Online Inventory Management System

Submitted by - M. Sc. In CSE (14th Batch)

Name	Roll No	Registration No
Md. Arif Uddin	02	203562
Bishamver Borman	14	203574
Most. Shela Khatun	19	203579
Most. Bilkis Begum	09	203569

Knowledge is Power

DIU

Estd. 7th April, 1995
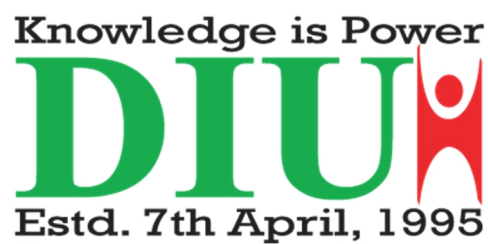
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DHAKA INTERNATIONAL UNIVERSITY

MAY – 2016

DECLARATION

We hereby declare that, the project "Online Inventory Management System" presented is the outcome of the investigation performed by us under the supervision of Mst. Jahanara Akhtar, Associate Professor, Department of Computer Science & Engineering, Dhaka International University, Dhaka, Bangladesh. We also declare that no part of this project and thereof has been or is being submitted elsewhere for the award of any degree.

Countersigned:

_____

Mst. Jahanara Akhtar

(Supervisor)

<u>Signature of the Project Member's:</u>

Name	Signature
Md. Arif Uddin	--
Bishamver Borman	--
Most. Shela Khatun	--
Most. Bilkis Begum	--

Acknowledgement

In the name of Allah, "The most gracious and the most merciful" All praise and glory to Allah, for given us the opportunity to propose the Master's degree in Computer Science and Engineering, in Dhaka International University.

First of all, our appreciation heartfelt gratitude goes to our project Supervisor Mst. Jahanara Akhtar Associate Professor department of CSE, Dhaka International University. We are very grateful to her for constant endeavor step by step guidance and moral support.

We are very much grateful to Prof. Dr. A.T.M. Mahabubur Rahman, Chairman, Department of Computer Science and Engineering, Dhaka International University for his guidance's facilities and the work environment provided to us.

Also, we would like to special thanks to our advisor Prof. Dr. Hafiz Md. Hasan Babu, Adviser of Dhaka International University for his valuable directives on standard course curriculum, dignified teacher and special attention.

Our parents are very much keen and hopeful in the best performance of the dissertation we are going to submit. We wish we could fulfill their aspiration. We also pay regards to our friends in the department who, through their interest and work, are our constant source of inspiration.

Finally, we thank to all teachers, stuffs, Students of different departments of Dhaka International University for their help, support & greetings throughout the whole period of the course that enable us to complete this project successfully.

Abstract

Now-a-days, Internet becomes an important relationship medium in information technology filed and a network of changing the method of human communication and people environments interaction. In addition, information system that using Internet have been upgraded all media and device. Now, Internet makes somebody easy and fast to reach all information. Hence, by using this technology, one project will be completely developed to overcome problems and to replace the manual system. The administrator, staff and lecturer in the Dhaka International University campus can use this Online Inventory Management System as known as web-based application in Internet environments.

This project is aimed at developing an Online Inventory Management System for any University. This system can be used to store the details of the inventory, update the inventory based on the item details, produce receipts and generate inventory reports periodically. This is one integrated system that contains both the user component (used by inventory managers) and the admin component (used by the administrators for performing admin level functions such as adding new items to the inventory).

Currently many small scale and medium scale organizations use a manual system to record the purchase and stock details of the industry, which is a time taking process and needs a lot of hard work which in turn implies a great burden on accounts and stock departments. This system is a complete business solution for those departments which records the information manually, it records the invoice, purchase, shipping details, billing details etc. The system uses modern information technology resources together with traditional management practice and methods to provide the users with the necessary information to manage their Company.

Table of Contents

Chapter 1

Basic Introduction

Most organization or institute involve a continuing flow of assets management. To avoid disruption of flow, most of them make an effort to keep track of the item's amounts on hand. In small organization or institute with small amount of inventories, it is possible for them to remember the approximate amount of their inventory records manually. And if they want to make a stock checking, they just count on the inventories manually as well. However, it becomes more difficult to count the inventories if the amount of inventory. When this condition occurs, some form of record keeping is needed. Manual inventory records can be in an efficient way for organization just to assure that they have sufficient materials. But, to be efficient, the procedures involved must be in a proper way. Usually, it involved several people in keeping records. There are three basic methods involved to overcome this issue [1].

> i) When material is received and added to the inventory, a new inventory balance is calculated by adding the receipt quantity to the prior on-hand balance.

> ii) When material is distributing from the inventory, another new balance is calculated by subtracting the quantity distribute from the quantity on hand prior to the distribute.

> iii) From time to time, for a variety of reasons, it may be found that the recorded on-hand balance differs from the quantity actually on hand. In these cases, the records are updated by adding an adjustment to the on-hand balance or subtracting and adjustment from it, as necessary [2].

These three steps can be combined into a single equation, called the perpetual inventory equation [3].

The equation is as below:

$$COH = POH + RQ - DQ$$

Where: COH = current on-hand balance,
POH = previous on-hand balance,
RQ = Received Quantity,
DQ = Distribute Quantity

1.1 SYSTEM ANALYSIS:

Existing System

Online Inventory Management System deals with online communicating between all users in this system. In the existing manual system huge expenditure and a lot of time is spent in collecting the inventory information. So, there is a need for an integrated automated system, which has some centralized control over the entire process [10].

The following drawbacks of the existing system emphasize the need for online inventory management system:

1. Conventional system makes use of huge amounts of time for providing the information about the inventory to the management.

2. Difficulty in tracking and retrieving data from the related inventory. With computerized systems paper work drastically reduces, data retrieval becomes easy, duplication of work is avoided [2].

Proposed System

The proposed system consists of full on-line data entry with online validations on field and referential checking. The goal of this system is to bring down the workload with the increased efficiency and to speed up the activities. The major activity of Online inventory management system is to provide online communication between the users of the system. Auto generation of emails as soon as the inventory manager or sales person or sales manager is required to send a mail to the administrator, intimating if any problem occurred due to their work.

The other advantages are:
1. Availability of the information immediately after data captures.
2. An integrated normalized relational database will be maintained for the process.
3. Predefined queries for generation of any specific enquiry purposes.

System Specifications

Hardware Requirements for server: -

Intel Core 2 duo or higher processor
RAM 2 GB for x86 and 4 GB for x64 architecture system
Hard Disk Drive 80 GB

Hardware Requirements for client: -
Any specification that can browse internet

MODULES

1. Administration
2. Purchase
3. Store and Distribution

MODULE DESCRIPTION

Name of the module 1: **Administration**

Description: Can login to the system through the first page of the application and Can create new user account for an employee and assign (if existing user) username and password. Admin can also add branch or sub office information, also can update as required.

Sub modules:
- Add / Update Employee Information
- Change / Update Password
- Add / Update Branch

Name of the module 2: Purchase:

Description: Can login to the system through the first page of the application and can do the following sub module task.

Sub modules:

- Add asset category, asset name, brand and model
- Add purchase information
- Add purchase items according to purchase information

Name of the module 3: **Store and Distribution:**

Description: Can login to the system through the first page of the application and can do the following sub module task.

Sub modules:

- Store status
- Store pending status
- Distribution

1.2 Motivation

The process that accounts for an individual intensity, direction and persistence of effort towards attaining a goal. It is the result of the interaction between an individual and the situation motivated person says, 'nothing is impossible'. He will put his best effort.

To manage the workforce in a retail company it is important to change their attitude as it directly affects the customers and growth of the company. By increasing the intrinsic factor employees are more likely to be satisfied and motivated to perform it [3].

Almost all the factories use Computer, Internet and IT related services to their day to day work. As they are export oriented industry and have to communicate with their buyer day to day basis via Email, Voice chat, Video Conference so on and so forth. So, this will not be a new book to them than a new chapter full of interesting issues.

As per my belief and market studies it will be a pretty much effective and helpful tools for the medium and small scale industry to use the Information Technology for the betterment of the company and also to achieve the goal of the company.

And if at least 1% of the market shares has been acquired then it will be an opportunity to pocket a handsome profit. And it is pretty much feasible and getable challenge. Thus, it will also be a positive improvement for this industry [10].

1.3 Objectives

The prime objective of the project is to gather all the relevant data of the organization into one single platform or into one single system. Then to process these data and convert into valuable information so that, management can have a clear picture of the current status and can be able to plan, scheduled, fine tune and management the process to achieve the goal of the organization.

There are many items in a store, which are distribute to branch office / department and purchased from supplier or vendors. A request is placed by the office or branch required details, which are listed below:

- o Item name
- o Quantity
- o Destination

1.4 Scope

Scope of the Software is pretty much wide and there is a huge market share for these organization.

