B2B MANAGEMENT SYSTEM FOR WHOLESALER AND RETAILERS

DESIGNED AND DEVELOPED

BY

Shubhkaram Singh Thind

THIRD YEAR BACHELOR OF SCIENCE

(INFORMATION TECHNOLOGY)

PRN : 2020430096

DEPARTMENT OF INFORMATION TECHNOLOGY

VPM's B.N.Bandodkar College of Science (Autonomous),Thane

(*Affiliated to University of Mumbai*)

THANE(W) - 400 601

MAHARASHTRA

ACADEMIC YEAR: 2022-23

CERTIFICATE OF B2B MANAGEMENT SYSTEM

To whomsoever it may concern

Respected Sir,

I hereby certify that the following namely one“ Mr. SHUBHKARAM SINGH THIND” has been visiting our “B2B MANAGEMENT SYSTEM” warehouse regularly for past 3 months to study the various activities related to our organization.

The information related to the software design and other data contained in this report does not necessarily reflect our own.

Our guidance has been primarily present in the day to day activities of the organization and helps him in preparing the project report. The format and details used on all the pages have been entirely designed to suit the project report.

Regards,

(Client Signature)

ABSTRACT

In an era where having a computer an integral part of your operations is a matter of prestige for some and necessity for others. Management of these projects is still an area which is ignored or done in traditional manner. We in our system called **B2B MANAGEMENT SYSTEM FOR WHOLESALERS AND RETAILERS** is integrated and automation software for wholesalers and retailers. The proposed system reduce the hardship for a client in maintaining & protecting the records in long term. Unlike excel all the data will be store in a secure data base which can easily be updated which in return help a client in saving time. All Data will be safe and secure with the client and it will be easy to access that data. All the client has to do is just go in the software and edit the information like of retailer, orders, stocks and they can also generate invoice by simply putting the information such as price i.e. in numbers and all calculations is done by itself. This receipt will be more detailed than old one and reduce most of the human errors and save the time of the client.

We hope that this package would prove to be an excellent environment for simpler for end user.

ACKNOWLEDGMENT

I am glad to present my project **B2B MANAGEMENT SYSTEM FOR WHOLESALERS AND RETAILERS**.

For everything I have achieved, the credit goes to all those who offered me invaluable assistance and guidance to make the project.

I take this opportunity to express my soulful gratitude management of

B. N. BANDODKAR COLLEGE OF SCIENCE for giving this opportunity to accomplish this project work.

I am thankful to our project guide MR. TEJAS JADHAV for most sincere, useful & encouraging contribution throughout the project span. Without their support we wouldn't be able to complete the project on time.

I am highly obliged to the teaching members of the Computer Science & IT who took efforts to make the project a successful endeavour. I would also like to thank non-teaching staff members.

I would also like to thank my project member ABHISHEK DUBEY for developing this project along side with me , appreciate his support to me and all his efforts to make the project successful. Last, I extend my sincere thanks and appreciation to my family for supporting me a lot in finalizing this project within the limited time frame.

DECLARATION

I here by the declare that the project entitled, "**B2B MANAGEMENT SYSTEM FOR WHOLESALERS AND RETAILERS**" done at B. N. Bandodkar College of Science, has not been in any case duplicate to submit to any other university for the award of any degree. To best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirement for the award of degree of BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as final semester project as part of our curriculum.

Name and Signature of the Student

SYNOPSIS

Introduction:

This is a Web Application of Inventory Management for Wholesalers and Retailers developed to reduce the problem in old manual system. The client has challenges for managing the information Of wholesale products, distributor, recent order placed, maintaining record, retailer's can easily track wholesalers in particular location. Our Software deals with these challenges. It is User friendly and client can easily use it.

Main Objective of Project:

For Wholesalers:

- To Store Data in Data Base.
- Smooth Business Communication with Retailers.

For Retailers:

- Different Wholesalers available on the platform for different items.
- Be able to contact and place orders to Wholesalers.
- Smooth Business Communication with Wholesalers.

EXISTING SYSTEM:

Currently the old method i.e., storing order information, retailer's information is being manually written or excel sheets is being used. After long duration of time it gets hard to manage and maintain manually written and stored data . With excel sheets after some time it will become very difficult for client to go through the recorded data. Retailers also face problems like not able to find wholesalers in areas for the product which they want.

PROPOSED SYSTEM:

The proposed system reduce the hardship for a client in maintaining & protecting the records in long term. Unlike excel all the data will be store in a secure data base which can easily be updated which in return help a client in saving time. All Data will be safe and secure with the client and it will be easy to access that data. All the client has to do is just go in the software and edit the information like of retailer, orders, stocks. This will reduce most of the human errors and save the time of the client.

SURVEY OF TECHNOLOGY:

Front-end:- Html, CSS, Javascript

Back-end:- Python/Django, PostgreSQL database

SOFTWARE AND HARDWARE REQUIREMENTS:-

Software:-

Windows 10 and higher

Hardware:-

RAM:- Minimum 4GB or higher

Storage:- 120GB or Higher

Processor:- 64-bit

TABLE OF CONTENTS

INTRODUCTION

1.1 Background:

The B2B management system is for wholesalers and retailers to get all required facilities regarding the b2b marketplace. The main aim of software is to develop the best management system for wholesaler to overcome the manual system which is more time consuming and costly due to which whole process tends to be slow & where a retailer can shop for the items in bulk more easily and efficiently. This application will help them by reducing the workload and by simplifying the business. The users want to change from paper based system that is manual system to computerized system. Currently the Client is unable to fulfill all the requirements of the customer thereby some ambiguity is left while providing the service. These traditional techniques are wasting of time and money . On top of that maintaining records on papers is very difficult because most of papers gets misplaced and even has great chance of getting ragged over a period of time and hence the client will have to rewrite the information again. Hence, this project will focus on canceling the paperwork and to digitalize information so it will never get misplaced or lost.

1.2 Feasibility Study:

The main objective of project is to provide better communication and better business management between wholesalers and retailer's . All projects are feasible given unlimited resources and infinite time. Unfortunately, the development of computer-based system in many cases is more likely to be plagued by scarcity of resources and delivery date. Hence, we have made use the concept of reusability which is achieved by Object-Oriented approach .The Feasibility report of the project holds the advantages and flexibility of the project.

Here are various measures of feasibility that helped us to decide whether our project is feasible or not. These measures include :-

Economic Feasibility:

Economic analysis is the most frequently used method for evaluating the effectiveness of the candidate system. More commonly known as cost/benefit analysis, the procedure is to be determining the benefits and savings that are expected from a candidate and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system.

The proposed system is economically feasible because-

- The system will have user friendly interfaces and very less user-training is required to use.
- The system will provide fast and efficient environment instead of slow and error prone manual system, thus reducing both time and man power spent in the running system.

Technical Feasibility:

Technical feasibility centres on the existing computer system (Hardware and Software etc.) and to what extend it supports the proposed system. The operating system can be windows or Linux. Hardware requirement is RAM memory minimum 4GB or higher, Processor is 64-bit and Storage is 120GB or higher, the version of python 3.6 or higher is required. In this project all the necessary cautions have been taken care to make it technically feasible. . The system is compatible with the current computer systems.

Schedule Feasibility:

If a project takes too long to finish before it is beneficial, it will fail. But this project can be completed within the timeframe provided and will be successful. The timeframe identified is 6 months.

Behavioural Feasibility:

The customer of the system i.e., wholesaler and retailers are bound to use the old system that is use of Registers, Excel sheets and contacting through phone calls which means they are inherently resistant to change, and computers have been known to facilitate change. An estimate should be made of how strong a reaction the user staff is likely to have towards the development of a computerized system. Therefore it is understandable that the introduction of such system will require special efforts to educate and train the staff. The software that is being developed is user-friendly and easy to learn. In this way, the developed software is truly efficient and can work on any circumstances, tradition, locales.

Operational Feasibility:

The user of the system i.e., wholesalers and retailers are the operators to use the system and use the results for themselves. We are able to create the operations that the client expects. The client i.e., wholesaler needs the software which can store the data for the longer term and make sure that the data is secure, data should be easy to access, detailed and error free invoice should be produced. Whereas, retailers require to easily locate different wholesalers and contact them in a more efficient way. Our system is capable of doing all these things mentioned, and for this reason our system is operational and address the client's needs through fixing their problems.

Legal Feasibility:

Legally this project meets the requirments of cyber laws and other regulatory compliances. The planned project will not violet any rules or regulations and it is lawful to proceed.

1.3 Objectives:

- The main objective of the project is to simplify the task of the Wholesaler and Retailer.
- This system must store all the records of retailer and wholesaler digitally, so it saves the time and this system is storing the records in digital manner(i.e. in data base) so paper and cost required for these papers should be save.
- The system must provide security that means only authenticate person can only have the access to the system.
- The system must create invoice for payment method which decreases the chances of error caused by wholesaler if the invoice made manually.
- System must keep the records of wholesaler, retailer and keep track of the stocks. It should be able for saving, deleting and updating the records.

1.4 Purpose:

The proposed system is fulfilling the requirements of client so that the data and the information of the wholesaler and retailer can be store for longer period of time with easy access to data and manipulation of the data. This system lead to error free and fast management of the stocks and all the records.

The project aims to computerized B2B management system for wholesaler and retailer that would be more effective and efficient than existing manual system. This proposed system will eliminate all the manual interventions and increase the speed of whole process. The invoice generated by the wholesaler will be more efficient than old manual system receipt which can contain

manual errors. All the records of stock ,retailer, payments, status of order will be secured with client and will easy to access.

1.5 Scope:

The scope of the project is to simplify the work of client, reduce the overall cost, access to customer information easily, reduce time consumption of the client. It will increase the business as the client will get more time. The aim of the project is to automate the current manual process. The proposed system could be used for maintaining the record of stocks, record of retailer details, retailer can easily place orders, smooth communication between wholesaler and retailer, and etc. The data is directly stored in the database in the hard disk of the pc.

1.6 Advantages:

Some advantages are-

- Not much manual work involved.
- The proposed system would easily overcome most of difficulties coming from the current system.
- The system will provide the user friendly interface for the operations.
- The system will save time and money of the client.
- The wholesale can keep track of required and available stocks, retailer can add items to cart and further proceed for buying.
- Manual errors created while billing by wholesaler can be discarded by digital system.

1.7 Applicability:

The trading between wholesaler and retailer can be efficient. Digitalizing existing manual paper form which increases the communication efficiency between wholesaler and retailer. Keeping track of stocks digitally, placing orders by retailer to wholesaler online, generating payment details cards digitally, order status, these all functionalities of the proposing system increases the efficiency of work management between wholesaler and retailer.

SURVEY OF TECHNOLOGIES

TOOLS :

The tools that we are using for this software are as follows:

- Microsoft Visual Studio Code
- PostgreSQL

➢ MICROSOFT VISUAL STUDIO CODE:

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

➢ PostgreSQL:

PostgreSQL is an open-source relational database management system. It organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like PostgreSQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

TECHNOLOGY :

Technologies that we have used in our project are as follows:

- Python
- Django
- SQL

➢ Python:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

➢ Django:

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. It takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.

Django was designed to help developers take applications from concept to completion as quickly as possible.

➢ SQL:

SQL stands for Structured Query Language, is a standard language for accessing and manipulating databases. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, PostgreSql and SQL Server use SQL as their standard database language.

REQUIREMENT AND ANALYSIS

3.1 Problem Definition :

The current system used by client is simple paper based register book. Wholesaler is using the register book to store the payment details of retailer, record the stocks availability and requirement. If there will be same retailer then user have to search in multiple register book which is time consuming and it becomes messy. Retailer have to visit the wholesaler shop ask for the products, enquiry regarding products, payment in offline mode getting payment receipt in paper form which can contain some manual errors. Old manual system creates wastage of papers, consuming time of both wholesaler and retailer. Each time wholesaler have to make an entry in register book after accepting payment from receipt and update records in register. Retailer need to travel to wholesaler shop for purchasing of goods which consumes time of retailers also. Due to all these situation the management between wholesaler and retailer is not properly managed.

3.2 Requirement Specification :

For making this system we are using following software and tools:-

- Microsoft Visual Studio Code
- PostgreSQL

3.3 Planning and Scheduling :

3.3.1 Life Cycle Of Project:

For developing any system there are various phases that must be carried out to finally get an efficient output. Choosing an appropriate approach to software development is mostly dependent on the nature of the project which needs to be developed. System Development Lifecycle(SDLC) is one of the traditional and most common methodology suited for this purpose. SDLC explains different steps that a developer must follow regardless of the approach he chooses. This methodology helps in improving the quality and overall productivity of the system. It also helps in planning and managing different tasks involved in the system development.

For development of the our proposed system the model we will be using is the **Iterative Waterfall Model.**

In practice, it is not possible to strictly follow the classical waterfall model for software development work. In this context, we can view the iterative waterfall model as making necessary changes to the classical waterfall model so that it becomes applicable to practical software development projects.

The various phases of Iterative Waterfall model are as follows:

1. Requirement Analysis:

For the development of the proposed system the requirements of the users are:

1. Wholesaler wants to store the details of the customers, order placed and needs a system for managing the Stocks.

2. Retailer wants to place orders in a bulk.

3. Wholesaler wants Invoices and retailer wants receipts.

2. Design:

In the design phase, we will be designing the software by the different diagrams like Class diagram, ER diagram, Use case diagram, Event table.

3. Coding and Unit testing:

Requirements are written in the coding language and transformed into computer programmes which are called Software..

6. Integration and System Testing:

After completing the coding phase, integration and system testing phase is performed to check the behaviour and validity of the developed product and if there are any error found then the process starts again from the requirement gathering.

7. Maintenance:

In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required. Maintenance involves debugging and new addition options.

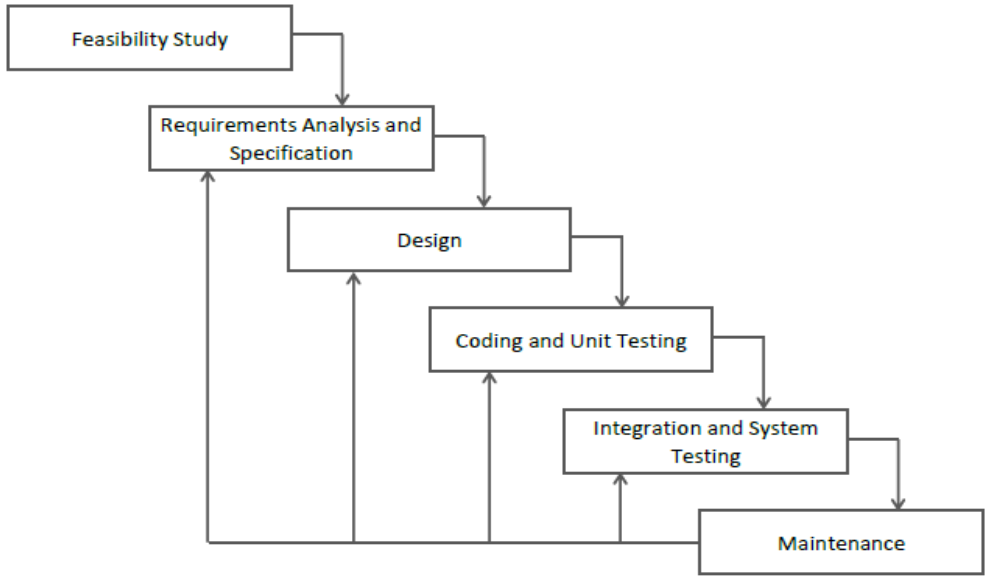


Figure 2: Iterative Waterfall Model

3.3.2 Gantt Chart :

A Gantt chart is a horizontal bar chart developed as a production control tool. Gantt charts are useful for planning and scheduling projects. They help you assess how long a project should take, determine the resources needed, and plan the order in which you'll complete tasks. They are also helpful for managing the dependencies between tasks. A Gantt chart is constructed with a horizontal axis representing the total time span of the project, broken down into increments (for example, days, weeks, or months) and a vertical axis representing the tasks that make up the project.

