A Project On

Integrated Web-based Inventory Management System for Academic Institutions

A Project Report submitted to the Department of Computer Science and Engineering, Islamic University in partial fulfillment of the requirements for the degree of M.Sc. in Computer Science and Engineering under the Evening Program.

Submitted By

Mohammad Harun-Or-Rashid	Md. Musa Kalimullah
Roll: 23214131	Roll: 23214118

Supervised By

Dr. Md. Habibur Rahman
Associate Professor
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

ISLAMIC UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ISLAMIC UNIVERSITY, KUSHTIA, BANGLADESH

FEBRUARY 2025

Letter of Transmittal

February, 2025

To
Dr. Md. Habibur Rahman
Associate Professor
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

Subject: Submission of Project Report on **“Integrated Web-based Inventory Management System for Academic Institutions”**

Dear Sir,
It is with great pleasure that we submit our project report titled **“Integrated Web-based Inventory Management System for Academic Institutions”**, which has been prepared as a mandatory requirement for the completion of our M.Sc. in CSE (Evening) program. We have put forth our best efforts to ensure that the report is well-structured, informative, and meets the objectives of the study.

We are deeply grateful to you for providing us with the opportunity to work on such a relevant and insightful topic. This project has not only enhanced our technical skills but also provided us with valuable practical experience in the field of web-based systems and inventory management. We sincerely believe that the knowledge and experience gained from this project will be instrumental in our future academic and professional endeavors.

Thank you for your guidance and support throughout this project. We hope that the report meets your expectations.

Sincerely,

…………………………………
Mohammad Harun-Or-Rashid
Roll: 23214131
Group-A
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

………………………
Md. Musa Kalimullah
Roll: 23214118
Group-A
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

Declaration

We, the undersigned, hereby declare that the work presented in the project titled **"Integrated Web-based Inventory Management System for Academic Institutions"** is the result of the research and investigation carried out by us under the guidance and supervision of **Dr. Md. Habibur Rahman**, Associate Professor, Department of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh.

We further declare that no part of this project has been submitted or is being submitted elsewhere for the award of any degree, diploma, or certification.

Submitted By

……………………………………
Mohammad Harun-Or-Rashid
Roll: 23214131
Group-A
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

……………………….
Md. Musa Kalimullah
Roll: 23214118
Group-A
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

Certificate of the Supervisor

This is to certify that **Mohammad Harun-Or-Rashid** (Roll: 23214131) and **Md. Musa Kalimullah** (Roll: 23214118) has successfully completed their **M.Sc. in Computer Science and Engineering (Evening)** project report titled **“Integrated Web-based Inventory Management System for Academic Institutions”** under my supervision.

To the best of my knowledge, this report is their original work and has not been submitted elsewhere for any other purpose.

I extend my best wishes for their future success.

Approved By

……………………………

Dr. Md. Habibur Rahman
Supervisor & Associate Professor
Department of Computer Science and Engineering
Islamic University, Kushtia, Bangladesh.

DEDICATION

We dedicate this research project to our beloved family, whose unwavering support and love have been our greatest strength. To our parents, whose sacrifices, guidance, and encouragement have shaped our path—this achievement is as much yours as it is ours. To our friends, for their constant motivation and companionship. And to our supervisor, for their invaluable guidance and belief in us. We are deeply grateful to you all for being part of this journey.

ACKNOWLEDGMENT

We begin by expressing our profound gratitude to the Almighty, whose blessings and guidance have enabled us to complete this internship report successfully.

We are deeply thankful to **Dr. Md. Habibur Rahman**, Associate Professor in the Department of Computer Science and Engineering (CSE), for his invaluable support, mentorship, and encouragement throughout this project. His expertise, dedication, and insightful feedback have been pivotal in shaping our work and ensuring its successful completion. Dr. Rahman's patience, enthusiasm, and willingness to share his knowledge have been a constant source of inspiration for us.

We are particularly grateful for the opportunity to work on the project titled **"Integrated Web-based Inventory Management System for Academic Institutions"** under his guidance. His vast experience in information technology and his ability to provide practical solutions to challenges have been instrumental in the progress of this project.

We could not have asked for a better mentor, and we are truly honored to have had the chance to learn from him. His unwavering support and encouragement have made this journey both rewarding and memorable.

Abstract

Inventory Management Software plays a crucial role in academic institutions and corporate organizations. Many companies possess numerous assets but lack a proper system to track their condition or quantity, often requiring manual counting. Our software aims to digitize asset management by providing an efficient solution for tracking and managing inventory.

Our Inventory Management System offers real-time information on departmental assets, stock reports, and an employee request system for items such as laptops. The application includes an approval workflow, ensuring a structured process where employees receive items only after necessary approvals.

To enhance user experience, we have designed an interactive and user-friendly interface. The application is highly secure, allowing access only to authenticated users. Initially, the software was developed with essential features, focusing on usability and security.

This is a fully web-based application developed using the following technologies:

1. **Programming Language** – PHP
2. **Database** – MySQL
3. **UI Technology** – Bootstrap 4
4. **Framework** – Laravel (a widely used PHP framework)

The application's responsive design ensures accessibility via mobile browsers. Faculty members in the CSE department, for example, can obtain login credentials from the Inventory Manager and request items such as laptops, chairs, or tables. The Inventory Manager reviews requests and, if items are available, can approve them or escalate them to the Department Chairman for higher-level approval. Once approved, users can collect their requested items.

Additionally, the system features role-based access control. Admin users can create accounts and assign specific permissions based on roles, ensuring secure and structured access to system functionalities.

Abbreviations

DFD = Data Flow Diagram

PHP = Hypertext Preprocessor

MySQL = My Structured Query Language

IMS = Inventory Management System

MVC = Model-View-Controller

HTML5 = Hypertext Markup Language 5

CSS = Cascading Style Sheets

VS Code = Visual Studio Code

ERP - Enterprise Resource Planning

CSE = Computer Science and Engineering

UAT = User Acceptance Testing

DB = Database

PK = Primary Key

FK = Foreign Key

UI = User Interface

ERD = Entity-Relationship Diagram

UX = User Experience

UML = Unified Modeling Language

CONTENTS

CHAPTER 1

INTRODUCTION

The Inventory Management System is a web-based application designed to streamline and digitize the inventory management process for organizations and educational institutions, such as the CSE department at IU. Accessible from any computer or mobile device via a browser, the application offers a user-friendly interface tailored to meet the needs of its users. The primary user roles include **Teacher**, **Inventory Manager**, **Department Chairman**, and **Admin**, each with specific permissions and responsibilities.

This system enables users to easily track and manage inventory, view available stock, and make informed decisions about purchasing new items. It also helps prevent theft by providing detailed information about stock items and supporting theft-prevention measures. With a simple database search, users can quickly locate available items in stock, eliminating the need for manual counting and reducing errors.

The application is particularly beneficial for organizations with numerous assets, as it provides real-time information about the condition and quantity of assets. It digitizes the entire asset management process, allowing employees to request items (e.g., laptops, chairs, or tables) through the system. Each request goes through an approval process, ensuring transparency and accountability. Once approved, the requested item is allocated to the employee.

To ensure a seamless user experience, the application features an interactive and responsive UI design, making it accessible and easy to use on both desktop and mobile devices. Security is a top priority, and only authenticated users with proper credentials can access the system.

Key Features:

1. Role-Based Access:
   - **Teachers** can request items after obtaining their login credentials from the Inventory Manager.
   - **Inventory Managers** can view and approve requests or forward them to the Department Chairman for higher-level approval.
   - **Admins** can create users, assign roles, and grant specific permissions to access certain menus.
   - **Department Chairmen** oversee higher-level approvals and decision-making.

2. Stock Management:
    - View real-time stock reports.
    - Track available items and their conditions.
    - Generate insights for inventory replenishment.
3. Request and Approval Workflow:
    - Employees can submit item requests.
    - Requests are routed through an approval process (Inventory Manager → Department Chairman, if necessary).
    - Approved requests result in item allocation.
4. Technology Stack:
    - **Programming Language**: PHP
    - **Database**: MySQL
    - **UI Framework**: Bootstrap 4
    - **Backend Framework**: Laravel

The application is designed with scalability in mind, starting with essential features and allowing for future enhancements. Its responsive design ensures compatibility across devices, while its robust security measures protect sensitive data. By digitizing inventory management, this system saves time, reduces errors, and improves overall efficiency for organizations and institutions.

CHAPTER 2

BACKGROUND & LITERATURE REVIEW

2.1 BACKGROUND TECHNOLOGIES

Inventory management is a critical component of organizational operations, ensuring the efficient tracking and distribution of assets. An effective inventory management system (IMS) digitizes asset management processes, enabling organizations to monitor stock levels, generate reports, and streamline asset distribution. The proposed inventory software provides functionalities such as tracking the number of items in each department, generating stock reports, and facilitating employee requests for assets (e.g., laptops) through an approval process. By automating these tasks, the system enhances productivity, reduces manual errors, and improves the overall quality of asset management.

The development of the Inventory Management System (IMS) leverages the following technologies:

a) Programming Language:
PHP is the primary programming language used for this project. PHP is a widely adopted open-source scripting language known for its flexibility, extensive community support, and availability of open-source tools and libraries. It is particularly suited for web-based applications, making it an ideal choice for developing the IMS.

b) Framework:
The PHP Laravel MVC Framework is utilized for building the application. Laravel is a robust and secure web application framework that simplifies development through its elegant syntax, built-in features, and extensive library support. It includes an Artisan command-line interface for efficient application management and adheres to the Model-View-Controller (MVC) architectural pattern, ensuring modular and maintainable code.

c) Database:
MySQL is employed as the relational database management system (RDBMS). MySQL is a popular open-source database solution known for its reliability, scalability, and compatibility with web-based applications. It efficiently handles data storage, retrieval, and management, making it a suitable choice for the IMS.

d) Frontend Technologies:

The frontend of the application is developed using HTML5, CSS3, Bootstrap 4, and JavaScript. These technologies ensure a responsive, user-friendly interface that is compatible with various devices and browsers. Bootstrap 4, in particular, provides pre-designed components and a grid system for creating modern and visually appealing layouts.

e) Code Editor:

Visual Studio Code (VS Code) is the chosen code editor for this project. VS Code is a lightweight yet powerful tool optimized for building and debugging modern web and cloud applications. Its extensive library of extensions and integrations enhances developer productivity.

f) Operating System:

The application is designed to be compatible with multiple operating systems, including Windows, Linux, and macOS, ensuring broad accessibility for users and developers.

Design Pattern: Model-View-Controller (MVC)

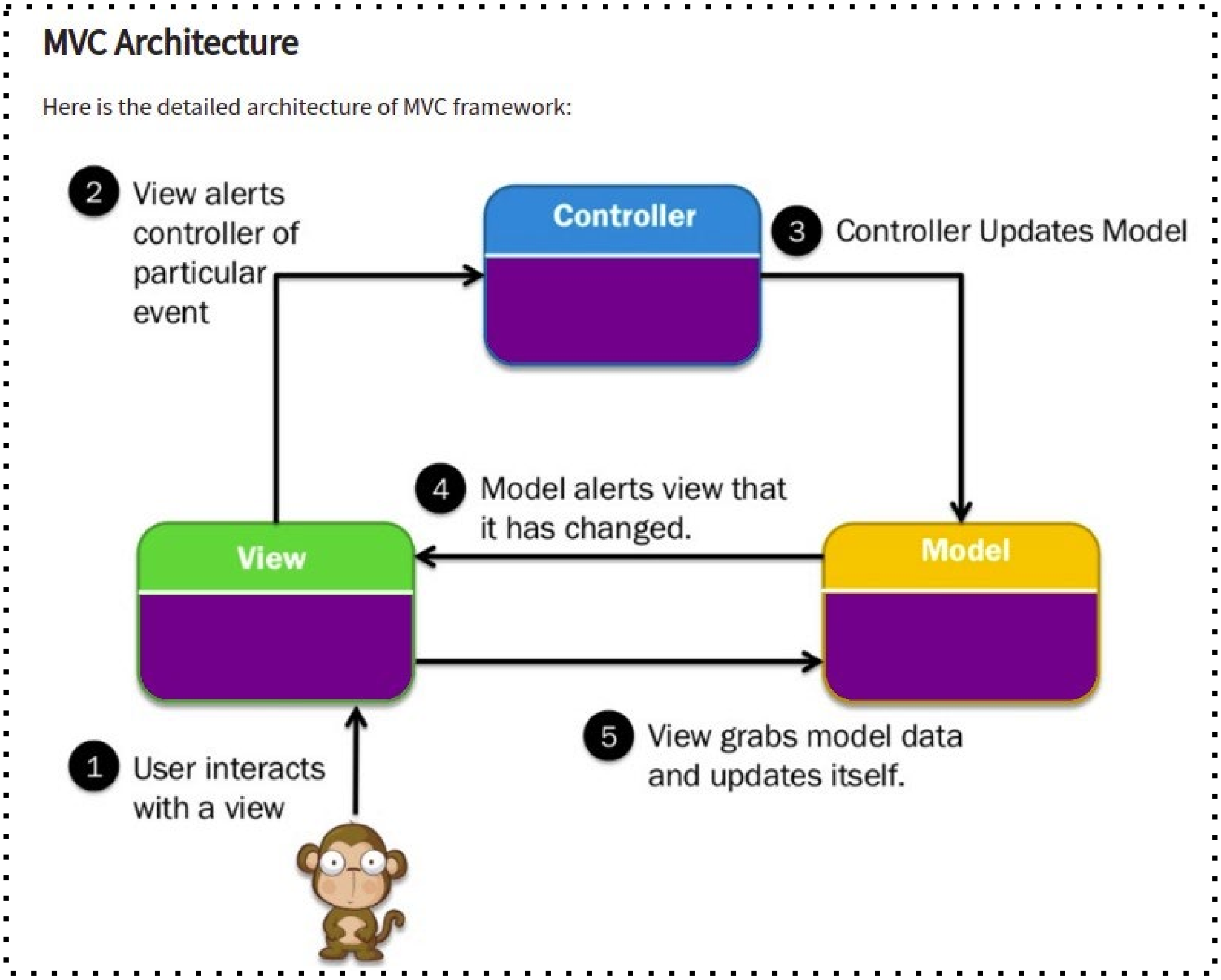
Selecting the appropriate design pattern is crucial for developing a scalable, maintainable, and reusable application. The Model-View-Controller (MVC) architecture is widely recognized as one of the best design patterns for web-based applications. It separates the application into three interconnected components:

- **Model**: Manages the data and business logic of the application.
- **View**: Handles the presentation layer and user interface.
- **Controller**: Acts as an intermediary between the Model and View, processing user input and updating the Model accordingly.

The MVC pattern is particularly advantageous for PHP-based applications, as it promotes code modularity, reusability, and ease of refactoring. Laravel, the chosen framework for this project, inherently supports the MVC architecture, making it an ideal choice for developing the IMS.

By adopting the MVC design pattern, the IMS ensures a clean separation of concerns, enabling developers to manage complex functionalities efficiently. This approach not only enhances the application's maintainability but also facilitates future scalability and integration with additional features or modules.

In summary, the proposed Inventory Management System leverages modern technologies and best practices, including PHP, Laravel, MySQL, and the MVC design pattern, to deliver a robust, secure, and user-friendly solution for organizational asset management.



MVC Architecture

Here is the detailed architecture of MVC framework:

2 View alerts controller of particular event
Controller
3 Controller Updates Model
4 Model alerts view that it has changed.
View
Model
5 View grabs model data and updates itself.
1 User interacts with a view

Figure 2.1: MVC Architecture Diagram.

The View

The View is responsible for presenting data to the user in a visually understandable format. It defines how the information is displayed, such as through text boxes, dropdown menus, or other UI elements. For example, in an item information view, the item name might be displayed in a text box, while the item category could be shown in a dropdown menu.

The Controller

The Controller acts as the intermediary between the user, the View, and the Model. It handles user interactions, processes input, and manages the flow of data. When a user performs an action, the Controller requests the necessary data from the Model and then passes that data to the View for presentation.

The Model

The Model represents the application's data and business logic. It interacts with the database to retrieve, store, and manipulate data. The Model is responsible for enforcing the rules and logic of the application. When the Controller requests data, the Model provides it, ensuring that the information is accurate and consistent.

2.2 Literature Review

An Inventory Management System helps track and prevent item theft by providing valuable insights into stock levels and the need for security measures. With an efficient inventory system, users can quickly check item availability through a simple database search.

Many corporations own numerous assets but often lack visibility into their condition or exact quantity. Without an automated system, they must manually count their assets, which is time-consuming and prone to errors.