And at the same time as it has different module for Inventory Management, Production Management, Human Resource Management & Accounts Management so, it will be very much possible to use the system to different domain's business some minor to mid-level changes [12].

As the software will use very much common and used framework/ environment along with very less dependency, so it will be very easy to use from Computer, Tablet, Smart Phone etc.

Our work area is to automate the above process or to generate a more efficient system.

The operational benefits of reduced inventory levels, improved auditing and financial control, elimination of paperwork, enhanced staff efficiency and shortened delivery time.

➢ Time savings

An automated inventory solution saves time by streamlining purchasing and inventory control. Tasks that once took hours or even days can be performed with a few clicks of a mouse. Staff no longer wastes time matching receipts with deliveries, figuring out overly complex invoices and keying in redundant information.

➢ Increased accuracy

An automated inventory and procurement solution increases accuracy. Because staff is no longer required to re-enter data from paper documents, clerical errors are dramatically reduced. Mistakes in ordering are also minimized.

1.5 Limitations

➢ Lack of comprehension of the respondents was the major problem that created much confusion regarding verification of conceptual question.

➢ Another important issue will be to taught the current manpower that how to use the software.

➢ It will be a definite challenge to make them (Organization's current stuff) interested to the use of the software.

➢ Insufficiency of necessary information & data is also a limitation.

➢ Rush hours of business were another reason that acts as an obstacle while gathering data.

➢ The final and the most important issue is to ensure that the system is running effectively will without any bug.

➢ Barcode has not implemented yet.

Chapter 2

Implementation

The implementation process begins with the Requirement Gathering and gradually will provide a Software Requirement Specification [11] Documents which trend to Design the software alone with Database Design and finally implement the software.

2.1 Requirement Gathering:

Use Case

A sequence of actions a system performs that yields a valuable result for a particular actor.

• What is an Actor?

A user or outside system that interacts with the system being designed in order to obtain some value from that interaction. Figure 2.1 shows the software architecture of the system.
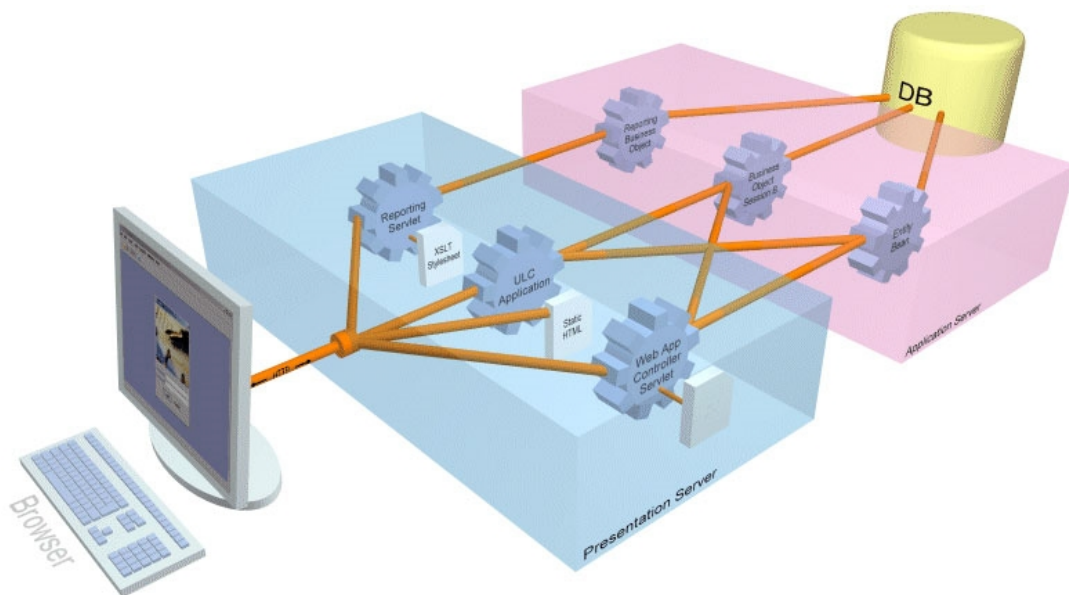

Figure 2.1: Software Architechture

2.2 Information gathered from 'Store'

Currently every organization has a 'Store' department with employees. They maintained the process by the following way:

The company's Inventories are mainly of two kinds:

1. Roll over inventory (which have been purchased and use monthly or yearly basis.)

2. Distribute wise inventory (Mostly asset items, which are purchased or distribute by user ad delivery basis.)

Requirement of Managerial Level:

As the whole process is manual and time consuming Managerial level cannot get his desired information in quick time. And it also makes some inventory untraceable. So, the requirements are as follows:

First of all, they want a very easy to understand interface so that employee can adopt to the environment in almost no time.

At any point of time he wants to see the current in hand inventory status.

He also wants to know the value of unused inventory.

He wants to get access of all this information from web (because some time he stays in abroad).

He wants that the data will be inputted from different workstation and will be stored in a central server.

2.3 Requirement analysis

The requirements gathering exercise started with a number of fairly informal discussions with researchers. These one-to-ones discussed the current project activity of the user to determine stages and processes for their activities. They were also asked about the systems that they used and an attempt was made to identify gaps in the provision of support systems.

Subsequent one-to-one interviews were more structured, with a view to determining whether the research followed similar processes or whether it diverged from those interviewed previously.

These interviews looked into more detail about the actual systems used, although still kept the flexibility to look into unused provision or gaps in research systems. The majority of interviewees were from the store Departments and Purchase. By limiting the number of departments involved, a degree of consistency and depth of understanding with regard to the Project process was achieved.

The major limitation of the methodologies employed in the requirement gathering phase of the work package is related to sample size. The resource available to the project limited the number of one to-one interviews that could be undertaken within the allotted timescale. Despite this, it was felt that increasingly repetitive answers indicated that at least some of the key requirements and main phases of research activity were already being identified.

Chapter 3

Software Requirement Specifications (SRS)

3.1 User requirements

Service and option that must be followed to the system are described below:

✓ **Login:**

A user name & a password will be required in order to log in to the system. There must be admin panel where admin can create user id with password and user can set different access permission to particular user account. And every user can able to change his own password.

✓ **Add Vendor:**

This function must have allowed user to add Vendor Name, Contact Number, Email, Country, Address, Contact Person information.

✓ **View Current Vendor:**

This function will list the entire current vendor alone with all the information of respective vendor.

✓ **Alter or Delete Vendor:**

This function will allow altering or deleting the current information of vendor.

✓ **Add Purchase:**

This function must have allowed user to add New Purchase to the system alone with following information: Work Order & Bill Number (must be unique), Date, Asset Category, Vendor Name. After Saving this information, user have to input item / asset information with its own breakdown such as Asset Name, Brand, Model, Serial / Tag no, Unit Price, Expire Date and warranty period.

✓ **View or Search Purchase:**

This function will allow user to view all current purchase information, user have to select the purchase ID from dropdown menu which shows the work order and bill number as purchase id.

✓ **Report:**

This function will allow user to get report on current store status.

3.2 System Requirements Specification

➢ Operating System Graphical OS like Ubuntu, Windows, MAC OS
➢ Web-Technology PHP, PHP Framework (CodeIgniter)
➢ Front-End   HTML5, CSS3, JavaScript, jQuery, Ajax, Bootstrap
➢ Back-End   MySQL or MariaDB
➢ Web Server   XAMPP (Cross-Platform, Apache, MySQL, PHP and Perl)

3.2.1 Personal Web Server (PWS)

Personal Web Server can do anything if the system put in the server. A web server can be the gateway to the web. It does not only allow people to visit the web page on the own server, it can also allow to temporary host large files or other types of materials. A web server provides a folder in which to store all files.

3.2.2 Hypertext Markup Language (HTML)

HTML is rather straightforward, and it is pretty simple to create a tag or two and throw some text in. It is great to create an online web page. There's much more to building a web page than just creating a couple of tags and adding some text [13].

3.2.3 Preprocessor Hyper Text (PHP)

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP: Hypertext Preprocessor [14].

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications [15].

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as a *de facto* standard. Since 2014 work has gone on to create a formal PHP specification [16].

3.2.4 Graphical Operating System

The GUI familiar to most of us today in either the Mac or the Windows operating systems and their applications originated at the Xerox Palo Alto Research Laboratory in the late 1970s. Apple used it in their first Macintosh computers, followed by Atari with their ST range, and Commodore with the Amiga. Later, Microsoft used many of the same ideas in their first version of the Windows operating system for IBM-compatible PCs. Examples of systems that support GUIs are Mac OS, Microsoft Windows, NEXTSTEP and the X Window System. The latter is extended with toolkits such as Motif (CDE), Qt (KDE) and GTK+ (GNOME) [17].

3.2.5 MySQL

MySQL [5] is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together.

