the business processes. The proposed system should be user-friendly and provide management reports to help support decision-making. The project scope is limited by the allocated time and resources, as well as the specific requirements and constraints identified by the client.

1.4 Project Objective

Several objectives of the proposed system are listed below.

- **Manage customers**

The system provides managerial staff able to add, delete, and modify customers.
Extend or reduce their credit limits. Create username password for customers to access the Web Based Customer Orders and Payments management System (**WBCOAMS**).

- **Manage customer orders**

System provides easy access to the Capital Hardware for registered customers to prepare their orders online and send it through the proposed **WBCOAMS** system and maintain the record of ordered items and add, modify and view their orders and its status.

- **Manage sales**

The system should be able to view their customer orders, invoices and handle their sales transactions. Furthermore the system should be able to maintain customer's payments and returned items records by add, modify and view.

- **Payments**

System should be able to maintain the customer's payments and settlements up to date by adding, modifying and deleting. Therefore customers are able to view their payments which are handed over to Capital Hardware sales representatives or sent through the postal service, by login into **WBCOAMS** system.

- **Cheques**

System should maintain the records of customer cheques details therefore the system provides easy to manage the customers cheques including adding new cheques, edit cheques and keep aware of bank account number of account closed cheques to restrict entering the same account cheques in future payment transactions.

- **Accounts**

System should be able to maintain the account transactions of customers similar to the customer ledger by adding transactions of their sales, payments, returned cheques and returned items which are the most common transactions added to customer ledger by Capital hardware. Therefore the registered customers can easily access the **WBCOAMS** system and compare their accounts with Capital hardware account transactions.

- **Generating relevant reports.**

System should provide the necessary reports to the management to make decisions and run the business successfully such as monthly sales reports, customer's payments reports, etc.

Up to date and accurate information should be available to create various reports within a short time period to get better management decisions.

Chapter 02: Analysis

The most important task in software development is gathering and analyzing requirements. In this chapter, the author focuses on the process of requirements gathering and analysis for the developed software system. Clients may have an idea of what they want, but they may not know exactly what the software should do. If the analysis of the system fails, the entire system may fail because it will not meet the client's expectations. Therefore, a significant amount of time is spent on requirements analysis to gain a better understanding of what the client wants from the system.

2.1 Existing system

Capital Hardware currently does not have an automated system to monitor their customer's business transactions. Customers place their orders through phone or fax, and the staff manually record the required items in a note. The company still uses the traditional manual file handling approach for their business processes, including manual bookkeeping to maintain their customer's records and business transactions.

At the end of each business day, the accounts staff add a new day book entry into the day book ledger and then transfer each transaction of the day book into relevant ledger accounts such as customer ledger, sales ledger, payments ledger, and general ledger using the double-entry accounting method. They also record each check given by customers for their invoice payments in a checkbook ledger. The managerial staff often calculates customer balances whenever it is needed by looking at customer ledger entries.

When a check is returned by the bank without being credited to the Capital Hardware bank account, the manager needs to manually search through customer check records in the checkbook for identification of the owner of the check and then notify the customer by telephone.

2.1.1 Problems of existing system

- Difficult to find the customer who owns the cheque when it is returned, sometimes the entire cheque book to be searched for a single entry.
- Some sales returned items may not be credited to the customer accounts when the customer returns their goods in cases of high price or quantities or physical damages.
- All the data relevant to the business are recorded in physical files over them and maintaining physical files ultimately leads to unnecessary paper and labour cost.
- Invoice and customer details are kept in physical files, this results in unnecessary time delays when finding a record for any verification processes of invoice or customer details.
- Customers cannot choose items in various brands because of no real time item details with graphics whenever they place an order.
- Capital Hardware faces difficulties while promoting their new products and items through the telephone to each customer when a new item comes to the stock. This may lead to unnecessary stock of unsold items.

2.1.2 Need for an online system

- Get Online Sales Orders from customers
- Easy Marketing new items
- Real time items information
- Generate Report as they needed
- Real time customer's and their accounts transactions information
- For a quality of customer service

2.2 Requirement gathering

Requirements gathering is one of the tough tasks in software system development, we have to arrange frequent meetings to identify exactly what the client wants. The author

completed the requirement gathering stage by using the following techniques to get the maximum output from clients about the system they needed.