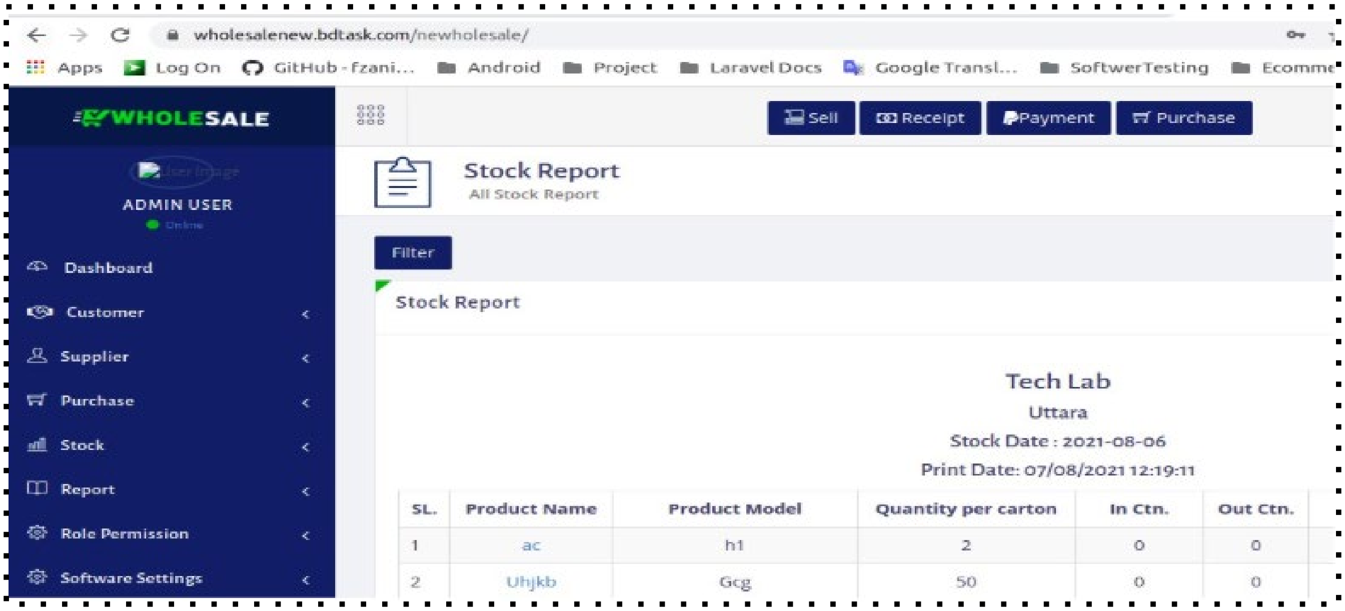
Inventory management is a critical concern for every organization, ensuring efficient asset tracking and resource optimization. Below are some key features of existing Inventory Management Systems.

1. BDTask Inventory Management System

BDTask is a Bangladeshi software company that offers an advanced Inventory Management System with various features, including:

- **User Management** – Manage users with role-based access control.
- **Role and Permission Management** – Define roles and set permissions for different users.
- **Customer Management** – Maintain customer records efficiently.
- **Supplier Management** – Track and manage supplier information.
- **Stock Management** – Monitor inventory levels and stock movements.
- **Sales Management** – Handle sales transactions and reports.

The following figure illustrates a stock report generated by the BDTask Inventory Management System.



Stock Report
All Stock Report

Stock Report

Tech Lab
Uttara
Stock Date : 2021-08-06
Print Date: 07/08/2021 12:19:11

SL.	Product Name	Product Model	Quantity per carton	In Ctn.	Out Ctn.
1	ac	h1	2	0	0
2	Uhjkb	Gcg	50	0	0

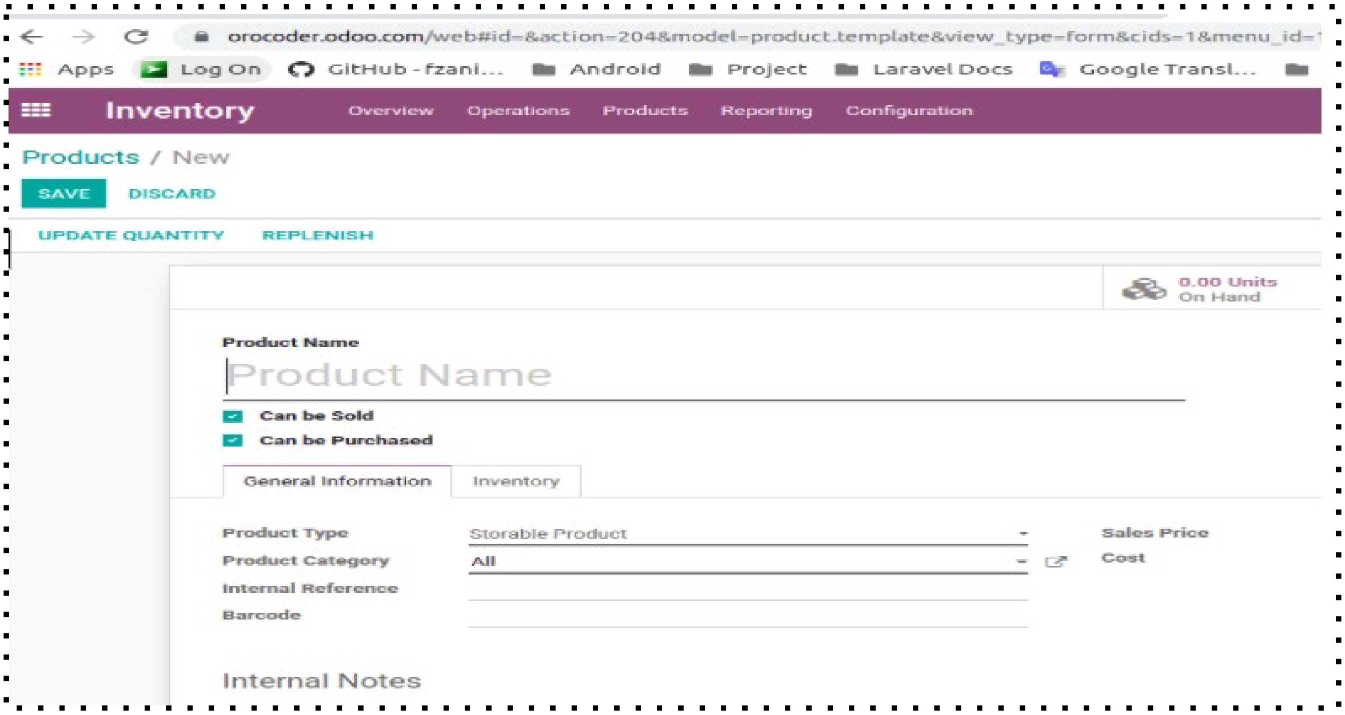
Figure: bdtask Inventory Management System

2. Odoo ERP Inventory Management System

Odoo is a comprehensive suite of business management software that includes a powerful Inventory Management System. Its key features include:

- **Product Management** – Efficiently manage product details and stock levels.
- **Inventory Adjustment** – Modify inventory counts to reflect actual stock.
- **Product Transfer** – Track and manage product movement between locations.
- **Warehouse Management** – Organize and optimize warehouse operations.
- **Product Categorization** – Classify products for better organization.
- **Comprehensive Reports** – Generate various reports for data-driven decision-making.

The following figure illustrates the Product Entry page in the Odoo ERP Inventory Management System.



Product Name

Product Name

☑ Can be Sold

☑ Can be Purchased

Product Type	Storable Product	Sales Price
Product Category	All	Cost
Internal Reference		
Barcode		

Internal Notes

Figure: Odoo ERP Inventory Management System

3. BASE IT Inventory Management System

BASE IT is a leading IT firm in Bangladesh that offers a comprehensive Inventory Management System. Its key features include:

- **Product Management** – Categorize and manage product entries.
- **Purchase Requisition** – Streamline purchase requests and approvals.
- **Purchase Payment** – Track and manage purchase payments.
- **Purchase Overview** – View all purchase records in one place.
- **Due Payment Management** – Monitor and manage outstanding payments.
- **Sales Tracking** – Maintain detailed records of all sales transactions.
- **Stock Reports** – Generate real-time inventory reports.
- **Branch Sales Reports** – Track sales details across different branches.
- **Comprehensive Reporting** – Access various reports for better decision-making.

This system helps businesses efficiently manage inventory, purchases, and sales while ensuring transparency and accuracy.

Many corporate companies have many assets but they don’t know what is the condition of their assets or how many assets are there. For this they need to manually count their assets.

Some organization asset management system are given below:

1. TechnoVista Ltd – A Software Company

TechnoVista Ltd owns a wide range of assets, including computers, printers, photocopy machines, laptop bags, chairs, tables, and more. However, the company currently relies on a manual system, such as notebooks, to track its inventory.

This outdated method makes it difficult to monitor asset usage. For example, during the COVID-19 period, when employees took laptops home for remote work, there was no centralized system to track who took which device.

Without an integrated Inventory Management System, the company lacks real-time visibility into the condition, location, and allocation of its assets, leading to inefficiencies and potential mismanagement.

2. Department of CSE, IU

The Department of Computer Science and Engineering (CSE) at Islamic University currently lacks an integrated Inventory Management System to efficiently track and manage their assets. At present, the department relies on manual processes to count and allocate items such as laptops, desktops, furniture, and other computer accessories to teachers and employees.

By adopting an Inventory Management System, the department could streamline this process and easily generate reports on asset distribution with just one click, saving time and reducing errors. This system would improve overall asset management and enhance operational efficiency.

Benefits of using Our Inventory Management System (IMS):

Our Inventory Management System (IMS) is designed to simplify and optimize inventory tracking with a range of powerful features, including:

- **Dual Language Support** – Available in both **English** and **Bangla**.
- **Dynamic Configuration** – Customize item categories and inventory settings.
- **Purchase Management** – Efficient tracking of item purchases.
- **Item Requests** – Employees, teachers, or departments can request items.
- **Approval Workflow** – Ensures proper authorization before item issuance.
- **Stock Reporting** – Generate real-time stock reports.

Asset and Allocation Management

- **Room-Wise Stock Reports** – Track inventory based on asset location.
- **Item Allocation Tracking** – Monitor item distribution to teachers and employees.
- **Return Management** – Manage item returns to the custodian.
- **Approval Flow for Allocations** – Proper authorization for allocations to teachers, students, or employees.

Supplier and Employee Management

- **Supplier Information** – Store and manage supplier details.
- **Purchase Records** – Maintain a history of purchases linked to suppliers.
- **Employee Information** – Track item allocations for each employee.

Reporting & User Experience

- **Comprehensive Reports** – Export reports in PDF, Word, or Excel format.
- **User Activity Log** – Track all system interactions for transparency.
- **Interactive User Interface** – User-friendly and easy to navigate.

And many more features to enhance efficiency, security, and control over inventory management!

CHAPTER 3

SYSTEM ANALYSIS

3.1 PROBLEM STATEMENT

3.1.1 The Problem: Many academic institutions have many assets but they don’t know what is the condition of their assets or how many assets are there. For this they need to manually count their assets.

For example, A company has many assets such as Computer, Printer, Photocopy Machine, laptop bag, Chair, Table and many more assets. But they maintain manual systems such as notebooks to count their assets. Even If anyone take any item such as laptop for COVID situation in home for Work from home office, They don’t know who took which laptop.

They have no Integrated Inventory Management System to quickly know the condition of their assets. Many companies or Educational institutions like the CSE department, IU have the same situation. In CSE Department, IU has no Integrated Inventory Management System to quickly know the Stock report , asset distribution status

3.1.2 The Solution: The Inventory Management System is complete solution to manage asset distribution flow and It has various report to know the current stock status report, Item Information, Item requisition etc.

3.1.3 Stakeholders: Inventory Manager, Employee, Teacher of CSE department.

3.1.4 User Roles: Multiple types of user roles have to be created for the better accessing of the end user through the system.

3.1.5 Risks: Unclear about the following issues: what constitutes the latest technologies?

- What will happen for the Multiple department stock report?
- What will happen if we provide the wrong information?
- How to dispose of a bad condition item?
- How to notify for new Item request
- To the system cannot work

3.1.6 Assumptions: Some obvious **assumptions**

- Inventory Manager may inexpert, Intermediate and expert person
- Item category Content
- Users of Inventory Management Systems have an alternative way when the system will not work properly.
- For the (chosen) solution to work, the following assumption is required even though we have no control of it.
  - If an unwanted situation is created to the system, then it must be restored to the system for users.
- Check the validity of the above assumptions.

3.2 SOLUTION ATTRIBUTES

- For sound knowledge for using web-based application
- Participate in UAT.
  - Supply UAT training materials (Manual, video etc.)
  - Admin can entry for an item request to the system on behalf of a teacher
- Characteristics of the provided information
  - Monitoring dashboard will help to the authority of Inventory Management System application
  - Simple and Clear
- All of these attributes reflect *decisions* made by management team and authority of sound system (i.e. user) or by the solution developer

3.2.1 Vision of the Solution: Vision of the solution objectives are.

- Eliminate Redundant Work
- Data Sharing instantly everywhere.
- Increase Accuracy through validation & instruction.
- Personal Assistance using reminder & alert

3.2.2 Vision statement: Inventory Management System is passionate about excellence asset management system where automation system will address the overall process of asset distribution through the dedicated features.

3.2.3 Vision Statement Template:

It aims to Cover the essence of the new system Market the software Keep the developers focused on the "core" essence of the system It should be brief

The various parts of a general vision statement are:

- **For** (target user, audience)
- **Who** (statement of the need or opportunity)
- **The** (product name) is a (product category)
- **That** (key benefit, compelling reason to build/buy)
- **Unlike** (primary competitive alternative)
- **Our product** (statement of primary differentiation)

Vision Statement Template for Inventory Management System:

For	Academic Institutions
Who	Have difficulty in asset distribution
The Integrated Web-based Inventory Management System for academic institution	is a Web based application
That	provides the ability to distribute Items among employee
Unlike	Currently available systems that have poor interface or many organizations have no integrated system.
Our product	Our Inventory Management System is custom dynamic and Interactive user interface to properly distribute Item among Teachers or other employee and can find stock report

Vision statement template of Inventory Management System

3.3 Types of Requirements

Functional requirements:

These requirements describe the core features and functionalities of the application. All the requirements across the three phases of the system development are considered functional requirements.

Nonfunctional requirements:

These requirements specify the performance, reliability, and behavior of the system, including:

- **Response Time:** The system must respond within 2 seconds.
- **Transaction Confirmation:** The system should display a confirmation message after each transaction.
- **Database Handling:** The system should be optimized to handle large databases efficiently.

Constraints (“Pseudo requirements”):

These are limitations or specifications imposed by the client or the environment in which the system will operate:

- The implementation language must be **PHP**.

3.4 Users

- Admin
- Inventory Manager
- Teacher
- Department Chairman and any other employee of department

3.5 List of Features of Inventory Management System

The Inventory Management System (IMS) is equipped with a comprehensive set of features designed to streamline inventory processes and enhance user experience. Below is a detailed list of the system’s key features:

1. **Dynamic User Management System:** Enables efficient management of user accounts and access.
2. **User Role and Permission Management System:** Allows administrators to define roles and permissions for secure and controlled access.
3. **Dynamic Setup Data Configuration System:** Provides flexibility in configuring system settings and data.
4. **Employee Information Management:** Stores and manages detailed information about employees.

5. **Responsive Design:** Ensures the system is accessible and functional across various devices and screen sizes.
6. **Login System:** Provides secure access to the system for registered users.
7. **Sign-Out Functionality:** Allows users to securely log out of the system.
8. **Forget Password Feature:** Enables users to reset their passwords in case they forget them.
9. **User Profile Management:** Allows users to view and update their personal information.
10. **Item Categories Configuration:** Facilitates the creation and management of item categories.
11. **Dynamic Item Configuration:** Enables customization of item details to meet organizational needs.
12. **Supplier Information Management:** Maintains records of suppliers and their details.
13. **Item Purchase from Supplier:** Tracks and manages purchases made from suppliers.
14. **Item Request System:** Allows employees to request items through the system.

15. **Requested Item Approval Process:** Ensures proper authorization before items are issued.
16. **Allocation of Requested Items to Employees:** Tracks items allocated to specific employees.
17. **Return Item to Inventory Manager:** Manages the process of returning items to the inventory.
18. **Stock Report:** Provides real-time information on stock levels and inventory status.
19. **Asset Location (Room) Wise Item Stock Report:** Tracks stock based on specific locations or rooms.
20. **User Activity Report:** Monitors and logs user activities for accountability and security.
21. **Downloadable Reports:** Allows all reports to be exported in PDF, Word, or Excel formats for easy sharing and analysis.
22. **User-Friendly Filter System:** Enables quick and efficient searching of items using advanced filters.
23. **Interactive UI Design:** Offers a visually appealing and intuitive user interface for seamless navigation.

Scope of phased release: There is three-phase release to the Inventory System. These are

Phase 1	**Phase 2**	**Phase 3**
1. Inventory Management System template design. 2. Completed database design. 3. Fixing sidebar for different type of role 4. Responsive design 5. Registration 6. Login 7. Sign out 8. Forget password	1. Employee information setup. 2. User's role and permission system setup 3. Both Bangla and English language switcher in whole application 4. Fixing Architecture 5. Main dashboard design. 6. Develop user activity report. 7. Common setup data configuration setup 8. Item categories setup 9. Developed Item setup 10. Supplier information setup.	1. Developed Item receive from supplier scope. 2. Developed Item request system 3. Developed Approval process on requested item. 4. Developed Item receive scope. 5. Developed Item return scope 6. Developed Stock report 7. Asset location wise stock report 8. Bug fixing 9. Incorporate feedback.

CHAPTER 4

SYSTEM DESIGN

4.1 COLLECTING DATA IN THE PROBLEM DOMAIN

To gain a comprehensive understanding of the challenges associated with the Inventory Management System, the following techniques can be employed:

- **Analysis of Asset Management Systems**: Study existing asset management systems in various organizations to identify common issues and best practices.
- **Questionnaires**: Conduct surveys to gather insights from stakeholders, including administrators, inventory managers, and end-users.
- **Prototype Experimentation**: Develop and test a prototype to evaluate its functionality and identify areas for improvement.
- **Observation**: Observe current inventory management processes to understand pain points and inefficiencies.
- **Document Inspection**: Review existing documentation, such as inventory logs and reports, to analyze current practices.
- **User Stories**: Collect user stories to understand the needs and expectations of different user roles within the system.