3.2.6 JavaScript, jQuery and Ajax

JavaScript [6] is an interpreted programming or script language from Netscape. It is somewhat similar in capability to Microsoft's Visual Basic, Sun's Tcl, the UNIX-derived Perl, and IBM's REXX. In general, script languages are easier and faster to code in than the more structured and compiled languages

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. **jQuery** is the most popular JavaScript library in use today, with installation on 65% of the top 10 million highest-trafficked sites on the Web. **jQuery** is free, open-source software licensed under the MIT License.

Ajax [7] is a client-side script that communicates to and from a server/database without the need for a post back or a complete page refresh. The best definition I've read for **Ajax** is "the method of exchanging data with a server and updating parts of a web page - without reloading the entire page."

3.3 System Architecture

- This software is web based so accessible from anywhere of the word, where there is Internet connection.

- Accessible from Desktop, Laptop, Tablet and Smart Phone.

- Accessible from Graphical OS Windows, Mac and Linux.

Chapter 4

Project Planning and Methodology

4.1 Project Planning

The project planning [9] in Online Inventory System development will be provided to estimate the resources, cost and schedule. These estimates are made within a limited time frame at the beginning the software development stage like Online Inventory System and if should be updated regularly as a project progresses. In addition, estimates or planning should first be attempted to define best-case and worst-case scenario so that the project outcomes can be anticipated. The numbers of useful techniques for time and effort estimation do.

Process and project matrix or project flow can provide a historical perspective and a powerful input to the generation of quantitative estimation. The past experience can aid immeasurably as estimate are developed and reviewed. To identify information technology requirement and an action plan or Work Breakdown Structure (WBS) is then developed to define the specific project to achieve the identified Online Inventory System goals. The development approach, benefits and a reasonable estimate of the development cost of the conceptual design are then determined.

4.2 Project Methodology

The Online Inventory System uses the waterfall model or life cycle model. In addition, the techniques can provide assistance for completing the deliverance. In the waterfall method, it suggests a systematic, sequential approach into system development that begins at the system level and progresses through the requirement analysis, design phase, implementation, testing phase and maintenance. The Waterfall model is shown in figure 4.1 [18]

A software development methodology is very formal and precise system development process that defines a set of activities, methods, best practices, deliverables and automated tools for software development. Project manager develops and maintains most or all information system and software. The method in this Online Inventory System project is designed to complete the coverage of the software development process by providing guidance from the analysis phase down to implementation process. The methodology can make the task easy, avoid or reduce confusion, save of time and instruct users of the system depending, on objectives. Evaluation and improvement are a continuous integrated process at every stage in this methodology. Below are the activities in that methodology:

a) Requirement Gathering

b) Estimation and Scheduling

c) Analysis and Design

d) Implementation

e) Delivery
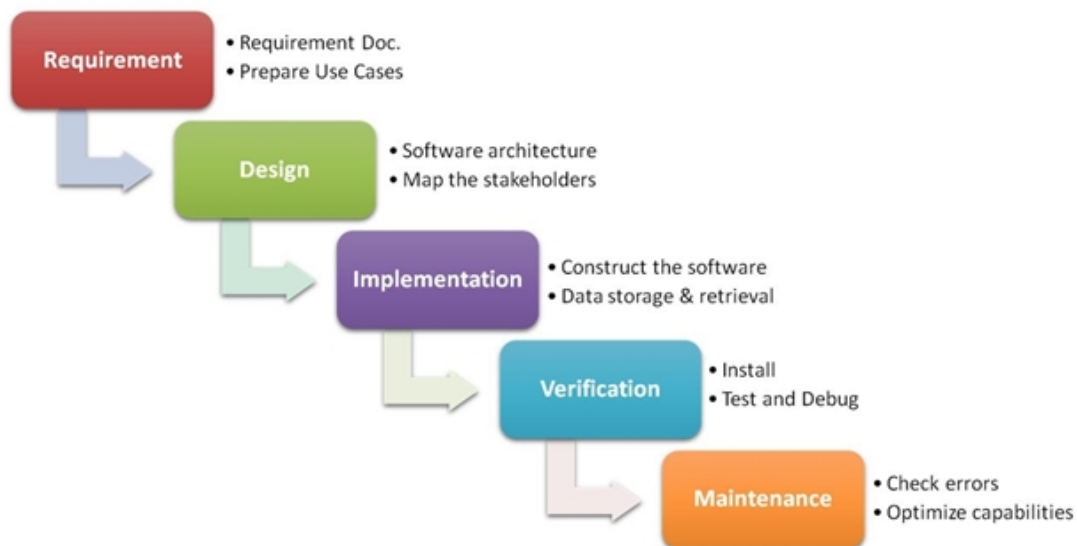
Figure 4.1: Waterfall Model

4.2.1 Requirement Gathering

The requirement gathering process is intensified and focused specifically on system. The Online Inventory System has required functions, behaviors, performance and interface designs. Requirements for this system are reviewed with investigation of past researches.

Gathering involves a detailed study of the current system, leading to the specifications of a new system. Gathering is a detailed study of various operations performed by a system and their relationships within and outside the system. During this phase, data is collected on the available files, decision points and transactions handled by the present system. Interviews, on-site observations and past researches are the tools used for system analysis. All procedures, requirements must be then analyzed and documented in the form of detailed data dictionary, logical data structures and miniature specifications.

4.2.2 Estimation and Scheduling

The estimation and scheduling techniques described are often implemented under constraint of a defined deadline. It best estimates indicate that the deadline is unrealistic, a competent for the project development.

4.2.3 Analysis and Design

System design is actually a multi-step process that focuses on four distinct attributions of a program data structure i.e.: software architecture, and interface representations, procedural details. The design process translates requirements into a representation of the system that can be evaluated for quality before starting the implementation phase [19].

Here the solution proposed by the developer is developed into a complete technical specification. Files, databases, data entry screens and reports in Online Inventory System are designed. Programming languages and a database management [21] system may be selected. Hardware and other software selections are included in the system design. Programs to be written are identified and each program's logic is completely spelled out. Throughout the design phase should develop test plans and procedures for quality assurance.

4.2.4 Implementation and Testing

The design must be translated into a machine-readable form. The implementation step performs this task. If the design is performed in a detailed manner, implementation with code can be accomplished mechanistically. The grammatical and spelling check should be over in this phase. In coding phase, it is now for turn to add code without disturbing the design. The designer may need to develop some graphic buttons whenever the necessary, especially while using some form buttons. Coding developer should generate necessary testing plans as well as technical documentation [4].

Once the implementation has been generated, program testing begins. The testing involves the internal and external user. The testing process focused on the internal logic of the system, to ensure that all statements have been tested and have been tested on functional externals. With the testing phase, conducting test to uncover errors and to ensure that the defined input will produce actual results that agree with required result.

After each program is certified as working properly as individual entity, the programs are then tested together in what is called a system test. Finally, if the programs integrate properly the system is stress tested. Large volumes of real data arc processed by the system and an evaluation is made about whether or not the system can handle the load. Any problems that are discovered are either sent back to the developers to be fixed before moving to the next stage or if considered not critical some problems may be deferred until the maintenance stage.

4.2.5 Delivery and Maintenance

The Online Inventory System delivery needs preparation of constant analysis and submitting in the server to run completely. System will undoubtedly undergo change after it is delivered to the user. Change will occur because errors have been encountered, it is because the system must be adapted to accommodate changes in its external environments. Once the Online Inventory System is operational, technical maintenance, content management and updating, site visit activity reports, staff training and mentoring are needed on a regular basis depending on the complexity of the system.

4.3 Methodology Justification

These methodologies are converging with project management techniques, process management techniques and provide a delivery system addressing many of application system development problems. There are significant productivity gains to be made by using a methodology for the system Online Inventory System development process. There are reasonable and included that system management can protect its procedural by ensuring that the project supports are well defined and likely to succeed. Methodologies frequently have deliverance in the business case that defines the economics, benefits, costs and resources need. Then, with the methodologies used to provide guidance, the business case is updated throughout the project waterfall model to reflect changes that affect the ongoing and non-recurring costs and benefit of the Online Inventory System.

With using the methodology, techniques and right instruments, the Online Inventory System development follows the plan, is built by deploying new and existing information system and the application. It can reduce the costs of the project by minimizing reworking and maintenance effort, reducing cycle time of projects by providing a consistent, repeatable process to follow. Then it can increase the system quality by providing superior solutions that meet user needs. In addition, this methodology can assist in managing project more efficient and providing guidelines to conduct a project and to track progress and highlight potential delay easily.

Chapter 5

System and Software Design

The Software Design and Development service includes software requirements analysis, software and system architecture design, coding, testing, implementation, documentation, training, support, and on-going software and system maintenance.

