

**Invoice System: Design and Implementation for handling
business using web application**

BY

**Md. Mehedi Hasan
ID: 151-15-225**

&

**Md. Mushfiqur Rahman
ID: 152-15-511**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science Computer Science and Engineering

Supervised By

Md. Mahfujur Rahman
Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Tasniya Ahmed
Lecturer
Department of CSE
Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
DECEMBER 2018**

APPROVAL

This Project titled “Invoice System: A new approach for finding jobs based on web application”, submitted by- Md. Mehedi Hasan ID No: 151-15-225 and Md. Mushfiqur Rahman. ID No: 152-15-511 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on August 6, 2019.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman

Dr. S M Aminul Haque
Assistant Professor & Associate Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Saif Mahmud Parvez
Lecturer

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Mohammad Shorif Uddin

Professor
Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Md. Mahfujur Rahman, Lecturer, Department of CSE** at Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by

Md. Mahfujur Rahman
Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:

Tasniya Ahmed
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Md. Mehedi Hasan
ID: 151-15-225
Department of CSE
Daffodil International University

Md. Mushfiqur Rahman
ID: 152-15-511
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First of all we express our heartiest thanks and gratefulness to almighty Allah for divine blessing makes us possible to complete the final year project successfully.

We really thankful to Md. Mahfujur Rahman Lecturer, Department of CSE at Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “Invoice System” to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Dr. Masud Rana Rashel Assistant Professor, Department of CSE at Daffodil International University. He give us concept about project.

We would like to thank our entire course mate in Daffodil International University who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

6 August 2019

ABSTRACT

Now a days world is becoming digital by bless of modern technology. Web application is a crucial part of technology. Today, developed country using web applications in all sectors. Especially they are using in their business to look after their business. A dynamic web application can handle business properly and accurately. So made this project named “Invoice System” is for handling business. This invoice system will keep record of selling products, buying products, seller info, buyer information, selling and buying history, quick search option to found selling and buying history. It will show profit of owner during selling product. Dynamically it will create report by analyzing the record of buying and selling. Most interesting thing is that, it will send notification to owner when his product quantity is lower than 5. Moreover, it will rank top 5 products, buyer based on sell. It will create chart bar for showing the business status graphically. This system will automatically create invoice after selling and buying. Owner is allowed to print out the invoice for ensuring seller or buyer. This system is also mange bank account. User can transfer balance digit in bank account so that, user can understand how much balance he has in his bank.

TABLE OF CONTENTS

CONTENTS	PAGE
Title page	i
Approval	ii
Declaration	iii
Acknowledgement	iv
Abstract	v
Table of Contents	vi - x

CHAPTER

CHAPTER 1: INTRODUCTION	1
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	1
1.4 Expected Outcome	1
CHAPTER 2: BACKGROUND	2-3
2.1 Introduction	2
2.2 Related work	2
2.3 Comparative Studies	2
2.4 Scope of the Problem:	3
2.5 Challenges:	3

CHAPTER 3: REQUIREMENT SPECIFICATION	4-14
3.1 Business Process Model:	4
3.2 Requirement Collection and Analysis:	4
3.2.1 User Requirements:	5
3.2.2 Technical and Legal Requirement:	5
3.2.3 Requirement Components:	5
3.2.3.1 Hardware Requirement:	5
3.2.3.2 Software Requirement:	6
3.3 Use Case Modeling and Description:	7
3.3.1 Use Case Model:	7
3.3.2 Use Case Description:	8-12
3.4 Logical Data Model:	13
3.5 Design Requirments:	13
3.5.1: System user Requirements:	14
3.5.2: Seller and buyer Requirements:	14
CHAPTER 4: DESIGN SPECIFICATION	15-
17	
4.1 Front-end Design:	15
4.1.1 Hypertext Markup Language:	15
4.1.2 Cascading Style Sheet:	15
4.1.3 JavaScript:	16
4.2 Back-end Design:	16
4.2.1 Hypertext Transfer Protocol:	17
4.2.2 Types of HTTP request:	17

CHAPTER 5: IMPLEMENTATION AND TESTING	18-
27	
5.1 Implementation of Database:	18
5.1.1 Entity Relationship Diagram:	18
5.2 Implementation of Front-end Design:	19
5.2.1 Dashboard Page:	19-20
5.2.2 Register seller and buyer:	20-21
5.2.3 Setup product's brand name and model name:	21-22
5.2.4 Sell page and buying page:	22-23
5.2.5 Invoice record list and invoice creating:	24
5.2.6 Returning Product for (seller or buyer):	25
5.2.7 Store balance in bank:	26
5.2.7.1 Bank Lists Display:	26
5.3 Implementation of Interactions:	27
5.4 Implementation of Testing:	27
5.4.1 Main System Testing:	27
5.4.2 Database Testing:	27
5.5 Test Results and Reports:	27
CHAPTER 6: CONCLUSION AND FUTURE SCOPE	28
6.1 Discussion and Conclusion:	28
6.2 Scope for Further Developments:	28
REFERENCES	29

LIST OF FIGURES

FIGURES	PAGE
Figure 3.1 Business Process Modeling	4
Figure 3.2 Use Case Model	7
Figure 3.3 Logical Data Model	13
Figure 5.1 E-R diagram of Database	18
Figure 5.2 System users dashboard (analysis day by day)	19
Figure 5.3 System users dashboard (graphical report view)	20
Figure 5.4 Register seller and buyer option	21
Figure 5.5 Setup product's brand name and model name	22
Figure 5.6 Selling page functionality	23
Figure 5.7 Invoice records and invoice creating	24
Figure 5.8 Return and returns calculation	25
Figure5.9 Live contest	26

LIST OF TABLES

TABLE NAME	PAGE
Table 3.1 Hardware Requirement:	5-6
Table 3.2 Software Requirement:	6
Table 3.3 Log In:	8
Table 3.4 Register seller & buyer name:	8-9
Table 3.5 Setup product name and model name:	9
Table 3.6 Record sell and buying history:	10
Table 3.7 Return product :	10-11
Table 3.8 Invoice creating:	11
Table 3.9 Receive invoice:	12
Table 3.10 Receive final report :	12

Chapter 1: Introduction

1.1 Introduction

This project is developed for handling business. So that business owner can look after his business accurately. This system will create invoice. User will allowed to print that to confirm selling and buying.

1.2 Motivation

Now, we are living in smart world. We are getting huge opportunity to make our work more simple and accurate. That's way we made this project for handling our business simply. In our country, we need 4-5 sells-man for a running a small business. It is costly for owner to bear cost of sells-man. On the other hand if we think about large business then owner need many sells man and they record there every selling and buying record into note book which can be cause of mistake for huge and large business. Even it is difficult to find out which product is demand is high or which buyer taking product more than others. For solving those problem we made this project where everything will be recorded dynamically and it will show status (ups and down) based on owners input and business condition.