- **Interviews.**

"This method is used to collect the information from groups or individuals. Analyst selects the people who are related with the system for the interview. In this method the analyst sits face to face with the people and records their responses. The interviewer must plan in advance the type of questions he/ she is going to ask and should be ready to answer any type of question. He should also choose a suitable place and time which will be comfortable for the respondent. " [1]

- **Review of documentations.**

"Document analysis is a technique of gathering requirements in which existing system related to current system is reviewed for collecting information regarding current system. Analyst should have to check different sources of documents for analysis process." [2]

- **Observation.**

"This is a skill which the analysts have to develop. The analysts have to identify the right information and choose the right person and look at the right place to achieve his objective. He should have a clear vision of how each departments work and work flow between them and for this he should be a good observer. " [1]

Requirements gathering are a very important stage in the software development life cycle. When a clear requirement is done, the client needed system can be clearly identified. It identified a number of problems of the existing system through the documents reviewing and interviews, in the stage of requirements gathering.

For the observation method author spend considerable amount of time to identify the day to day business process of the Capital hardware to get a clear understand of their business, furthermore author identified and understand how the capital hardware receive orders from customers, prepare the invoices and how they are dispatched the customer ordered items to

their destinations and furthermore how customer payment settlements are entered into the transaction records whenever a payment received.
At the requirements gathering stage a great guidance and support was given to the author from the Capital hardware staff and management.

2.3 Requirements Classifications

Normally at the system analysis stage, software system requirements are classified into two distinct groups, namely functional requirements and non-functional requirements and the requirements analysis involves frequent communication with system users to determine specific feature expectations, resolution of conflict or ambiguity in the gathered requirements as demanded by the various role of users, avoidance of feature creep and documentation of all aspects of the project development process from start to finish.

2.3.1 Functional requirements

The functional requirements are the requirements which the client actually expected from the system. The client specified the following are the major requirements which they expected from the **WBCOAMS** system.

System security

- System should provide facilities for add, update, view, and delete entries of the managerial staff, administrator and support staff depending on their roles.
- System should have proper login methods.
- System should distinct users and view user specific data and action on the screen.

Customer information and records

- System should provide functions to facilitate add, update, view, search and delete customer's details.
- View payable balances of customers whenever needed.
- Ability to track customer information easily.
- Ability to view the customer transactions details such as orders, invoices and payments

Items

- System should support adding, updating, view, search and delete merchandise items.
- Items should have their own set of pictures and descriptive information.
- Promotes items on the home page.
- Customers are able to pick/add items to their order by fewer processes.

Orders

- System should support creating an order of items through online.
- Provide easy picking items to customer orders.
- Support to add additional remarks of the order such as request special discounts, point out the importance of on time order delivery etc.

Payments

- System should provide an interface to add, view and delete payments details of customers to managerial staff.
- Summarized view of customer payments.
- Customers are able to view their payments information on Capital hardware.

Cheques

- System should provide an interface to facilitate add, update, view, search and delete cheque details.
- System should detect account closed cheques while entering a cheque to the payment because its already resulted cheque was returned due to the account was closed.
- System should always notify if there are available returned cheques to remind the customer.
- List out day to day clearance cheques.
- Sumersible view of cheques.
- Customers are able to view their cheques details and status.

Accounts

- System should create a customer account automatically when a customer is created.
- Summarized view of customer accounts to the managerial staff.
- Customers are able to view their Capital hardware accounts with all their updated transactions.

User management

- Able create, delete and modify system users and customers.
- Users should be able to change their login password.
- Users able to view their personal info.

2.3.2 Nonfunctional requirements

“a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. This should be contrasted with functional requirements that define specific behavior or functions. ” [3]

Usability

System should provide easy to use simple graphical user interfaces (GUI) and it should be able to use easily with its user with minimum training by the administrative user. Proper error messages should give by the system while end user interaction the system for minimize the occurrences that user can make errors.

Reliability

Information presented by the system should be accurate and up to date. System should ensure integrity of data when performing some actions such as entering, updating and deleting etc. of the data.

Security

System should provide a secure login facility to its users. Unauthorized pages containing sensitive data should not be disclosed to end users unless they have required access level.