5.1 ER Diagram

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems [2]. Figure 5.1 shows the ER Diagram of this software.
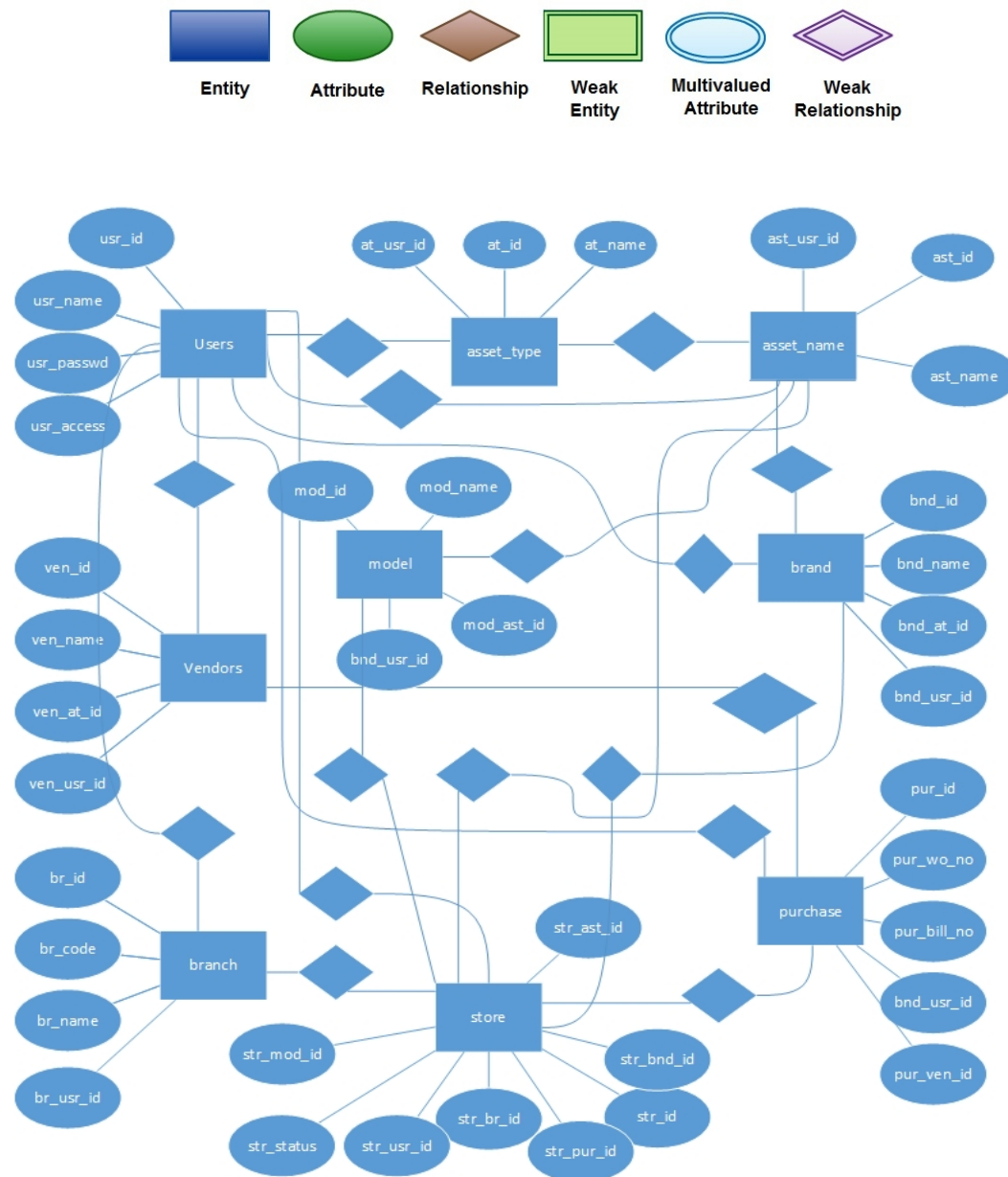
Figure-5.1: ER Diagram

5.2 Software Design

After analyzing all requirements, the whole system is divided into 4 Module:

1. Login System Module.

2. Application Setup Module

3. Purchase Module.

4. Store Management Module.

Login System Module:

➢ This module will compose same Login window for admin and user login.

➢ Administrator of the system will define the role and permission level of user.

➢ Also will be able to Create or Delete User Account, Reset Password.

Application Setup Module

➢ This module will setup all initial entry for user related issue.

➢ Also add sub-office or branch information.

Purchase Module:

➢ This module will compose of Accessories and Roll over inventory

➢ Add product category, item name, related brand and model no

➢ Add basic purchase information like work order number, bill number, category and category wise vendor information.

➢ Add all asset related information.

Store Management Module:

➢ This module will compose of Store related information.

➢ Preview pending issue to adding assets to store

➢ Also distribute assets to another location or branch.

The module descriptions are as below:

Module: Login (Before)

Valid Users of the Software will get access to Software through this Login Form. Each user will login first time through 'Username' and 'Password' provided by **'System Administrator'.**

Module: Login (After)

After Successful Login each user force to change password if they are not change their initial password. And no menu will show except the user menu for logout and other support.

Module: Users Information

This Module "**Users**" is used to create user, set access level for different Users.

Module: Purchase Management

This module manage all assets related information, vendor information and purchase items. First have to insert asset category, then category wise item, then item wise brand, then brand wise model. After that, add the vendor information or enlist the vendor to this system according category. After adding vendor information, use have to add purchase information like work order number, bill number, asset category and vendor selection

Module: Store Management

This module maintains store information for all assets. Find the pending items, matching and accept the assets to store, find and distribute assets according to their serial / tag / holding number.

5.3 Database Table Description

The Database design of our "Online Inventory Management System" is a pretty uncomplicated. Below is the short description of the database tables that we need for our software:

TABLE 5.1: ASSET_TYPE

This table contains asset type like ICT, Furniture, Land etc.

COLUMN NAME	DATA TYPE
AT_ID	INT(5)
AT_NAME	VARCHAR(30)
AT_REMARKS	VARCHAR(250)
AT_USR_ID	INT (5)
AT_UPDATE	TIMESTAMP

TABLE 5.2: ASSET_NAME

Table 5.2 contains name like CPU, Monitor etc.

COLUMN	DATA TYPE
AST_ID	INT(5)
AST_NAME	VARCHAR (50)
AST_AT_ID	INT (5)

AST_USR_ID	INT (5)
AST_UPDATE	TIMESTAMP

TABLE 5.3: BRAND

Table 5.3 contains Asset Name wise brand name like HP, DELL, ASUS etc.

COLUMN NAME	DATA TYPE
BND_ID	INT (5)
BND_NAME	VARCHAR(50)
BND_AST_ID	VARCHAR(5)
BND_USR_ID	VARCAHR(5)
BND_UPDATE	TIMESTAMP

TABLE 5.4: MODEL

The MODEL table contains Brand wise model number of all assets

COLUMN NAME	DATA TYPE
MOD_ID	INT (5)
MOD_NAME	VARCHAR(50)
MOD_BND_ID	VARCHAR(5)
MOD_USR_ID	VARCAHR(5)
MOD_UPDATE	TIMESTAMP

TABLE 5.5: VENDORS

Information about Vendor details, vendor name, vendor address, phone, email etc. are the contents of VENDORS table.

COLUMN NAME	DATA TYPE
VEN_ID	INT(5)
VEN_NAME	VARCHAR(50)
VEN_ADDR	VARCHAR(250)
VEN_DIS_ID	INT(5)
VEN_TEL	VARCHAR(15)

VEN_MOB	VARCHAR(15)
VEN_EMAIL	VARCHAR(35)
VEN_FAX	VARCHAR(15)
VEN_WEB	VARCHAR(35)
VEN_KCP_NAME	VARCHAR(35)
VEN_KCP_MOB	VARCHAR(15)
VEN_KCP_EMAIL	VARCHAR(35)
VEN_AT_ID	INT(5)
VEN_USR_ID	INT(5)
VEN_UPDATE	TIMESTAMP

TABLE 5.6: BRANCH

The branch or remote office and department information is recorded in this table.

COLUMN NAME	DATA TYPE
BR_ID	INT(5)
BR_CODE	VARCHAR(6)
BR_NAME	VARCHAR(50)
BR_ADDR1	VARCHAR(100)
BR_ADDR2	VARCHAR(100)
BR_DIS_ID	VARCHAR(1000
BR_ZIP	VARCHAR(10)
BR_TEL	VARCHAR(16)
BR_MOB	VARCHAR(16)
BR_FAX	VARCHAR(16)
BR_EMAIL	VARCHAR(30)
BR_KCP_NAME	VARCHAR(150)
BR_KCP_MOB	VARCHAR(16)
BR_KCP_EMAIL	VARCHAR(30)
BR_USR_ID	INIT(5)

BR_UPDATE	TIMESTAMP

TABLE 5.7: PURCHASE

All purchase related information like work order number, work order data, bill number, bill date, asset category, vendor name etc. are stored in PURCHASE table.