These techniques will help in defining the problem domain and designing a system that addresses the specific requirements of inventory management.

4.2 FORMAT OF USER STORY

The User Story format has been chosen to capture the requirements of the Inventory Management System. Below are examples of user stories that outline the system's functionality:

1. **Admin Access:**
   - As an Admin, I need access to the dashboard so that I can monitor system data effectively.
2. **User Login:**
   - As a registered user, I am required to log in so that I can access the system securely.
3. **Password Recovery:**
   - As a forgetful user, I can request a password reminder so that I can log in if I forget my password.
4. **Item Request:**
   - As a teacher, I want to request an item (e.g., laptop, chair) so that I can use it for my work.

5. **Stock Management:**
    - As an Inventory Manager, I want to receive purchased items from suppliers so that I can update the stock report accurately.
6. **Stock Reporting:**
    - As a department chairman, I want to view the stock report so that I can understand the current distribution of assets.
7. **Item Category Setup:**
    - As an Admin, I want to add item categories so that I can use them during item setup.
8. **Item Information Management:**
    - As an Admin, I want to add item details so that I can generate item-wise reports.

These user stories provide a clear and structured way to define the system's features and ensure that the needs of all user roles are addressed.

4.3 Data Flow Diagram (DFD)

Figure 4.1 shows the data flow diagram of the system. Here we see that Inventory managers purchase products or items from suppliers and at a time stock is increased. Then an employee request for an item then inventory manager approved item

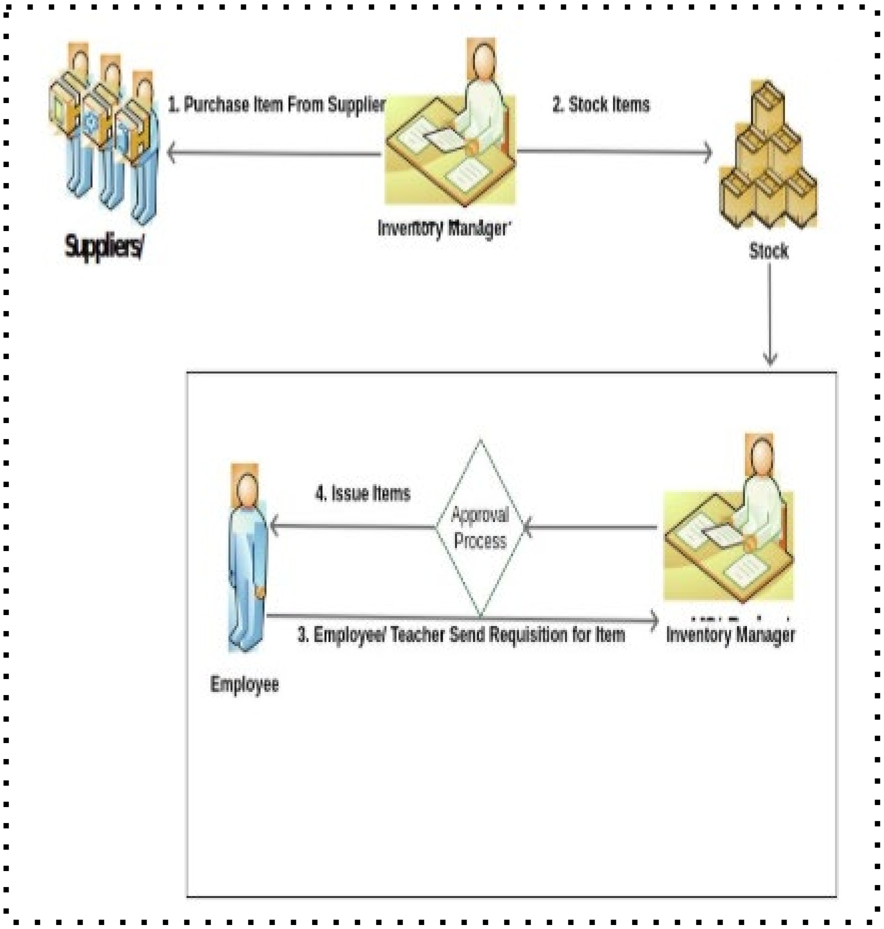


Figure 4.1: DFD diagram of Item Purchase and Item Request by Teacher

The figure 4.2 shows the full inventory management system features with data flow diagram.

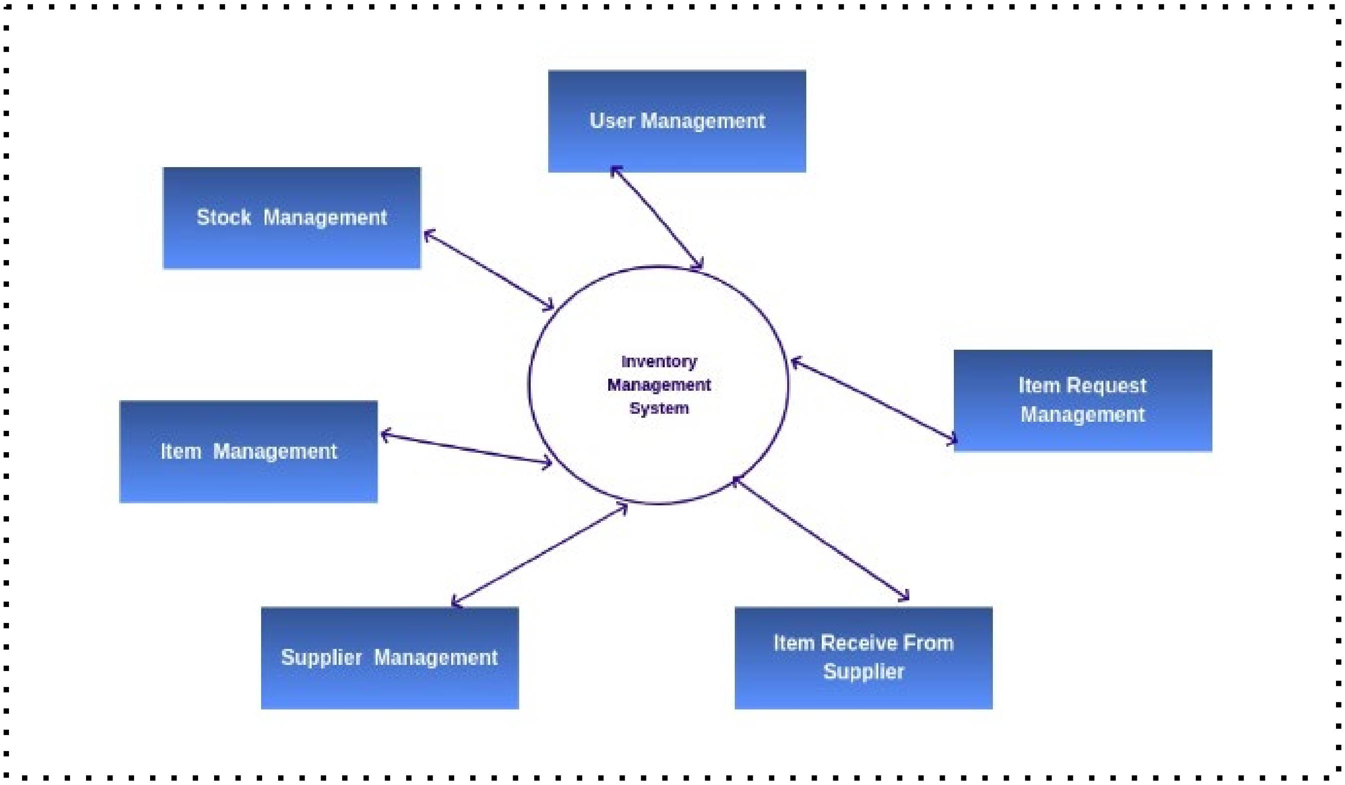


Figure 4.2: DFD diagram of Full Inventory System

4.4 Use Case Diagram for Inventory System

The figure 4.3 shows the Use Case diagram with a set of use case and actors and their relationships for Inventory Management System

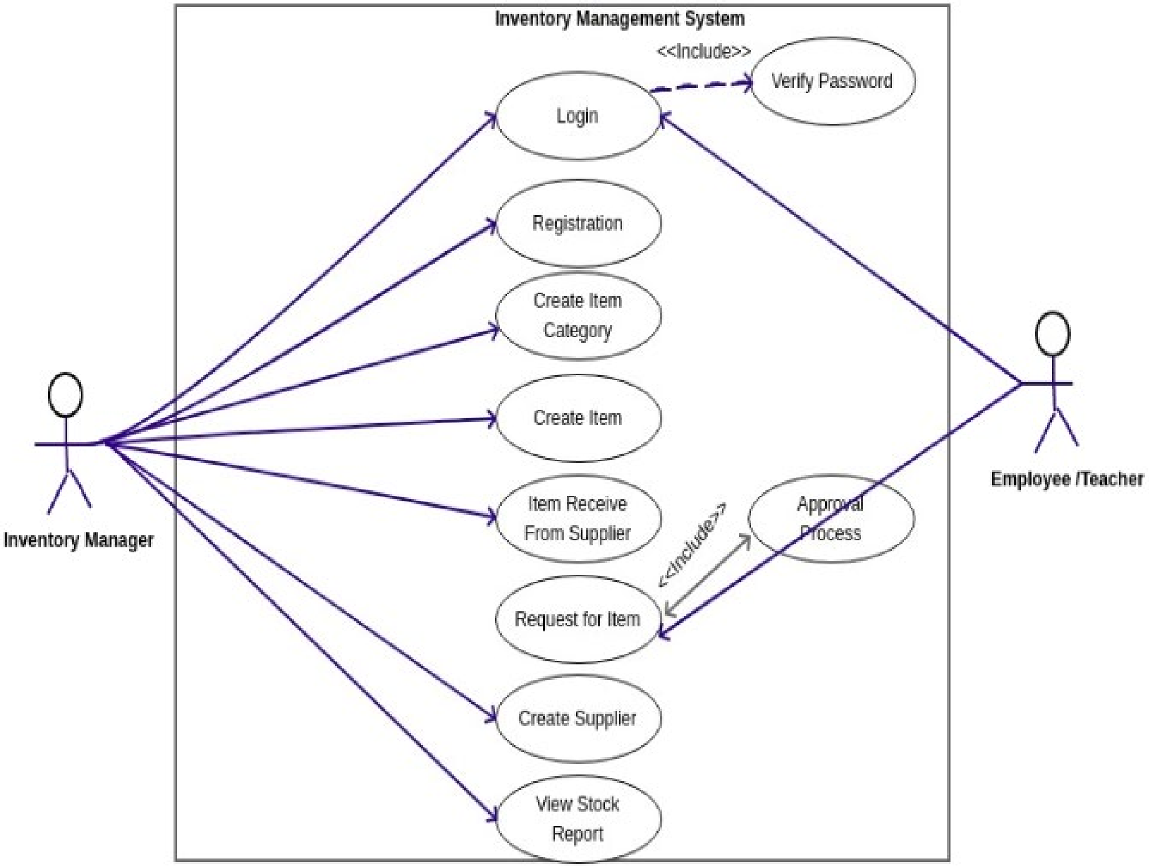


Figure 4.3: Use Case Diagram

4.5 Database Design of Inventory Management System

Database design is the most important part of system design. All activities of users store by using database.

4.6 DATA DICTIONARY OF INVENTORY MANAGEMENT SYSTEM Table

Table Name : users

Primary Key : id

Name of Attribute	**Description** ***(field level Business Rule)***	**Data Type** ***(Field Size)***	**Mandatory/ Optional**
Name English	Text	Text (100)	M
Name Bangla	Text	Text (100)	M
Login Id	Text	Text (20)	M
Password	Encrypted/Hashed text, validated with confirm password. Password must contain combination of [Aa-Zz], [0-9] and at least one special character	Text (100)	M
Email	Valid Email Address	Text (50)	M
Mobile No.	Text	Text (20)	O
Is Active	Yes or No	Boolean	M
User Level	Text	Text (100)	O
Designation	Derived from designation setup	int	O
Location	Asset Location. Ex. Room No 1	int	O
Blood Group	Blood group	Text (100)	O
Role [Multiple]			
Role Id	Derived from Role	Number (FK)	O

Table Name : inv_item_information

Primary Key : id

Name of Attribute	Description *(field level Business Rule)*	Data Type *(Field Size)*	Mandatory/ Optional
Name (নাম)	Unique text	Text (100)	M
Name in Bangla (নাম বাংলায়)	Text	Unicode Text (100)	O
Code (কোড)	Unique text like: "IU-001"	Text (20)	M
Code in Bangla (কোড বাংলায়)	Text	Unicode Text (20)	O
Asset Type	Derived[cc] Fixed Asset, Accessories, Consumable/ Moveable	Number (FK)	M
Category	Derived from Category	Number (FK)	M
UoM	Derived[cc] from Unit of Measurement	Number (FK)	O
Manufacturer	Derived[cc] from Manufacturer	Number (FK)	O
Model	Text	Text (100)	O
Part Number	Text	Text (100)	O
Minimum Re-order qty.	Minimum re-order quantity	Number (10)	O
Is Serialized	Default false, if true serial is required for all further transactions	Boolean	M
Is Active	Default true, if false will not available in further "Add New" transactions	Boolean	M
Remarks	Text	Text (200)	O

Table Name : inv_item_category_sub_category_information

Primary Key : id

Name of Attribute	**Description** ***(field level Business Rule)***	**Data Type** ***(Field Size)***	**Mandatory/** **Optional**
Name (নাম)	Unique text like "Computer, Air Condition (A/C)"	Text (100)	M
Name in Bangla (নাম বাংলায়)	Text	Unicode Text (100)	M
Code (কোড)	Unique text like: "AC"	Text (20)	O
Code in Bangla (কোড বাংলায়)	Text	Unicode Text (20)	O
Remarks	Text	Text (200)	O

Table Name : inv_item_receive_from_supplier_information

Primary Key : id

Name of Attribute	**Description** ***(field level Business Rule)***	**Data Type** ***(Field Size)***	**Mandatory/** **Optional**
Receive ID	Unique, auto generated from DB-PK	Text (100)	M
Receive Date	Date	Date	M
Supplier	Derived from Supplier	Number (FK)	M
PO Number	Either Package-Lot number or unique PO if purchased directly without Package [Optional for Accessories]	Text (20)	M
Invoice Number	Text	Text (50)	O
Invoice Date	Date	Date	O

Received By	Logged in User Id	Number (FK)	O
Remarks	Text	Text (200)	O
Item Detail Information [Multiple]			
Item Id	Derived Item from Package- Lot or Search Items UI	Number (FK)	M
Item Status	Derived from Item Status, default value is 1=Good (Is-Good)	Number (FK)	M
Quantity		Number (10)	M
Serial	Mandatory for Fixed Asset and Is-Serialized items	Text (50)	O
Fixed Asset ID		Text (50)	O
Remarks		Text (100)	O
Update Current Stock			

Table Name : inv_request_items

Primary Key : id

Name of Attribute	**Description** ***(field level Business Rule)***	**Data Type** **(Field *Size)***	**Mandatory/ Optional**
Receive ID	Unique, auto generated from DB-PK	Text (100)	M
Request Date	Date	Date	M
Receive Date	Date	Date	M
Location	Derived from Asset location setup	Number (FK)	M
is_requested	0 or 1	Tinyint (4)	M
is_approved	0 or 1	Tinyint (4)	M
is_received	0 or 1	Tinyint (4)	M
Requested By	Logged in User Id	Number (FK)	O
Approved By	Logged in User Id	Number (FK)	O
Received By	Logged in User Id	Number (FK)	O
Remarks	Text	Text (200)	O
Item Detail Information [Multiple]			
Item Id	Derived Item from Package-Lot or Search Items UI	Number (FK)	M
Item Status	Derived from Item Status, default value is 1=Good (Is-Good)	Number (FK)	M
Quantity		Number (10)	M
Serial	Mandatory for Fixed Asset and Is-Serialized items	Text (50)	O
Fixed Asset ID		Text (50)	O
Remarks		Text (100)	O
Update Current Stock			

Table Name : inv_suppliers

Primary Key : id

Name of Attribute	Description *(field level Business Rule)*	Data Type *(Field Size)*	Mandatory/ Optional
Name (নাম)	Unique text	Text (100)	M
Name in Bangla (নাম বাংলায়)	Text	Unicode Text (100)	O
Code (কোড)	Unique text like: "IU-001"	Text (20)	M
Code in Bangla (কোড বাংলায়)	Text	Unicode Text (20)	O
Contact No	Text	Text (20)	M
Email	Valid Email	Text (20)	O
Website	Website	Text (20)	O
Address	Supplier Address	Text	M
Remarks	Text	Text (200)	O

Table Name : inv_return_items

Primary Key : id

Name of Attribute	Description *(field level Business Rule)*	Data Type (Field *Size)*	Mandatory/ Optional
Returned Date	Date	Date	M
Created Date	Date	Date	O
Created By	Logged in User Id	Number (FK)	O
Remarks	Text	Text (200)	O
Item Detail Information [Multiple]			
Item Id	Derived Item from Package-Lot or Search Items UI	Number (FK)	M
Item Status	Derived from Item Status, default value is 1=Good (Is-Good)	Number (FK)	M
Quantity		Number (10)	M
Serial	Mandatory for Fixed Asset and Is-Serialized items	Text (50)	O
Fixed Asset ID		Text (50)	O
Remarks		Text (100)	O
Update Current Stock			