Chapter 03: Design

3.1 Introduction

“Software design is a process of problem-solving and planning for a software solution. After the purpose and specifications of software are determined, software developers will design or employ designers to develop a plan for a solution. “ [3]

3.2 Brief description of WBCOAMS Customers

The **WBCOAMS** is developed for the customers of Capital Hardware who are granted to purchase merchandise items for long term or short term credit payments. Therefore the system did not include a facility to register random users to become a customer of Capital Hardware.

For register with **WBCOAMS** system a customer need to clarify with their sales activities with the Capital hardware and make request to management about their interest in online purchase or . Once the customer approved to do the credit business with the Capital hardware they will get the username and password in order to use the **WBCOAMS**.

3.3 System development life cycle

“The systems development life cycle (SDLC) is a conceptual model used in project management that describes the stages involved in an information system development project, from an initial feasibility study through maintenance of the completed application. Various SDLC methodologies/ life cycle models described in following subsections have been considered to guide the processes involved.” [4]

3.3.1 Selected System development life cycle

Author chose a waterfall model for the system development since the requirements are straightforward and the system should be delivered once it has fully developed.

The waterfall model derives its name due to the cascading effect from one phase to the other. In this model each phase has a well defined starting and ending point, with

identifiable deliveries to the next phase. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project. [Sommerville, 2007]

The Waterfall model was the first model introduced and used widely in software engineering. In this model, the whole process of developing software is divided into separate process phases such as requirement analysis & definition, system & software design, implementation & unit testing, integration & system testing and operations & maintenance as shown in Figure 3.1. All the phases are cascaded to each other and the next phase starts when the previous phase is signed off and achieved by a set of goals defined in it.

Flexibility of this model is much less than other models, because it does not fully support back tracking. All the requirements should be fully captured before moving into the design phase in the model and it is very difficult to integrate new requirements into the system once moved into design. Hence, it is important to clearly identify system boundaries to gather requirements in full and define system requirements clearly, when using a waterfall model.

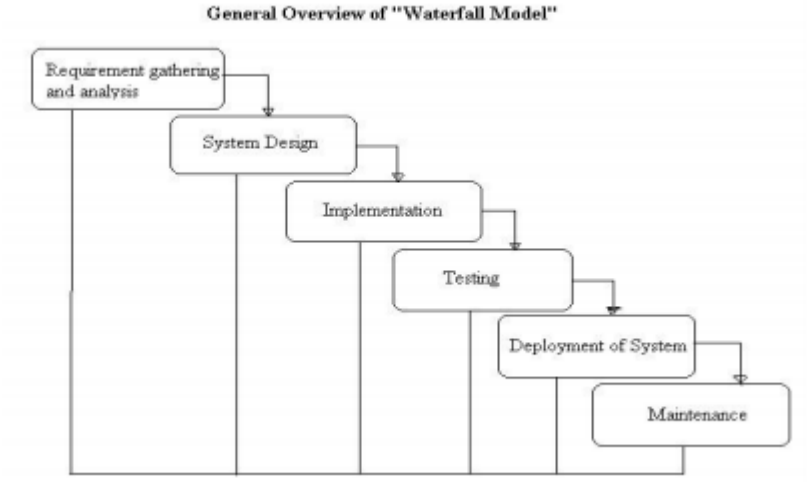

Figure 3.1 Waterfall model

3.4 Object oriented designing

“Object-oriented design is the process of planning a system of interacting objects for the purpose of solving a software problem. It is one approach to software design.” [2]

Object oriented design uses object based approach to create the system design and it is the process of developing object oriented models to implement requirements discovered earlier. Therefore, object oriented analysis (OOA) should be performed prior to this.

Unified Modeling Language (UML) is a visual modeling language, which is used as the standard language for object oriented modeling. UML provides several diagrams to model different aspects of a system such as functionalities, static structure, interaction between objects, state transition of objects and implementation structure.

3.4.1 Use Case Diagram

Use Cases are typically used to describe the typically visible interactions that the system will have with users and external systems. Typically, they are used to describe how a user would perform their role using the system, and as such form an essential part of the development process.

A use case diagram has contained use cases, actors, interactions and system boundaries. An actor is a user and use cases are a top level representation of the intended functionality of the system.