1.3 Objectives

- Handle whole business accurately.
- Reduce time & cost of business owner.
- Store all records of seller and buyer.
- Analyze data dynamically for current trends product.
- Create invoice with accurate calculation
- Provide notification for lower quantity products

1.4 Expected Outcome

We developed this dynamic system with interactive design so that a business man (owner) can handle his whole business using a single application. It will save owner's money, time and it will more accurate answer.

Chapter 2: Background

2.1 Introduction

This system is a web application. We used modern technology to make this project. We used Hml5, Css3, Java Script, and JQuery to complete our front end. For making our application more secured we used Laravel framework. Our system is so fast for searching data we used there JS data table plug in library. Another important thing is when system taking data from database, the data passing via Ajax which save user's time.

2.2 Related work

In the field of web there is lot of work already have been done. And there are some specialized Systems have been created by some company. Some honorable mention are <https://invoicely.com/> [1]. But in some cases we are totally different than others.

2.3 Comparative studies:

Purpose: The purpose of this paper was to compare with invoicely.com. But this site only allowed to create invoice all inputs are given by owner, we worked on this so that system take some input dynamically.

Design/Methodology/Approach: This system will give opportunity to business owner to take control over his business

Originality/Value: The paper will be of value to those interested for handling business accurately and want to get business status to take action to improve his business.

2.4 Scope of the Problem:

Scopes for the system are as follows: -

- Handling business
- Reduce cost and man power

2.5 Challenges:

Now a days people are more interested for android application. Business owner is a busy person so, many time he stay outside from is shop. If this application could be integrate in mobile application then they could know about their business from fair. So we have an idea to integrate this web app into mobile app.

Chapter 3: Requirement Specification

3.1 Business Process Model:

Business process modeling is the graphical representation of a company's business processes or workflows, as a means of identifying potential improvements. This is usually done through different graphing methods, such as the flowchart, data-flow diagram, etc.

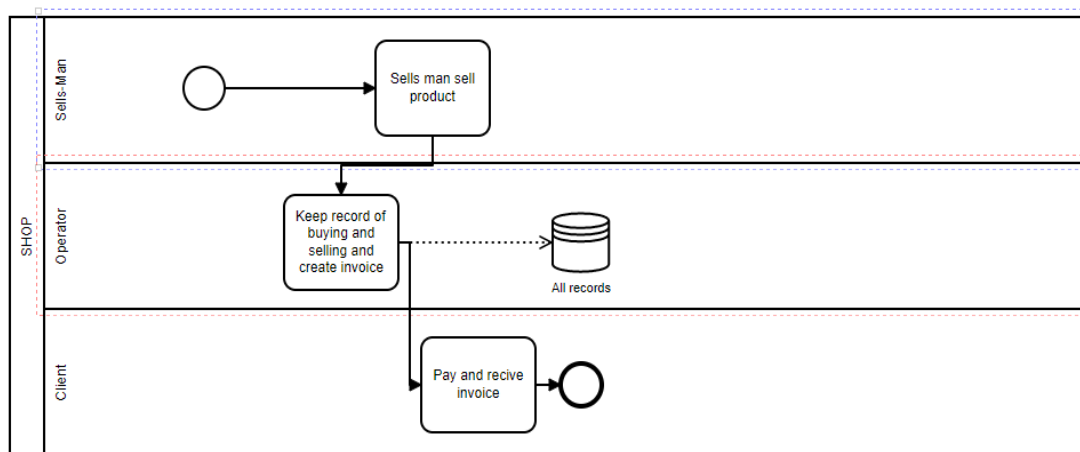


Figure 3.1: Business Process Modeling.

3.2 Requirement Collection and Analysis:

This project is invoice system. This application is created and developed based on some requirements.

3.2.1 User Requirements:

- User is allowed to register name of seller and buyer by clicking on register button. User must fill all input type such as name and location and mobile no. For every sell and buy user must record history into system.

3.2.2 Technical and Legal Requirement:

The system should be design as to ensure that they continue to work efficiency that the observation with relevant legislation and to check that they are safe guarded from threats such as virus and hackers. The requirement are listed below:

- Data protection facilities and Security data transmission facilities.
- Protection facilities against hacker and cracker and Protection against virus.
- Protection Company or Candidate Data From Hacker

3.2.3 Requirement Components:

To carry out project we need hardware and software and those are given below:

3.2.3.1 Hardware Requirement:

To run this project minimum one pc and INTERNET connection need.

Processor	Intel Pentium/AMD processor(500 MHZ)
Motherboard	Any
Ram	1 GB
Lan	Any
AGP Card	Any
Sound Card	Any

Hard Disk	60GB
Floppy Disk(not mandatory)	1.44 MB
Casing	ATX
Monitor	Any Color Monitor
Keyboard	Any
Mouse	Any
CD ROM	52X

Table 3.1 Hardware Requirement.

3.2.3.2 Software Requirement:

Different Type of software need to developing and maintaining projected website.
Details Given below:

Software	Usage
Any version of windows operating system	To start up computer and coordinate all hardware components, application and customized software.
Browser	Google chrome is recommended but any kind of web browser which support JavaScript and CSS3.

Table 3.2 Software Requirement

3.3 Use Case Modeling and Description:

Use case diagram is a behavioral UML diagram type and frequently used to analyze various systems. They enable you to visualize the different types of roles in a system and how those roles interact with the system