Table Name : employees

Primary Key : id

Name of Attribute	Description *(field level Business Rule)*	Data Type *(Field Size)*	Mandatory/ Optional
Employee ID		Text (10)	M
Employee Name (In English)		Text (100)	M
Employee Name (In Bangla)		Text (100)	M
Employee Photo		Image	O
Date of Birth		Date	M
Father's Name		Text (100)	O
Mother's Name		Text (100)	O
Gender	Derived (Male/Female)	Text (20)	M
Mobile		Text (15)	O
Religion	Derived	Text (20)	M
Designation	Derived (Ex: Senior Officer)	Text (50)	O
Employee Type	Derived (Ex: Officer)	Text (100)	M
Employee Category	Derived (Ex: Contractual)	Text (100)	M
Employee Class	Derived	Text (10)	M
Joining Date	Date and Time	Date	M
Release/Resign/Retirement Date	Date and Time	Date	O
Is Active	Yes or No	Boolean	M
Picture		Image	O

Table Name : access_logs

Primary Key : id

Name of Attribute	Description *(field level Business Rule)*	Data Type *(Field Size)*	Mandatory/ Optional
User Id	Logged in User	INT	M
Login IP		Text (100)	M
Login Datetime		date	M
Logout Datetime		date	O
User Agent	Browser Info	Text	M

Table Name : common_labels

Primary Key : id

Name of Attribute	Description *(field level Business Rule)*	Data Type *(Field Size)*	Mandatory/ Optional
Data Type	type of data used to categorize content	Text	M
Name English		Text (100)	M
Name Bangla		TEXT (100)	M
Order		int	O
Status	0 or 1	Tinyint (1)	M

4.7 ER-DIAGRAM OF INVENTORY MANAGEMENT SYSTEM

The figure 4.4 shows the Entity Relationship Diagram (ERD) for inventory management system. We can see the all entity of Inventory system with their attributes and relationship among entity

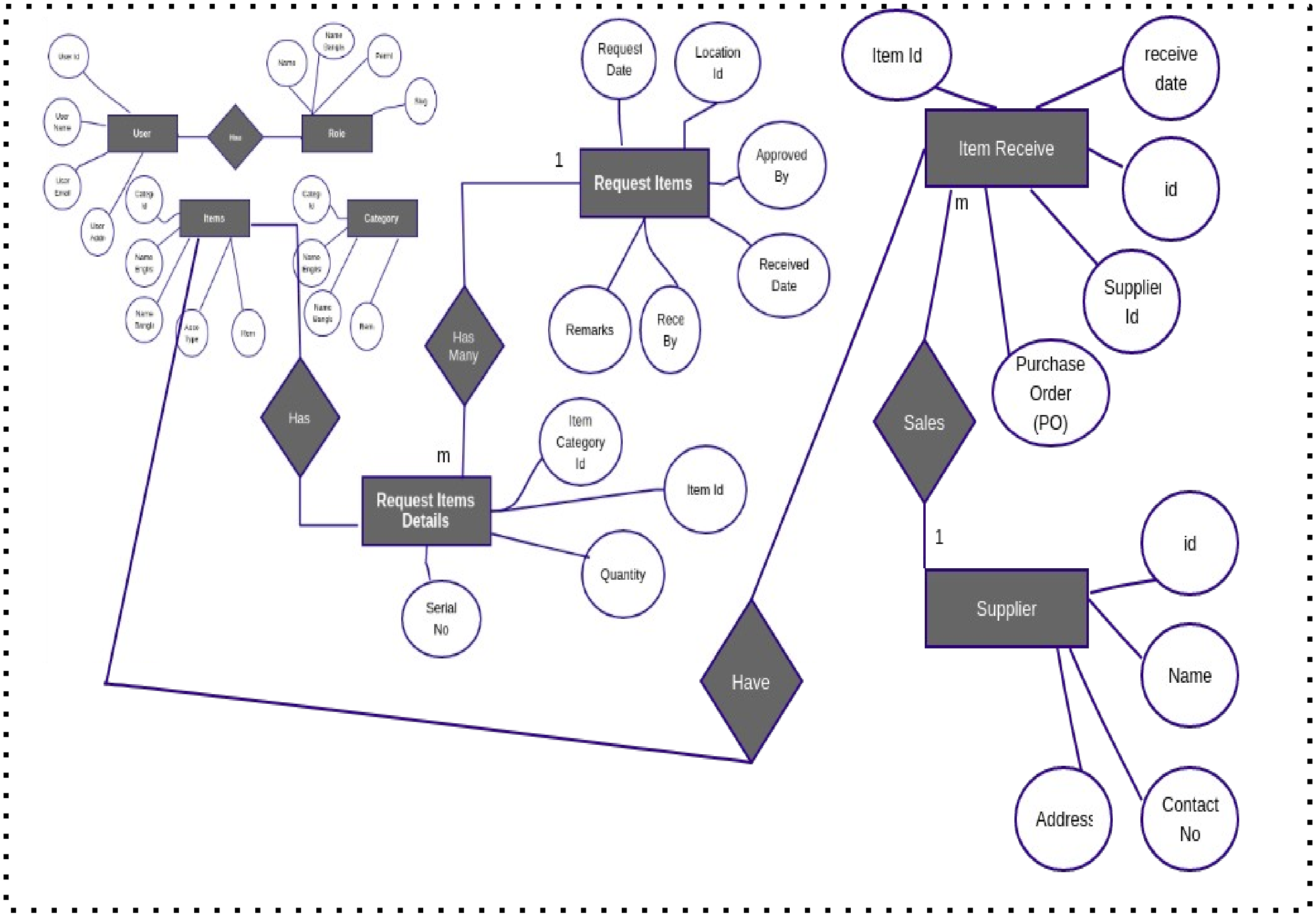


Figure 4.4: ER-Diagram for whole Inventory System

CHAPTER 5

SYSTEM IMPLEMENTATION AND EVALUATION

5.1: A SUCCESSFUL APPLICATION STARTS WITH A PROPER IDEA

To create a successful application, the first thing we need to keep in mind is:

Identify a problem which can be resolved by our application: Inventory Management System(IMS) has been developed for mid or large Organization and Institution to manage their asset distribution. Users will be able to access the application on Laptop and Desktop computers and android devices (smartphones or tablets) through Browser(Firefox, Google Chrome etc). The user- interface and experience is developed to consider user needs and requirements. User roles are Inventory Manager, Dept. Chairman, Other Teacher of CSE dept, IU and Admin:

The application should provide an organization/institution like our University with tangible benefits including Item distribution, Item stock report, Employee database etc.

5.2: IDENTIFICATION / CLARIFICATION

To create a successful application, we need to identify or be clear about:

- **Application target users** – An application should always be developed keeping in mind the target users of an application. So to the Inventory Management System application users are Inventory Manager and Other top management of organization and Admin
- **Multiple OS and devices to be supported** – An web application should be supported in Latest version of all popular browsers.
- **Interactive User Friendly UI** – And web application User Interface (UI) should be user friendly and mobile responsive so that users easily can use it.
- **Load Balancing**– When we start developing a web application, we need to keep in mind that our application should work properly for large scale data and many user.

5.3: Wire framing of Inventory Management System Application

Designing our application is another important factor responsible for the success of an application. Remember, a good UX design and good UI-UX means good discovery ability. An application developer should concentrate on the UI design and consider platform design standards as well. Today if the UI design of an application is beautiful and mobile responsive, then the application will be a successful application.

5.4: Development Technology

Purpose	Tools & Technologies
Backend Programming	PHP, PHP Laravel MVC Framework
Web Server	Apache
Database	MySQL
Front-End	HTML, CSS, JavaScript, Jquery, Bootstrap
Operating System	Windows, Linux, MacOS
Browser	Firefox, Google Chrome, Opera, Safari
Design	UML, Pencil, Photoshop etc.

5.5 Application Screenshots

Login Screen: After entering username and password in the login screen, the user will be redirected to the dashboard if username and password is correct. If username or password is wrong then will through error message

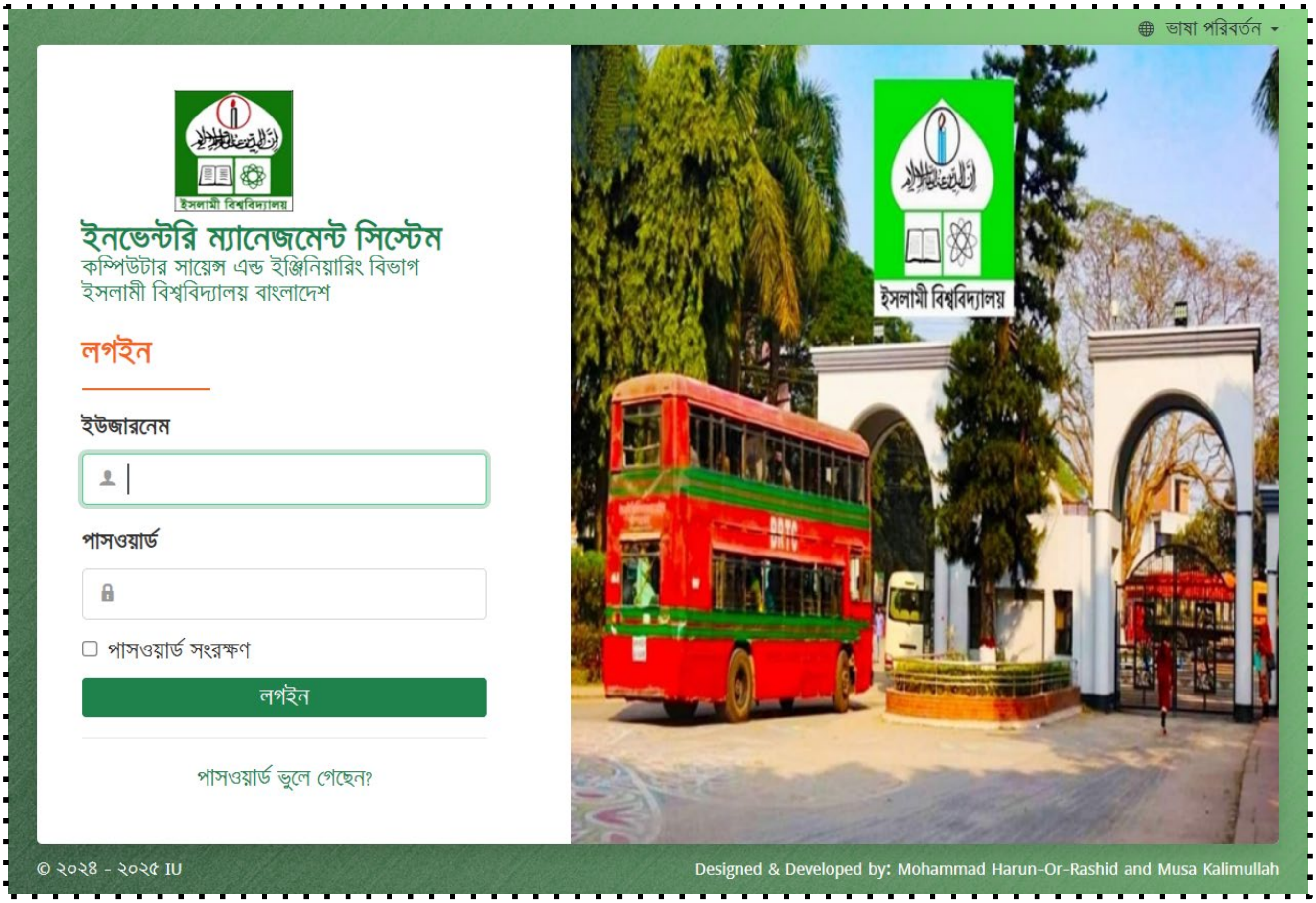

ভাষা পরিবর্তন
ইসলামী বিশ্ববিদ্যালয়
ইনভেন্টরি ম্যানেজমেন্ট সিস্টেম
কম্পিউটার সায়েন্স এন্ড ইঞ্জিনিয়ারিং বিভাগ
ইসলামী বিশ্ববিদ্যালয় বাংলাদেশ
লগইন
ইউজারনেম
পাসওয়ার্ড
□ পাসওয়ার্ড সংরক্ষণ
লগইন
পাসওয়ার্ড ভুলে গেছেন?
ইসলামী বিশ্ববিদ্যালয়
© ২০২৪ - ২০২৫ IU
Designed & Developed by: Mohammad Harun-Or-Rashid and Musa Kalimullah

Figure 5.1: Login screen

Main Dashboard: Users will be redirected to this dashboard after successful login. In the sidebar, there is User management menu and Settings menu and in the body of the page there is Inventory module and user management shortcut icon.

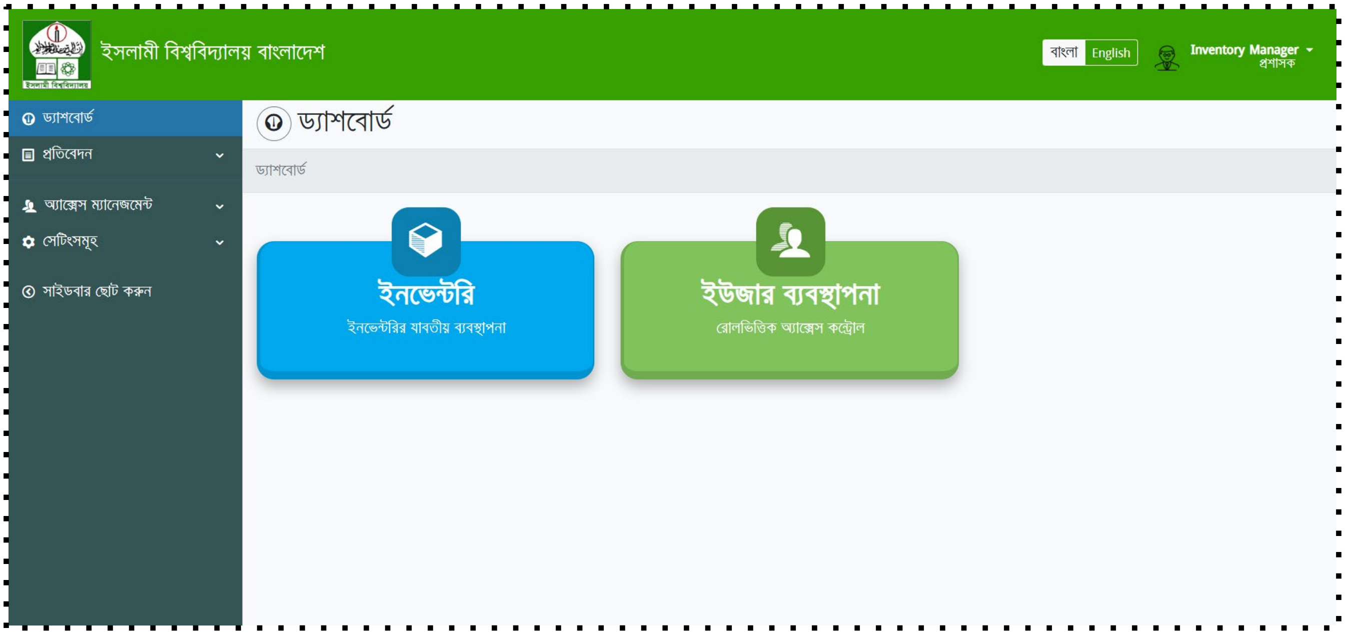
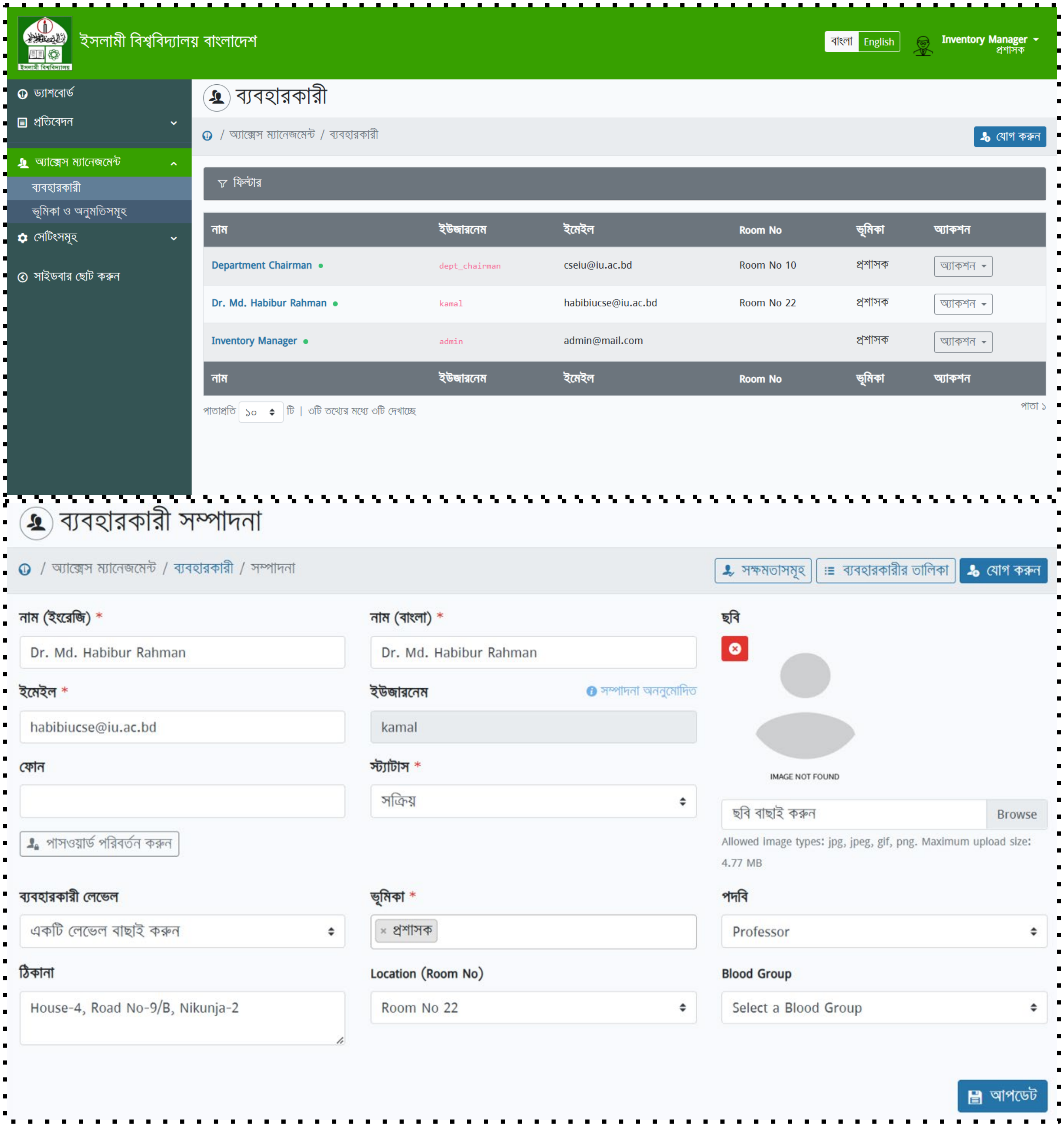


Figure 5.2: Main Dashboard

User List: Whole system users are in this list. From this list we can create new users or edit existing users and also delete existing users. We can also assign user roles and permission.