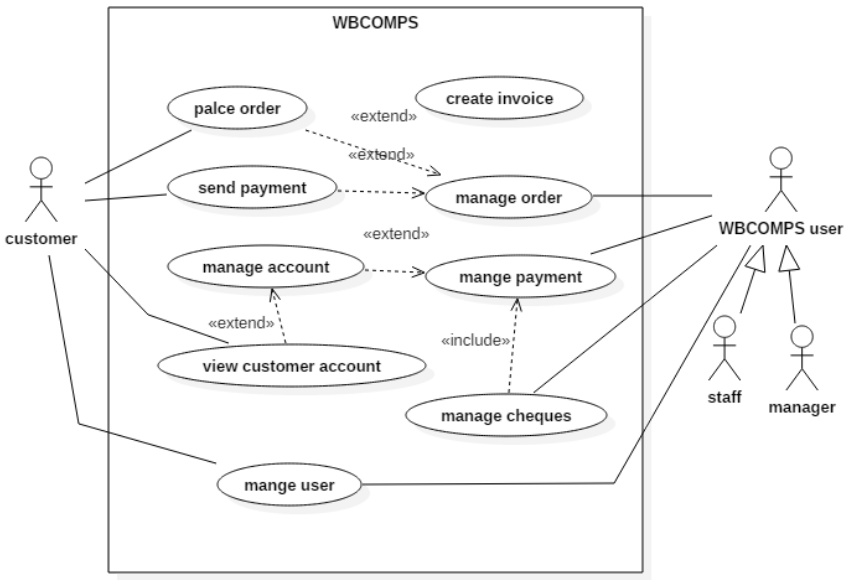

Figure 3. 4Top level use case diagram of the proposed system

Use case diagram for user management

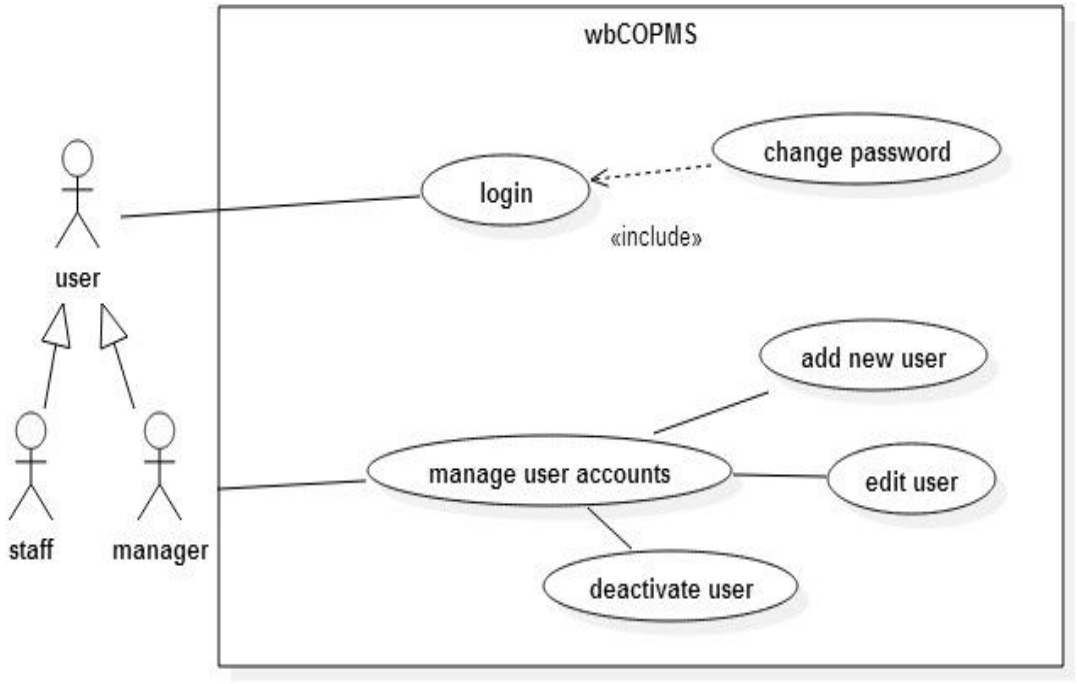

Figure 3. 5 Use case diagram for user management

Use case name	Add new user	
Actors	Administrator	
Description	Create a new user	
Pre-Condition	System user should be logged in	
Typical course of events	**Action**	**System response**
	1. Enter valid user name and password	
	2. Click Add button	System display "successfully created" message