COLUMN NAME	DATA TYPE
PUR_ID	INT(5)
PUR_WO_NO	VARCHAR(30)
PUR_WO_DATE	DATE
PUR_BILL_NO	VARCHAR(30)
PUR_BILL_DATE	DATE
PUR_AT_ID	INT(5)
PUR_VEN_ID	INT(5)
PUR_USR_ID	INT(5)
PUR_UPDATE	TIMESTAMP

TABLE 5.8: STORE

This table contains information about asset details and have fields like, Asset Name, Brand, model, Unit price, Quantity, warranty period, expire date etc.

COLUMN NAME	DATA TYPE
STR_ID	INT(11)
STR_PUR_ID	INT(5)
STR_AST_ID	INT(5)
STR_BND_ID	INT(5)
STR_MOD_ID	INT(5)
STR_SL	VARCHAR(35)
STR_QNTY	INT(5)
STR_UNT_PRICE	DECIMAL(8,0)
STR_EXP_DATE	DATE

STR_W_PRE	INT(5)
STR_LOC	VARCHAR(6)
STR_PRE_LOC	INT(5)
STR_STATUS	INT(5)
STR_USR_ID	INT(5)
STR_UPDATE	TIMESTAMP

TABLE 5.9: ORGANIZATION

The records about Company-Info detail organization name, address, phone, email, etc. are the content of this table.

COLUMN NAME	DATA TYPE
ORG_ID	INT(5)
ORG_NAME	VARCHAR(50)
ORG_NAME_S	VARCHAR(30)
ORG_ADDRESS	VARCHAR(250)
ORG_DIS_ID	INIT(5)
ORG_TEL	VARCHAR(15)
ORG_MOB	VARCHAR(15)
ORG_EMAIL	VARCHAR(35)
ORG_WEB	VARCHAR(50)
ORG_FAX	VARCHAR(15)

Chapter 6

Testing

This section describes the objectives and extent of the tests. The goal is to provide a framework that can be used by managers and testers to plan and execute the necessary tests in a timely and cost-effective manner.

a) Test objectives

The objective of the test suite is to provide adequate coverage metrics, requirements validation and system quality data such that sufficient data is provided for those making the decision to release.

b) Extent of tests

The tests referenced herein are written to validate use cases, requirements (both functional and non-functional), system architecture, and object design. The structured tests for object design will be run first as the components of the system are developed. The structured tests to validate the system architecture will be run next as the system is integrated in bottom-up fashion during integration test.

6.1 Testing Plan

Testing is an important part of the systems design. It involves checking for correctness, which is one of the focuses of this project. The way to test to see if the system is working correctly is to use black box testing (or functional testing). What black box testing does is to feed in inputs and obtain relevant outputs. These inputs and outputs are then studied and checked to make sure that they are desired the result, in order to make sure that the system is behaving properly [20].

We planned to perform test parallel with the implementation of the new system and we are going to follow some testing procedures defined in the software engineering literature.

First, we will build my database and insert sample data in it. These sample data were taken from Wear & Style record section. As a result, the quality of those sample data will be the same as real environment. The functionalities of the database will be tested with sample operations such as queries, updates, insertions and deletions. Actually, all of these operations will be performed by custom written methods which will be called from various sections of our software. We just need to check out that the database is appropriate to store results of these operations. The most important thing for proper operation is to ensure that the database is in correct location at all times.

Then we will build our software interface. We will start from the user interface in order to identify the components and objects that will need. Then we will define the appropriate classes and methods. we will test the operations of each function with sample data independently. Then we will test the functionality of each with my database. We also need to check the format of the data entered by the user which are fixed such as Order quantity. Quantity can only contain numbers. Any characters into Quantity data will make an error and the user will be notified accordingly.

6.2 Testing Schedule

Testing of the system was predominantly "Black Box" based. The test schedule followed the implementation schedule and was carried out incrementally. Testing was a mixture of 'bottom-up' and 'top-down'. Each unit was tested first independently. Then the whole system was tested with all components. The final test of the system was completed successfully.

Black Box testing is a software testing technique whereby the tester does not know the internal workings of the items being tested. For example, in a black box test in software design, the tester only knows the inputs and what the expected outputs should be and not how the program arrives at those outputs. The tester does not ever examine the programming code. More specifically, this technique checks that the program behaves accordingly to the specification, meaning that the program produces the correct output for each input. In this part, we are going to analyze the testing schedule and the techniques we have planned to use.

Code-Walkthrough:

As the developer of this software, it is completely my responsibility to check the code for each part of the system. This is seen as far better to find a problem at this low level rather than later in this testing cycle where several components have just grouped together.

This should be completed as soon as the coding for each section of the system has been completed. Things to look for in the code are syntax errors, incorrect/invalid function or variable names, and inefficient data structures. Once the changes are completed, a second walkthrough should occur to check for any problems created by changes made to the code.

Component Testing:

This should occur on the development machine, immediately after the cod has been given the "all clear" in the cod-walkthrough. The code for individual functions are compiled and executed to see whether it performs the functions that it was designed to. If a section of code does not produce the required outcome, then the code should be checked against the requirement specification given for that section, and rewritten as required.

Bottom-up Integration Testing:

Integration testing is where all the components will be combined together and checked to see if they all work correctly. If they don't, then there are problems with the code and unit testing will have to review again in order to pinpoint the problem and fix it. According to Somerville, the integration should ideally be completed on a machine that was not used during productions of any individual sections of code.

System Test:

In this part, the objective is to make sure that the system can perform the way the user or the visitor has asked it to and that the system design has been implemented correctly. Testing process of the system is shown in figure 6. The two major issues those need to be taken into consideration here are:

Function Testing: Functional testing checks that once the system has been integrated together, it works as desired, by checking that the functional requirements have been satisfied.

Performance Testing: Functional testing checks that the non-functional requirements have been satisfied. This will include security matters, speed, accuracy and reliability.

The two major issues those need to be taken into consideration are:

i) Fails incorrect functionality fails

ii) Fails acceptance test/new functions to be added
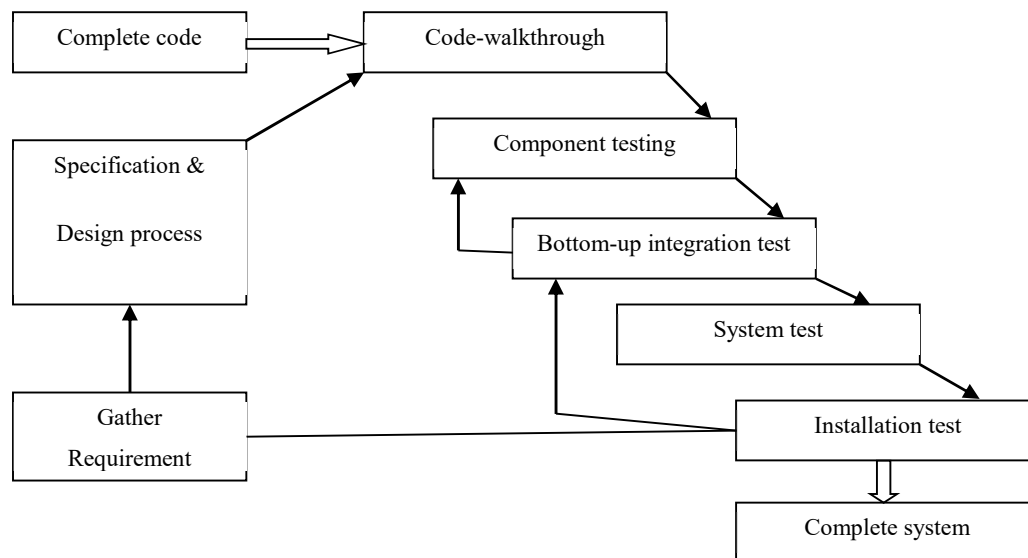
Figure-6: Testing Process of the System

The testing schedule that used in order to test the system was the following:

 - ➢ Unit/component testing
 - ➢ Integration testing
 - ➢ System testing

6.3 Unit Testing

Here is a brief description of how we have tested our system. During **i**mplementation, each unit was built and tested in parallel, making sure it meets its requirements to the fullest extent.

The first testing task that was performed was to check whether the database connection is correct and SQL queries work properly. The next text included adding, deleting, modifying and searching records in the database.

Similar tests were performed for all functions those were used to build this software. Some errors were found and they were fixed. The database was also tested to check that values fit into it nicely. The values entered by user in the quantity and in/Out section were also checked for valid format.

6.4 Integration Testing

In this part, units had to be tested to see if they work together properly. The general test performed on each section was to make sure that values entered in the relevant textboxes were updated the database according to the functions performed by the uses, and at the same time, updates all the pages in the software.

6.5 System Testing

Once the integration test has been made, we set out to make sure that our system worked as specified in the design document. We checked together that the system works as desired, by checking that the functional requirements have been satisfied.