ব্যবহারকারী

/ অ্যাক্সেস ম্যানেজমেন্ট / ব্যবহারকারী

নাম	ইউজারনেম	ইমেইল	Room No	ভূমিকা	অ্যাকশন
Department Chairman ●	dept_chairman	cseiu@iu.ac.bd	Room No 10	প্রশাসক	অ্যাকশন ▾
Dr. Md. Habibur Rahman ●	kamal	habibiucse@iu.ac.bd	Room No 22	প্রশাসক	অ্যাকশন ▾
Inventory Manager ●	admin	admin@mail.com		প্রশাসক	অ্যাকশন ▾
নাম	ইউজারনেম	ইমেইল	Room No	ভূমিকা	অ্যাকশন

পাতাপ্রতি ১০ টি | ৩টি তথ্যের মধ্যে ৩টি দেখাচ্ছে

পাতা ১

ব্যবহারকারী সম্পাদনা

/ অ্যাক্সেস ম্যানেজমেন্ট / ব্যবহারকারী / সম্পাদনা

নাম (ইংরেজি) *: Dr. Md. Habibur Rahman

নাম (বাংলা) *: Dr. Md. Habibur Rahman

ছবি

ইমেইল *: habibiucse@iu.ac.bd

ইউজারনেম: kamal (সম্পাদনা অননুমোদিত)

ফোন:

স্ট্যাটাস *: সক্রিয়

Allowed image types: jpg, jpeg, gif, png. Maximum upload size: 4.77 MB

ব্যবহারকারী লেভেল: একটি লেভেল বাছাই করুন

ভূমিকা *: × প্রশাসক

পদবি: Professor

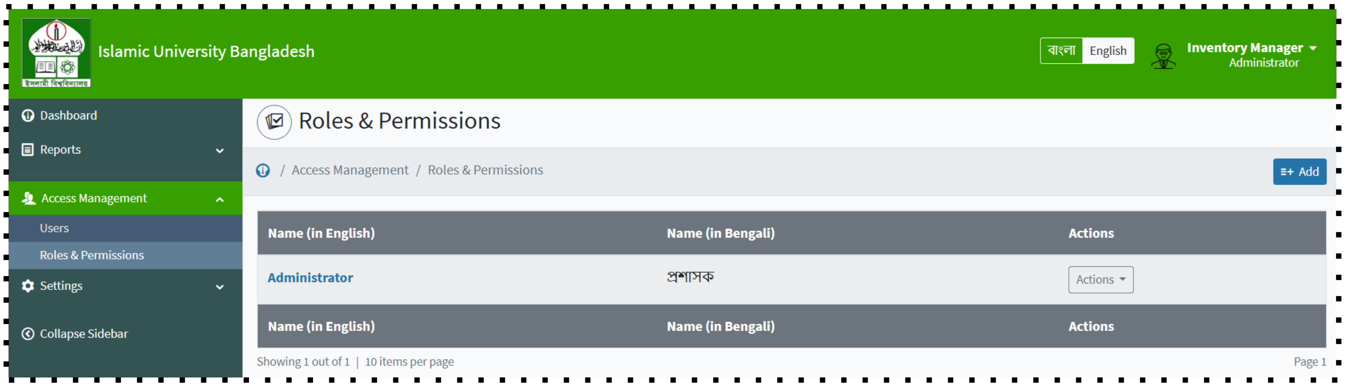
ঠিকানা: House-4, Road No-9/B, Nikunja-2

Location (Room No): Room No 22

Blood Group: Select a Blood Group

Figure 3: User List and User Entry

User Role: Assign user role from this menu. We can create new roles or update existing roles.



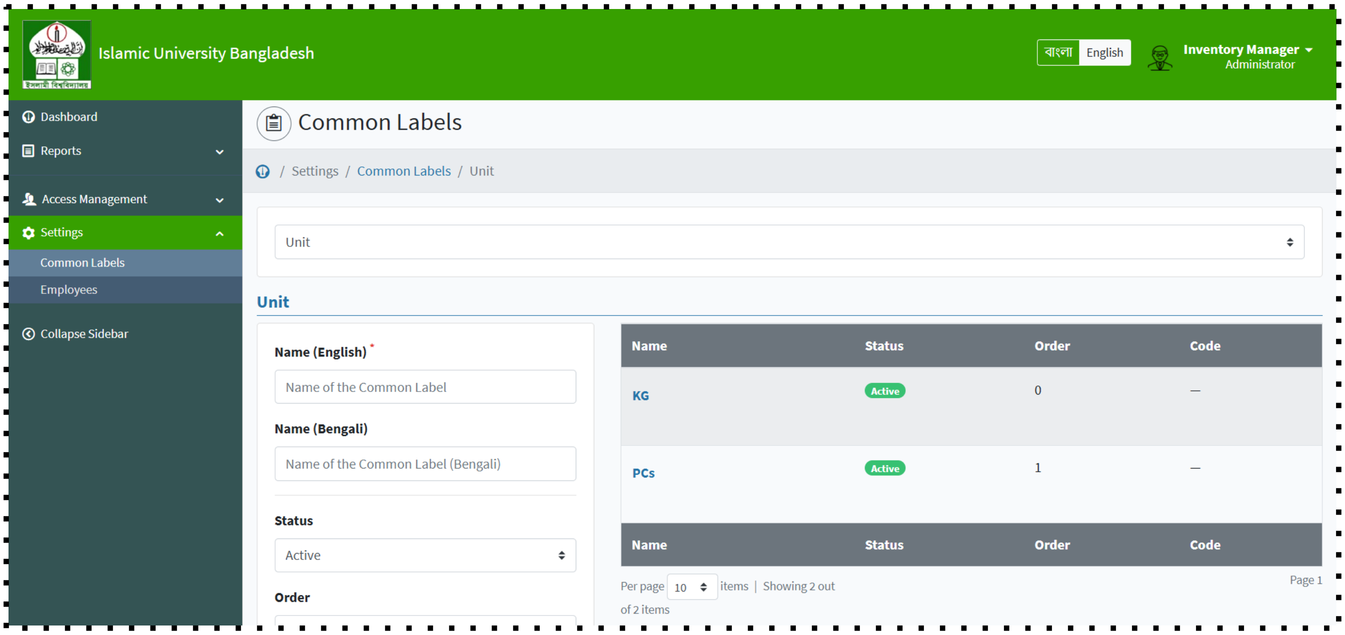
Name (in English)	Name (in Bengali)	Actions
Administrator	প্রশাসক	Actions
Name (in English)	Name (in Bengali)	Actions

Showing 1 out of 1 | 10 items per page

Page 1

Figure 5.4: User Role and Permission

Common Labels: Whole system common setup data is here. From this menu anyone can add or update setup data



Unit

Unit

Name (English) *

Name of the Common Label

Name (Bengali)

Name of the Common Label (Bengali)

Status

Active

Order

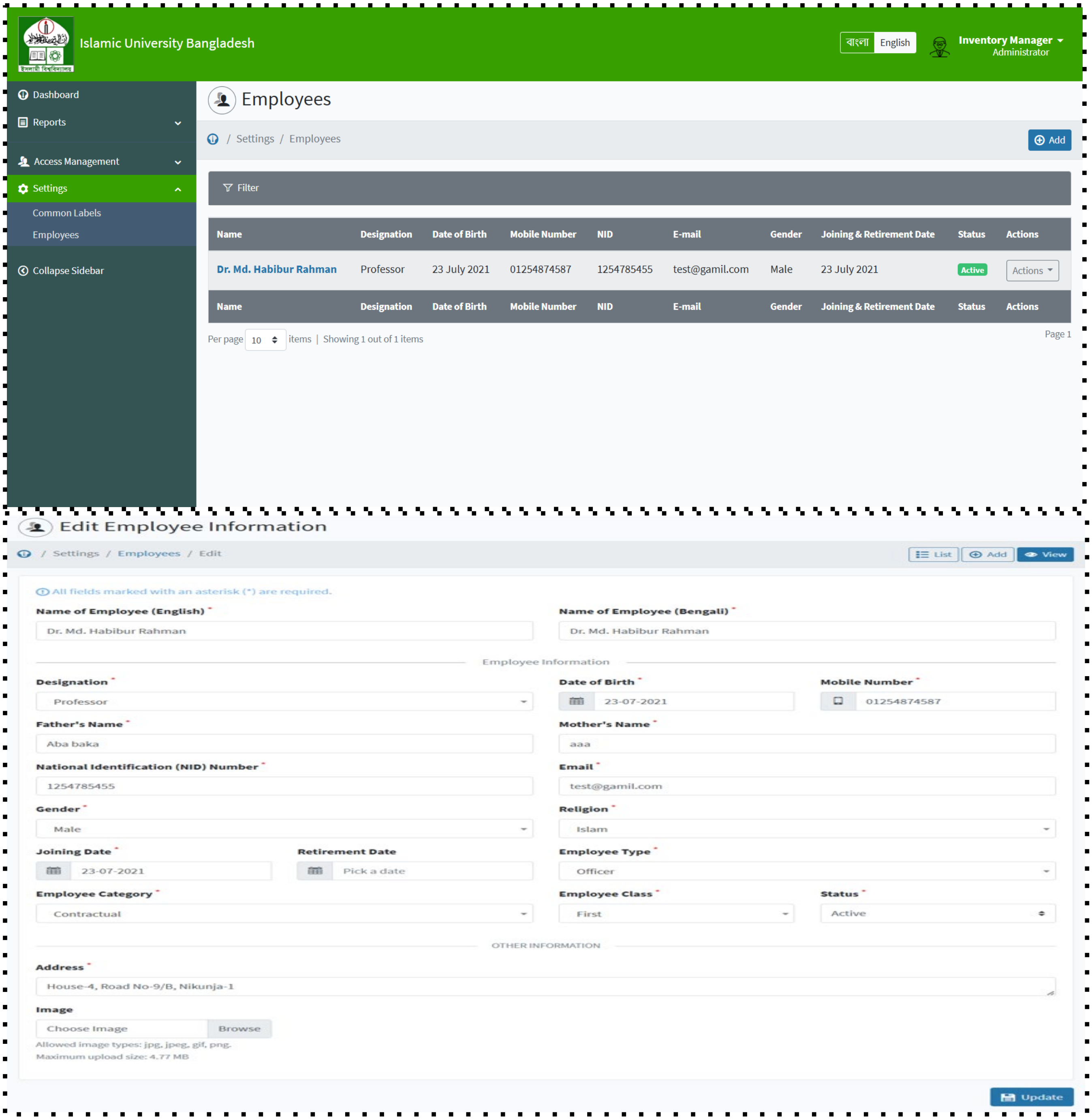
Name	Status	Order	Code
KG	Active	0	—
PCs	Active	1	—
Name	Status	Order	Code

Per page 10 items | Showing 2 out of 2 items

Page 1

Figure 5.5: Common Setup Data Screen

Employees: Employee information is in this menu. We can create new employees and edit existing employee information. We also view employee information



Name	Designation	Date of Birth	Mobile Number	NID	E-mail	Gender	Joining & Retirement Date	Status	Actions
Dr. Md. Habibur Rahman	Professor	23 July 2021	01254874587	1254785455	test@gamil.com	Male	23 July 2021	Active	Actions
Name	Designation	Date of Birth	Mobile Number	NID	E-mail	Gender	Joining & Retirement Date	Status	Actions

Per page 10 items | Showing 1 out of 1 items   Page 1

All fields marked with an asterisk (*) are required.

Name of Employee (English) *: Dr. Md. Habibur Rahman

Name of Employee (Bengali) *: Dr. Md. Habibur Rahman

Employee Information

Designation *: Professor

Date of Birth *: 23-07-2021

Mobile Number *: 01254874587

Father's Name *: Aba baka

Mother's Name *: aaa

National Identification (NID) Number *: 1254785455

Email *: test@gamil.com

Gender *: Male

Religion *: Islam

Joining Date *: 23-07-2021

Retirement Date: Pick a date

Employee Type *: Officer

Employee Category *: Contractual

Employee Class *: First

Status *: Active

OTHER INFORMATION

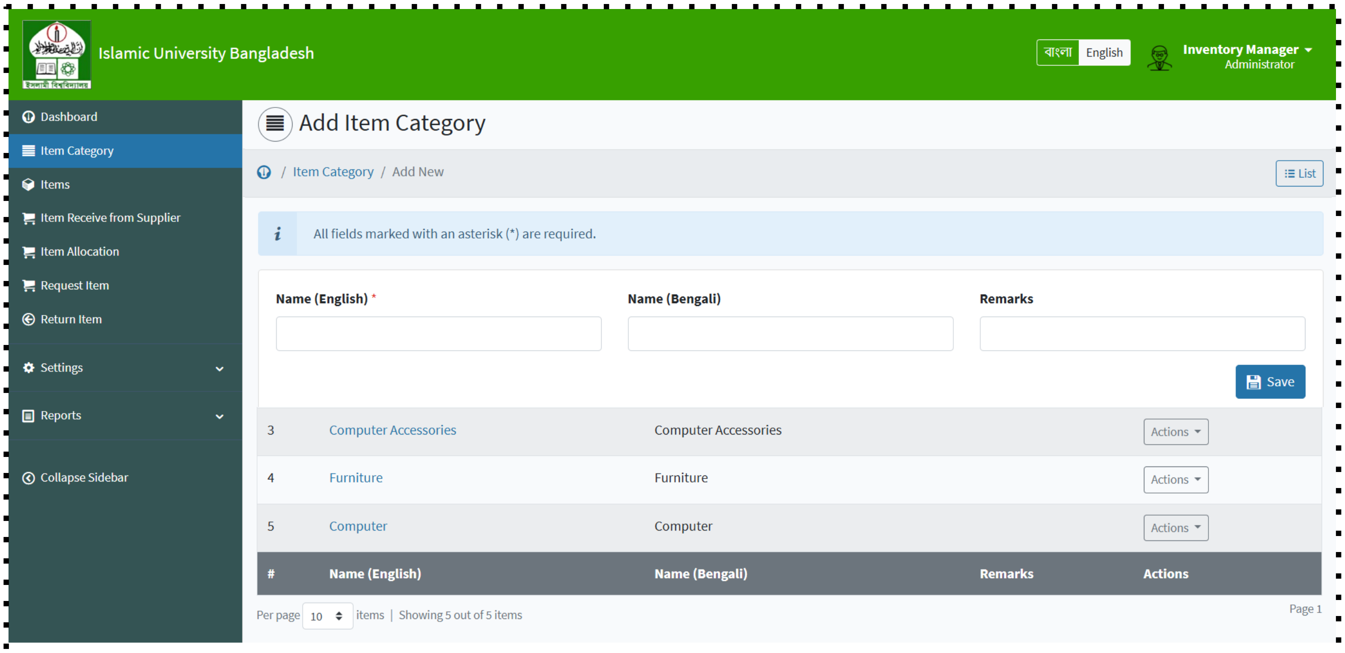
Address *: House-4, Road No-9/B, Nikunja-1

Image: Choose Image Browse

Allowed image types: jpg, jpeg, gif, png.
Maximum upload size: 4.77 MB

Figure 5.6: Employee List and Employee Entry form

Item Category: Item category list is here. New Item category will be inserted from this menu and update the existing category. Category name will be both English & Bangla.



Add Item Category

All fields marked with an asterisk (*) are required.