Alternatives	System displays error messages
Conclusion	Creates a new user for this system
Post condition	Data saved in Database

Table 3. 1Use case description for add new user

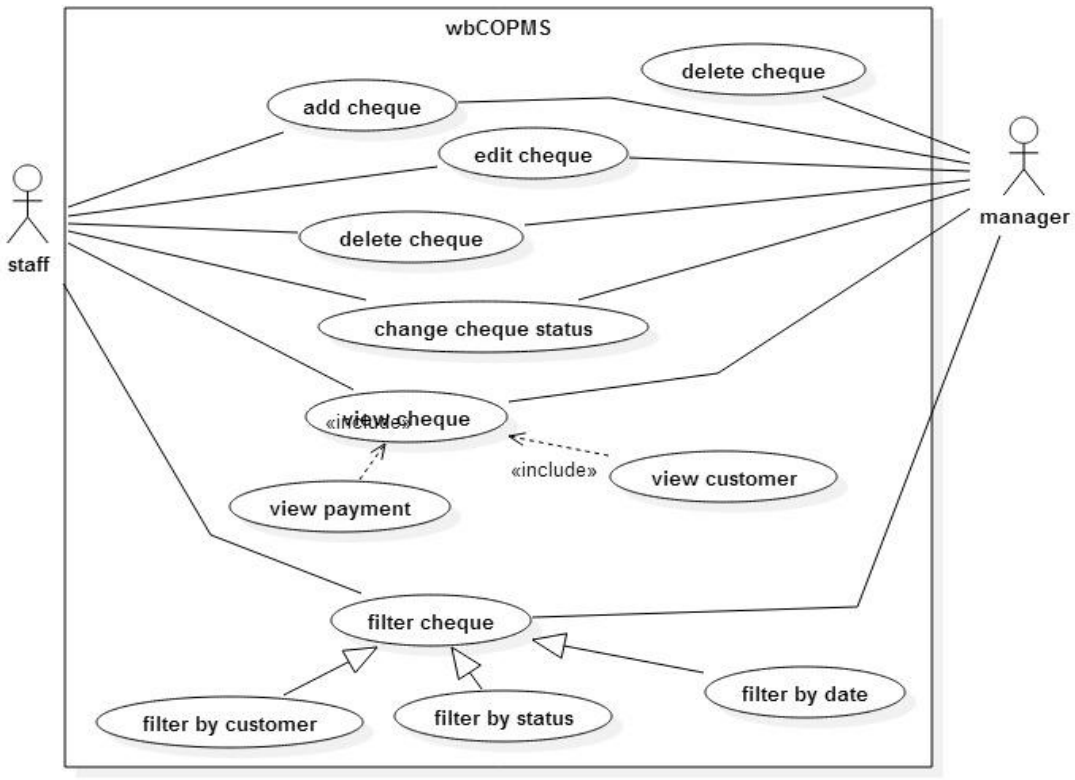

Figure 3. 6Use case diagram for cheque management

Use case name	Add cheque
Actors	Manager, sales man
Description	Actors add a new customer Cheque
Pre-Condition	1. System user should be logged in 2. Customer should be registered to the system

Typical course of events	**Action**	**System response**
	1. select the customer from list while entering customer number or name	System fills the customer number and name.
	2. enter valid cheque details	
	3. enter bank account number	If system detect account is closed then display message ”this bank account is closed”
	4. Click Add button	System display “successfully Added” message
Alternatives	System displays error messages	
Conclusion	Creates a new cheque entry into system	
Post condition	Data saved in Database	

Table 3. 2Use case description for add new cheque

3.4.2 Class Diagrams

Class diagram is a collection of static elements such as classes relationship connected graph as each other in a class diagram, the classes are arranged in groups that share common characteristics. A class diagram resembles a flowchart, in which classes are portrayed as boxes, each box having three rectangles inside.

The top rectangle contains the name of the class; the middle rectangle contains the attributes of the class; the lower rectangle contains the methods, also called operations, of

the class. Lines, which may have arrows at one or both ends, connect the boxes. These lines define the relationships, also called associations, between the classes.

Figure 3. 7 class diagram for domain classes

3.4.3 Activity Diagrams

Activity diagram in which states are activities represent performance of operation transaction triggered by completion operation.