3.3.1 Use Case Model:

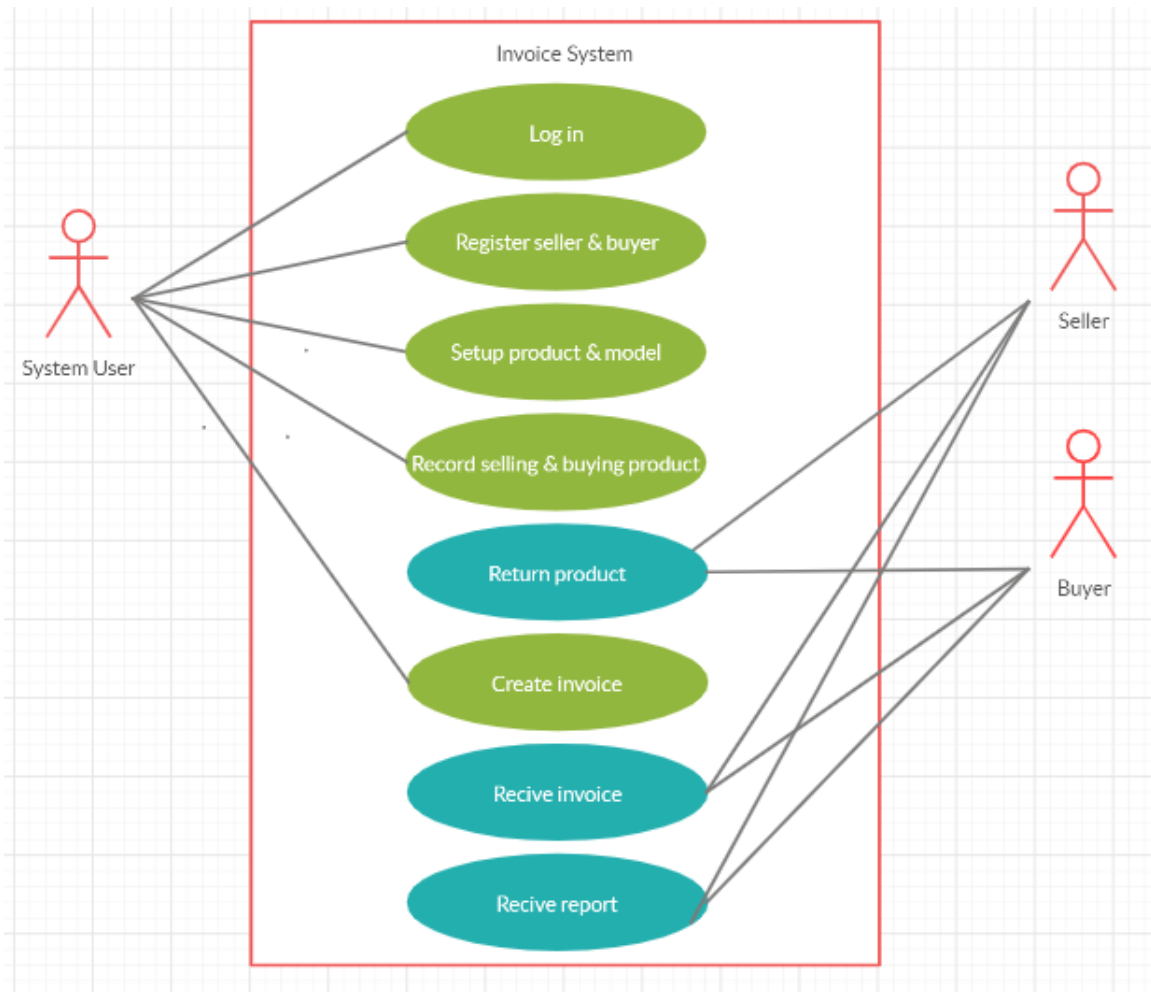


Figure 3.2 Use Case Model

3.3.2 Use Case Description:

Table 3.3 Log In

Use Case Id:	1
Use Case Name:	Log In
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow user to log in in system for controlling his business via application
Primary Actor:	System user
Secondary Actor:	None
Precondition:	None
Post condition:	The system will display the Log In page

Table 3.4 Register seller & buyer name

Use Case Id:	2
Use Case Name:	Register seller & buyer name
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow user to register seller and buyer name, address.
Primary Actor:	System user
Secondary Actor:	None

Precondition:	None
Post condition:	System will display seller and buyer name, address in list and selling panel

Table 3.5 Setup product name and model name

Use Case Id:	3
Use Case Name:	Setup product name and model name
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow user to set up product name and model name
Primary Actor:	System user
Secondary Actor:	None
Precondition:	None
Post condition:	The system will product with model no in selling and buying page

Table 3.6 Record sell and buying history

Use Case Id:	4
Use Case Name:	Record sell and buying history
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow system user to record selling buying products
Primary Actor:	System user
Secondary Actor:	System
Precondition:	Setup product and model name
Post condition:	None

Table 3.7 Return product

Use Case Id:	5
Use Case Name:	Return product
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow to return product of seller and buyer
Primary Actor:	Seller or buyer
Secondary Actor:	System

Precondition:	Save selling and buying record
Post condition:	It will display in return list

Table 3.8 Invoice creating

Use Case Id:	6
Use Case Name:	Invoice creating
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow to create invoice seller and buyer
Primary Actor:	System user
Secondary Actor:	system
Precondition:	Seller or buyer should sell or buy something from user
Post condition:	System will create invoice

Table 3.9 Receive invoice

Use Case Id:	7
Use Case Name:	Receive invoice
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow to receive invoice for seller or buyer
Primary Actor:	Seller or buyer
Secondary Actor:	None
Precondition:	Buying or selling something
Post condition:	Display selling or buying records in invoice list

Table 3.10 Receive final report

Use Case Id:	8
Use Case Name:	Receive final report
Created By:	Mehedi Hasan
Date Of Creation:	6-7-2019
Description:	This use case will allow to receive report for seller or buyer
Primary Actor:	Seller or buyer
Secondary Actor:	None
Precondition:	Buying or selling something
Post condition:	Display selling or buying records in report list

3.4: Logical Data Model:

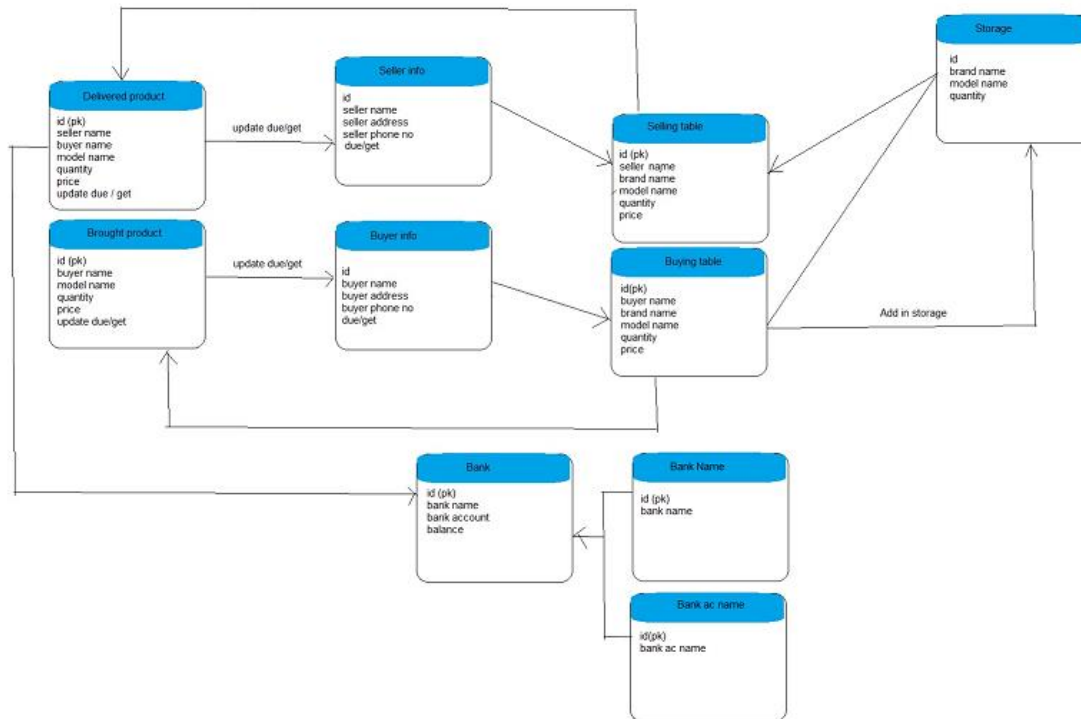


Figure 3.3 Logical Data Model

3.5: Design Requirements:

An **invoice** is a document sent by a seller to a buyer that officially requests payment for goods or services supplied. The issuing of an **invoice** indicates that the buyer must now settle the amount due with the seller according to the set out payment terms.

3.5.1: System user Requirements:

- Register seller
- Register buyer
- Setup product's brand name
- Setup product's model name
- Deliver product to sell
- Deliver product to sell

3.5.2: Seller and buyer Requirements:

- Return product
- Collect invoice
- Collect report

Chapter 4: Design Specification

4.1 Front-end Design:

For finishing front-end improvement, we've used three primary languages in my project.

- HTML5
- CSS & CSS3
- Bootstrap framework
- JavaScript
- libraries like JQuery

4.1.1 Hypertext Markup Language:

HTML elements unit of measurement the building blocks of Web pages. With markup language constructs, pictures and alternative objects like interactive forms are also embedded into the rendered page. HTML provides a way to form structured documents by denoting structural linguistics for text like headings, paragraphs, lists, links, quotes and alternative things. HTML components square measure diagrammatical by tags, written exploitation angle brackets. Tags such as `` and `<input />` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page [5].

4.1.2 Cascading Style Sheet:

It is a decent practice to isolate page structure from its plan. To do as such, CSS is utilized as a styling dialect. It characterizes format, shading, estimate and different traits of page components. CSS report is a content record containing an arrangement of CSS

rules, where rule comprises of selector and presentation square. Each standard influences page content indicated in selector guiding either toward: All components of one kind in a DOM, e.g. all labels [10].

- Elements of an equivalent class or id (both are label traits). In CSS, these standards begin with a spot (.) and hash (#) for classes and ids, separately, e.g. ".list" and "#examplebtn".

4.1.3 JavaScript:

JS is interpreted programming language generally used at client side to control user input, control browser actions, load additional content to a browser and make changes to the DOM. In short, it gives functionality to a page [10]. JS code can be added to HTML document in two ways

- Inline JavaScript, JS code is written as content of HTML document between opening and closing <Script> tag.
- External JavaScript, JS code is written in external file and reference to the file is defined in <script> tag attribute. Any content enclosed by the <script> tag is not executed [6].

When HTML <script> element is reached, during parsing of HTML document, the JS code defined by the tag is immediately executed in a browser using built-in JS interpreter.

4.2 Back-end Design:

To find, save, or change data and serve it back to the user in front-end code, we've used

- Server-side languages- Laravel
- Tools like MySQL Server

4.2.1 Hypertext Transfer Protocol:

HTTP is a protocol describing how browsers and servers communicate with each other over the Internet in order to exchange data. The process works as follows:

- Establishment of a connection between client and WA server.
- If the connection is successful request is sent by the client. It is a message requesting data from a WA server. The data can be of various types - web pages, images, client-side scripts, confirmation of user authentication and more [10].
- The server then sends response which contains requested data together with a status code. There are several types of status codes depending on success or failure of the requested operations.
- Last step closes the connection by either both parties. Both, request and response, are text based messages, each message has three parts :
 - I. an initial line, defines mainly source of information,
 - II. Header, zero or more lines specifying additional parameters of the request.
 - III. An optional message body contains data sent by client/server.

4.2.2 Types of HTTP request:

When user accesses the WA for the first time, browser sends an unconditional HTTP requests to a server. Server sends back the requested data and the browser may cache the data for later use, if the response's header allows it [10].

Chapter 5: Implementation and Testing

5.1 Implementation of Database:

The database implementation or deployment is the process of installation of database software, configuration and customization, running, testing, integrating with applications, and training the users [11]. Its different stages and processes are:

5.1.1 Entity Relationship Diagram:

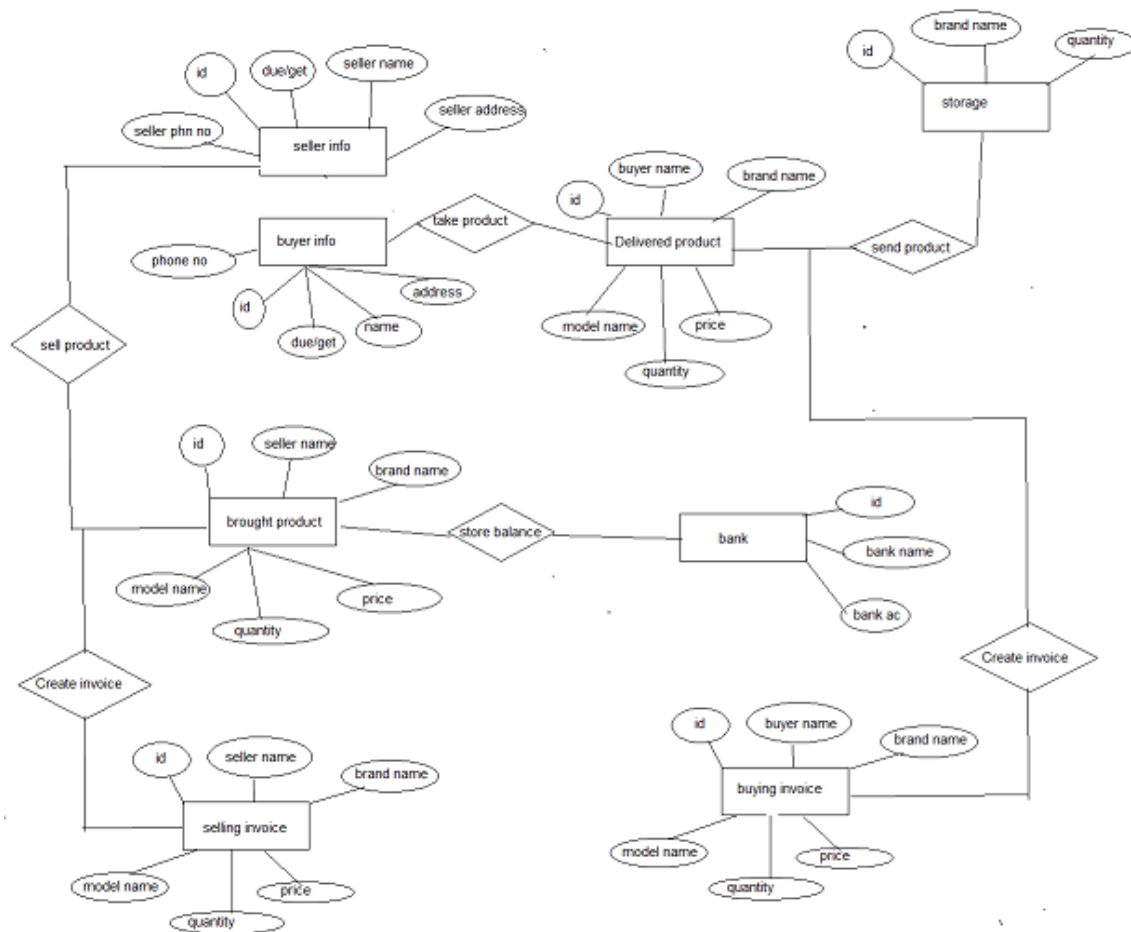


Figure 5.1 E-R diagram of Database

5.2 Implementation of Front-end Design:

Improvement of the customer side of WA is called front-end advancement and front-end (FE) itself is then comprehended as each substance client can see and can connect within a program. It is typically a blend of Hypertext Markup Language (HTML), Cascading Style Sheet (CSS) and JavaScript (JS). Every one of these dialects is deciphered and controlled by an internet browser bringing about a website page - interface client can cooperate with. Set of all accessible pages is called site.

5.2.1 Dashboard Page:

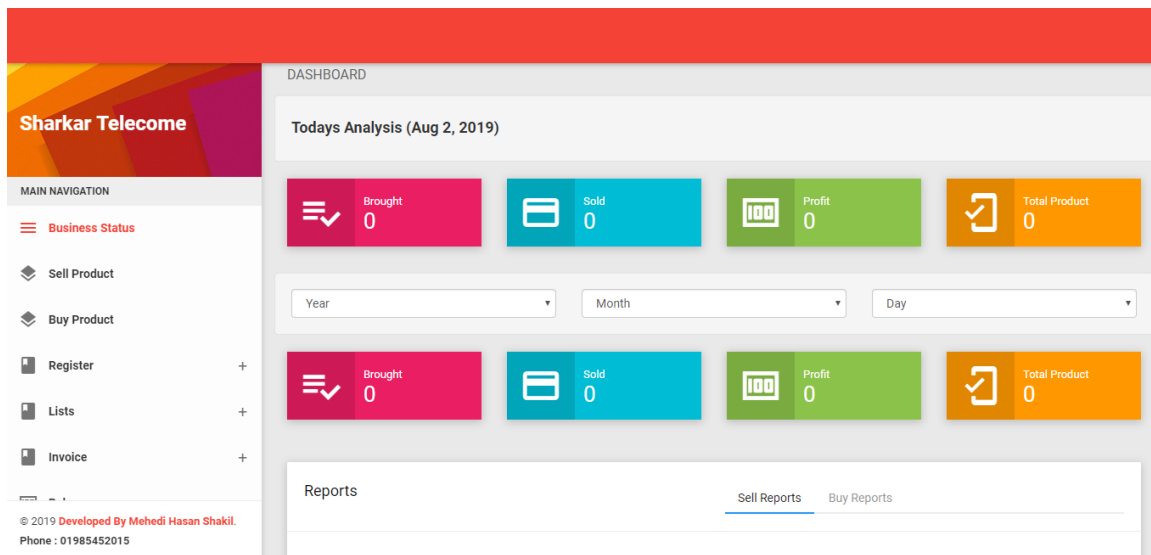


Figure 5.2 System users dashboard (analysis day by day)

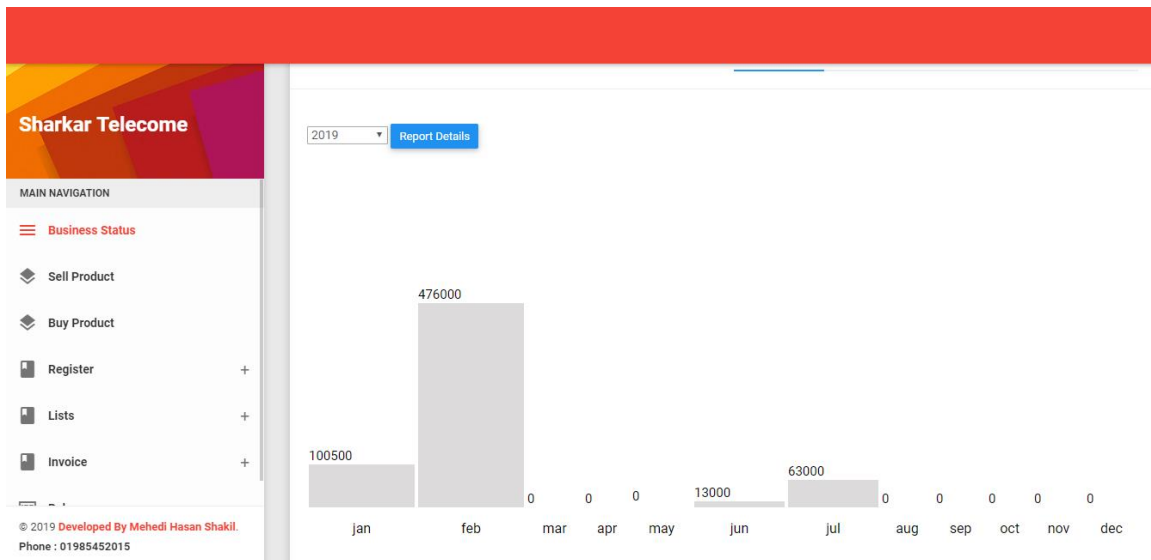


Figure 5.3 System users dashboard (graphical report view)

Output: This is a dashboard page for system users where system user can analyze every day's sells and buy and also can see graphical view of business statistics.

