

## SSR COLLEGE OF ARTS, COMMERCE & SCIENCE SILVASSA

**(Affiliated to Savitribai Phule Pune University, NAAC Accredited with B+ Grade)**

**Submitted to the partial fulfillment of**

T.Y BBA(CA)

### 2022-2023

Project Work

“GOLDEN BAKERY”

# Guided By: Submitted By:

AMAN GUPTA

MR. IMTIYAZ PADARSHI



## SSR COLLEGE OF ARTS, COMMERCE & SCIENCE SILVASSA

**(Affiliated to Savitribai Phule Pune University, NAAC Accredited with B+ Grade)**

**Sayli, Silvassa-396230, D&N.H**

Department of Computer Application

**CERTIFICATE**

This is to certify that Mr. **AMAN KUMAR GUPTA of** T.Y.B.B.A

[Computer Application] has successfully completed his project work

on the topic **GOLDEN BAKERY** in the academic year 2022-2023.

**Project Guide** **H.O.D**

**Seal of the College**

**Internal Examiner External Examiner**

**ACKNOWLEDGEMENT**

It is a great pleasure to acknowledge and express our deep sense of gratitude to **SSR College of ARTS COMMERCE & SCIENCE** for giving an opportunity to such project.

I extremely grateful and remain indebted to our guide **Mr IMTIYAZ PADARSHI** for being a source of inspiration and for his constant support in the Design, Implementation and Evaluation of the project. I am thankful to him, for his constant constructive criticism and valuable suggestions, which benefited us a lot while developing the project on **GOLDEN BAKERY**. He has been a constant source of inspiration and motivation which helped us to complete this project successfully.

I would like to thank Savitribai Phule Pune University for providing us an opportunity to apply our knowledge and skills in a practical environment as a part of curriculum for **T.Y.B.B.A [Computer Application].**

Lastly but significantly, we express sincere gratitude to all our friends and fellow students at SSR College for their help and timely advice on various occasions during this project.

**Project Associates:**

**AMAN KUMAR GUPTA**

**ABSTRACT**

The project titled **“GOLDEN BAKERY”** is designed with IDE-Visual STUDIO (HTML, CSS, JAVASCRIPT, PHP) as front end and XAMPP as back end.

There are different items available in bakery; many customers will order at different time from the catalogue online, after ordering a confirmation of response will be shown on next page.

This system is implemented to reduce the manual work and enhances the accuracy of work in a bakery. This system manages and maintains the records of customers and their order.

The Webpages are made in user-friendly interface to grab the screen quickly. The system is totally built at users and admin end.

The design form and the data report screenshots, advantages and limitations data dictionary, and future enhancement have been included.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **TOPIC** | **PAGE.NO** |
| 1 | **INTRODUCTION**  1.Introduction to System 2.Scope Of The System | 6-8 |
| 2 | **TOOL INFORMATION**   1. Front End Tool 2. Back End Tool | 9-11 |
| 3 | **ANALYSIS**   1. Feasibility Study | 12-13 |
| 4 | **SOFTWARE AND HARDWARE REQUIREMENT** | 14-15 |
| 5 | **SYSTEM DESIGN**   1. DFD 2. USE CASE DIAGRAM 3. CLASS DIAGRAM 4. ACTIVITY DIAGRAM 5. DATA DICTIONARY | 16-31 |
| 6 | **INPUT AND OUPUT DESIGN**  1.Screenshots | 32-45 |
| 7 | **REPORTS** | 46-50 |
| 8 | **ADVANTAGES AND LIMITATIONS** | 51-52 |
| 9 | **FUTURE ENHANCEMENT** | 53-54 |
| 10 | **BIBLIOGRAPHY** | 55-56 |

**1. INTRODUCTION**

**1.1 INTRODUCTION TO SYSTEM**

“GOLDEN BAKERY” is a collection of webpages. This system is developed to automate day today activity of a bakery. Bakery is a kind of business that serve people with readymade foods.

This system can be used by employees in a bakery to handle the clients, their orders and can help them easily find the menu or place orders. The services that are provided is food ordering and customer information management, menu information management. The bakery menu is organized by categories (bread, cake, pastries, cookies) are its items.

Main objective to build the system is to provide ordering and billing service to the customer. Each menu item has a name and price.

Online Bakery system is the system for managing the bakery business. After successful login the user can access the catalogue page with items listed.

The project has been developed because many bakery have a lot difficulty to manage the business. By using manual method it is difficult to keep the correct customer information and may be loss the customer information.