Chapter 7

Software overview

7.1 Screen shots

Login:

Login is most essential issue to any secure, database oriented software. It is ensuring that only authorized user can access, visit, make change and get report from this software. Figure 7.1 shows the login screen of this software.
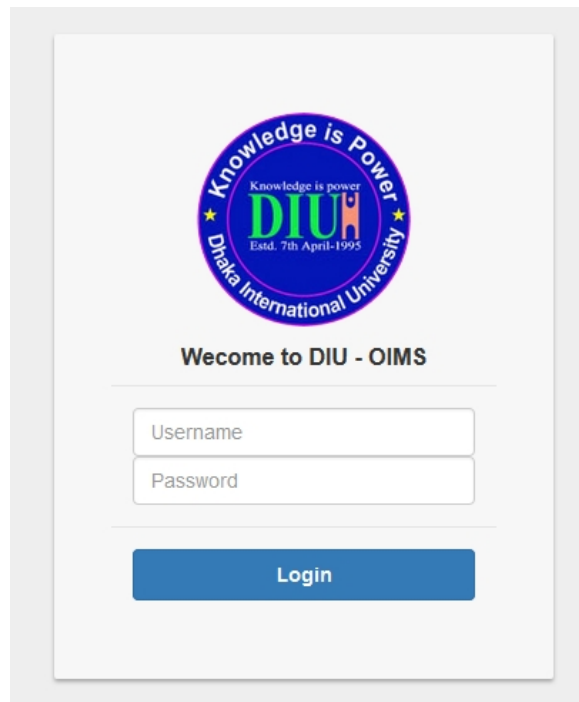
Figure 7.1: Login

After successful login user / admin will see the following home page with the menu of their access level. Here figure 7.2 shows the home page
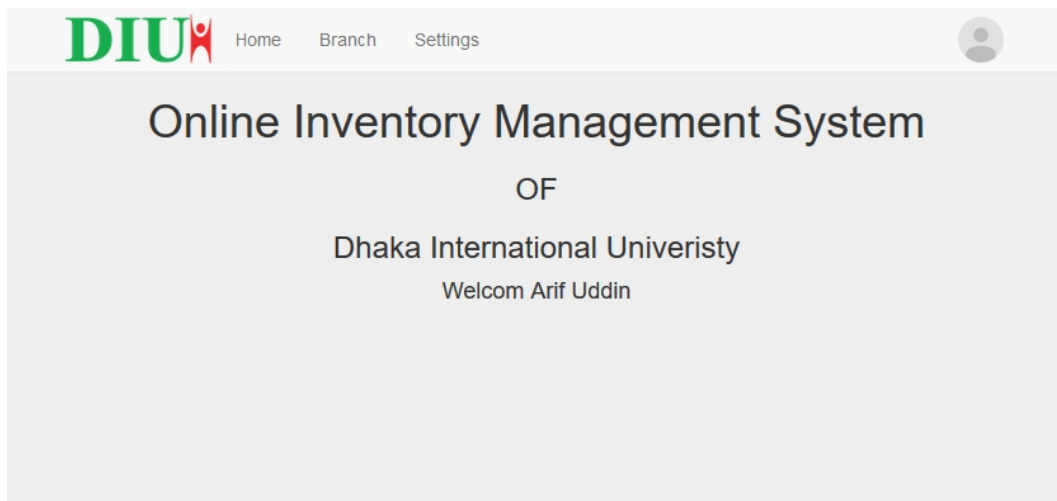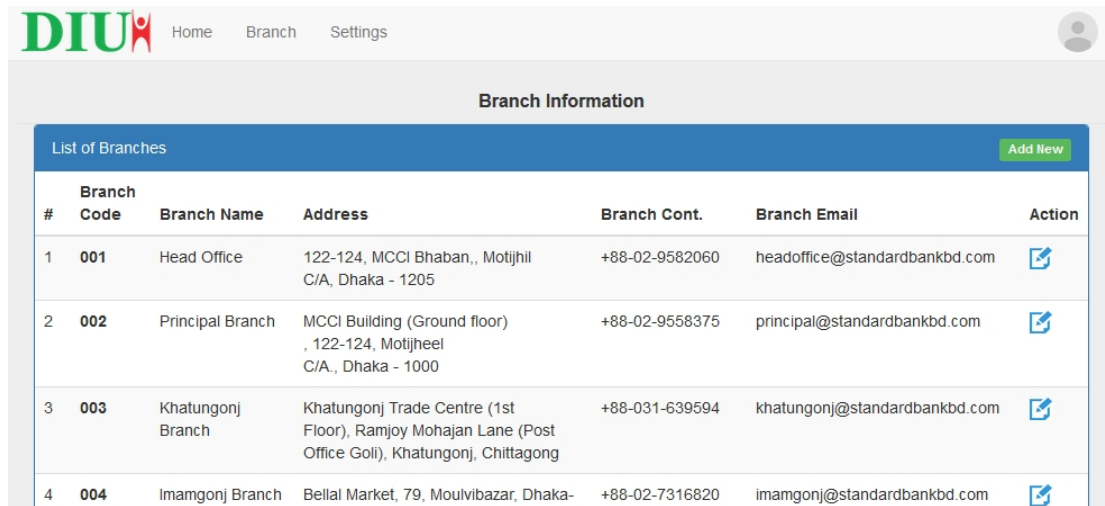
Figure 7.2: Home page

Branch:

You may have some or more branch office, from where you also want to get informed the asset information, inventory status and transfer or receive goods. For that you need to maintain the branch information, address, contact number, email address. You can do this from this Branch menu as you required. Figure 7.3 shows the branch information of the software.
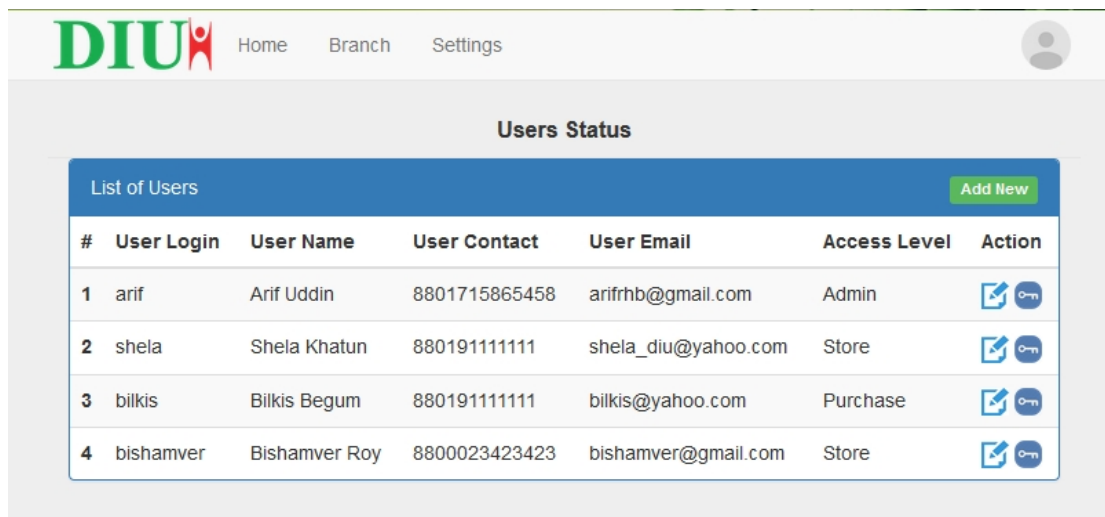
Figure 7.3: Branch

Users:

Settings menu contain a sub-menu Users. From here admin can create and update user information. Admin also can change password of any users as required. The user information page is shown figure 7.4.
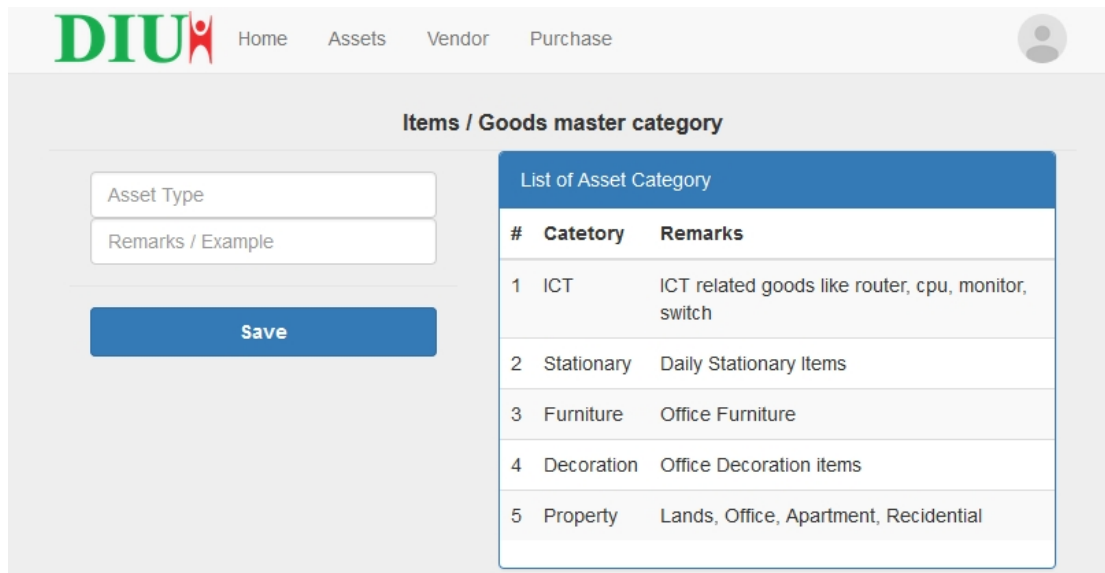
Figure 7.4: Users