5.2.2 Register seller and buyer:

The screenshot shows the 'Register Seller' form on the Sharkar Telecom dashboard. The form is titled 'Please Register Seller Info' and contains the following elements:

- Input field for Seller Name
- Input field for Seller Address
- Register Seller button (green)

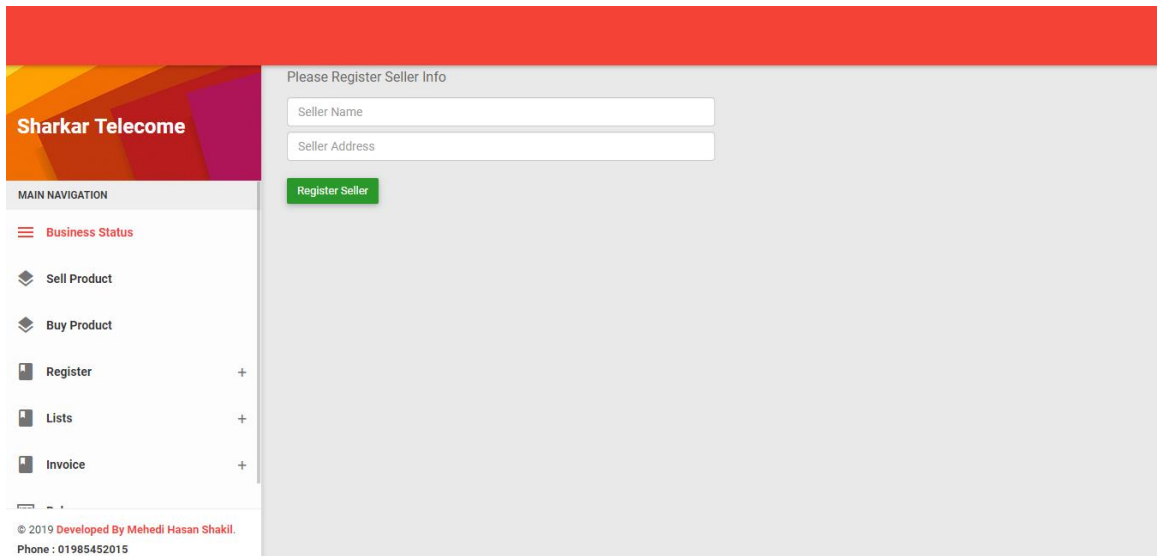
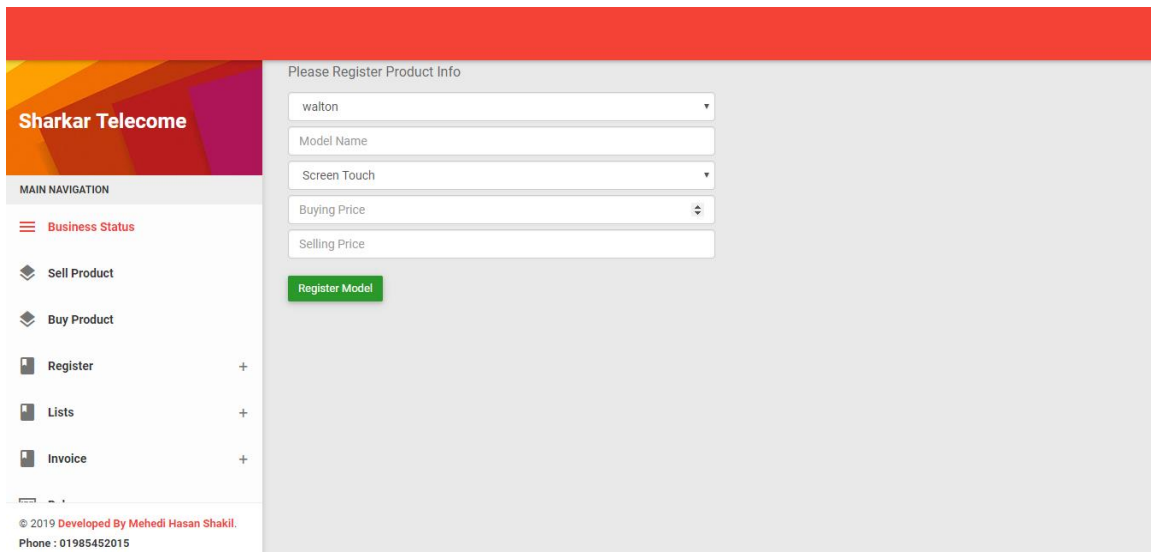


Figure 5.4 Register seller and buyer option

Output: Here all seller and buyer has been store by registration in a database.

5.2.3 Setup product's brand name and model name:



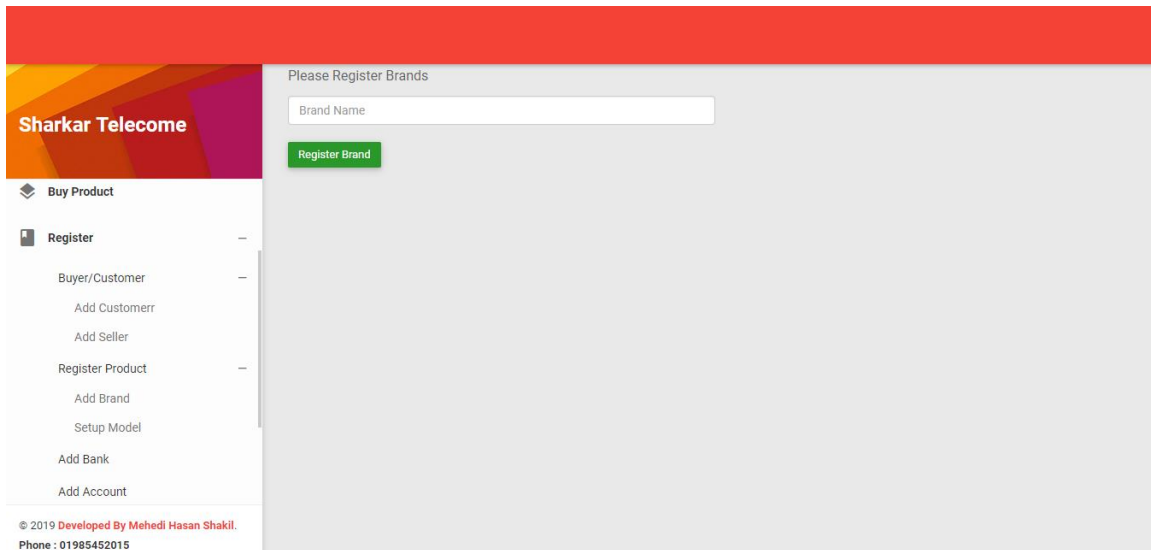
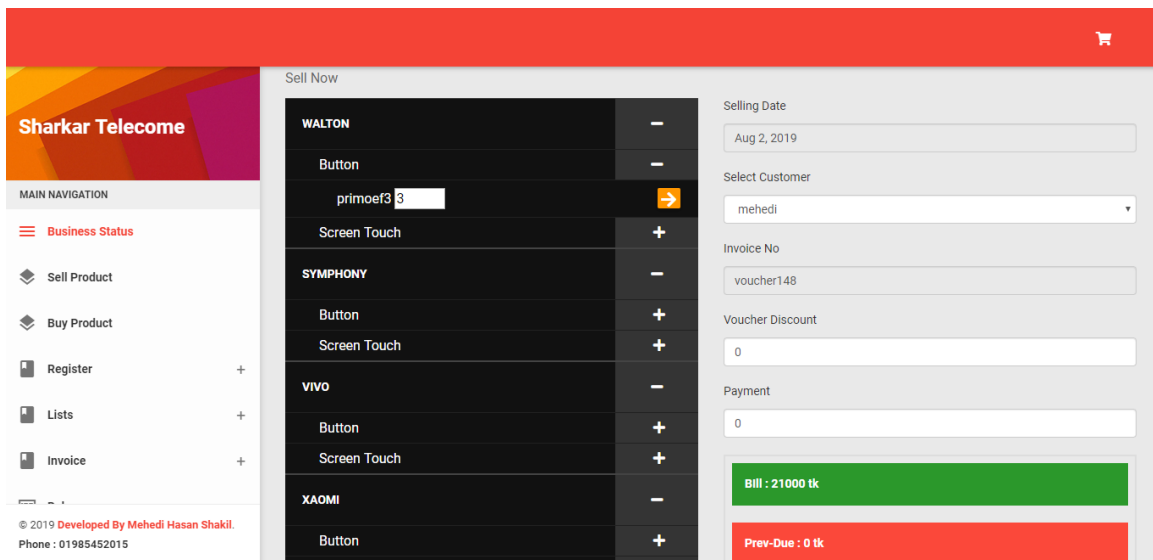