**1.2 SCOPE OF THE SYSTEM**

The webpage deals with creating a bakery system which will automate the major bakery operations such as billing, keeping tracks of records of daily transactions.

Scopes that have been considered during the development of the project are as follows:

1. User Friendly
2. Easily Upgradable
3. Updating, Insertion, Creating, Deleting
4. System Consistency
5. Allowing Keyboard Inputs
6. Easy Database Handling

**2. TOOL INFORMATION**

**2.1 FRONT END TOOL**

**MICROSOFT VISUAL STUDIO**

It is an Intergrated Development Environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services, and mobile apps. It supports 36 different programming languages. The most basic edition of Visual Studio, the Community edition, is available free of charge. Under which we wrote html, css, php pages.

Visual Studio includes a code editor supporting IntelliSense(the code completion component).

It makes it very easy to get the user interface portion of your application up and running. It provides a fast way to work under web development.

To develop any system with the back-end tools which provide access to the database and also solves the database queries, to make system interactive with the user, the use of front end tools comes into the picture. The front-end tools make the user interface with the system easier and also provide a user-friendly environment to the system.

**2.2 BACK END TOOL**

**XAMPP v3.3.0**

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

It has got the following advantages:

* Define a database
* Query the database
* Add, update and delete the data
* Modify the structure of the database
* Secure data from public access.
* Communicate within network
* Export and import data

SQL is a structured query language that we use to communicate with XAMPP server.

**3 .ANALYSIS**

**3.1 FEASIBILITY STUDY**

The objective of the feasibility study is to solve the problem and to acquire the ease of its scope. Feasibility means practicable. The feasibility analysis focuses on the fact that whether the project will be acceptable or not. If the feasibility analysis confirms that the project is feasible, it can be taken up for development. In preliminary investigation we found that project feasible.

**Types of Feasibility:**

**a. Operational Feasibility**

Operational feasiblility makes a mark on whether the project can be done with full requirements that the clients provide or not. Includes following:

1. Analysing all requirements.

2. Can be done in given period of time or not.

**b. Technical Feasibility**

Technical feasibility takes into account the technical aspect of the project includes the following:

1. Whether the project can be implemented with the existing technology or not.

2. Whether the project technically compatible or not.

**c. Economical Feasibility**

It determines whether the requirement software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team estimated cost of hardware and cost of performing feasibility study and so on.

Studying the feasibility of the project, the project can be implemented with the given period of time and with the existing or specified technology. Hence, we can conclude that this project is feasible.

**4. SOFTWARE AND**

**HARDWARE**

**REQUIREMENTS**

**4.1 SOFTWARE REQUIREMENT**

Software used in the designing (code) of the system:

Operating system: Windows 10

Application Softwares:

* Microsoft Visual Studio [Front-End Tool]
* XAMPP [Back-End Tool]

**4.2 HARDWARE REQUIREMENT**

To run the application software of the system in the computer, the minimum hardware configuration required is as below:

* 1.7 GHz Pentium processor or other compatible
* Intel chipset motherboard
* 4 GB RAM
* Color Monitor or LCD
* Keyboard
* Mouse
* Printer

**5. SYSTEM DESIGN**

**5.1 DATA FLOW DIAGRAM**

The data flow diagram are pictorial or graphical representation of the system study. The data flow covers all the processes and data storage area, which takes place during any transaction in the system. The data flow diagrams are functionally into context level, zero level diagrams.

a. **Process:** Here flow of data is transformed

b. **External Entity:** A source or destination of data, which is external to the system.

c. **A Data Flow:** It is a packet of data. It may be in the form of document, letter, etc.

d. **Data storage:** Any storage of data but with no reference to physical memory of

storing.