Category:

Category menu contains the sub-menu of Assets menu. From here we can maintain / control our assets category like Information & Communication Technology Items (ICT), Furniture, Decoration, Land / Property, Stationary and etc. Figure 7.5 shows category information.
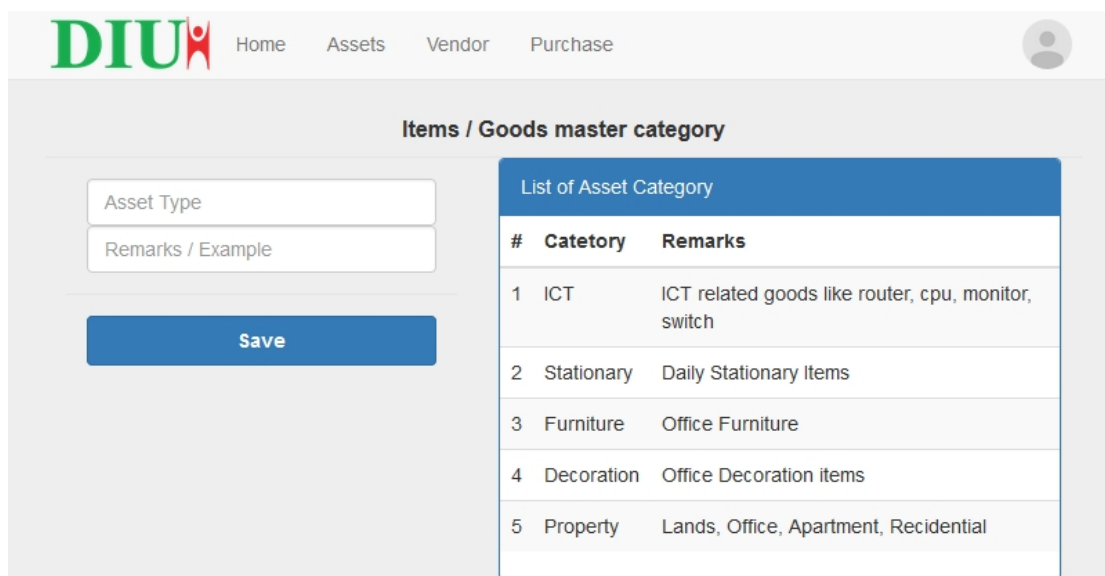
Figure 7.5: Category

Item:

Form this menu user can add Category wise items and can see the item list that already listed to this database. First have to select the Category name, which you want to add item or want to see the item list from Select Category dropdown list. Figure 7.6 shows item menu.

Figure 7.6: Item

Brand:

Brand menu contain as a sub-menu of Assets menu. Every asset items have some brand or manufactural brand, user have to input this from here and also can get update list from this menu. To see any item brand, user first have to select item category, then select item name and then input Brand name or see the list of item wise brand. Figure 7.7 shows Brand information page of this system.

Figure 7.7: Brand
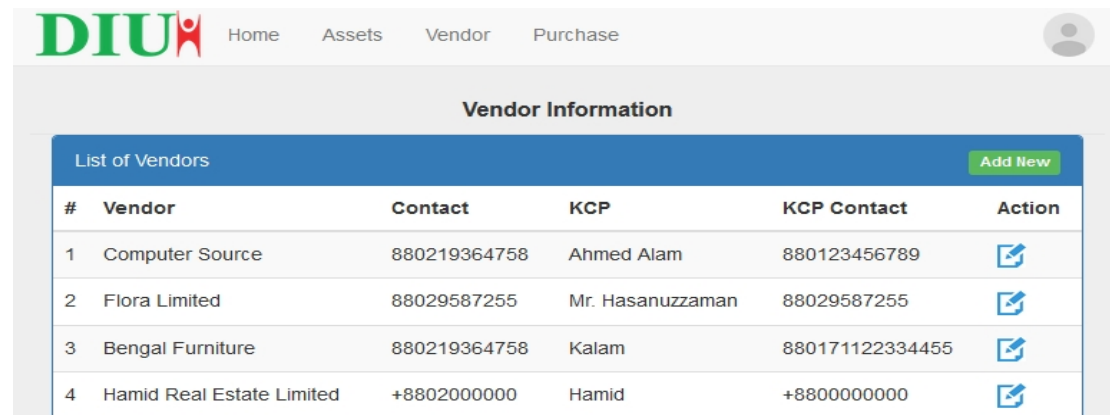
Model:

Model menu contain as a sub-menu of Assets menu shows in figure 7.8. Every asset items also have some brand wise model, user have to input this from here and also can get update list this menu. To see any item model, user first have to select item category, Item name, Brand and then have to enter Model or See the list of brand wise Models. At the beginning, user first see the category list, after select any category from the Select Category dropdown menu, user see the item list of the selected category. After select item name from Select Item dropdown menu, user can see the available brand name as a list. After select any brand, can see the available models of this item.

Figure 7.8: Model

Vendors:

From Vendor Status, you will find all available vendor information according to your Asset Category. A new vendor information can be adding, edit, view details and manage enlisted vendors for any organization or institute. Figure 7.9 shows Vendors information.
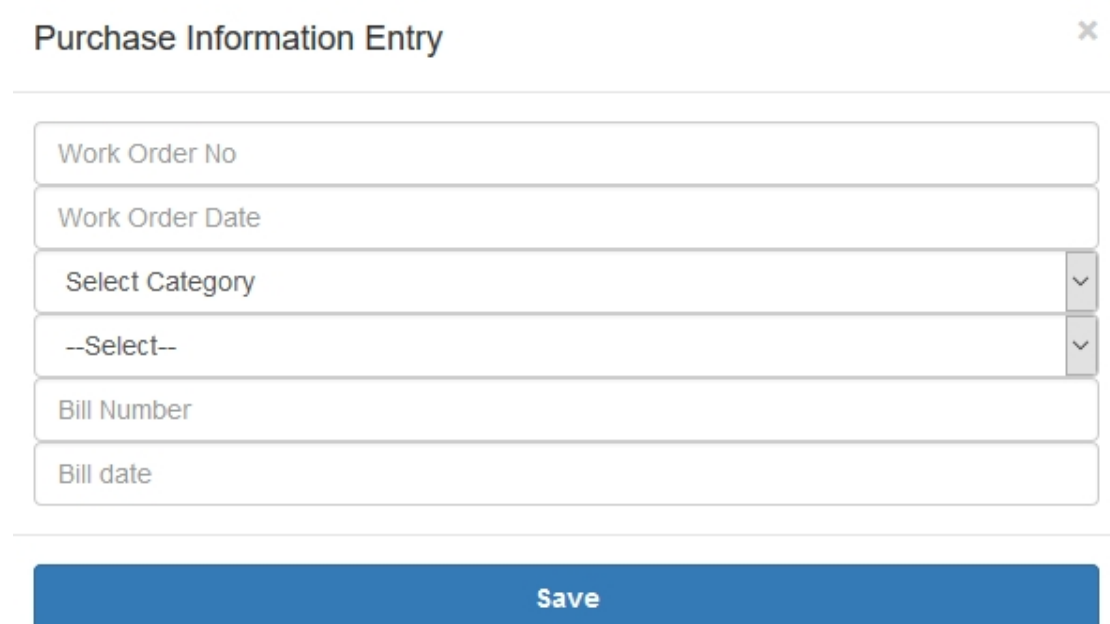
Figure: 7.9: Vendor

Purchase:

To maintain inventory system, user need to make stock or buy assets. For buying any assets, user first enter the work order number, bill number, assets type, then add assets / items to this system. Here every items show based on work order and bill number. Here figure 7.10 shows purchase information entry form.

Figure 7.10: Purchase information

First we add this basic information of any purchase. After that we select work order and bill number as combined for the dropdown menu, then we add the buying items as asset name, brand, model, serial/tag/holding no, unit price, quantity and then submit it by

click add button and we will find that all information added to the bellow table shown in figure 7.11.