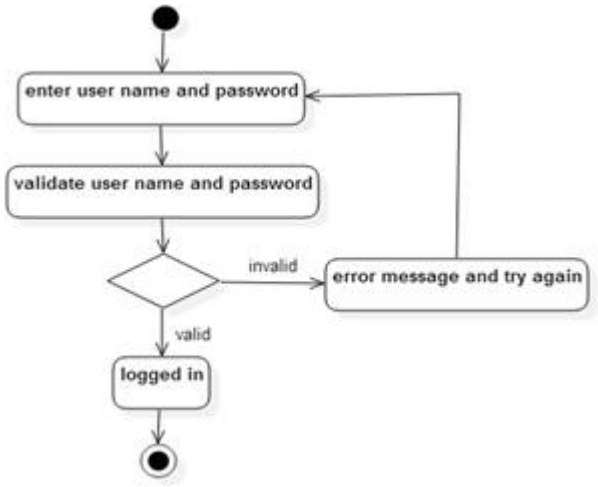

Figure 3. 8 Activity diagram of user, login to the system

3.7 Database Design

Databases play a critical role in almost all areas where computers are used. A database is a collection of related data. Data means known facts that can be recorded and that have implicit meaning.

Good database design is vital to build a robust system, because all data related to business should be recorded accurately while preserving their completeness, availability and security.

A centralized database was designed to implement the proposed system. One of main objectives of developing the proposed system was introducing a database with minimum data redundancy and easy maintenance. Maintenance overheads and redundancy in centralized databases are much less than compared to distributed databases.

3.6.1 Database Normalization

Normalization is a process of decomposing unsatisfactory relations to smaller relations. Normalization helps eliminate redundancy, organizes data efficiently and reduces potential anomalies during data operations.

First normal form (1NF)

The first normal form states that domains of attributes must include only atomic (simple, indivisible) values and the values of any attribute in a record must be single value. The 1NF also disallows composite attributes that are themselves multi valued. These are called nested relations because each record can have a relation with a relation.

Second normal form (2NF)

2NF was preformed to remove partial dependencies (non-key attribute functionally depends on just part of the key attribute).

Third Normal Form (3NF)

3NF was performed to eliminate transitive dependencies (non-key attribute functionally depends on another non-key attribute).

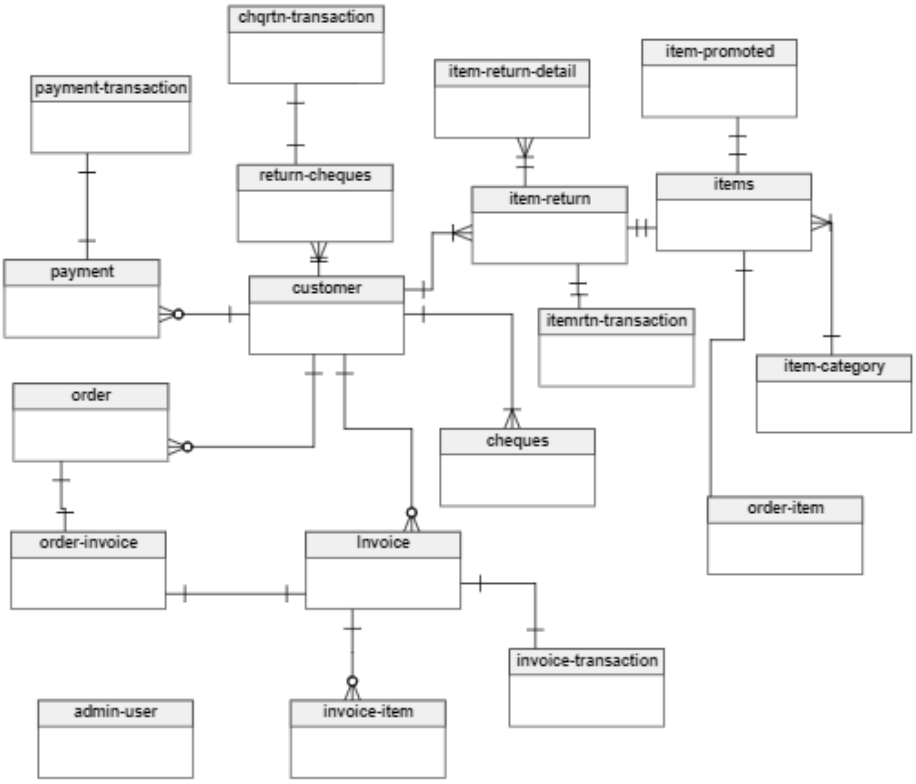

Figure 3. 9 Database diagram of WBCOAMS

3.6.2 Relational Schema

The description of the database is called as the database schema. The following set of tables shows about the relation schema in this **WBCOAMS**.