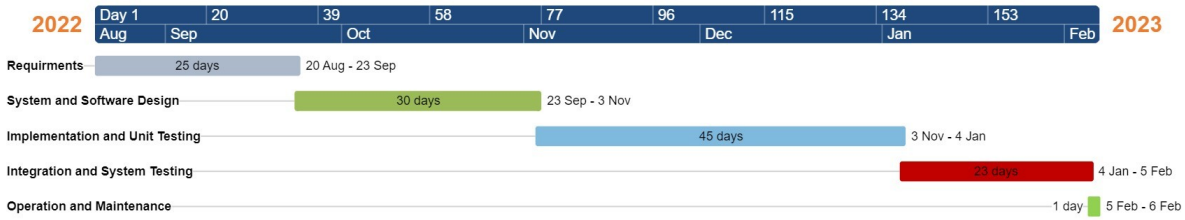


Fig 3.2 Gantt chart

3.4 Software and Hardware Requirement :

❖ **Software Requirements :-**

➢**Operating System : Microsoft Windows 8 and higher**

➢ **Front-end : Visual Studio Code**

➢ **Back-end: SQL Server**

❖ **Hardware Requirements :-**

➢ **Memory : Minimum 1 GB RAM**

➢ **Hard disk : Minimum 1 GB**

➢ **Processor : Minimum 32-bit processor**

3.5 Event Table :

Sr. No.	Event	Trigger	Source	Activity	Response	Destination
1.	Wholesaler Login	Login	Wholesaler	Check for valid username and password	Opens the account if the validation is successful	Dashboard
2.	Retailer Login	Login	Retailer	Check for valid username and password	Opens the account if the validation is successful	Dashboard
3.	Add Listings	Add	Wholesaler	Adding new listings	New listings added	Dashboard
4.	Edit Listings	Update	Wholesaler	Updating Listing details	Listing details updated	Dashboard
5.	Delete Listings	Delete	Wholesaler	Deleting Listings	Listing deleted	Dashboard
6.	Search for Listings	Search	Retailer	Search Listings	Listings Searched	Featured Listings
7.	Add to Favourites	Add	Retailer	Adding Listings to Favourites	Listings added to Favourites	Dashboard
8.	Make an Inquiry	Buy	Retailer	Placing order of goods	Order Placed	Dashboard
9.	Reply to Inquiry	Reply	Wholesaler	Reply to Retailer	Reply Sent	Dashboard
10.	Edit Retailer	Update	Retailer	Updating retailer details	Retailer details updated	Dashboard
11.	Wholesaler Logout	Logout	Wholesaler	Logging out	Navigate back to Login page	Login Page
12.	Retailer	Logout	Retailer	Logging	Navigate	Login Page

	Logout			out	back to Login page	

SYSTEM DESGIN

4.1 Basic Module:

The system has modules as follows:

1.Register Module:

Register module will save all the information necessary from wholesaler and retailers which will help in validation of the account and later on to upload listings and place orders.

2.Login Module:

Login module validate the usernames and passwords of the wholesalers and retailers. With this only authorized users can use the system, it will alert for the invalid password entered by the users. After login a wholesaler will enter the dashboard.

3.Dashboard Module:

With dashboard module wholesaler can create listings(upload a product),update the listings, delete a listing. Retailers favourites will also show up here.

4.Buying/Make an Inquiry Module:

In this module a retailer can place order by making an enquiry for the desired product and can contact the owner of that listing for further related information. An email will be sent to the wholesaler and then the whole communication can be carried forward by emails.

5.Search Module:

With this a retailer can search for products across the marketplace placed by the wholesalers throughout the nation. The retailer can search for listings with filters consisting of product category, price, city and state.

6.Logout Module:

Logged in users can safely and easily exit from the website without any worries.

4.2 Data Design :

By using the data design, the designer determines what data must be stored and how the data elements are related to each other. With this information, they can begin to fit the data to the database model. Database design involves classifying data and identifying interrelationships between the data.

Our project includes mainly three tables:

- **Users Table:-**

This table is used to store information about wholesaler and Retailer.

Fields	Data Type	Relationship
Id	Int(10)	Not Null
Username	Varchar(30)	Primary Key
First name	Char(100)	Not Null
Last name	Char(100)	Not Null
Email	Varchar(255)	Not Null
Favourites	Char(200)	Not Null

- **Listings Table:-**

This table is used to store information about Listings.

Fields	Data Type	Relationship
Id	Int(10)	Primary Key
Owner	Varchar(30)	Foreign Key
Title	Varchar(30)	Not Null
Category	Varchar(200)	Not Null
Address	Int(10)	Not Null
City	Varchar(100)	Not Null
State	Char(50)	Not Null
Pincode	Int(6)	Not Null
Price	Int(10)	Not Null
Description	Char(300)	Not Null
Photo_main	Image field	Not Null
Photo_1	Image field	Not Null
Photo_2	Image field	Not Null
Photo_3	Image field	Not Null
Photo_4	Image field	Not Null
Photo_5	Image field	Not Null
Photo_6	Image field	Not Null
List_date	Date field	Not Null

- **Inquiry Table:-**

This table is used to store information about inquiries being made to listings.

Fields	Data Type	Relationship
Listing	Char(100)	Not Null
Listing_id	Int(500)	Not Null
Name	Int(20)	Not Null
Email	Int(10)	Primary Key

Phone	Varchar (255)	Not Null
Message	Text(500)	Not Null
Contact_date	Date field	Not Null
User_id	Int(10)	Foreign Key
Owner_id	Int(10)	Foreign Key

4.3 Data Integrity and Constraint :

As a management system is being made and put into place it is very important to handle the data securily. There is a genuine need for validation. In addition to that, all phone numbers provided must be numeric making it sure that number should not exceed more that ten numbers and specifying that the current details is mandatory with an asterisk mark .The password must always be in hidden format along with the characters being in capital, this will ensure that the data entered is sensible and reasonable.

4.4 Logical Diagrams:

A logical diagram provides a graphical view of the structure of an information system, and helps you analyse the structure of your data system through entities and relationships, in which primary identifiers migrate along one-to-many relationships to become foreign identifiers, and many-to-many relationships can be replaced by intermediate entities. Logical diagram is used to ensure the client understands our proposed system.

4.4.1 Class diagram :

A Class diagram is the Unified Modelling Language (UML) is type of static structure diagram that describes the attributes and the constraints impose on system. It shows the collection of classes, interface, association, collaboration and constraints. They are used to show the different objects in a system, their attributes, their operations and the relationships among them. The following diagram represent the B2B Wholesaler and Retailer Management System with different operations and attributes, including the parent class login. The login class has the two child classes which is Retailer and Wholesaler. These both class have identical attributes and operations.

The below diagram consist of classes. Login class having two child classes namely retailer and wholesaler which are both associated with Stock class.

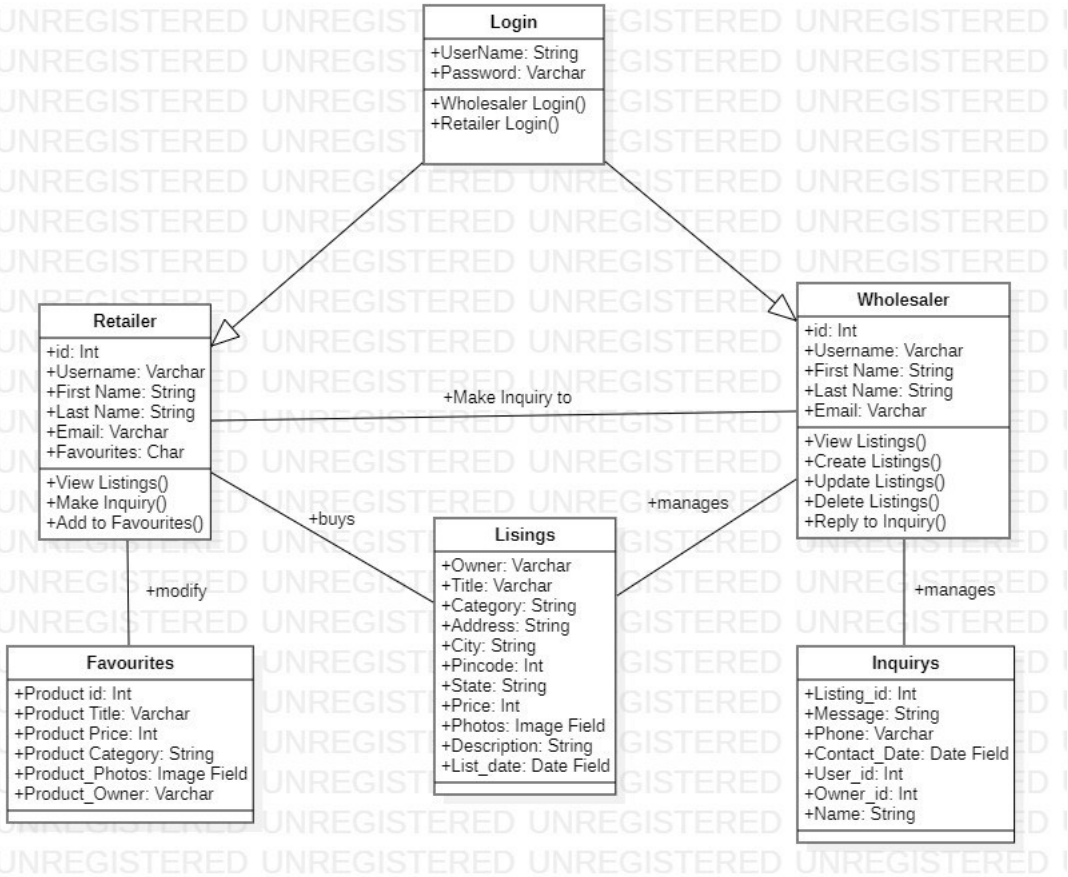


Fig 4.1 Class Diagram

4.4.2 ER diagram :

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and the relationship between them. Following diagram represent the B2B Management system for Wholesalers and Retailers , have the entities like wholesaler, retailer, stocks, orders, cart. It has the attributes. The wholesaler has the attributes like

name, address, contacts, email etc. The retailer entity has attributes like name, address, contact, etc. The order which is placed by retailer has attributes like date, total items, name of the retailer etc.

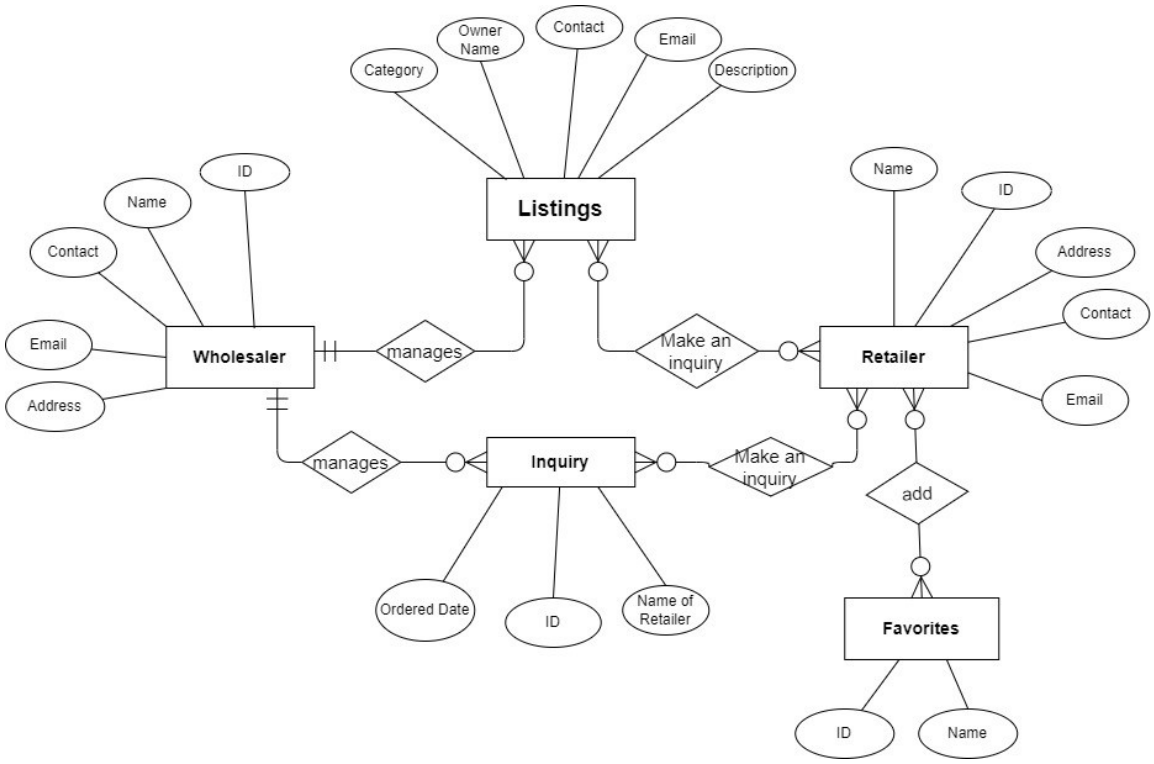


Fig 4.2 ER Diagram

4.4.3 Use case diagram:

A use case diagram at its simplest is a representation of user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by the other types of diagrams as well.The actors(end-users) involved in the use cases, a use case diagram and the detail use case description are

provided. The use cases that find representation are wholesaler and retailer. The retailer can place order for the stocks and can modify his or her cart.The wholesale gets to manage the stocks and the order which was placed my the retailer.