Name (English) *

Name (Bengali)

Remarks

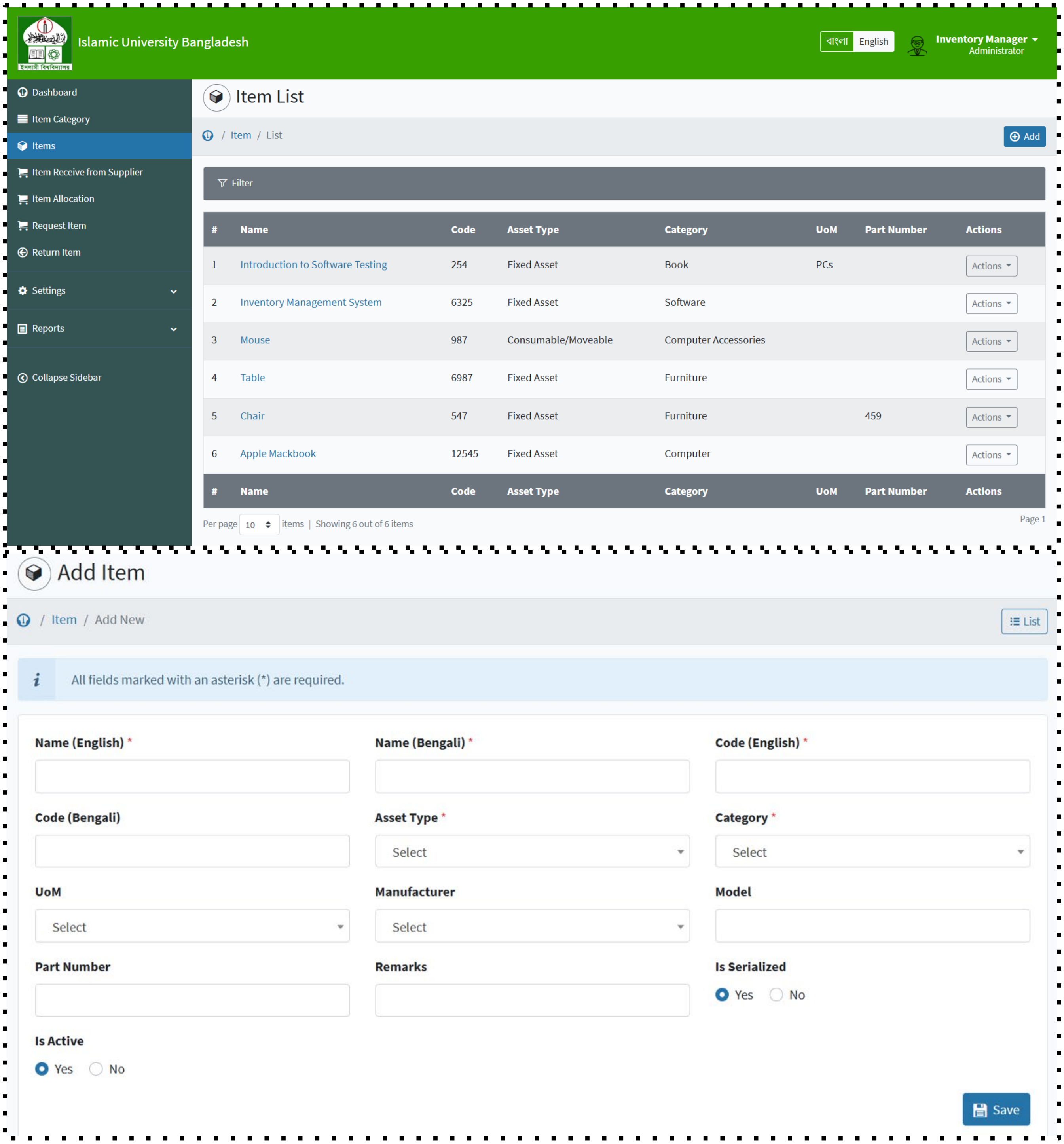
#	Name (English)	Name (Bengali)	Remarks	Actions
3	Computer Accessories	Computer Accessories		Actions
4	Furniture	Furniture		Actions
5	Computer	Computer		Actions

Showing 5 out of 5 items

Page 1

Figure 5.7: Item Category List and Entry form

Item: We can create or update Category Wise Item



Item List

Item / List

Add

Filter

#	Name	Code	Asset Type	Category	UoM	Part Number	Actions
1	Introduction to Software Testing	254	Fixed Asset	Book	PCs		Actions
2	Inventory Management System	6325	Fixed Asset	Software			Actions
3	Mouse	987	Consumable/Moveable	Computer Accessories			Actions
4	Table	6987	Fixed Asset	Furniture			Actions
5	Chair	547	Fixed Asset	Furniture		459	Actions
6	Apple Mackbook	12545	Fixed Asset	Computer			Actions
#	Name	Code	Asset Type	Category	UoM	Part Number	Actions

Per page 10 items | Showing 6 out of 6 items

Page 1

Add Item

Item / Add New

List

All fields marked with an asterisk (*) are required.

Name (English) *

Name (Bengali) *

Code (English) *

Code (Bengali)

Asset Type * Select

Category * Select

UoM Select

Manufacturer Select

Model

Part Number

Remarks

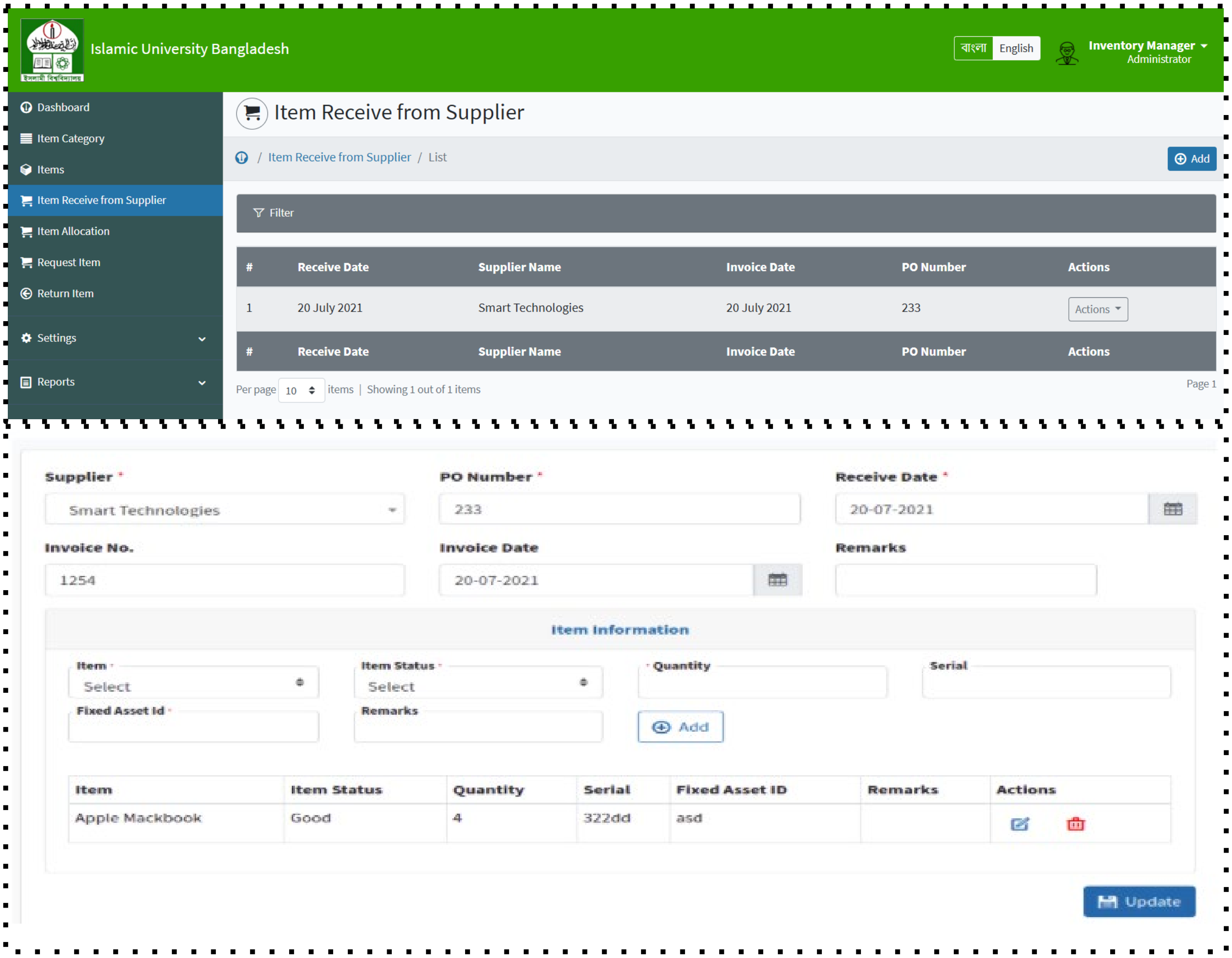
Is Serialized ● Yes ○ No

Is Active ● Yes ○ No

Save

Figure 5.8: Item List and Item Entry form

Item Receive from Supplier: Enter new purchased item in this menu. This menu shows the Item received from the supplier list.



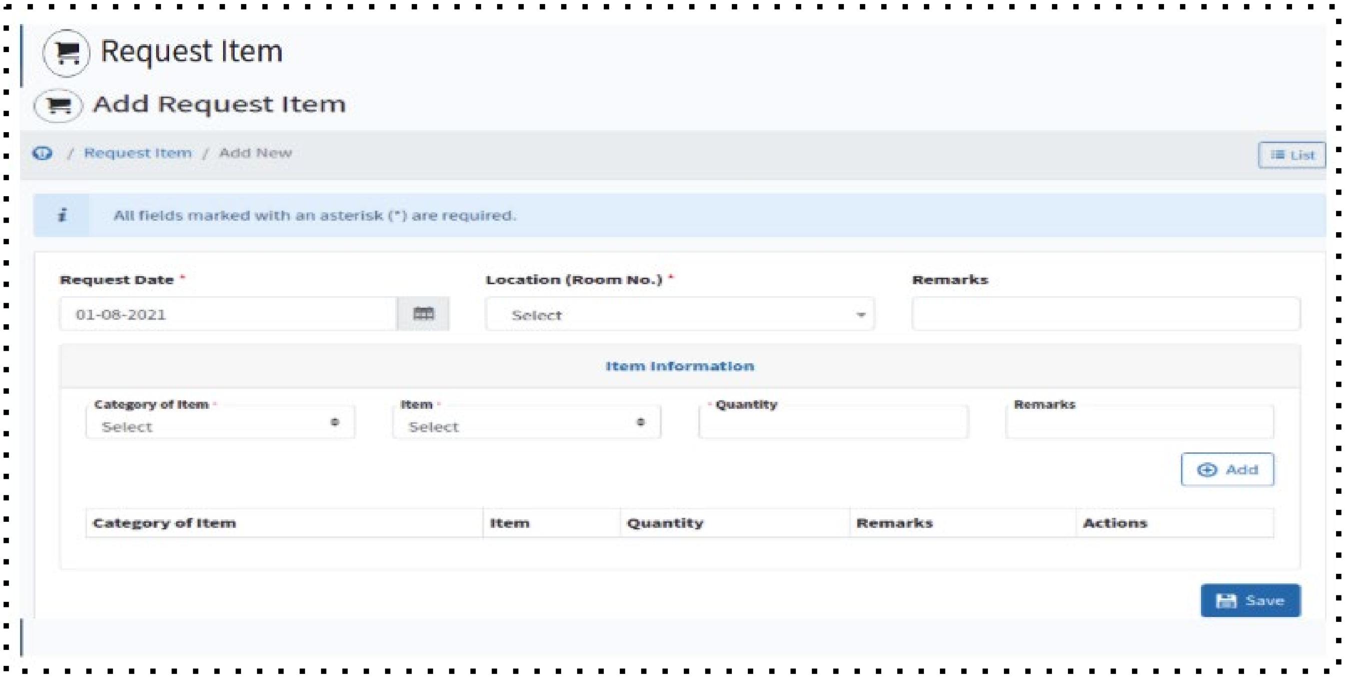
#	Receive Date	Supplier Name	Invoice Date	PO Number	Actions
1	20 July 2021	Smart Technologies	20 July 2021	233	Actions

Supplier *	Smart Technologies
PO Number *	233
Receive Date *	20-07-2021
Invoice No.	1254
Invoice Date	20-07-2021
Remarks	
Item *	Select
Item Status *	Select
* Quantity	
Serial	
Fixed Asset Id *	
Remarks	

Item	Item Status	Quantity	Serial	Fixed Asset ID	Remarks	Actions
Apple Mackbook	Good	4	322dd	asd		

Figure 5.9: Item Receive from Supplier List and Entry form

Item Request: Any employee or teacher can request for Item. The following figure shows the Item request list and entry from



All fields marked with an asterisk (*) are required.

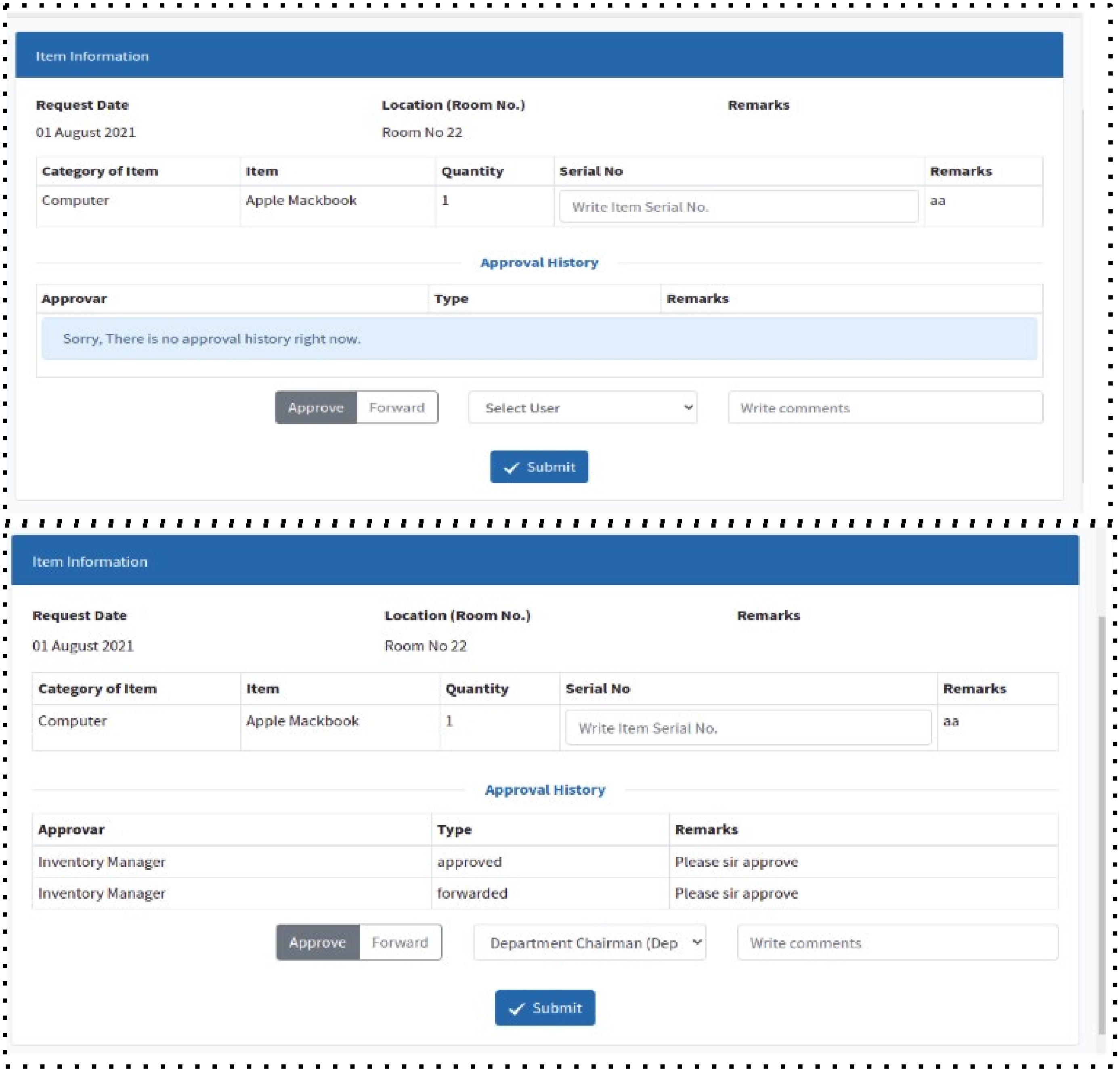
Request Date *	Location (Room No.) *	Remarks
01-08-2021	Select	

Category of Item *	Item *	* Quantity	Remarks
Select	Select		

Category of Item	Item	Quantity	Remarks	Actions

Figure 5.10: Request for Item List and Request Page

Item Request Approval process: Admin user or top management approve requested item. There is a dynamic approval process.



Item Information

Request Date	Location (Room No.)	Remarks
01 August 2021	Room No 22	

Category of Item	Item	Quantity	Serial No	Remarks
Computer	Apple Mackbook	1	Write Item Serial No.	aa

Approval History

Approvar	Type	Remarks
Sorry, There is no approval history right now.		