Figure 5.5 Setup product's brand name and model name

Output: System user will be allowed to register product's brand name and model name and this data will come into selling page to sell.

5.2.4 Sell page and buying page:



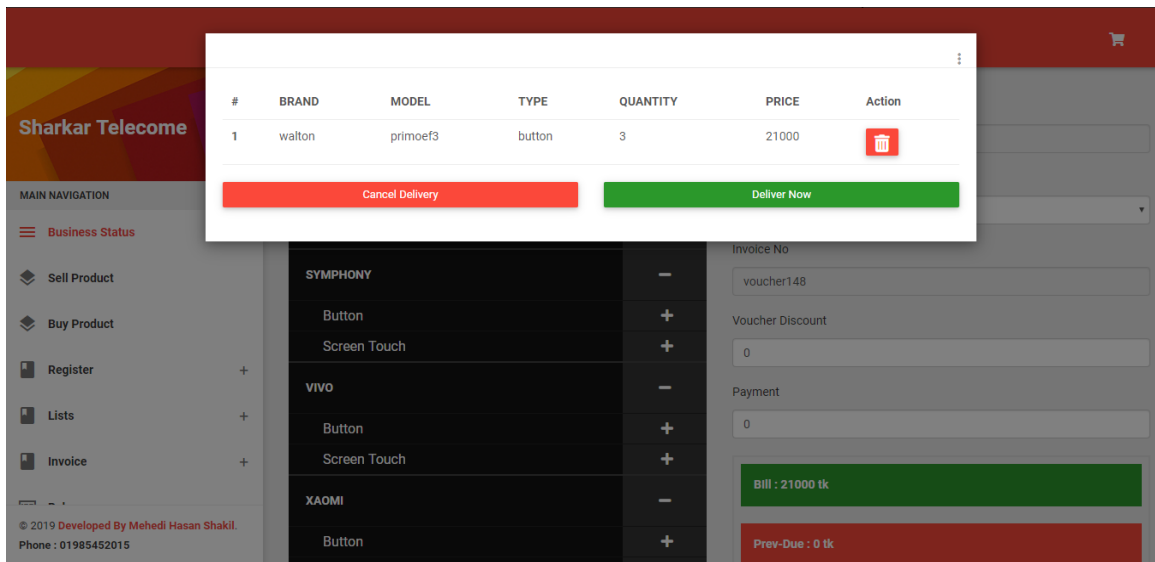
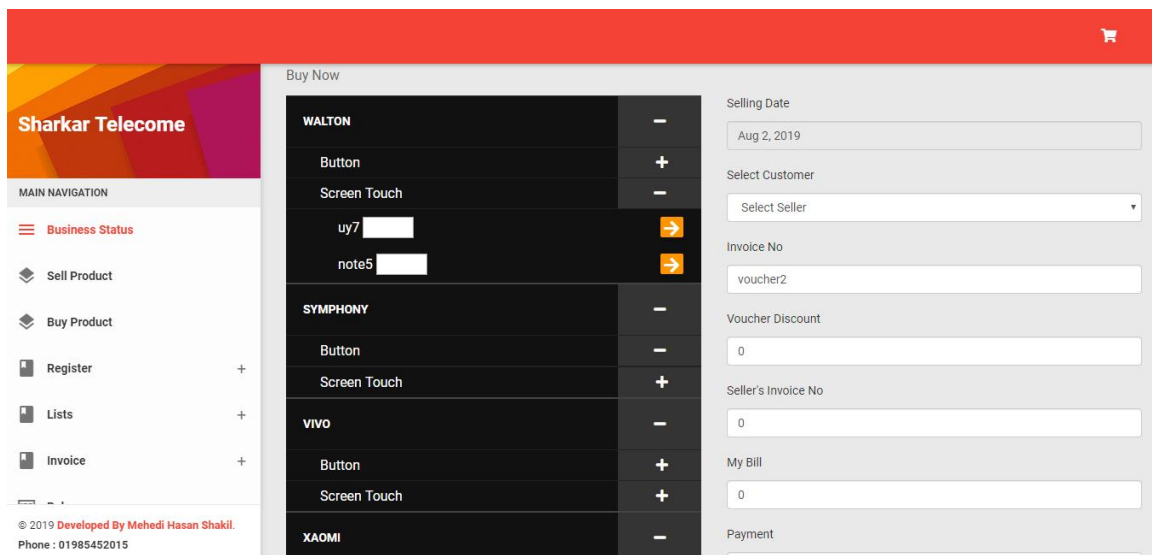


Figure 5.6 Selling page functionality



Output: System user will sell and buy product using those page.

5.2.5 Invoice record list and invoice creating

The image displays two screenshots of the Sharkar Telecom web application interface. The top screenshot shows the 'Boucher List' page, which features a sidebar with navigation options like 'Business Status', 'Sell Product', 'Buy Product', 'Register', 'Lists', and 'Invoice'. The main content area contains a table of vouchers with columns for Customer Name, Voucher No, Date, and Actions (View, Edit, Delete). The bottom screenshot shows the 'Invoice No : voucher139' detail view, which includes a summary of the invoice (Customer Name, Invoice No, Delivery Date, Invoice Discount, Total Bill, Paid amount, and Still Due status) and a table of items with columns for Brand, Model Name, Type, Quantity, and Bill.