**ZERO LEVEL DATA FLOW DIAGRAM**

Order Purchase Details

Customer Info

Customer Order and

Payment Details

**1.0**

**GOLDEN BAKERY SYSTEM**

3.0

ADMIN

2.0

CUSTOMER

Customer Info

Add and Modify

Existing Customer

Change Password

Admin and Customer

Payment Info

**FIRST LEVEL USER DATA FLOW DIAGRAM**

USER

2.0

CUSTOMER

AUTHENTICATION

2.2

LOGIN

login

2.2.6

ADD TO CART

2.1

REGISTER

2.2.1

USER ORDER SECTION

2.2.7

LOGOUT

register

2.2.3

CATALOGUE

2.2.4

MY CART

2.2.2

FEEDBACK

orders

feedback

2.3.5

CHANGE PASSWORD

2.3

PAYMENT

payment

register

2.2.1

RETURN TO USER LOGGED IN ORDER SECTION

2.2

LOGIN

2.3.1

LOGOUT

**SECOND LEVEL USER DATA FLOW DIAGRAM**

2.2.3

SEE CATALOUGE

2.2.1

USER ORDER SECTION

2.2.4

MY CART

2.3

PAYMENT

2.4

EMPTY CART

payment

**THIRD LEVEL ADMIN DATA FLOW DIAGRAM**

feedback

3.2.5

MESSAGE

register

admin

admin

2.3.2.2

CUSTOMER

2.3.2.1

ADMIN

register

2.3.4.2

MODIFY EXISTING USER

register

2.3.1.1

ADD NEW ADMIN

2.3.4.1

ADD USER

3.2.2

PASSWORD

3.2.3

DASHBOARD

3.2

NAVIGATION PAGE

ADMIN

login

3.0

ADMIN

AUTHENTICATION

feedback,

payments,

register,

orders

SIGNOUT

3.2.1

NEW ADMIN

3.3.4

CUSTOMERS

3.1

LOGIN

**5.2 USE CASE DIAGRAM**

A use case diagram is a graphical depiction of a user’s possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

The use cases are represented by either circles or ellipses. It specify the expected behavior and not the exact method of making it happen.

A use case diagram is usually simple. It does not show the detail of the use cases:

* It only summarizes some of the relationships between use cases, actors, and systems.
* It does not show the order in which steps are performed to achieve the goals of each use case.

GOLDEN BAKERY SYSTEM

ADMIN

USER

**5.3 CLASS DIAGRAM**

**What is Class Diagram?**

Class diagrams are a type of static structure diagram that represent the structure of a system by showing the classes, their attributes, and their relationships.

A class diagram is used to model object-oriented programming languages such as Java, C++, and Python, and is an essential tool for software development.

The purpose of a class diagram is to providea high-level overview of the system's structure and how it functions, which is useful for designing, analyzing, and communicating about the system.

A class diagram consists of classes, attributes, methods, and relationships between classes, which are represented using different symbols and notations.

Class diagrams are a critical part of the Unified Modeling Language (UML), which is a standard modeling language used in software engineering to design anddocument software systems.