Item Information

Request Date	Location (Room No.)	Remarks
01 August 2021	Room No 22	

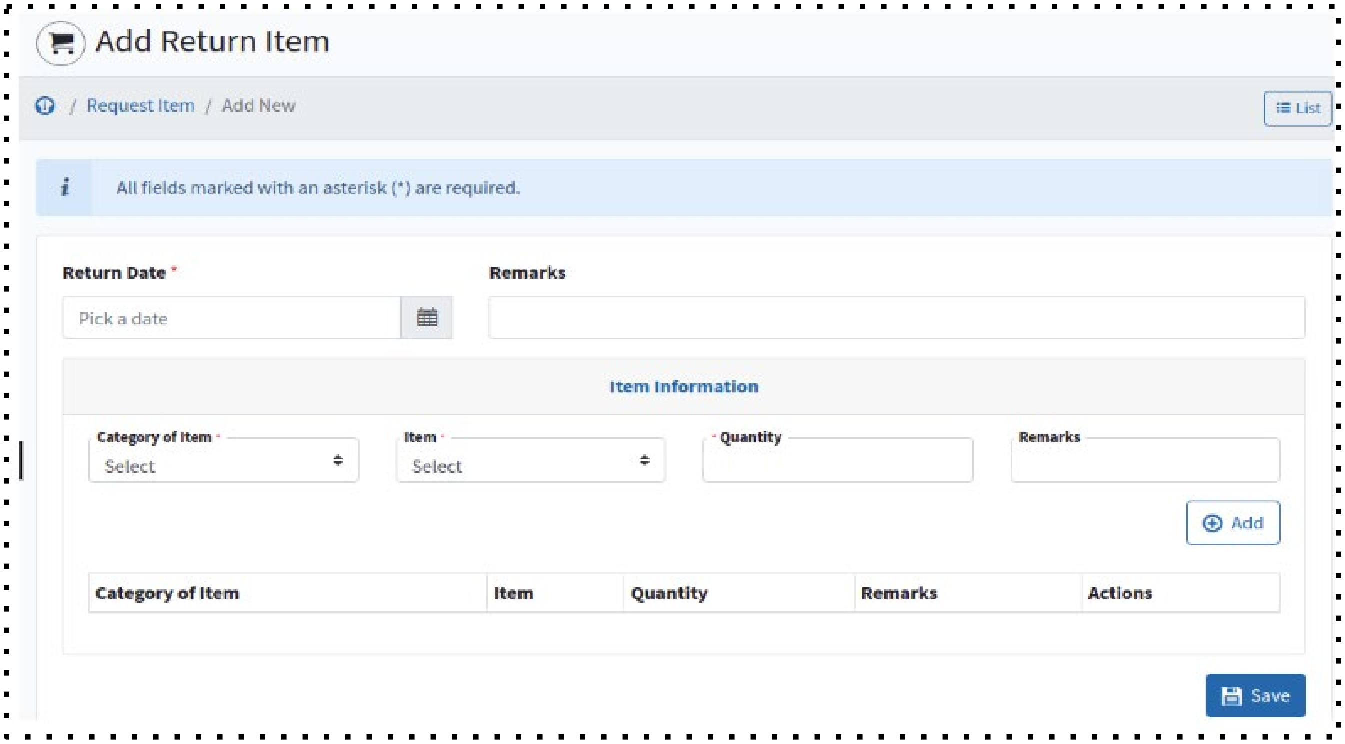
Category of Item	Item	Quantity	Serial No	Remarks
Computer	Apple Mackbook	1	Write Item Serial No.	aa

Approval History

Approvar	Type	Remarks
Inventory Manager	approved	Please sir approve
Inventory Manager	forwarded	Please sir approve

Figure 5.11: Requested Item Approval Process page

Item Return: Any user can return their allocated item by this scope.

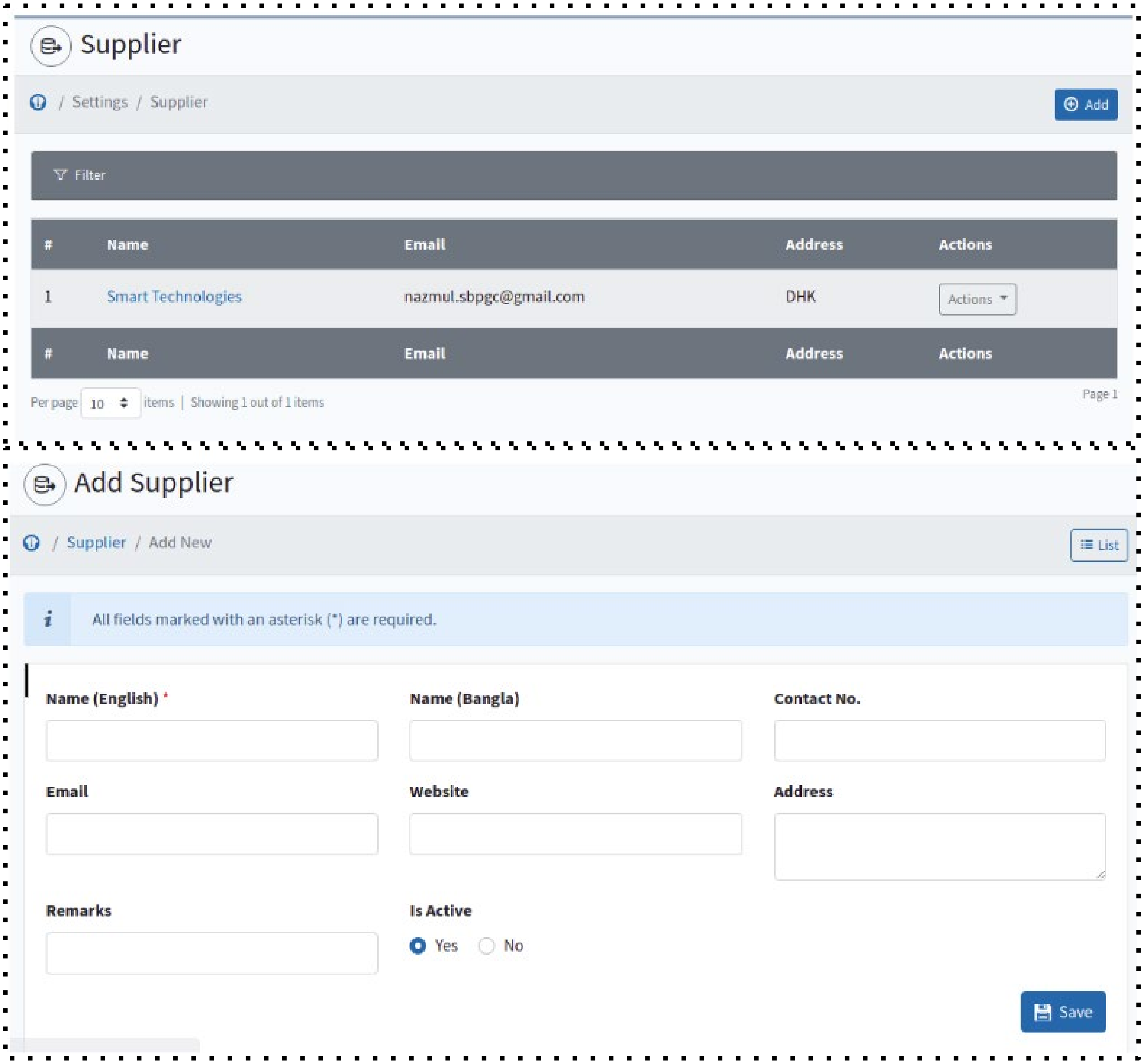


All fields marked with an asterisk (*) are required.

Category of Item	Item	Quantity	Remarks	Actions

Figure 5.12: Item Return screen

The figure 5.13 shows the supplier information list. We can create new supplier info or update existing suppliers from this scope.



Supplier

#	Name	Email	Address	Actions
1	Smart Technologies	nazmul.sbpgc@gmail.com	DHK	Actions ▾
#	Name	Email	Address	Actions

Per page 10 items | Showing 1 out of 1 items

Page 1

Add Supplier

All fields marked with an asterisk (*) are required.

Name (English) *

Name (Bangla)

Contact No.

Email

Website

Address

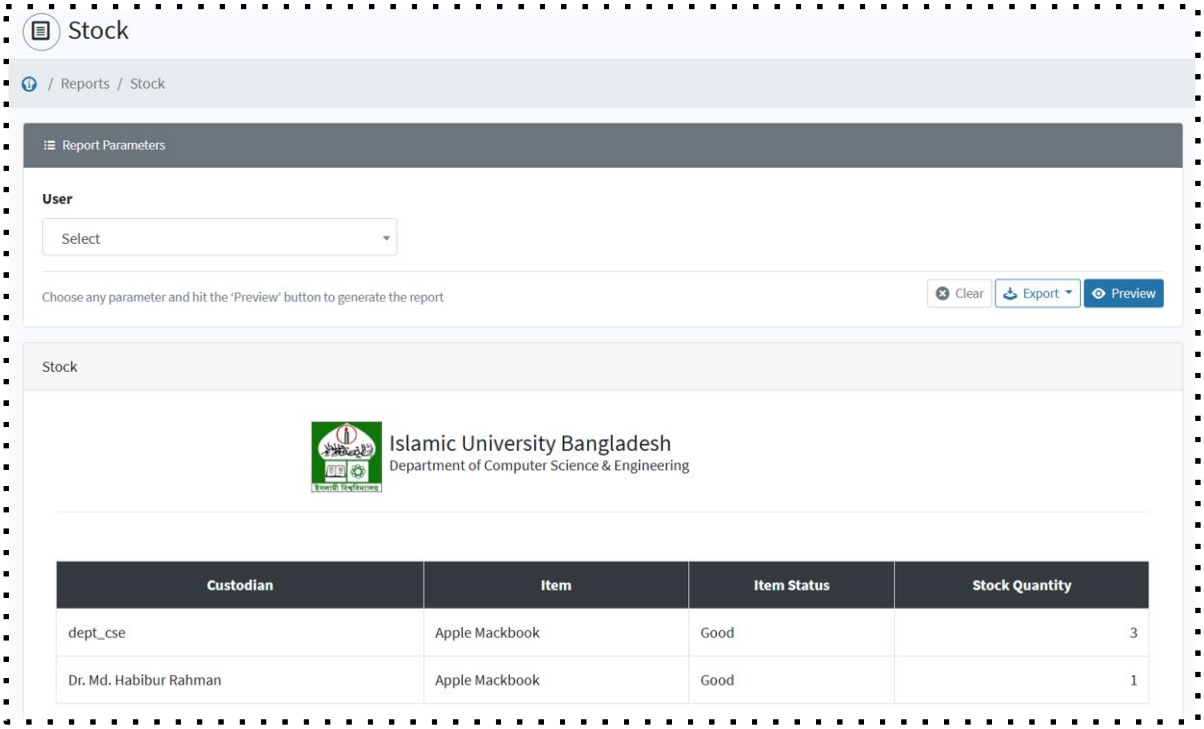
Remarks

Is Active

● Yes ○ No

Figure 5.13: Supplier List and Entry Screen

Stock Report: The following figure 5.14 and 5.15 shows that Item stock quantity with custodian wise or department stock. We can export report as PDF, XLS or Word File



Stock

/ Reports / Stock

Report Parameters

User

Select

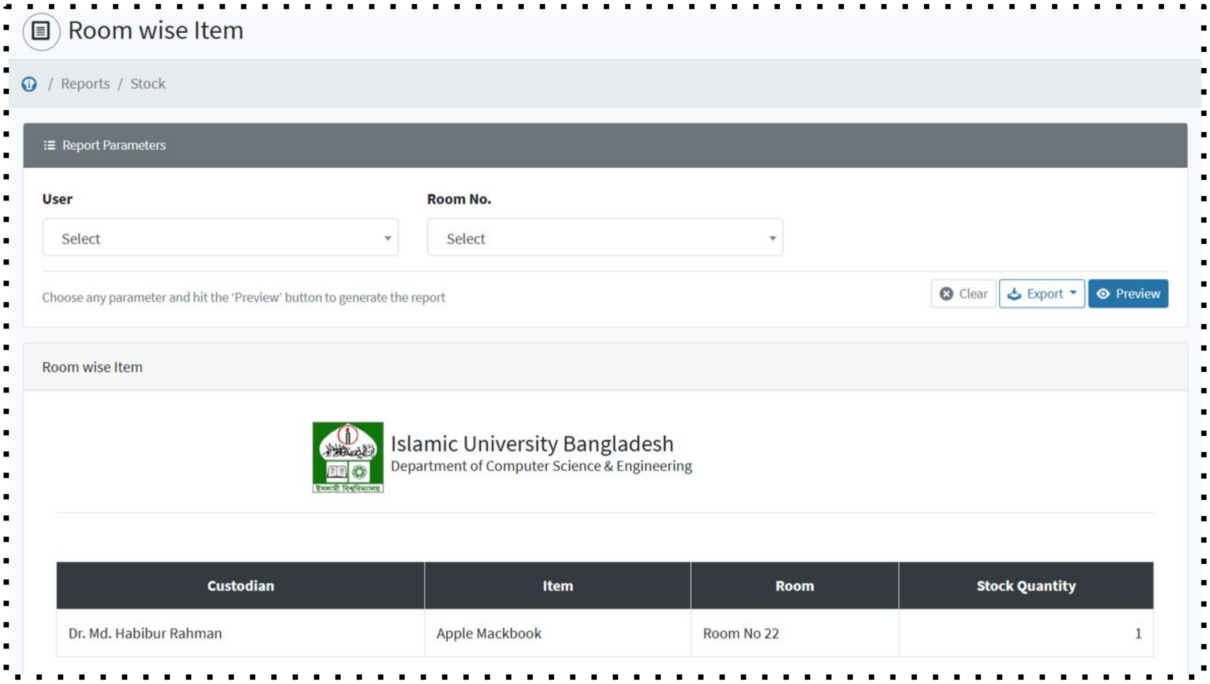
Choose any parameter and hit the 'Preview' button to generate the report

Stock

Islamic University Bangladesh
Department of Computer Science & Engineering

Custodian	Item	Item Status	Stock Quantity
dept_cse	Apple Mackbook	Good	3
Dr. Md. Habibur Rahman	Apple Mackbook	Good	1

Figure 14: Stock Report



Room wise Item

/ Reports / Stock

Report Parameters

User

Select

Room No.

Select

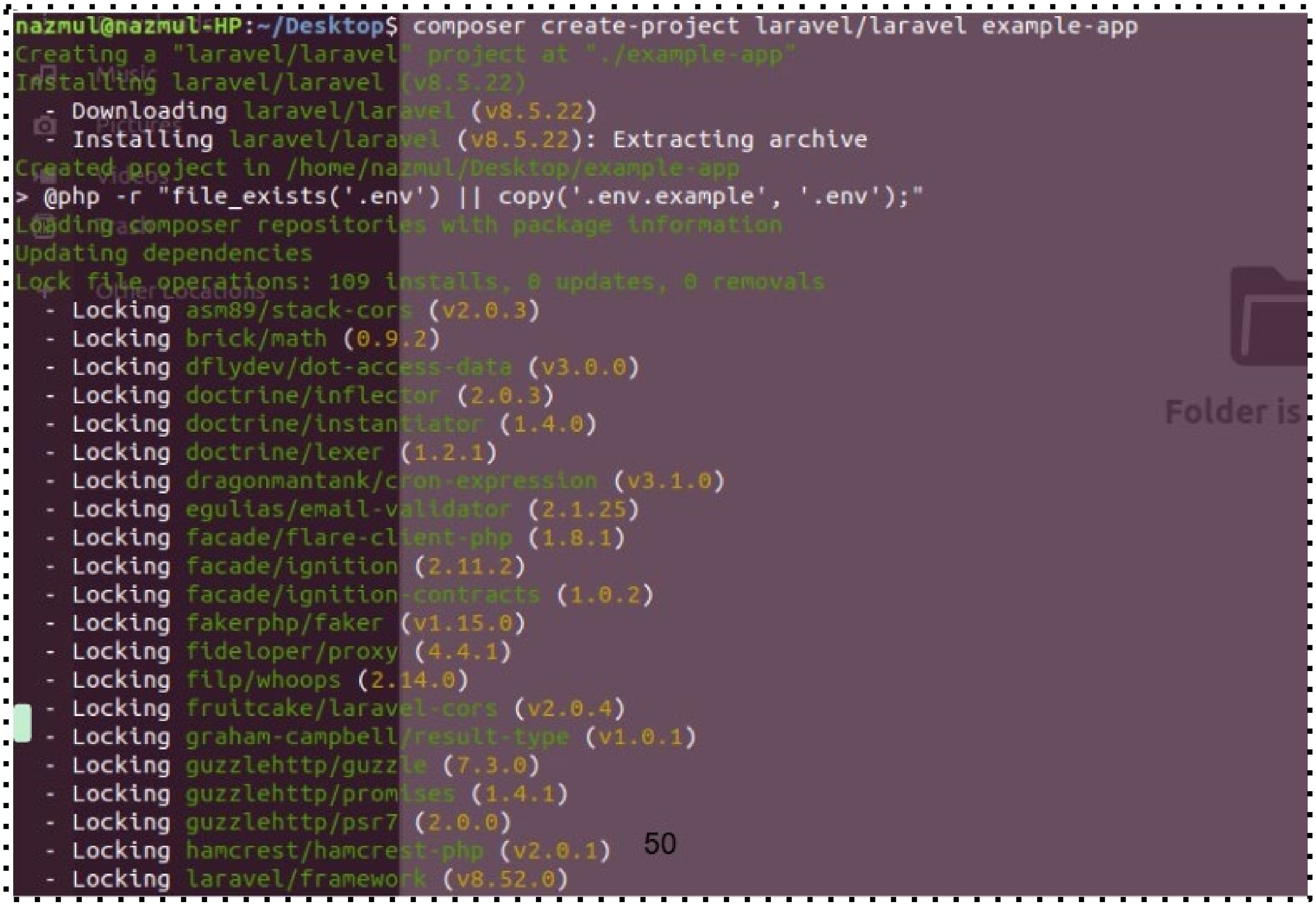
Choose any parameter and hit the 'Preview' button to generate the report