admin-user

id	name	role	created	userName	password	deleted

customer

id	name	address	city	telephone	nic	mobile	creditLimit	password	userId	image	email	s

payment

id	cusId	created-date	type	cash	remarks

cheques

id	cusId	amount	payId	bank	accountNo	status	remark	user	chqNo

cheque-return

id	chqId	reason	user

invoice

id	cusId	invoice-date	remark	user

order-invoice

orderid	invoiceId

invoice-item

id	itemId	invoiceId	price	quantity

invoice-order

invoiceId	orderId

items

id	name	description	catId	price	lastModifiedUser	status

item-promoted

id	itemId	oldPrice

item-category

id	description

order

id	cusId	orderDate	remarks

order-item

id	orderId	itemId	price	qty

item-return

id	createdDate	user

item-return-detail

id	returnId	itemId	price	qty

Invoice-transaction

id	invoiceId	amount

chqrtn-transaction

id	chqId	amount

payment-transaction

id	paId	amount

itemrtn-transaction

id	rtnId	amount

Figure 3. 10 Rational Schema

3.7 Interface Design

Interface design is an essential part of system design, because it models the main interaction between system and users. Good interface design is vital to success of any kind of a system, because major judgments about the system are done based on looking at interfaces and they also improve the usability of system.

The following important points are considered when designing user interfaces.

- Simple interface design with consistence look and feel over the system to improve user friendliness.
- Minimize colour combination while choosing colours which suit to eyes and make text easy to read.
- Easy navigation through the system while making important functions clearly visible to system users.
- Try to avoid user errors by using proper error messages, necessary field identification and clearly indicate what values can be entered to relevant fields.
- Use meaningful elements and avoid too many pictures to improve performances.

Login screen of the **WBCOAMS** system represented in figure **3.6** this screen provide access to the valid users by entering user name and password.

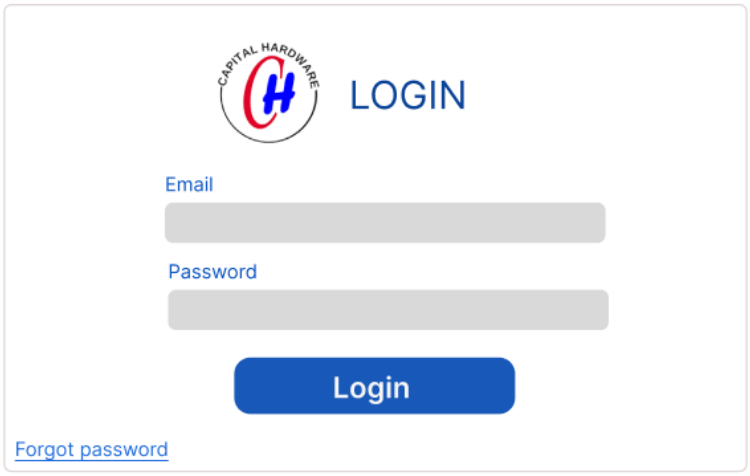

Figure 3. 11Login screen

3.7.1 Home page

The main screens of the **WBCOAMS**, was designed to make sure that user can navigate from one tab to other tab easily as possible.