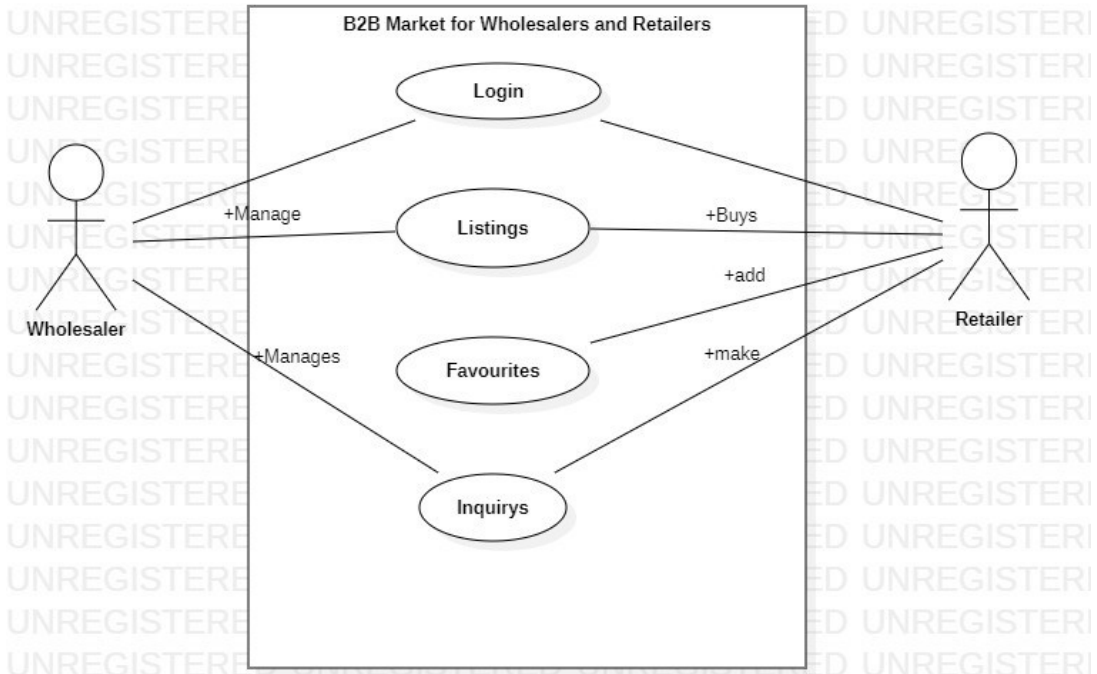


Fig 4.3 Use Case Diagram

Implementation and Testing

5.1 Testing Approach:

Software Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. It can also provide an objective, independent view of the software to allow the business to appreciate and understand the risk of software implementation. The basic purpose of testing is to detect the errors that may be present in the program. Testing as the process of executing a program with the intent of finding errors.

The Box Approach :-

Software testing methods are traditionally divided into white-box and black-box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

1) White-box Testing :-

In White box testing which is also known as clear testing, glass testing, we tested the internal perspective of the system such as code, logical conditions, loops, etc. By using different types of inputs the appropriate outputs are determined through which we also tested the working of external functionality of the software. In this testing we tested the if-else, else-if conditions, for and end for loops, internal programming logics of the system, internal settings of the software, connection of the different types of files within the system and also the performance of external functionalities of the system are also tested.

2) Black-box Testing :-

In Black box testing we treated the software as a “black box”, by only examining the functionally of the software without any knowledge of internal implementation, without seeing the source code. In this testing we are only aware of what the software is supposed to do, not how it does it. In this testing we tested the external functionality of the software like proper registration, login, logout of the users, ability of users to place inquiry for any product of there need and wish, view of dashboard for the user logged in, view of listings for the users, create, update, delete functionality of the software and many more without testing the source code and internal implementations.

Levels of Testing :-

The levels of testing are as follows :

1) Unit Testing

2) Integration Testing

3) System Testing

4) Acceptance Testing

1. Unit Testing:

This testing carried out during programming stage itself. As the programming of the software progresses the testing of that particular part is being carried out simultaneously. In unit testing we tested the each module individually instead of the whole system at one go. Each module is found to be working satisfactorily as regard to the expected output from the module. All textboxes are having validation by which they will not remain empty and all work properly

as expected. Due to the unit testing the efficiency of the working of system increases.

2. Integration Testing:

After unit testing we perform Integration testing in which all the individual modules are united together and then tested accordingly to check the performance of all the individual modules in unity. For example the make an inquiry page, mailbox and inquires page. In this example all the three individual modules are united together to work as a whole integrated system in which a user uses inquiry page o send mails and mailbox is used to receive and show the mails of inquiries for a particular product and inquiries page show the inquiry made for any product by a particular user. In the manner all the modules are combined and tested as a whole.

3. System Testing:

In this all the unit modules are integrated together and a system is build. Now, the testing of this entire system takes place by giving required inputs and determining various outputs on that. The system built by integrating all the modules is the final product which is to be delivered to the customer. So in this, the system is tested by registering, logging in, logging out, placing an inquiry for a specific product, sending mails for inquiry, receiving mails for inquiries, etc.

4. Acceptance Testing:

The acceptance testing is done by the end-user who used the software so that any error occurred will be eliminated there itself.. Before delivering the final product to the customer this acceptance testing was performed in which the software developed was given to an end user so that if any minor error occurs while using the

software by an end user in actual scenario will be eliminated there itself.

5.2 Coding Details and Code Efficiency:

In the software, to validate the form there are many if else and validation statements used. The main or core part of software code is to perform database operation, sending mail, form validation. we are using SQL technology for database connection with Python

- All the fields such as Phone no, Email ID etc. are validated and does not take invalid values.
- Each form for Wholesaler registration, Retailer registration, Lisitngs form will not accept blank value fields.
- Avoiding errors in data.
- Integration of all the modules/forms in the system.
- Preparation of the test cases.
- Preparation of the possible test data with all the validation checks.
- Actual testing done manually.
- Functionality of the entire module/forms.
- Validations for user input.
- Checking of the Coding standards to be maintained during coding.
- Testing the module with all the possible test data.

Login.html:

```
{% extends 'base.html'%}
{% block title %} Login {% endblock%}
{% block content %}
<section id="login" class="bg-light py-5">
    <div class="container">
      <div class="row">
        <div class="col-md-6 mx-auto">
          <div class="card">
            <div class="card-header bg-dark text-white">
              <h4>
                <i class="fas fa-sign-in-alt"></i> Login</h4>
                {% include 'partials/_alerts.html'%}
            </div>
            <div class="card-body">
              <form action="{% url 'login' %}" method="POST">
                {% csrf_token %}
                <div class="form-group">
                  <label for="username">Username</label>
                  <input type="text" name="username" class="form-control" required placeholder="Enter your Username">
                </div>

                <div class="form-group">
                  <label for="password">Password</label>
                  <input type="password" name="password" class="form-control" required placeholder="Enter your Password">
                </div>

                <input type="submit" value="Login" class="btn btn-secondary btn-block">
              </form>
              <br>
              <a href="{%url 'password_reset' %}" class="btn btn-primary">Forgot password</a>
            </div>
          </div>
```

Login Views.py:

```
def userlogin(request):
    if request.method == 'POST':
        username = request.POST['username']
        password = request.POST['password']
        user = authenticate(username=username, password=password)
        if user is not None:
            login(request, user)
            messages.success(request, 'You are now logged in!')
            return redirect('home')
        else:
            messages.error(request, 'Invalid Credentials!')
            return redirect('login')
    else:
        return render(request, 'acounts/login.html')
```

Register.html:

templates > acounts > register.html

```
{% extends 'base.html' %}
{% block title %}register{% endblock  %}
{% block content %}
<section id="register" class="bg-light py-5">
    <div class="container">
      <div class="row">
        <div class="col-md-6 mx-auto">
          <div class="card">
            <div class="card-header bg-dark text-white">
              <h4>
                <i class="fas fa-user-plus"></i> Register</h4>
            </div>
            <div class="card-body">
              {% include 'partials/_alerts.html' %}
                <form action="{% url 'register' %}" method="POST">
                  {% csrf_token %}
                  <div class="form-group">
                  <label for="first_name">First Name</label>
                  <input type="text" name="first_name" class="form-control" required placeholder="Enter your First Name">
                </div>
                <div class="form-group">
                  <label for="last_name">Last Name</label>
                  <input type="text" name="last_name" class="form-control" required placeholder="Enter your Last Name">
                </div>
                <div class="form-group">
                  <label for="username">Username</label>
                  <input type="text" name="username" class="form-control" required placeholder="Enter a Username">
                </div>
                <div class="form-group">
                  <label for="phone">Contact Number</label>
                  <input type="text" name="phone" class="form-control" required placeholder="Enter your Contact number">
                </div>
```

```
                  <label for="phone">Contact Number</label>
                  <input type="text" name="phone" class="form-control" required placeholder="Enter your Contact number">
                </div>
                <div class="form-group">
                    <label for="email">Email</label>
                    <input type="text" name="email" class="form-control" required placeholder="Enter an Email">
                </div>
                <div class="form-group">
                  <label for="password2">Password</label>
                  <input type="password" name="password" class="form-control" required placeholder="Enter a Password">
                </div>
                <div class="form-group">
                  <label for="password">Confirm Password</label>
                  <input type="password" name="password2" class="form-control" required placeholder="Enter the Password again">
                </div>
                <input type="submit" value="Register" class="btn btn-secondary btn-block">
              </form>
            </div>
          </div>
        </div>
      </div>
    </div>
  </section>
```

Register View.py:

acounts > views.py > register

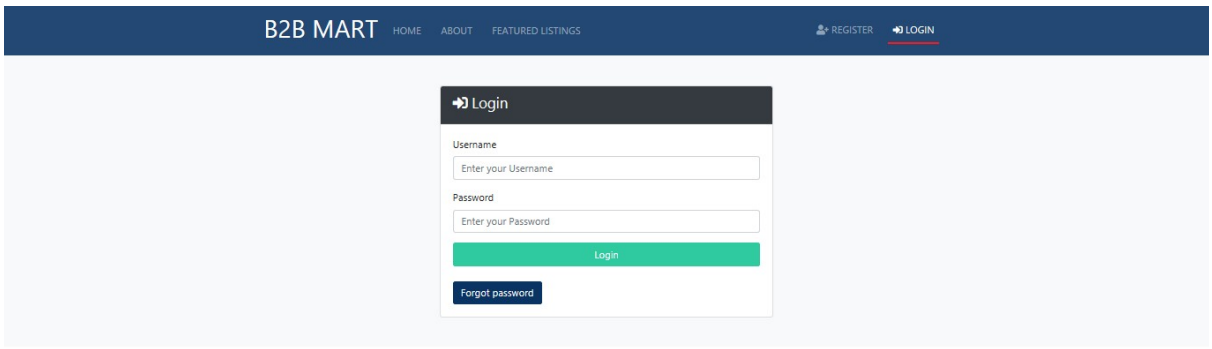
```python
from django.shortcuts import render, redirect, get_object_or_404
from core.models import User
from listings.models import Listing
from django.contrib.auth.decorators import login_required
from django.contrib.auth import login, authenticate, logout
from listings.models import Listing
from inquiry.models import inquiry
from django.contrib import messages
from django.core.mail import send_mail
from django.contrib.auth.hashers import check_password
from listings.forms import UpdateForm

def register(request):
    if request.method == 'POST':
        first_name = request.POST['first_name']
        last_name = request.POST['last_name']
        username = request.POST['username']
        email = request.POST['email']
        phone = request.POST['phone']
        password = request.POST['password']
        password2 = request.POST['password2']
        user = User

        context ={
            'first_name': first_name,
            'last_name': last_name,
            'email': email,
            'username': username,
            'phone': phone,
            'password': password,
        }

```

```python
    if password == password2:
        if user.objects.filter(username=username).exists():
            messages.error(request, 'Username is already taken')
            return redirect('register')
        else:
            if user.objects.filter(email=email).exists():
                messages.error(request, 'Email is already in use')
                return redirect('register')
            else:
                if user.objects.filter(phone=phone).exists():
                    messages.error(request, 'Contact no. is already in use')
                elif len(phone)!=10:
                    messages.error(request, 'Contact no. is not valid')

                    return redirect('register')
                else:
                    user = user.objects.create_user(username=username, email=email, password=password,  phone=phone, first_name=first_name, last_name=last_name
                    user.save()
                    login(request, user)
                    messages.success(request, 'You  are now logged in!')
                    return redirect('home')


    else:
        messages.error(request,'passwords do not match')
        return redirect('register')

else:
    return render(request, 'acounts/register.html')
```

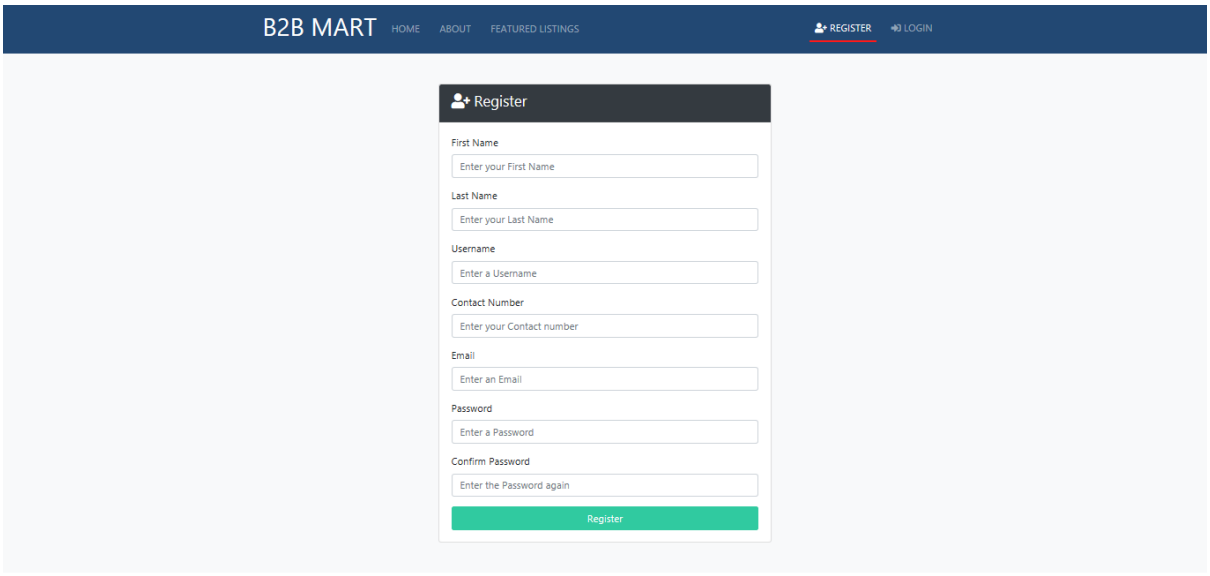
Results

Forms:

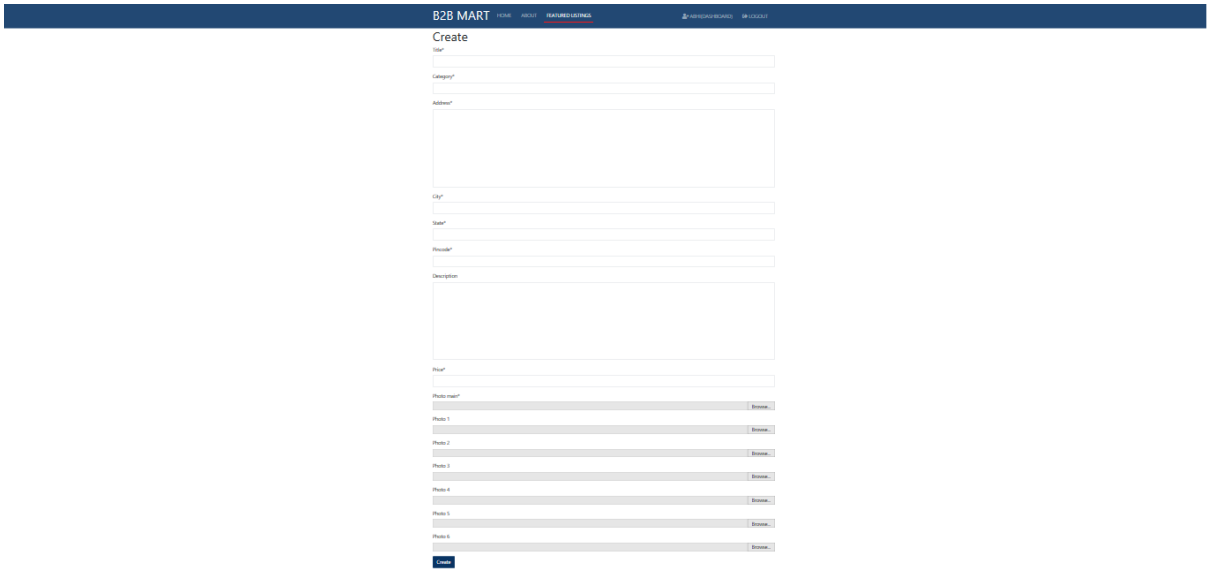
Login Form:



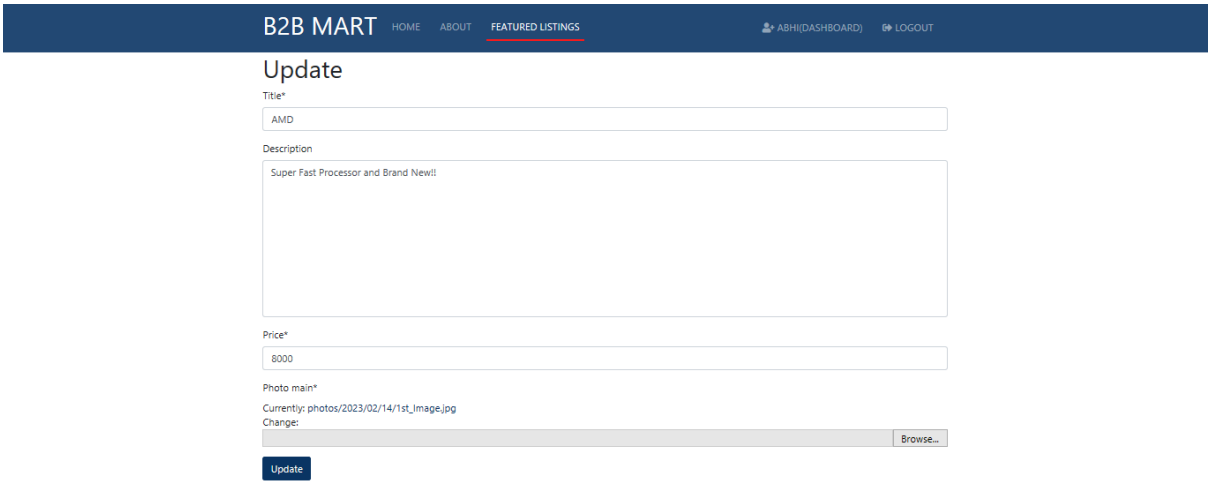
Register Form:



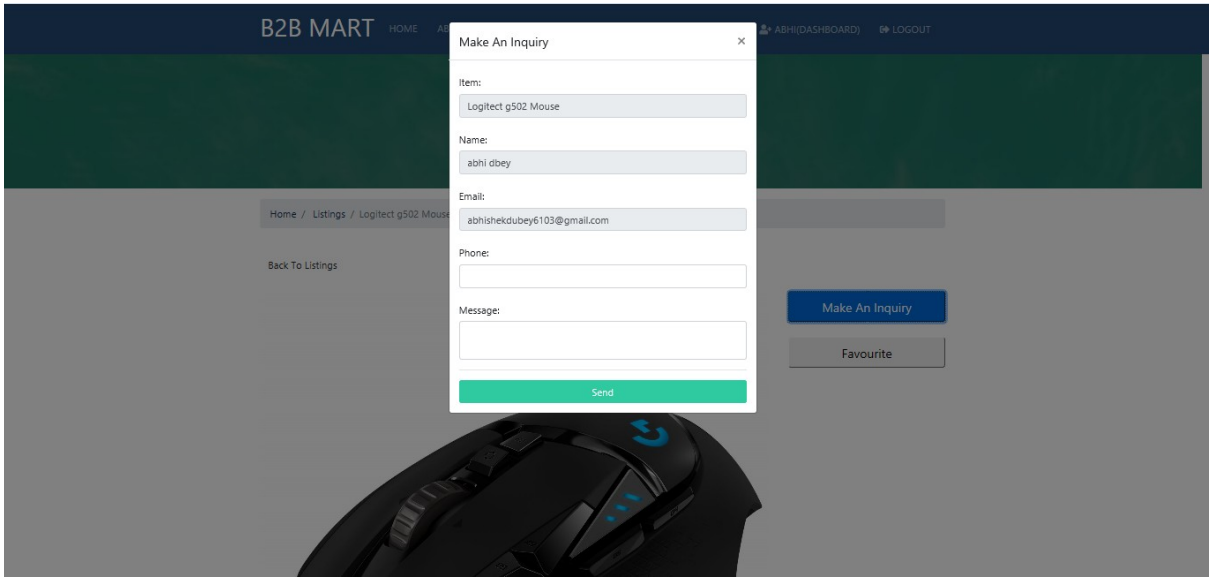
Create a Listing:



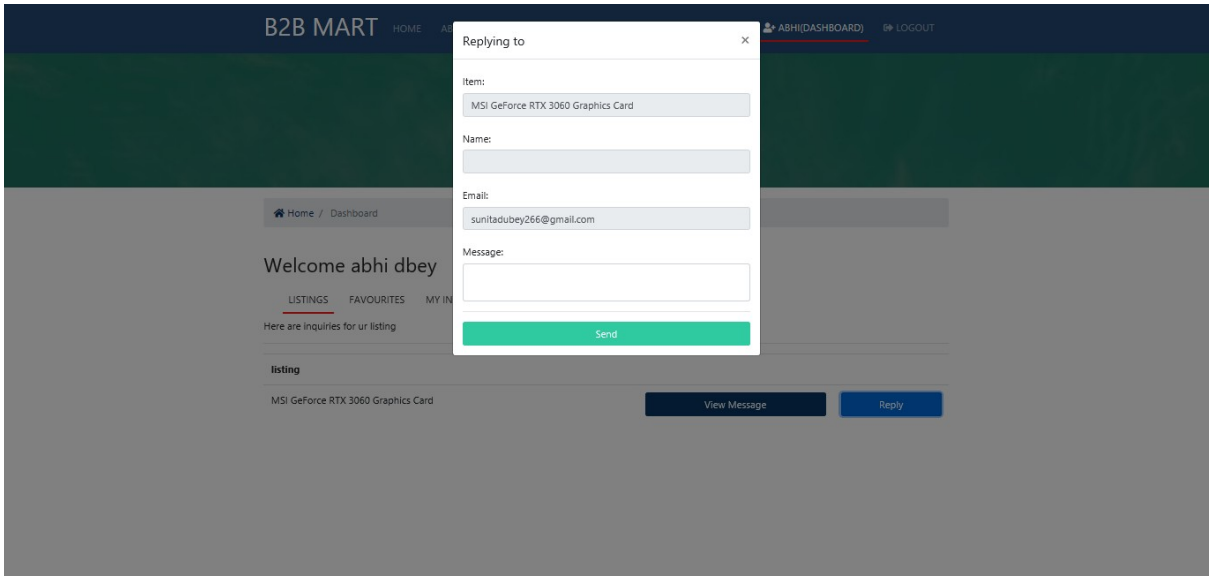
Update a Listing:



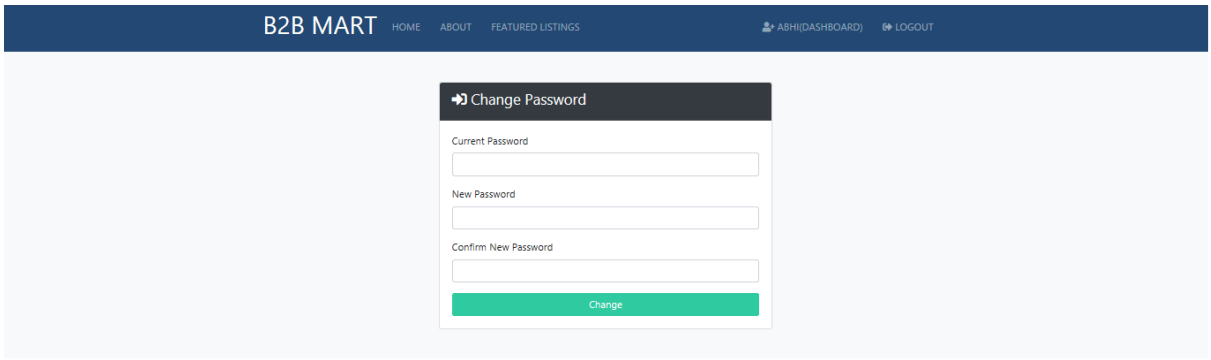
Make an Inquiry:



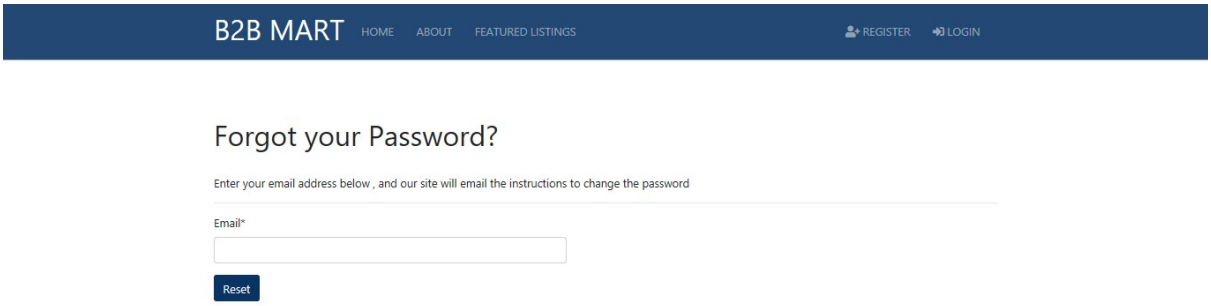
Reply to an Inquiry:

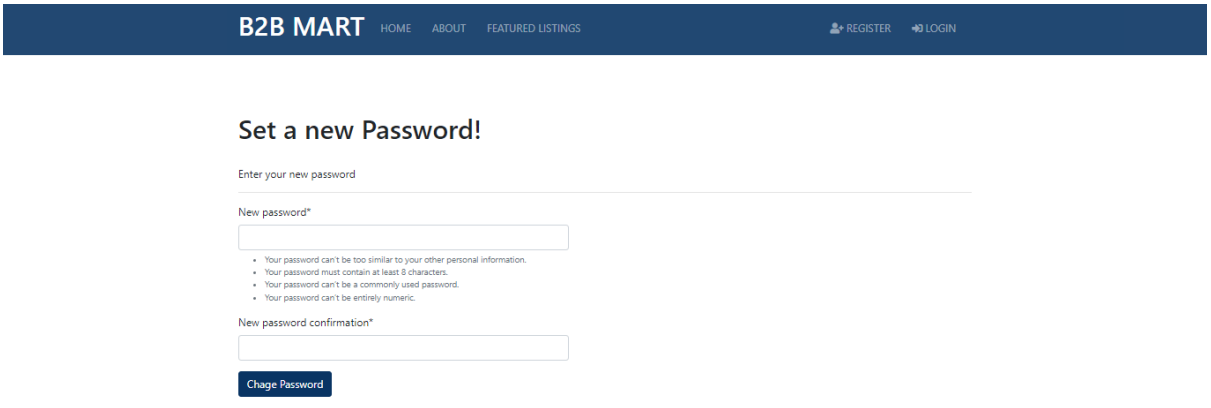


Change Password:



Forgot Password:

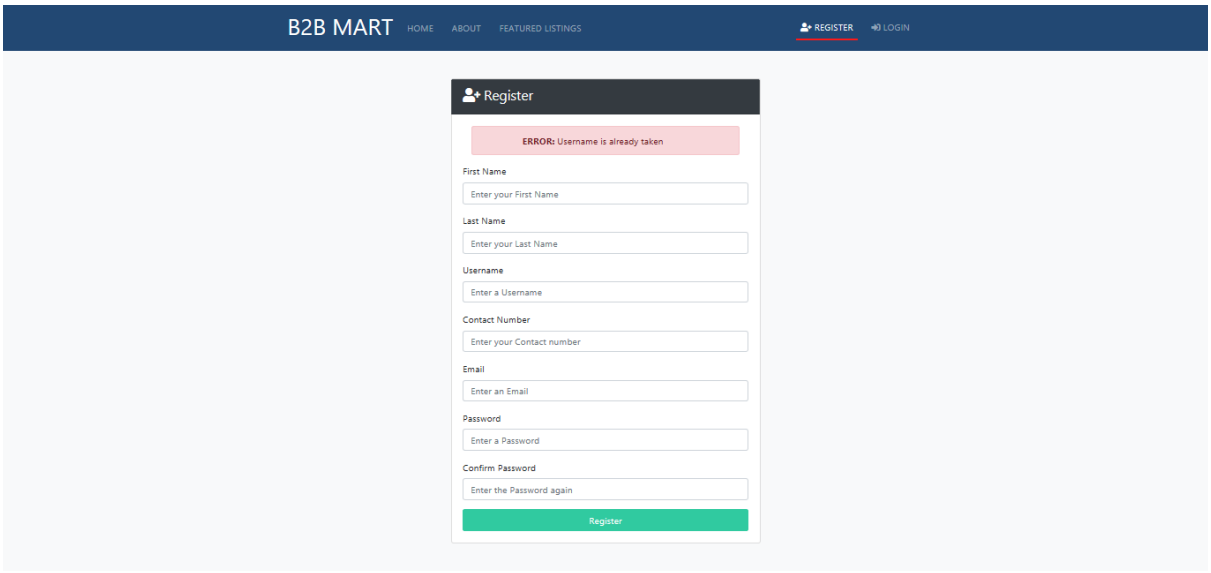




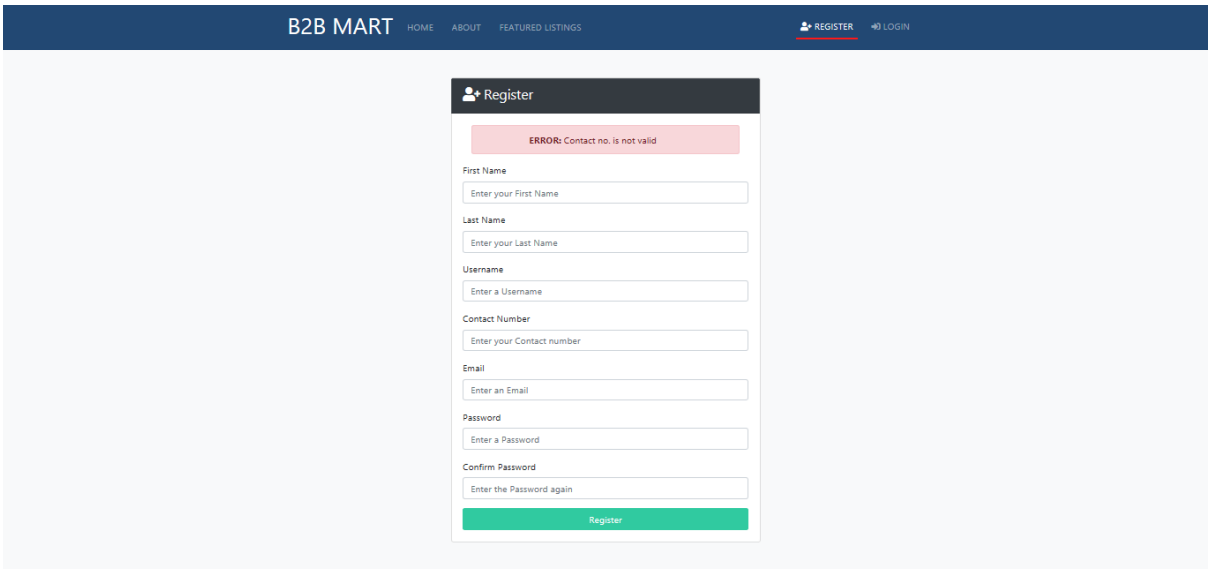
Validations:

These error messages will show up if:

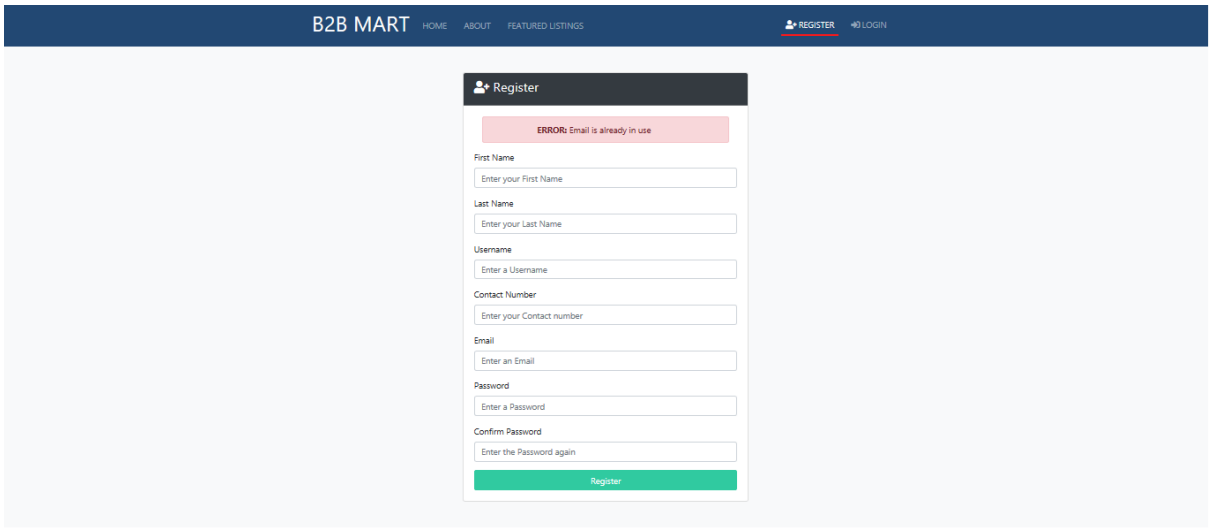
In Registration the Username is already taken:



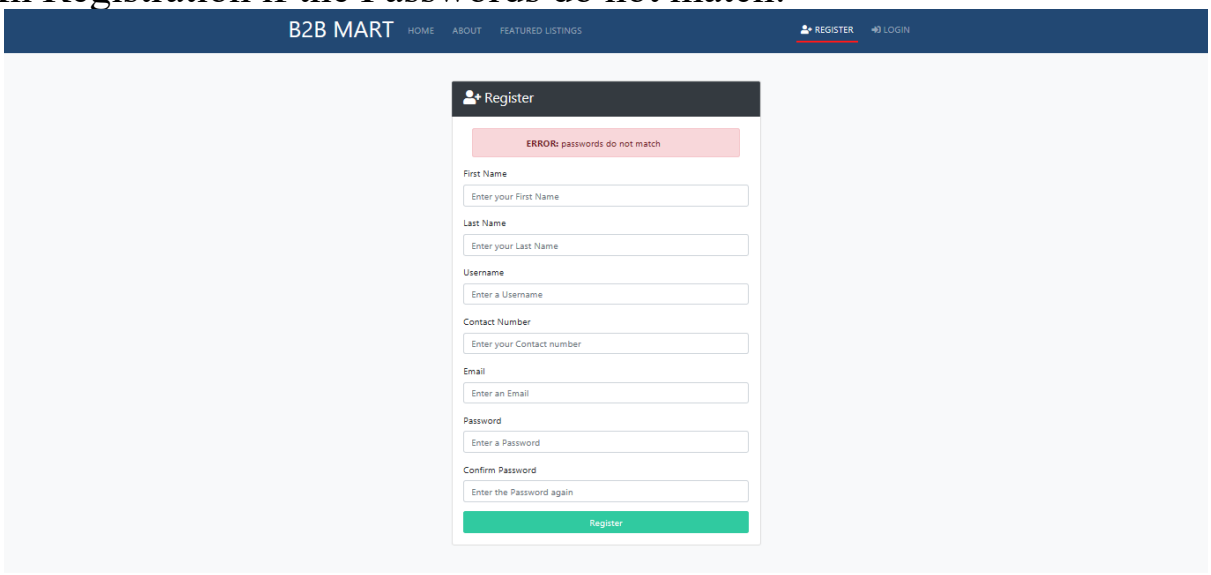
In Registration the Contact number is not valid:



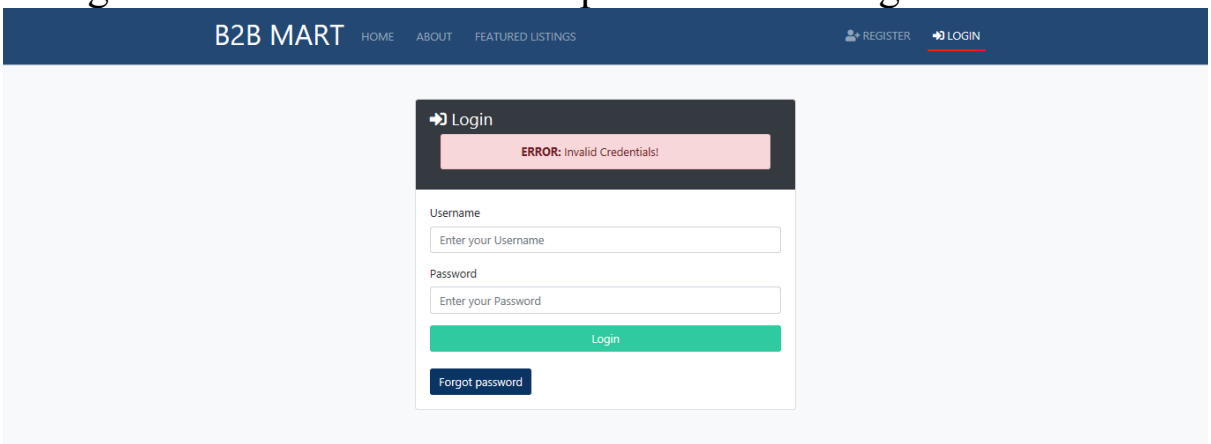
In Registration if the Email is already in use:



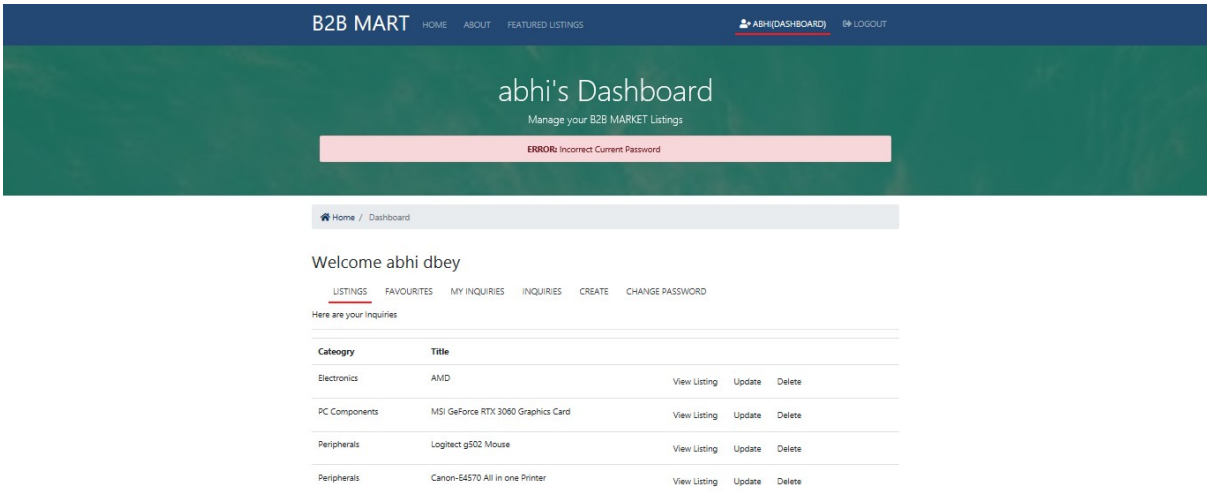
In Registration if the Passwords do not match:



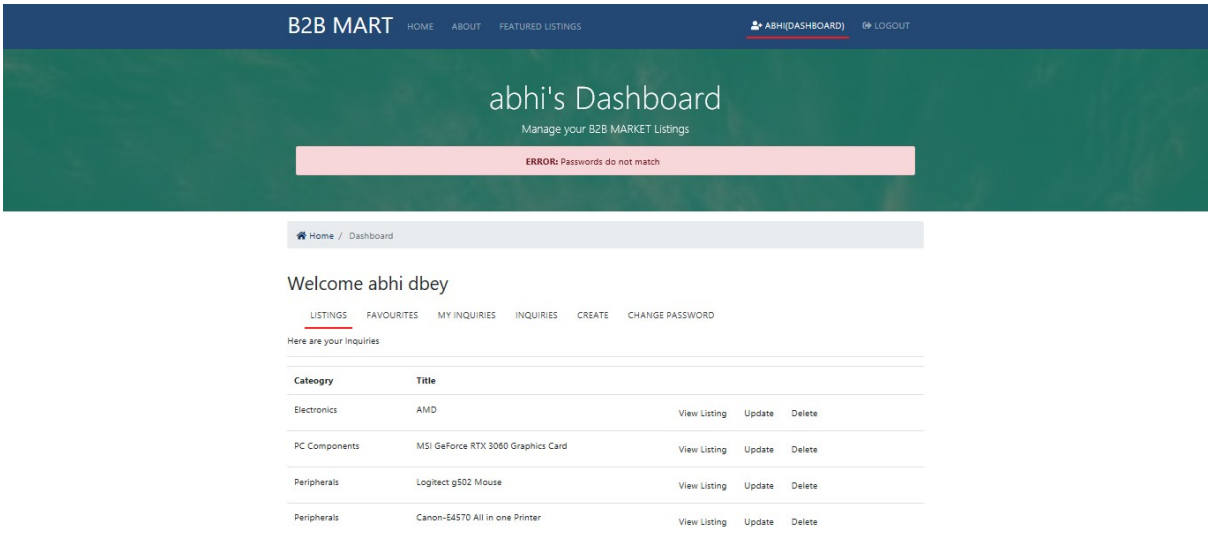
In Login if either the username or password is wrong:



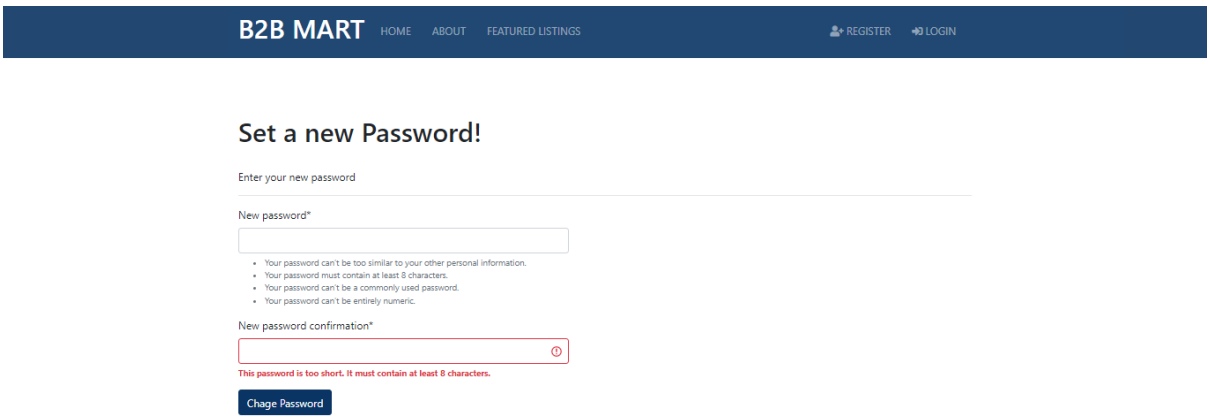
While changing password if the current password of the user doesn't match:



While changing password if the new password doesn't match:

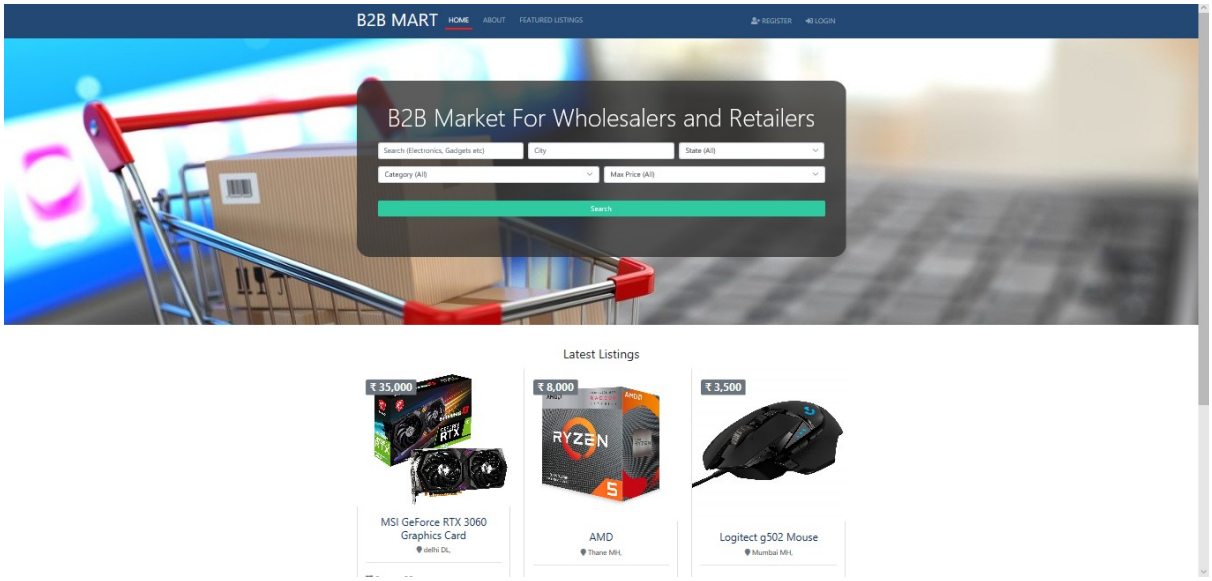


Validations while setting a new password:

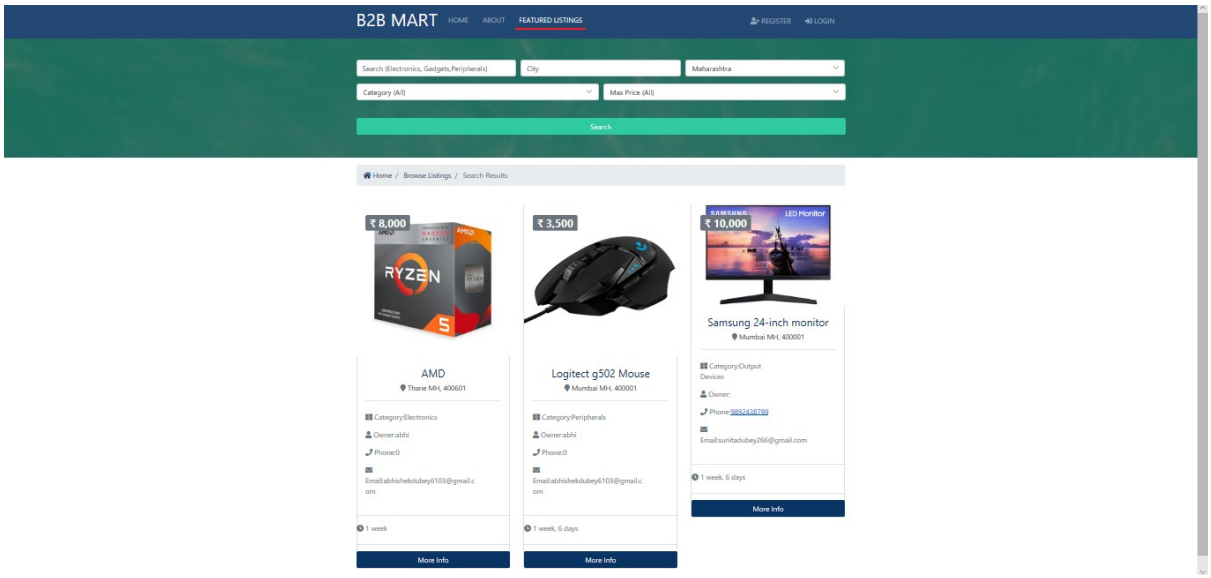


Working:

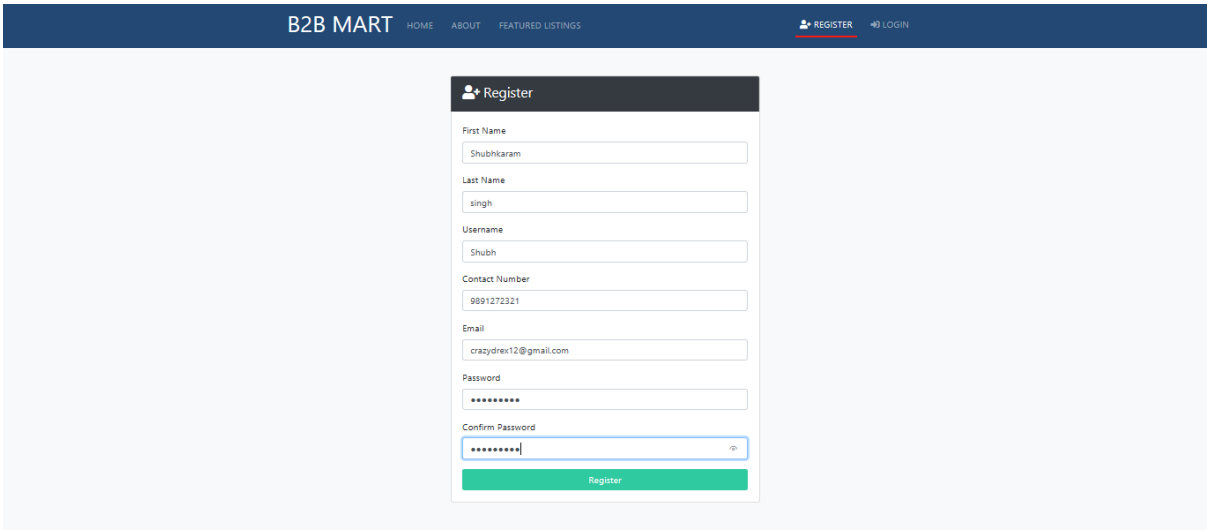
This is the Home screen from where a retailer can search for available listings.



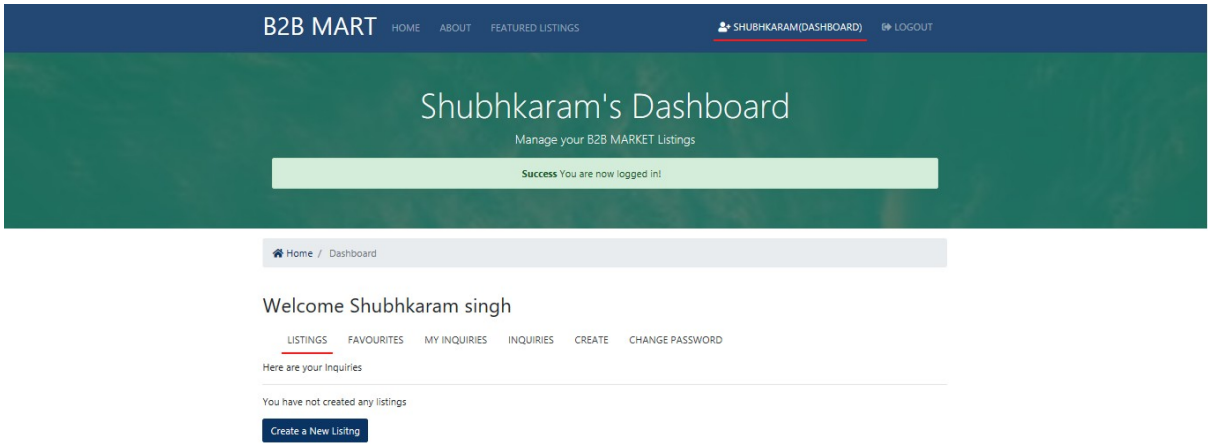
By using the search filter the listings will show up for example by choosing Maharashtra as state the following listings show up:



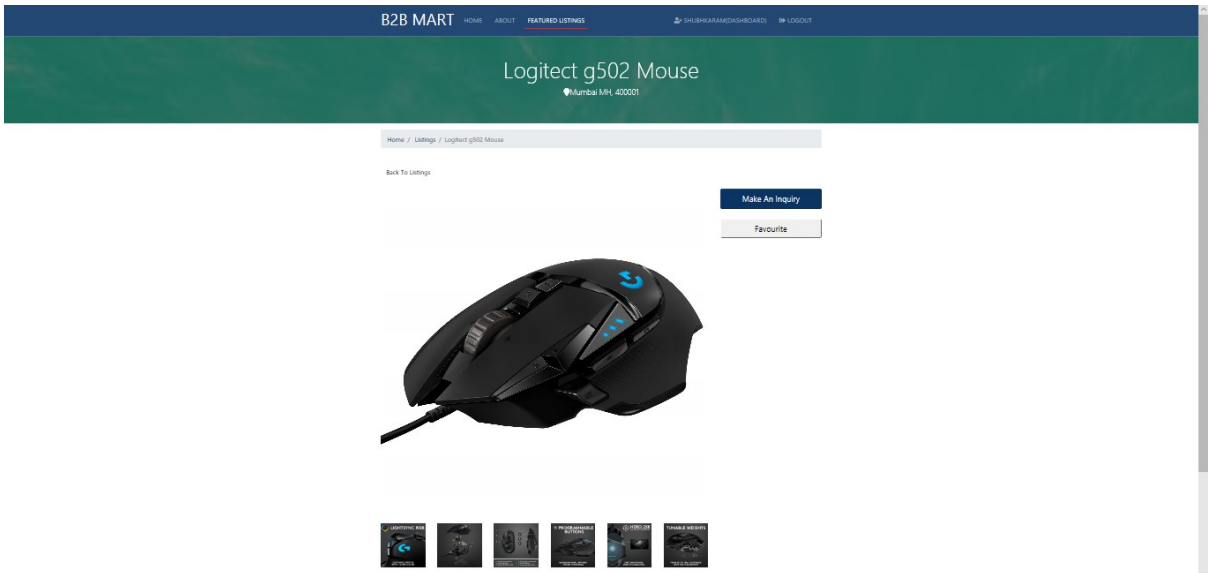
For Making an inquiry a retailer first needs to register:

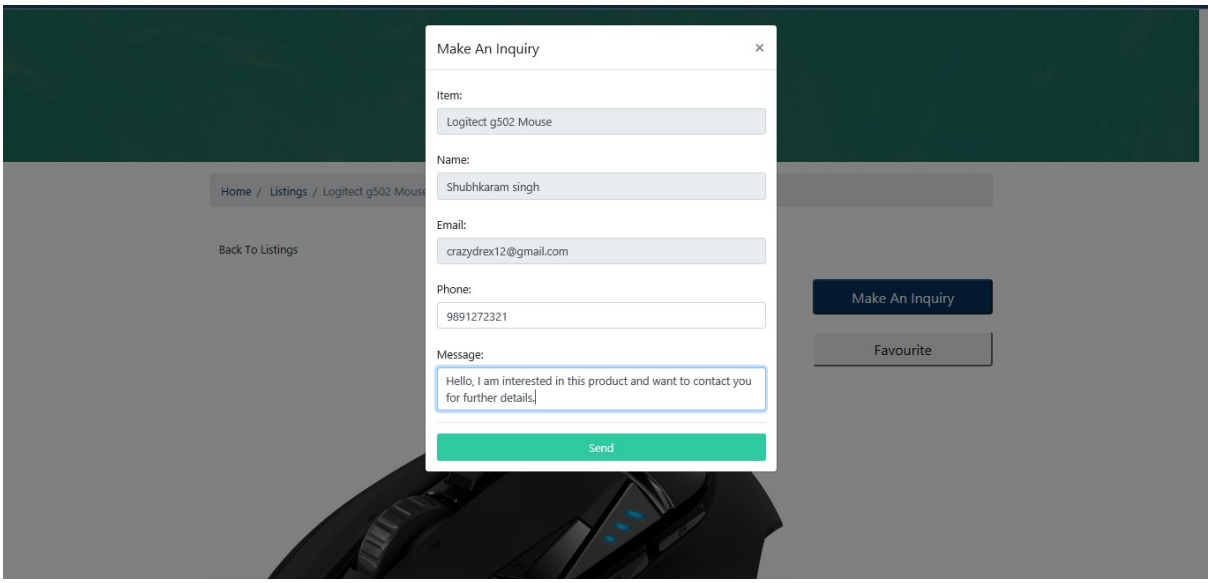


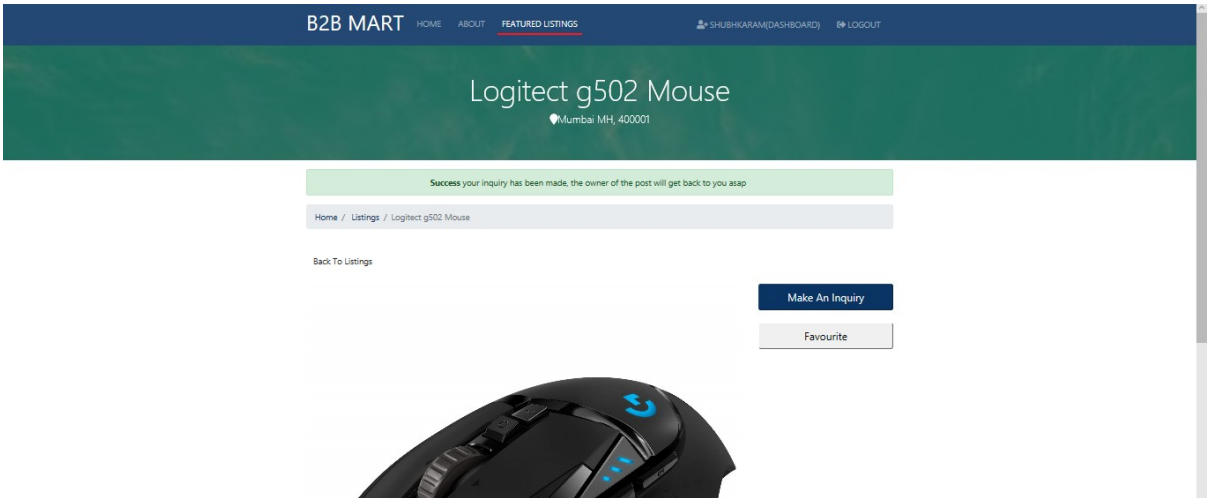
After making an account, it will lead the retailer to the dashboard directly where all the sent inquiries, added favorites will also be displayed:



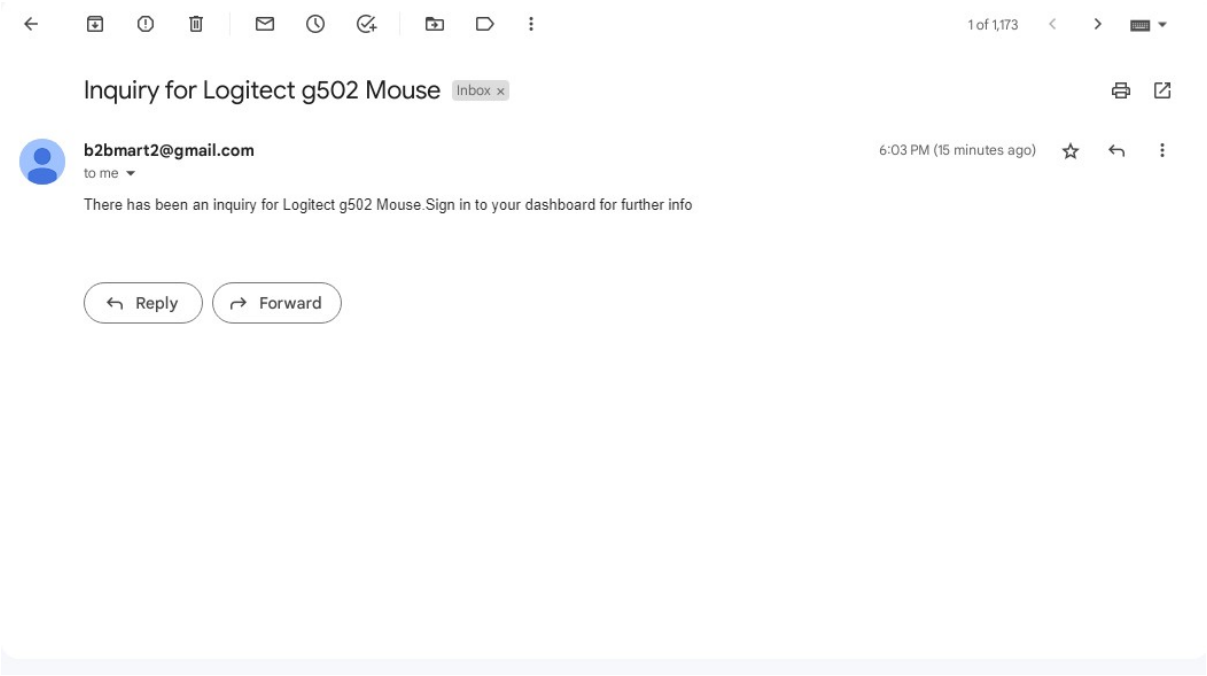
Retailer can click on make an inquiry button and send the wholesaler a message if interested:



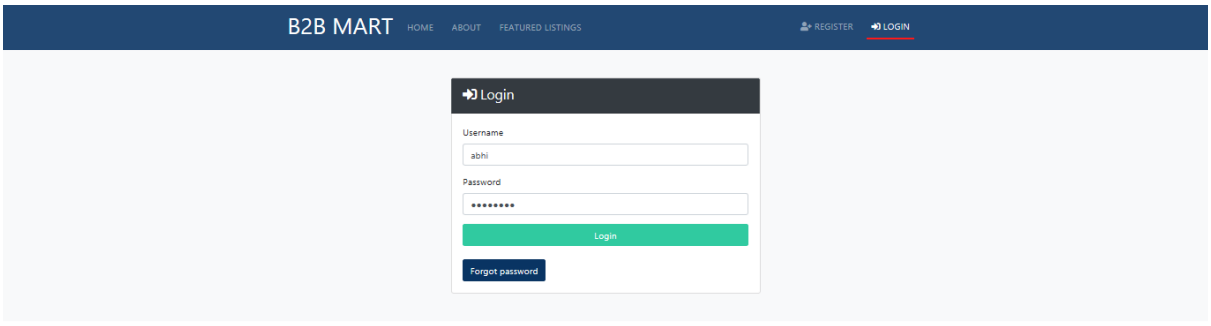




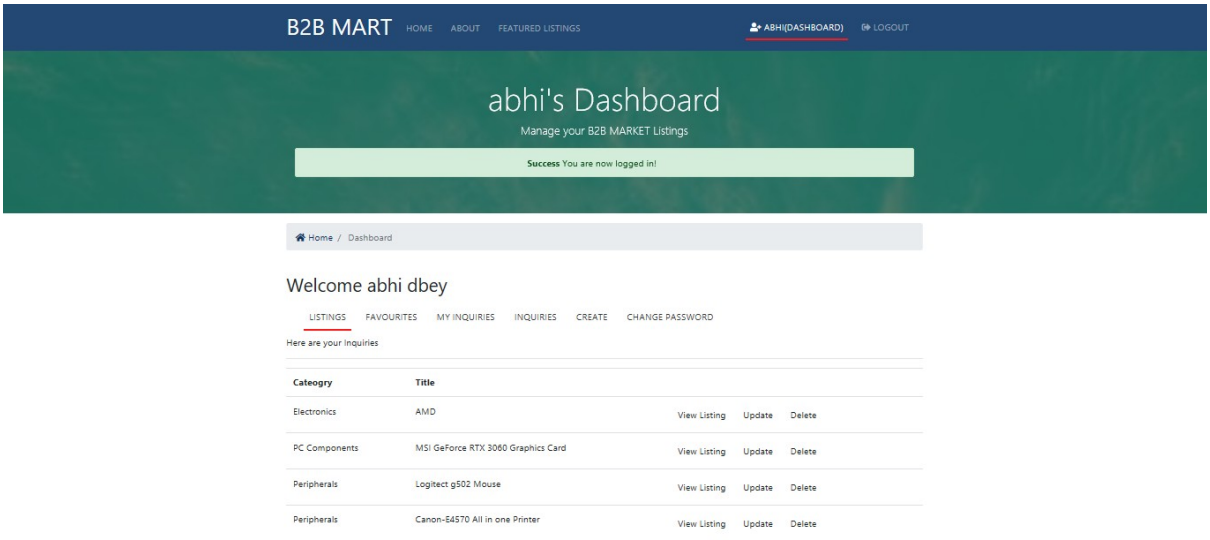
After making an Inquiry the wholesaler will get an email and let them know that an inquiry is available:



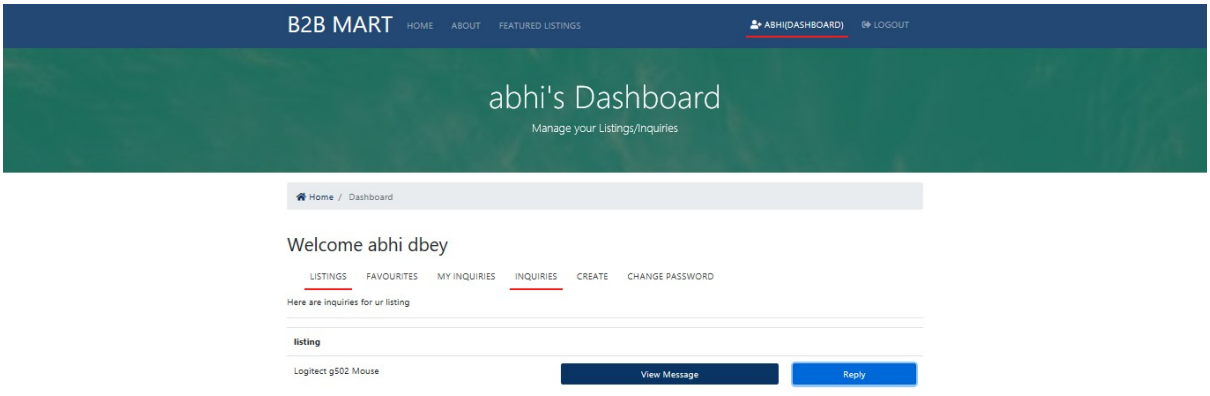
The wholesaler can login to view all the listings he/she has made, create listings, delete listings, update any current listings, check inquiries, reply to inquiries as well:



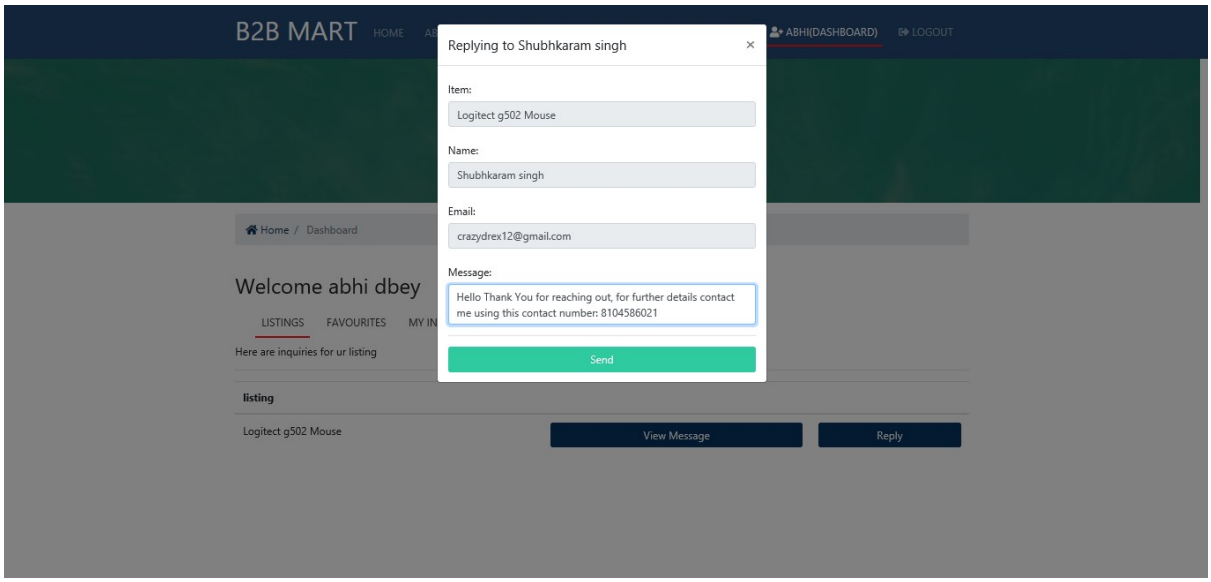
Password can also be changed from the Dashboard itself:

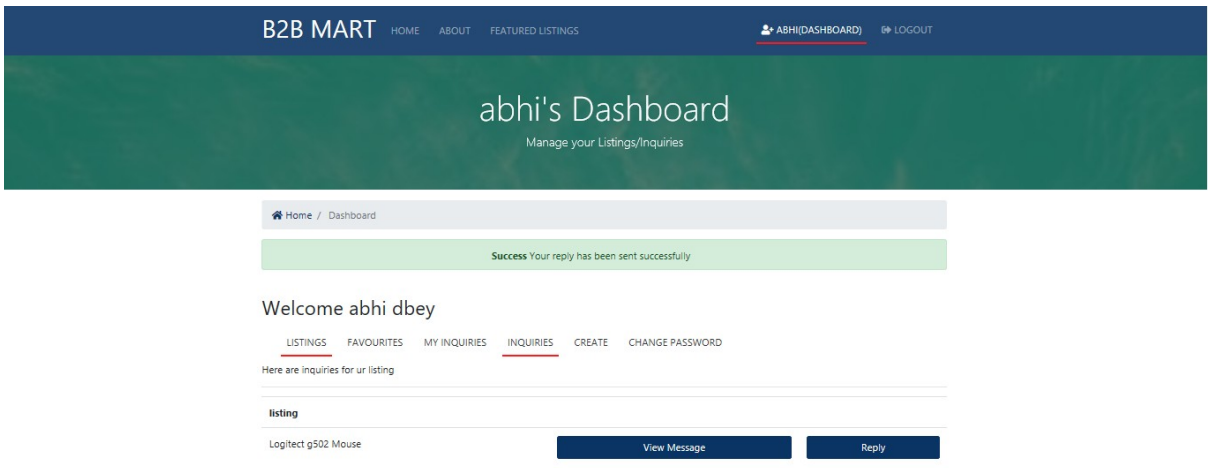


We can see the inquiry which was sent by the retailer is now being displayed on the wholesalers dashboard:

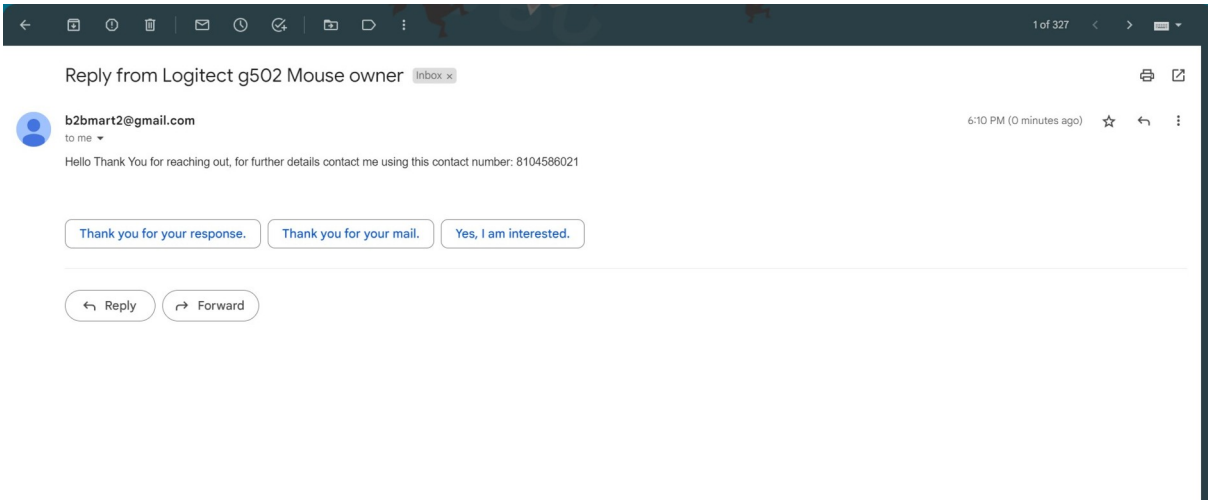


The wholesaler may now choose to reply to the retailer directly from the dashboard:

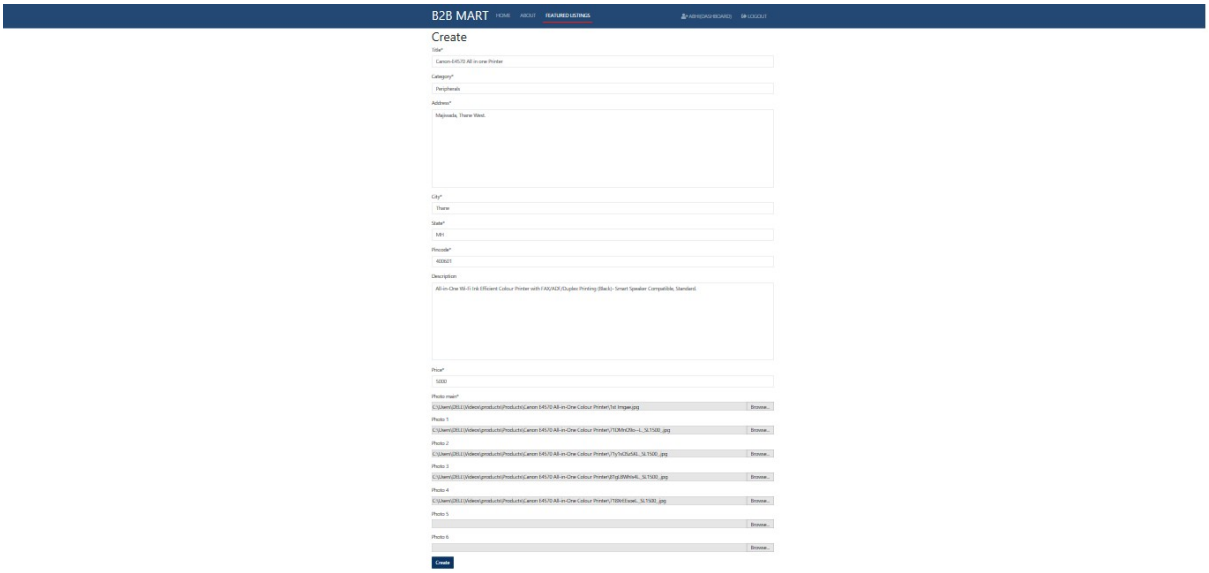




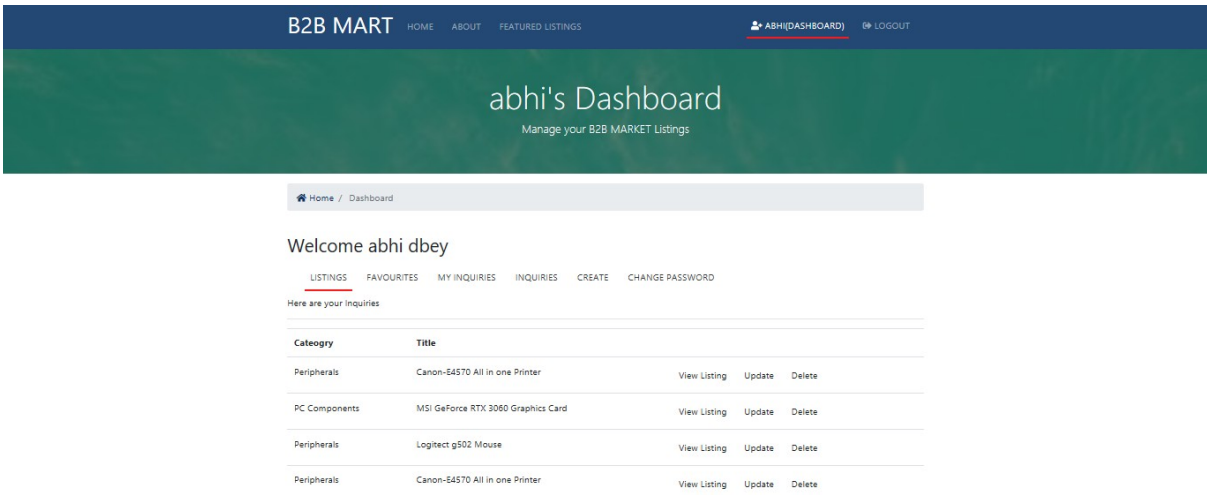
The retailer will now receive the reply via an email from the wholesaler:



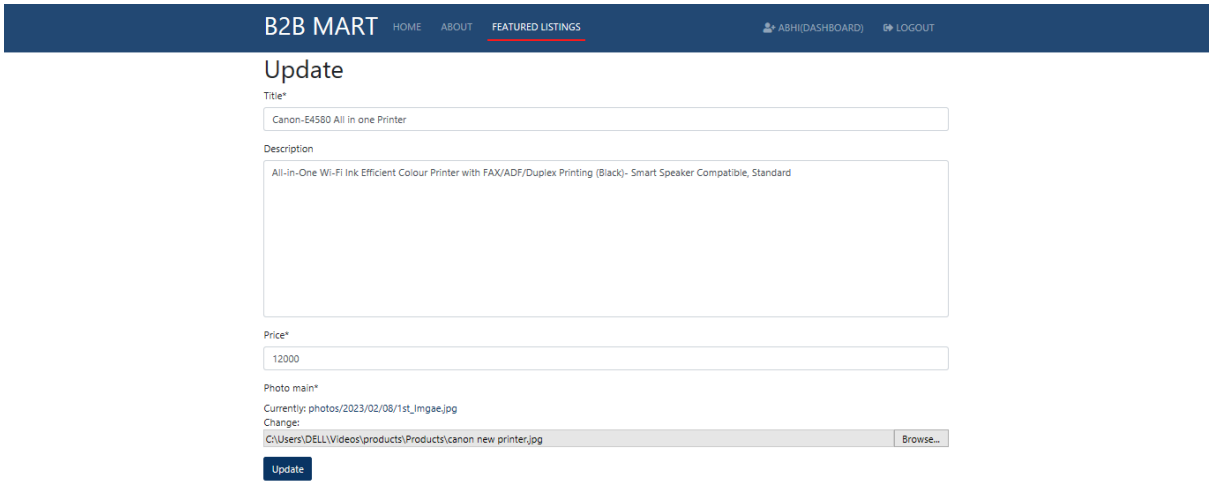
Now if the Wholesaler wants to create a new listing he/she can click on the create button on the dashboard:



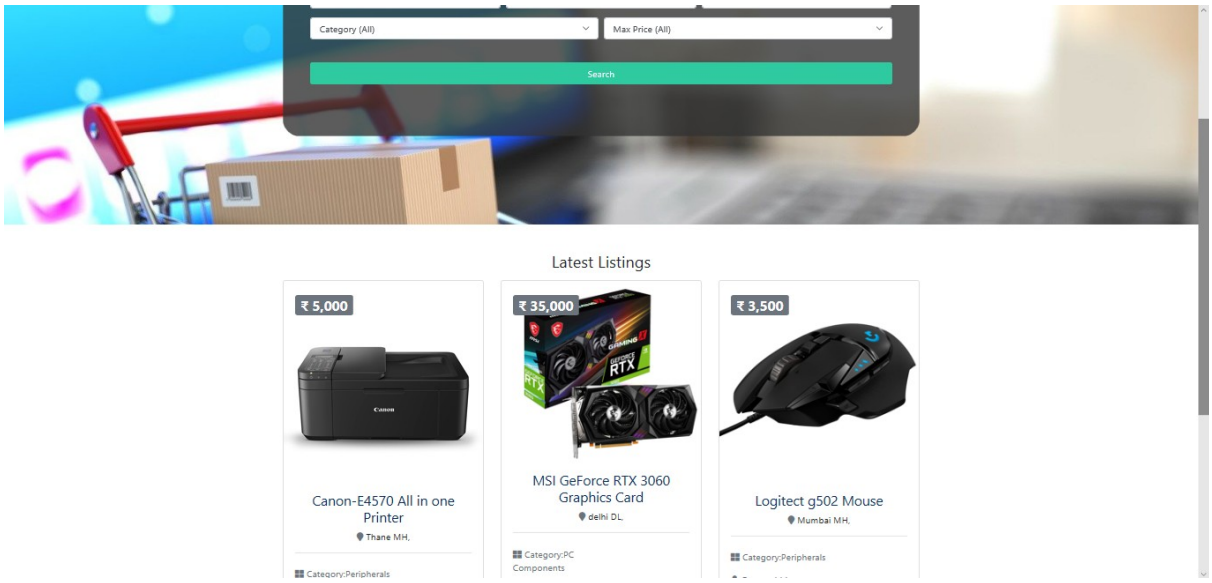
We can see that the Canon-E4570 All in one Printer listing has be created:



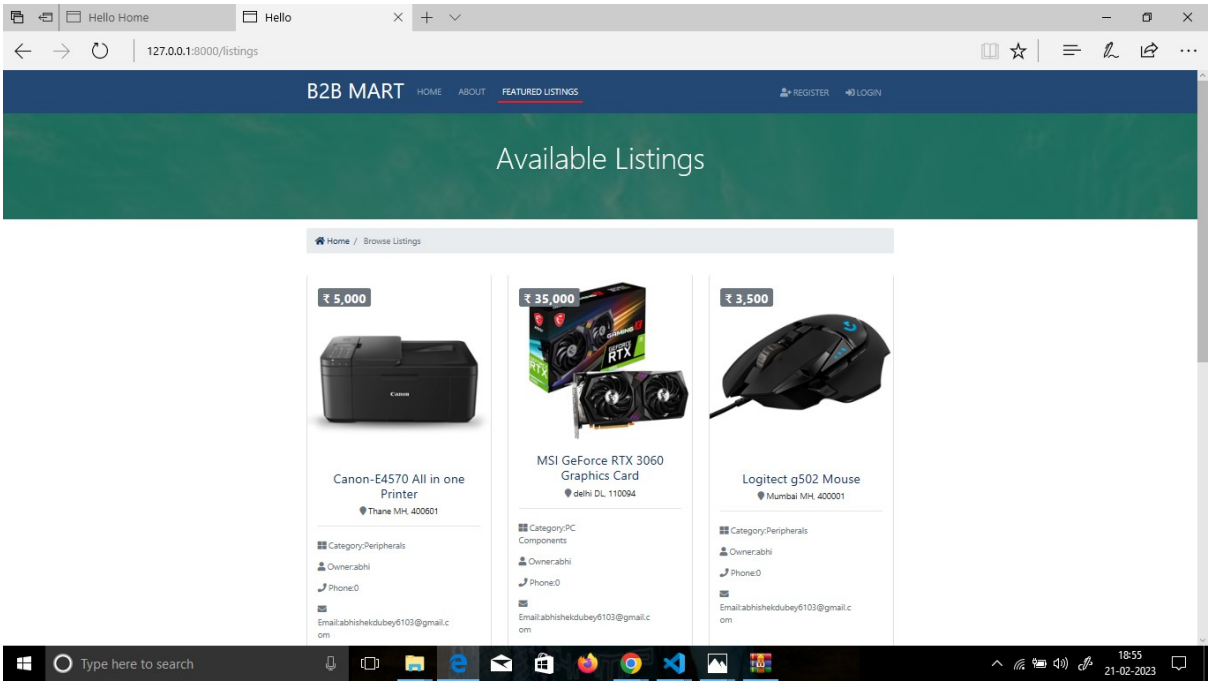
We can also see that there are 2 Canon-E4570 All in one Printer listings which can be also corrected by updating the listing:



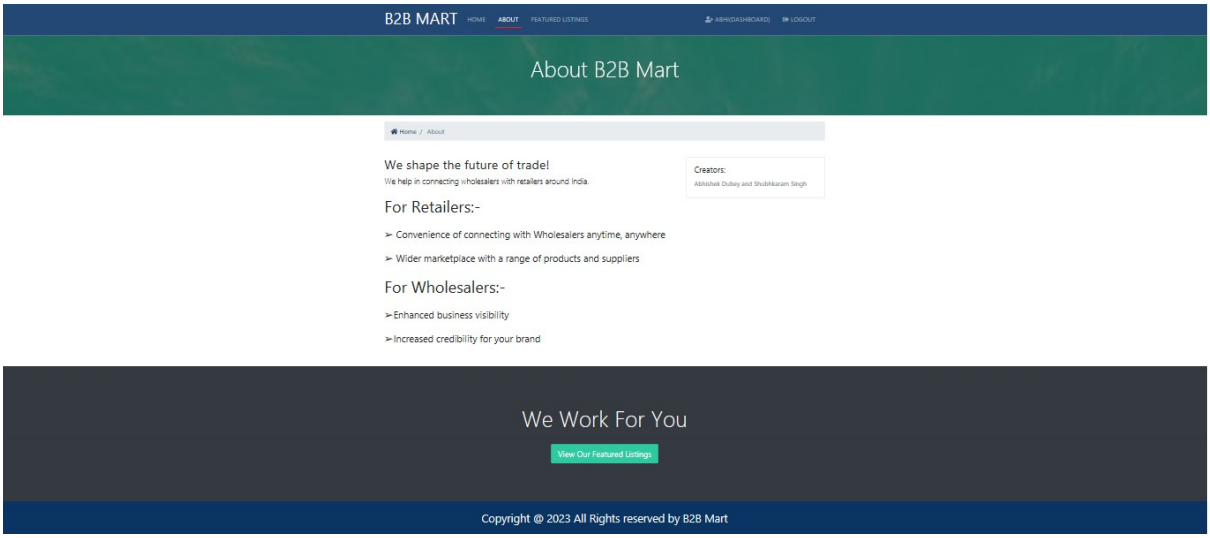
Now the listing is being shown on the home screen under the Latest Listings:



All the available Listings:



About Us Page:



Conclusion and Future Work

7.1 Advantages over Current System:

In the Existing system the work are done only manually but in proposed system we can do our with computerized system using this application. Existing system includes following points:-

- Lack of security of data.
- More man power.
- Time consuming.

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work.

- Security of data.
- Ensure data accuracy's.
- Less Human Errors
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.
- Minimum time required.

7.2 Future Enhancements:

This project also has some future enhancements in the evergreen and booming IT industry.

Further enhancement of the software will have the following features:

- There will be a product tracking feature available which the wholesaler can use to let the retailer know about the current details of the shipment by mentioning Shipping soon, Shipped, In-transit, Out for delivery and Delivered.
- In future there will also be an inbuilt payment gateway for the ease of transaction.
- Generate an invoice for the wholesaler for each order placed.
- A Rating System for the products.
- Other possibilities.

7.3 Conclusion :

- An attempt is made in all its earnest towards the successful completion of the project. The system is verified with valid as well as invalid data.
- The system is user friendly as it has been developed with Django framework using visual studio code.
- Upgradation of the system can be done without affecting the current proper functioning of the system.

Although I have put my best efforts to make the software flexible, easy to operate but limitations cannot be ruled out even by me.

List of limitation which is available in project are:

- Eventually both the wholesaler and retailer have to contact each other either via email or contact number.
- Since this is a web application it cannot work in offline mode and the constant need of an internet connection a must.

7.4 System Maintenance:

The software will definitely undergo changes once it is delivered to the customer. There can be many reasons for these changes to occur. Change could be because of some unexcepted input values into the system. The software should be developed to accommodate changes that could happen during post implementation period.

In Maintenance we will be doing the following:

- Fixing bugs if at all anything found during actual working.
- Any minor changes that is required when the client working with it will be done.
- Periodic checking of software at regular intervals.
- Make better use of existing tools and techniques.

References

To prepare this project we required information regarding how to develop efficient & proper software on B2B management system.

Reference Websites:

1. https://www.google.com
2. https://www.youtube.com
3. https://stackoverflow.com
4. https://docs.djangoproject.com
5. https://www.w3schools.com