**Purpose of the Class Diagram:**

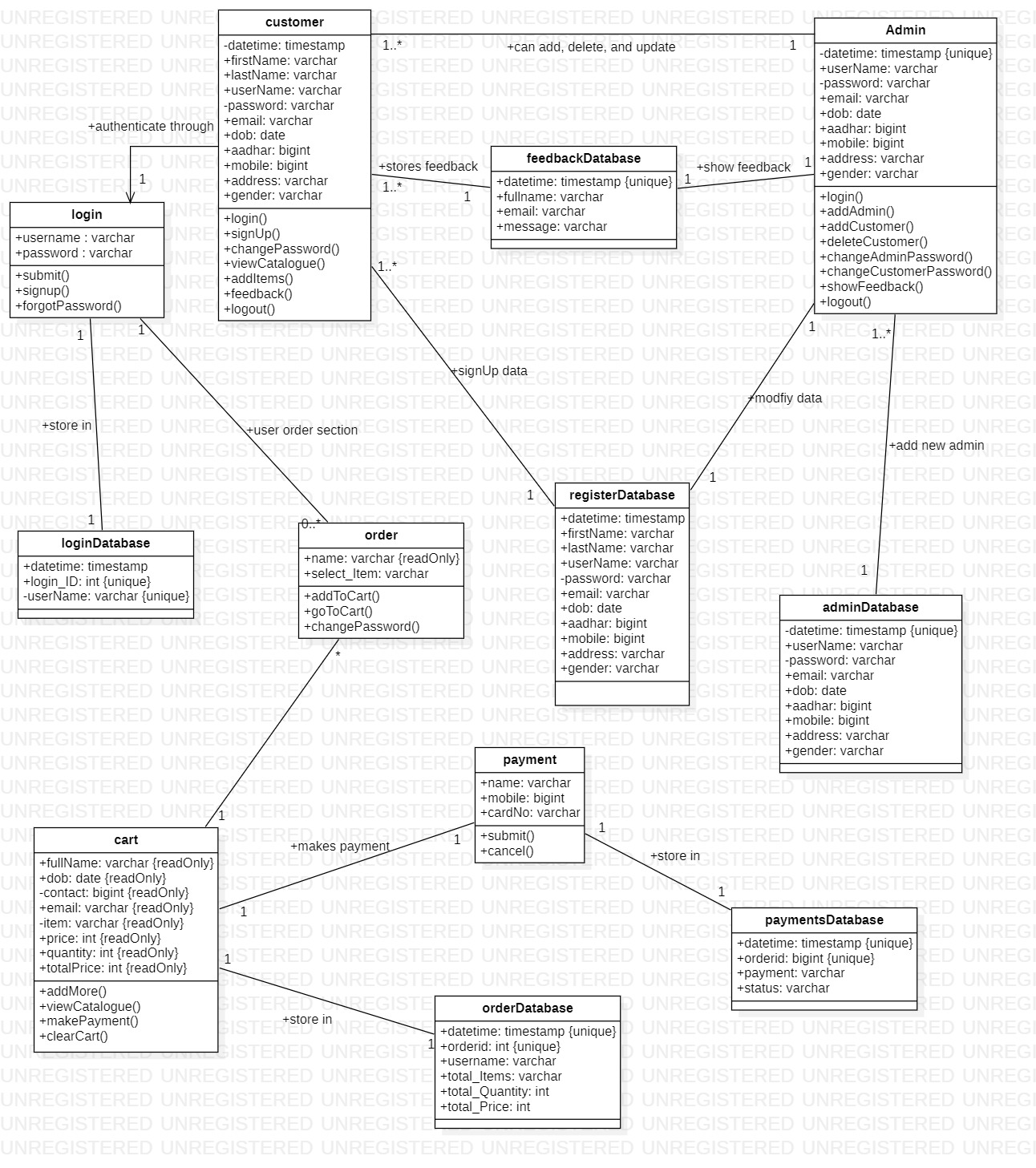
1. Analysis and design of the static view of an application.
2. Describes responsibilities of a system.
3. Base for component and deployment diagrams.
4. Forward and reverse engineering.

**Notations:**

**Notations : Table**

|  |  |  |
| --- | --- | --- |
| **Name** | **Symbol** | **Description** |
| **1. Class** |  | A class is a template or blueprint for creating objects that defines its attributes and behaviors. |
| **2. Visibility** |  | Visibility represents the accessibility of attributes and methods from outside the class. There are three types of visibility: public, private, and protected. |
| **3. Attribute** |  | An attribute is a data field that represents a characteristic or property of a class. |
| **4. Operation** |  | An operation is a method or function that performs a specific task or action. |
| **5. Responsibility** |  | Responsibility represents the behavior or task performed by a class within a system. |
| **6. Association** |  | Association represents the relationship between two classes that indicates how they are connected. |
| **7. Aggregation** |  | Aggregation represents a type of association where one class is composed of one or more instances of another class. |
| **8. Composition** |  | Composition represents a stronger form of aggregation where the lifetime of the composed object is dependent on the lifetime of the composing object. |
| **9. Generalization** |  | Generalization or inheritance represents the relationship between a superclass and its subclasses, where the subclasses inherit the attributes and behaviors of the superclass. |
| **10. Realization** |  | Realization represents the relationship between an interface and its implementing class. |
| **11. Dependency** |  | Dependency represents a relationship between two classes where one class depends on the other class to perform its function. |
| **12. Multiplicity** |  | Multiplicity represents the number of instances of a class that can be associated with another class. It is denoted using a range, such as 0..1, 1..\*, or 2..4, etc. |

**CLASS DIAGRAM OF THE SYSTEM**

****

**5.4 ACTIVITY DIAGRAM**

**What is Activity Diagram?**

An activity diagram is a type of UML diagram used to model the flow of activities or processes within a system.

It provides a visual representation of the sequence of actions and decisions taken by actors or objects in a system or process.

Activity diagrams are useful for analyzing, designing, and documenting complex workflows or business processes.

The diagram consists of nodes, edges, and symbols that represent various elements such as actions, decisions, conditions, and swimlanes.

Activity diagrams can be used for both high-level and detailed level design of a system or process.

**Purposes of Activity Diagrams:**

The purpose of an activity diagram is to model and illustrate the flow of activities and actions within a system or process.

It provides a visual representation of the steps, decisions, and paths taken by actors or objects to accomplish a specific goal or task. Activity diagrams can help in analyzing, designing, and documenting business processes, software systems, and other complex workflows.