The customer home page which shown after customer logged in to the system

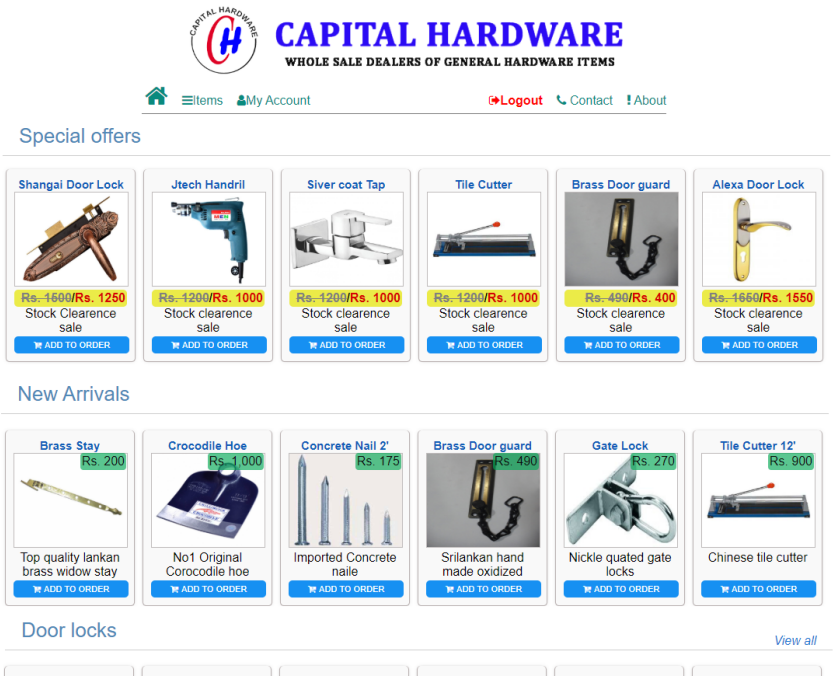

Figure 3. 2 Home page of WBCOAMS in customer view

Home page of administrative view of the system

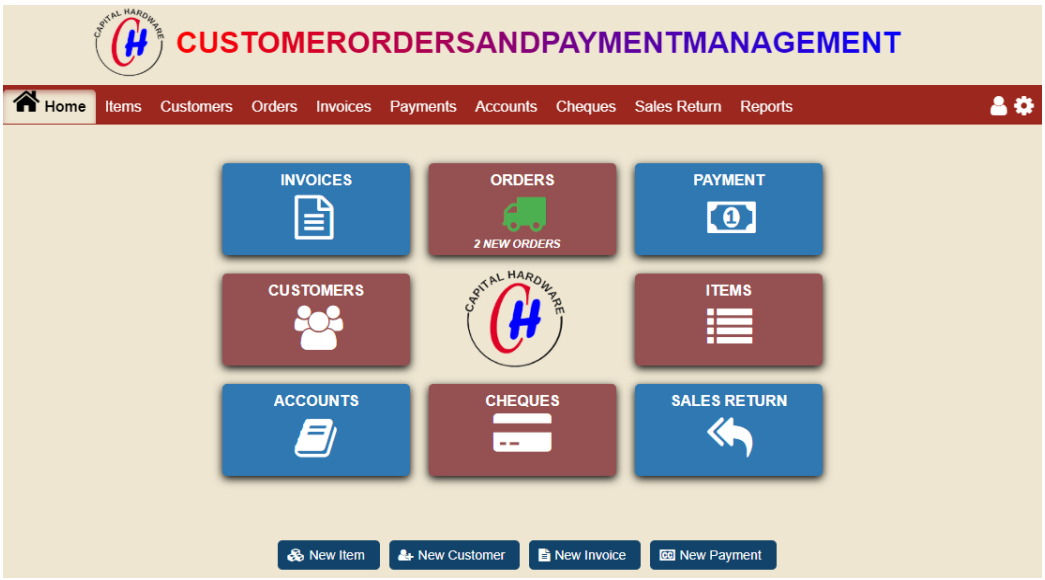

Figure 3. 23 Home page of WBCOAMS in administration

References

[1] "Systems Analysis and Design/Introduction," wikibooks.org, [Online]. Available: https://en.wikibooks.org/wiki/Systems_Analysis_and_Design/Introduction#Rapid_Application_Development_(RAD). [Accessed 20 12 2022].

[2] mariosalexandrou, "systems development life cycle," mariosalexandrou.com, [Online]. Available: http://www.mariosalexandrou.com/methodologies/systems-development-life-cycle.asp. [Accessed 15 1 2023].

[3] "Software design," wikipedia.org, [Online]. Available: http://en.wikipedia.org/wiki/Category:Software_design. [Accessed 20 1 2023].

[4] "http://business-analysis.in/?p=590," [Online]. Available: http://business-analysis.in/?p=590. [Accessed 22 1 2023].