Figure: 7.11: Purchase asset entry form
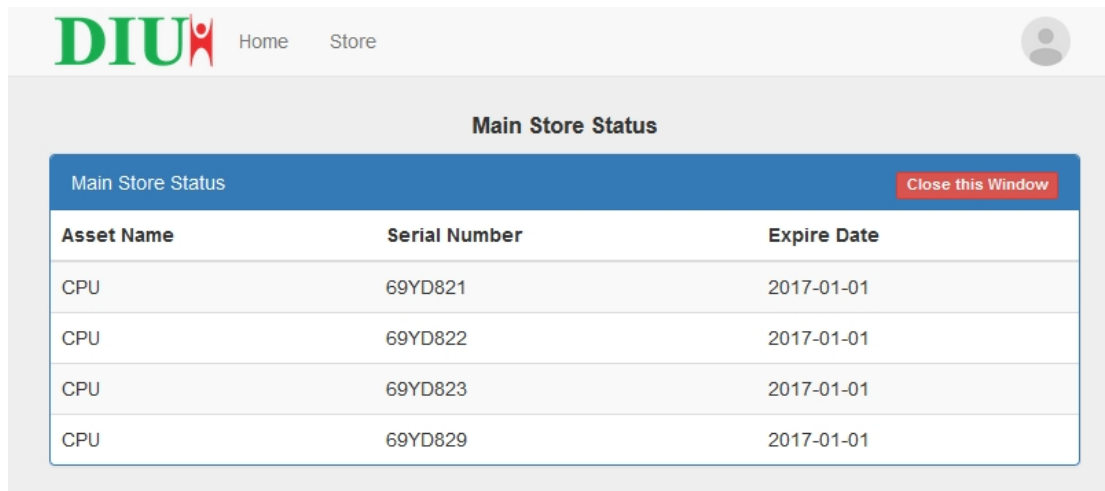
Store status:

From store status, store department user can easily find the current asset status. Here is a selection menu, after select the asset type from selection menu, user can get asset type wise status that show on figure 7.12. First, user can get item name wise status, then click on action menu, he can get details status of that item wise asset information. As an example, if the use clicks on action button of CPU, then he will find the current available CPU information like figure 7.13.

Figure: 7.12: Store Status
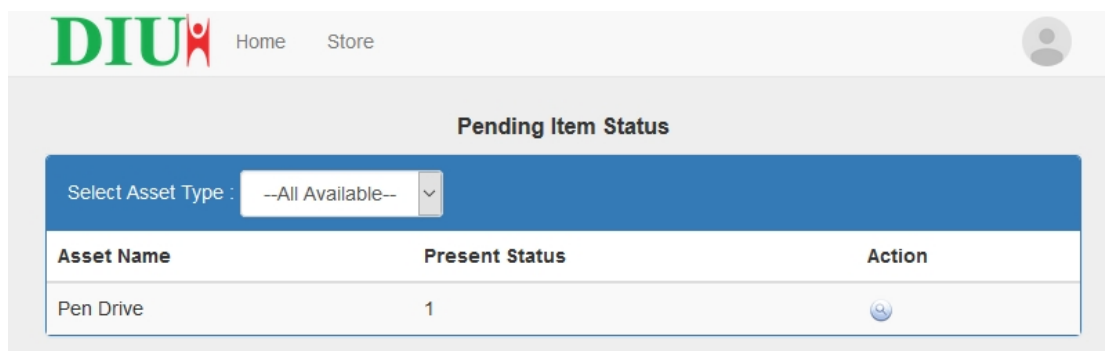
Figure: 7.13: Item wise store status

Store Pending:

From store pending status, store department can see the pending item status that purchase department already purchase and input to this software & send the items to store department. User will check the items that on pending state like shows on figure 7.14, after click on action button they can see the details about the item types as figure 7.15. They match the items with serial number and click on Received button to confirm.
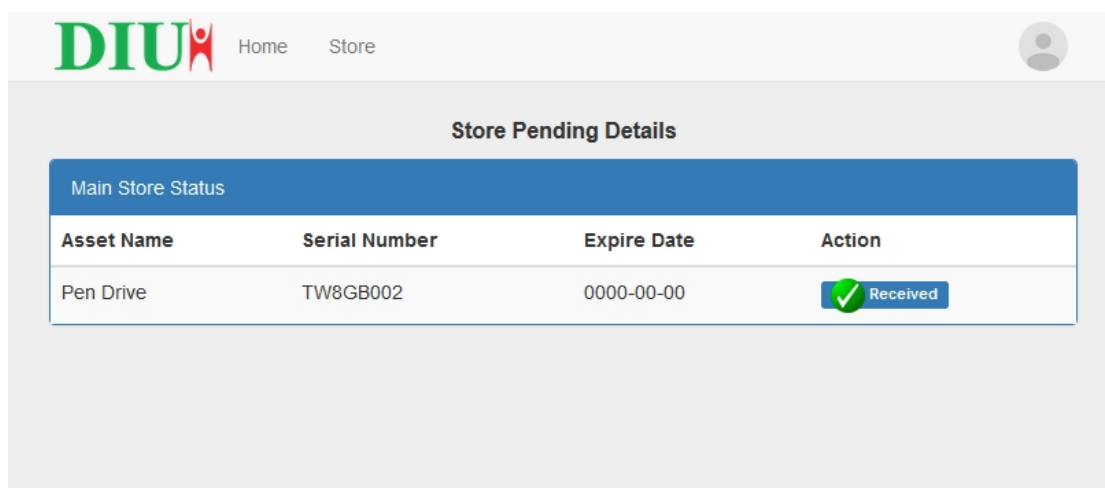
Figure 7.13: Pending Item name

Figure 7.15: Pending Item Details

Distribution:

To distribute any item, user have to visit Store Distribute menu. Here first user search using asset serial number and can view the details of the asset item. Then user select destination location form distribute to selection menu and click on Distribute to confirm distribution the item that show on figure 7.16

Figure 7.16: Item Distribution

Chapter 8

Conclusion

From the commercial aspect, these systems have a high value in software market. It is because this system well to do in universities and educational organization, government and private sector. Using this system, some organization can cut operation cost and use that money to add more products instead.

The Online Inventory System delivery needs preparation of constant analysis and submitting in the server to run completely. System will undoubtedly undergo change after it is delivered to the user. Change will occur because errors have been encountered, it is because the system must be adapted to accommodate changes in its external environments. Once the Online Inventory System is operational, technical maintenance, content management and updating, site visit activity reports, staff training and mentoring are needed on a regular basis depending on the complexity of the system.

This software has some limitations. Online requisition with approval system not instigated yet. We can be included online requisition system with approval from management, making purchase order and automatic send purchase order to the respective vendors. The system is artificially not so intelligent to automatically select best vendors. Intelligent system can be included to discover best vendor according to their given quotation and information. In future we will include SMS gateway to send automation mail and SMS notification system to notify any kind of change of inventory system. It's totally web based application, but currently most of the users have android smart phone, so we will make an android version of this software that user can easily access from anywhere. As we think a big market and want to cover on single platform, we will make a single cloud based platform for those organization which doesn't want to have their own server to maintain inventory management system.

Figures

References:

[1] Steven M. Bragg, "Inventory Accounting: A Comprehensive Guide", 1st Edition.

[2] http://hotelexecutive.com/business_review/2044/top-five-reasons-to-automate-inventory-and-procurement

[3] http://www.accountingtools.com/retail-inventory-method

[4] IEEE STD 1016-1998, IEEE Recommended Practice for Software Design Descriptions

[5] Paul DuBois, "MySQL (Developer's Library)", 5th Edition.

[6] Robin Nixon, "Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5", 4th Edition.

[7] Paul Deitel and Harvey M. Deitel, "AJAX, Rich Internet Applications, and Web Development for Programmers", 1st Edition.

[8] Steve Holzner, "Ajax Bible", 1st Edition.

[9] James Lewis, "Project Planning, Scheduling, and Control: The Ultimate Hands-On Guide to Bringing Projects in On Time and On Budget", Fifth Edition.

[10] https://en.wikipedia.org/wiki/Inventory_management_software

[11] IEEE STD 830-1998, IEEE Recommended Practice for Software Requirement Specification

[12] http://smallbusiness.chron.com/scope-limitations-inventory-systems-80920.html

[13] https://en.wikipedia.org/wiki/HTML

[14] https://en.wikipedia.org/wiki/PHP

[15] https://en.wikipedia.org/wiki/Graphical_user_interface

[16] https://en.wikipedia.org/wiki/PHP_Standard_Recommendation

[17] https://en.wikipedia.org/wiki/Operating_system

[18] https://en.wikipedia.org/wiki/Waterfall_model

[19] Alan Dennis, "Systems Analysis and Design", 5th Edition

[20] https://en.wikipedia.org/wiki/Test_plan

[21] Abraham Silberschatz, Henry F. Korth, "Database System Concept", 6th Edition.

Appendix – I