Boucher List

Customer Name	Voucher No	Date	Actions
fdgfg	voucher139	Jun 30, 2019	View Edit Delete
ioiouyi	voucher145	Jul 9, 2019	View Edit Delete
mehedi	voucher136	Jun 30, 2019	View Edit Delete
mehedi	voucher141	Jul 1, 2019	View Edit Delete
mehedi	voucher144	Jul 9, 2019	View Edit Delete
mehedi	voucher146	Jul 11, 2019	View Edit Delete
mehedi	voucher147	Jul 13, 2019	View Edit Delete

Invoice No : voucher139

Sharkar Telecom

Customer Name : fdgfg
 Invoice No : voucher139
 Delivery Date : Jun 30, 2019
 Invoice Discount : 0 %
 Total Bill : 476000 TK
 Paid : 0 TK
 Still Due : Customer is deleted !

Brand	Model Name	Type	Quantity	Bill
walton	primoeF3	button	68	476000

Figure 5.7 Invoice records and invoice creating

Output: After selling and buying system will create invoice automatically and it will also make list of all invoices.

5.2.6 Returning Product for (seller or buyer)

The screenshot displays the 'Sell Now' screen in the Sharkar Telecom app. The interface is divided into three main sections:

- Navigation Menu (Left):** Includes 'Business Status', 'Sell Product', 'Buy Product', 'Register', 'Lists', and 'Invoice'.
- Product Selection Table (Center):** A table with columns for brand, product type, and quantity. Brands listed include WALTON, SYMPHONY, VIVO, and XAOMI. Product types are 'Button' and 'Screen Touch'. A search bar with 'primoeof3' and a search icon is present.
- Transaction Form (Right):** Contains fields for 'Selling Date' (Jun 30, 2019), 'Selected Customer' (mehedi), 'Voucher No' (voucher136), 'Voucher Discount' (0), 'Paid Before' (0), and 'Payment' (0).

Below the form, a summary of the transaction is shown:

- Current Price : 34000 tk
- Prev-Due : 0 tk
- Total with due : 34000 tk
- Owner-get: 34000 tk

A 'Check Profit' button is located at the bottom of the form.

Figure 5.8 Return and returns calculation

Output: System user allowed to edit after buying and selling products.

5.2.7 Store balance in bank

The screenshot shows the Sharkar Telecom dashboard. At the top, there are three colored boxes: 'Get: 0 Tk' (orange), 'cash: 53567 Tk' (green), and 'Due on market: 0 Tk' (red). Below these, a form prompts the user to 'Please first select Bank name and then select ac'. The form includes three input fields: 'Select Bank' (a dropdown menu), 'Select Account' (a dropdown menu), and 'amount' (a text input field). To the right of the form, a green box with a banknote icon displays 'Total amount (Bank) : 53008'. On the left, a sidebar contains a 'MAIN NAVIGATION' menu with items: Business Status, Sell Product, Buy Product, Register, Lists, and Invoice. At the bottom of the sidebar, it says '© 2019 Developed By Mehedi Hasan Shakil. Phone : 01985452015'.

5.2.7.1 Bank Lists Display:

The screenshot shows the 'Bank Balance List' section of the Sharkar Telecom dashboard. At the top, there are buttons for 'Copy', 'CSV', 'Excel', 'PDF', and 'Print', along with a search bar. Below these is a table with the following data:

Bank Name	Account No	Balance	Delete
prime	1215478	50000	Delete Bank Delete Account
prime	123	3008	Delete Bank Delete Account

Below the table, it says 'Showing 1 to 2 of 2 entries' and 'Previous 1 Next'. On the left, the same 'MAIN NAVIGATION' menu is visible, and at the bottom, it says '© 2019 Developed By Mehedi Hasan Shakil. Phone : 01985452015'.

Figure 5.9 Live contest

Output: System user can submit balance into bank account and all bank account and balance will display as list.

5.3 Implementation of Interactions

Programming frameworks created as of late are ending up progressively intense, yet by and large they tend to desert the client to manage the multifaceted nature of the framework alone. Implementing the software design into the code and form design is the most significant part of the software [8]. This is the development phase of the application. In this phase codes are written and necessary requirements are assembled to build the software. There is a gigantic requirement for frameworks with individual, setting delicate help. The paper talks about the improvement of such an emotionally supportive network with exceptional spotlight on the UI. The augmentation of a current and complex program.

5.4 Implementation of Testing

5.4.1 Main System testing

Being the - landing page of the deployment entryway was tried for appropriate network to the database. Inappropriate advised linkage to the database was instantly corrected and confirmation was made to see that information were made enough recovered and exhibited without mistakes.

5.4.2 Database testing

Database testing was annoyed tested for network and capability. The test began with the plan organize, where activities were guaranteed that the right information representation was made.

5.5 Test Results and Reports

The performance of the system was very near the expected result .Throughout this structure, the advancement was done in modules. Every module was executed independently and tried. Extremely arrange involves troubleshooting, which was done and all blunder settled. The whole framework was coordinated by joining every module to work with the principle framework" At the finish of the combination, the whole framework was tried and confirmed Ok [6].

Chapter 6: Conclusion and Future Scope

6.1 Discussion and Conclusion

An online based application is always best. This is an online based application but we want to make it more dependable on online so that it can give more flexibility to the user of the system. We are working on it more to make more and more powerful.

6.2 Scope for Further Developments

A project is always incomplete. However, we want to bring more new feature for this application so that this system can take challenge and can compete with other application related this application.

- Online payment system
- Online order system
- Online booking system for product
- More graphical view for better understanding the business condition
- Payment integration system

REFERENCES

- [1] Taken concept to create in voice system application. Idea was taken from
<http://bigprof.com/appgini/applications/online-invoicing-system> [Accessed 8 august 2019].
- [2] How to design database for invoice system.
<https://itextpdf.com/en/resources/books/zugferd-future-invoicing/3-simple-invoice-database> [Accessed 8 august 2019].
- [3] Taken ides form Online invoice
<https://www.onlineinvoices.com/>
- [4] Invoicely" invoicely.com" .:
[https:// invoicely.com](https://invoicely.com) [Accessed 7 september 2019].
- [5] Beautiful business (How to make a invoice)
<https://www.xero.com/us/resources/small-business-guides/invoicing/invoicing-survival-guide/how-to-make-an-invoice/>