Room wise Item

Islamic University Bangladesh
Department of Computer Science & Engineering

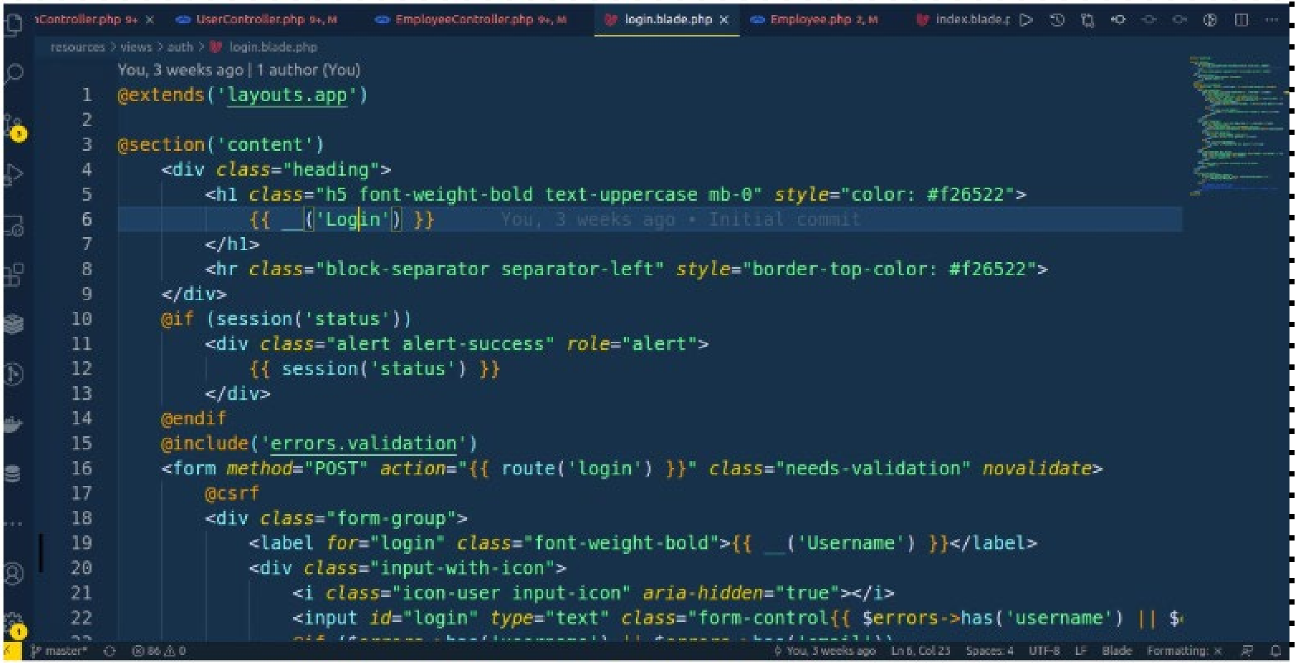
Custodian	Item	Room	Stock Quantity
Dr. Md. Habibur Rahman	Apple Mackbook	Room No 22	1

Figure 5.15: Room Wise Item Stock Report



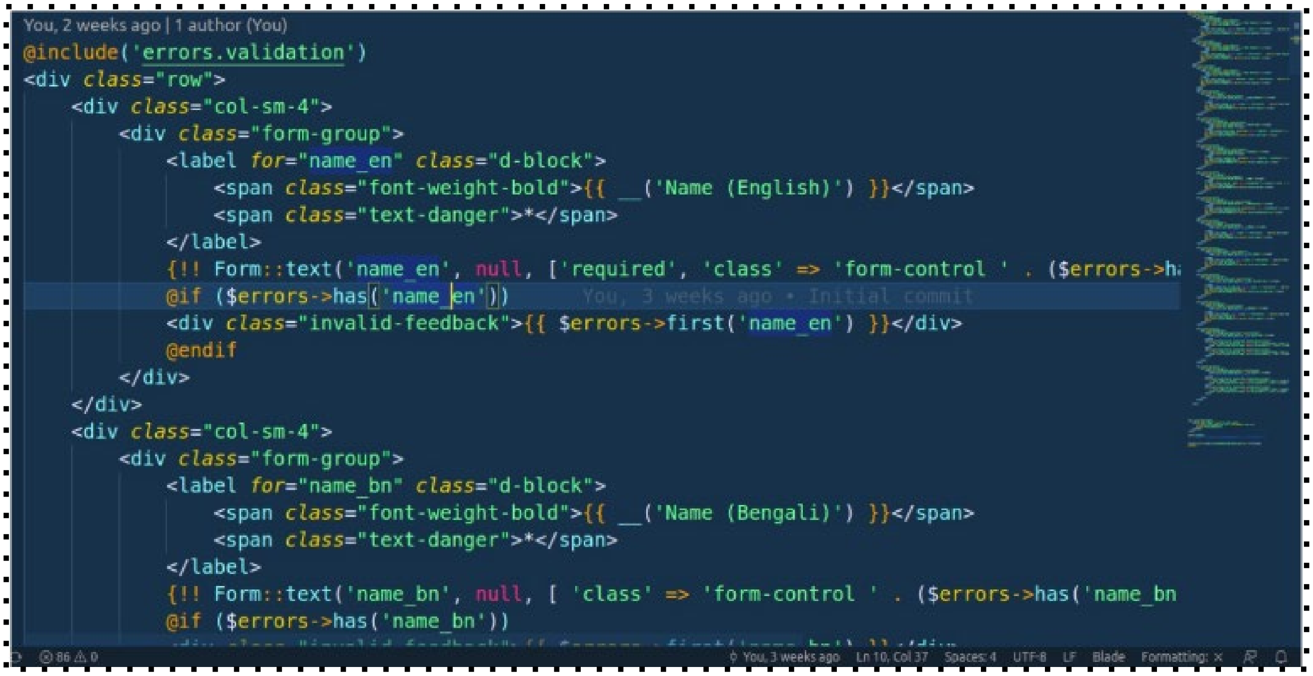
```
nazmul@nazmul-HP:~/Desktop$ composer create-project laravel/laravel example-app
Creating a "laravel/laravel" project at "./example-app"
Installing laravel/laravel (v8.5.22)
  - Downloading laravel/laravel (v8.5.22)
  - Installing laravel/laravel (v8.5.22): Extracting archive
Created project in /home/nazmul/Desktop/example-app
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 109 installs, 0 updates, 0 removals
  - Locking asm89/stack-cors (v2.0.3)
  - Locking brick/math (0.9.2)
  - Locking dflydev/dot-access-data (v3.0.0)
  - Locking doctrine/inflector (2.0.3)
  - Locking doctrine/instantiator (1.4.0)
  - Locking doctrine/lexer (1.2.1)
  - Locking dragonmantank/cron-expression (v3.1.0)
  - Locking egulias/email-validator (2.1.25)
  - Locking facade/flare-client-php (1.8.1)
  - Locking facade/ignition (2.11.2)
  - Locking facade/ignition-contracts (1.0.2)
  - Locking fakerphp/faker (v1.15.0)
  - Locking fideloper/proxy (4.4.1)
  - Locking filp/whoops (2.14.0)
  - Locking fruitcake/laravel-cors (v2.0.4)
  - Locking graham-campbell/result-type (v1.0.1)
  - Locking guzzlehttp/guzzle (7.3.0)
  - Locking guzzlehttp/promises (1.4.1)
  - Locking guzzlehttp/psr7 (2.0.0)
  - Locking hamcrest/hamcrest-php (v2.0.1)
  - Locking laravel/framework (v8.52.0)
```

Figure 5.16: PHP Laravel Framework setup and configuration project.



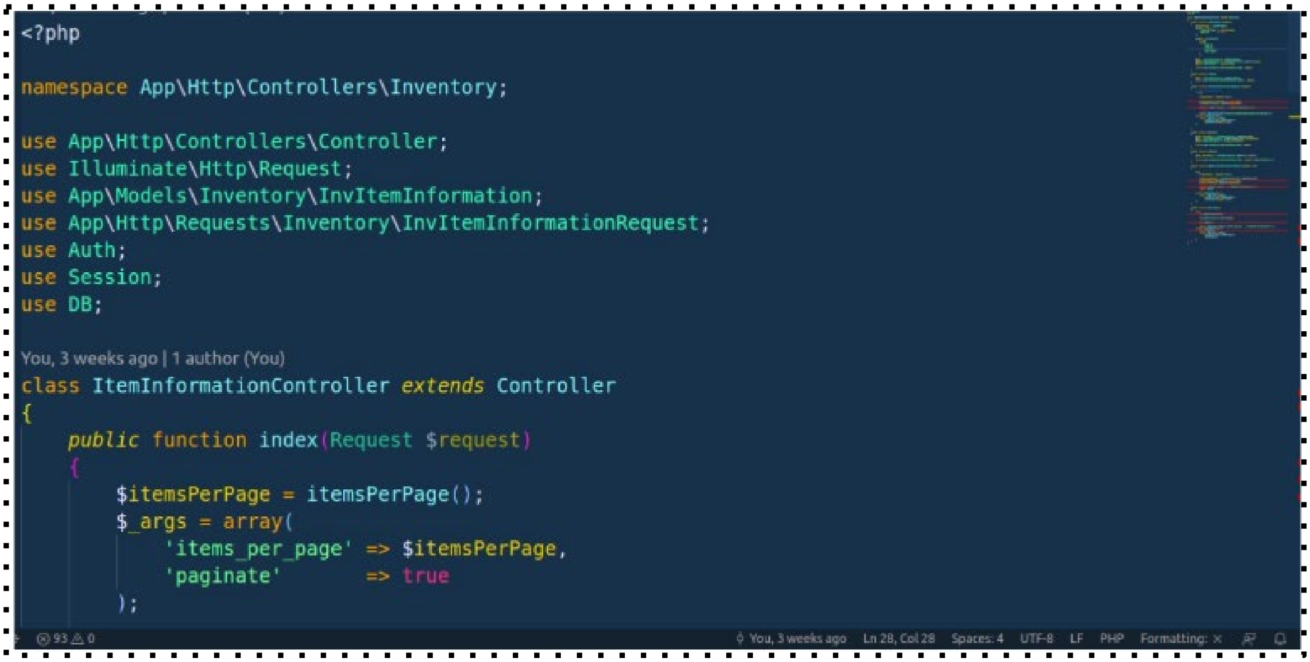
```
@extends('layouts.app')

@section('content')
    <div class="heading">
        <h1 class="h5 font-weight-bold text-uppercase mb-0" style="color: #f26522">
            {{ __('Login') }}        You, 3 weeks ago • Initial commit
        </h1>
        <hr class="block-separator separator-left" style="border-top-color: #f26522">
    </div>
    @if (session('status'))
        <div class="alert alert-success" role="alert">
            {{ session('status') }}
        </div>
    @endif
    @include('errors.validation')
    <form method="POST" action="{{ route('login') }}" class="needs-validation" novalidate>
        @csrf
        <div class="form-group">
            <label for="login" class="font-weight-bold">{{ __('Username') }}</label>
            <div class="input-with-icon">
                <i class="icon-user input-icon" aria-hidden="true"></i>
                <input id="login" type="text" class="form-control{{ $errors->has('username') || $
```

Figure 5.17: Login page design with Laravel Blade Template, Html, CSS.



```
@include('errors.validation')
<div class="row">
    <div class="col-sm-4">
        <div class="form-group">
            <label for="name_en" class="d-block">
                <span class="font-weight-bold">{{ __('Name (English)') }}</span>
                <span class="text-danger">*</span>
            </label>
            {!! Form::text('name_en', null, ['required', 'class' => 'form-control ' . ($errors->ha
            @if ($errors->has('name_en'))        You, 3 weeks ago • Initial commit
            <div class="invalid-feedback">{{ $errors->first('name_en') }}</div>
            @endif
        </div>
    </div>
    <div class="col-sm-4">
        <div class="form-group">
            <label for="name_bn" class="d-block">
                <span class="font-weight-bold">{{ __('Name (Bengali)') }}</span>
                <span class="text-danger">*</span>
            </label>
            {!! Form::text('name_bn', null, [ 'class' => 'form-control ' . ($errors->has('name_bn
            @if ($errors->has('name_bn'))
```

Figure 5.18: Item information entry page designing



```php
<?php

namespace App\Http\Controllers\Inventory;

use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Models\Inventory\InvItemInformation;
use App\Http\Requests\Inventory\InvItemInformationRequest;
use Auth;
use Session;
use DB;

You, 3 weeks ago | 1 author (You)
class ItemInformationController extends Controller
{
    public function index(Request $request)
    {
        $itemsPerPage = itemsPerPage();
        $_args = array(
            'items_per_page' => $itemsPerPage,
            'paginate'       => true
        );
```

Figure 5.19: Item Information Backend PHP Controller Class php file

5.6: Identify testers: Listen to them and incorporate relevant feedback

All feedback is incorporated basically these raised from application testing. Given below the overall feedback’s categories

1. UI feedback
2. Feedback based on functionalities.
3. Database level feedback

End Users Feedback: Basically, when I will release this application then end users feedback need to incorporate to the system

5.7: Release/Deploy the application

App will be released to the Organization provided Hosting Server .

5.8: Upgrade application with improvements and new features

After release, the application. Based on users demand new modules or existing module enhancement and further deploy as a new version.

CHAPTER 6

CONCLUSION

6.1 CONCLUSION

Inventory management is a critical aspect of asset management, requiring careful control and optimization. Numerous scientific models and strategies are available in the literature to help organizations select the most effective inventory policies. The Inventory Management System (IMS) developed in this project aims to address these needs by providing a streamlined solution for managing asset distribution within departments.

The system allows teachers and employees to easily request items such as laptops, mice, chairs, tables, and more through the application. Additionally, management can efficiently track all items in stock and monitor which items have been allocated to specific employees using the stock report feature. This system enhances transparency, reduces manual effort, and ensures better control over inventory.

6.2 FUTURE WORK

While the current version of the application provides a solid foundation for inventory management, there is significant scope for improvement and additional features. It is not feasible to implement all possible enhancements in the first iteration, but the following improvements are planned for future versions:

1. **Item Disposal Information:**

   The current version does not include functionality for tracking item disposal. This feature will be added in the next update to provide a complete lifecycle management solution for assets.

2. **Notification System:**
   A notification system will be integrated to alert inventory managers when an item request is submitted. This will ensure timely responses and improve the overall efficiency of the request process.
3. **Item Transfer Between Departments:**
   The ability to transfer items from one department to another will be implemented in the future. This feature will facilitate better resource sharing and utilization across the organization.
4. **Receiving Transferred Items:**
   A mechanism to receive and acknowledge items transferred from another department will also be developed. This will ensure proper tracking and accountability during inter-departmental transfers.

These planned enhancements aim to improve user experience, expand functionality, and provide a more comprehensive inventory management solution.

References

1. Lynn Beighley & Michael Morrison, 2008, “Head First PHP & MySQL”, 2nd Edition, O'Reilly Media, Inc.
2. Daginn Reiersol, Chris Shiflett, and Marcus Baker, 2007, “PHP In Action: Objects, Design, Agility”, 1st Edition, Manning Publications.
3. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, 1994, “Design Patterns: Elements of Reusable Object-Oriented Software”, Addison-Wesley Professional.
4. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 2020, “Database System Concepts”, 7th Edition, McGraw-Hill.
5. Sikha Bagui and Richard Earp, 2012, “Database Design Using Entity-Relationship Diagrams”, 2nd Edition, CRC Press.
6. “Development of Inventory Management System”, Conference Paper, 16-18 April 2010 [Accessed: June 1, 2023].
7. “PHP Manual”, Internet: https://www.php.net/manual/en/index.php, January 10, 2001 [Accessed: May 1, 2023].
8. “Laravel Documentation”, Internet: https://laravel.com/docs/10.x, June 09, 2011 [Accessed: June 1, 2023].
9. “The Apache HTTP Server Project”, Internet: https://httpd.apache.org, April 1995 [Accessed: April 1, 2023].
10. “Ultimate Entity Relationship Diagram Tutorial (ER Diagrams)”, Internet: https://creately.com/blog/diagrams/er-diagrams-tutorial, March 14, 2021 [Accessed: July 1, 2023].
11. “Design Pattern Overview”, Internet: https://www.tutorialspoint.com/design_pattern/design_pattern_overview.htm [Accessed: July 1, 2023].
12. “World Wide Web”, Internet: https://developer.mozilla.org/en-US/docs/Glossary/World_Wide_Web, July 23, 2021 [Accessed: July 24, 2023].
13. “Performance Evaluation of HTTP/TCP on Asymmetric Networks”, Internet: https://www.researchgate.net/publication/2925201_Performance_Evaluation_of_HTTPTCP, May 2004 [Accessed: July 10, 2023].
14. https://www.devopsschool.com/blog/complete-tutorials-of-php-oop-classes-objects-with-example-code, May 31, 2021 [Accessed: June 2, 2023].
15. “Pattern-based Analysis of UML Activity Diagrams”, Internet: https://www.researchgate.net/publication/228913582_Pattern-based_Analysis_of_UML_Activity_Diagrams, January 2004 [Accessed: July 10, 2023].