**Notations:**

**Table: Notations in Activity Diagram**

|  |  |  |
| --- | --- | --- |
| **Name** | **Symbol** | **Description** |
| **1. Start/Initial Node** |  | Represents the starting point of the activity diagram where the process or activity begins. |
| **2. Final/Exit Node** |  | Represents the end point of the activity diagram where the process or activity is completed. |
| **3. Action** |  | Represents an operation or task performed during the process. |
| **4. Activity** |  | Represents a series of actions or tasks performed in a particular order to accomplish a specific goal. |
| **5. Transition/Edge/Path** |  | Represents the flow of control from one activity to another activity. |
| **6. Fork Node** |  | Represents a point where the process splits into multiple parallel paths, each executing independently. |
| **7. Join Node** |  | Represents a point where the parallel paths of a fork node merge back into a single path. |
| **8. Condition** |  | Represents a logical expression that determines the path to be followed based on the current state of the system. |
| **9. Decision/Branch** |  | Represents a point where a decision needs to be made based on a condition or criteria, leading to different paths or outcomes. |
| **10. Note** |  | Represents additional information or comments related to an activity or decision in the diagram. |
| **11. Swimlanes** |  | Represents the organizational units or actors responsible for performing the activities in the diagram. |
| **12. Flow Final Node** |  | Represents a point where the flow of control terminates without any further activity. |

**ACTIVITY DIAGRAM OF THE SYSTEM**

Customer

Admin

**Start**

Add Item

Catalogue

Change Password

Logout

Feedback

About

End

Password

Dashboard

Customer

Cart

Go to Cart

SignOut

Payment

Proceed to checkout

Clear Cart

View Catalogue

Add More

No

No

Yes

Yes

Customers

Admin

Modify Existing

Add New

Messages

Add Admin

login

Userpage

login

register

**5.5 DATA DICTIONARY**

Data dictionary is a repository that contains description of all the data objects consumed By the software. It is a list of names used by the system alphabetically. As well as the name, the dictionary should include a description of the named entity and, If the name represents of a composite object, there may be description of the name entity. Other information such as the date of creation, creator and the representation. Entity may also include depending on the type of module, which is being developed. The data dictionary software can check for name uniqueness and tell requirements analyst duplication.

It serves as store of organizational information which can link analysis, design, implementation and evolution. As the system is developed, information is taken to inform the development. New information is added in it. All information about entity is in one place.

**DATA REPORTS :**

TABLE : **register**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| Datetime | Timestamp | Auto Stores users input time. |
| firstName | varchar(30) | Stores users first name |
| lastName | varchar(30) | Stores users last name |
| dob | Date | Stores users D.O.B |
| Email | varchar(30) | Stores users email |
| contact | bigint(10) | Stores users mobile no. |
| aadhar | bigint(12) | Stores users aadhar no. |
| username | varchar(30) | Stores users username (Primary Key) |
| password | varchar(255) | Stores users password in hash format |
| gender | varchar(6) | Stores users gender |
| address | varchar(40) | Stores users address |

TABLE: **login**

|  |  |  |
| --- | --- | --- |
| Field Name | Data type | Description |
| Datetime | Timestamp | Auto stores users input time. |
| Login\_ID | int(11) | Auto increment |
| username | varchar(14) | Foreign key of column - username (register table) |

TABLE : **feedback**

|  |  |  |
| --- | --- | --- |
| Field Name | Data type | Description |
| Datetime | Timestamp | Auto Stores users input time. |
| fullname | varchar(20) | Stores users fullname |
| email | varchar(15) | Stores users email-id |
| msg | varchar(50) | Stores users message |

TABLE : **orders**

|  |  |  |
| --- | --- | --- |
| Field Name | Data type | Description |
| Datetime | Timestamp | Auto stores users input time |
| Username | varchar(30) | Stores users username |
| OrderID | int(11) | Stores orderId (Primary Key) |
| Total\_Items | varchar(50) | Stores total items |
| Total\_Qty | int(11) | Stores total quantity |
| Total\_Price | int(11) | Stores total price |

TABLE : **Payment**

|  |  |  |
| --- | --- | --- |
| Field Name | Data type | Description |
| Datetime | Timestamp | Auto stores users input time. |
| OrderID | int(11) | Foreign key of column - orderId (orders table) |
| Payment | varchar(20) | Stores “paid” if user submit the card details else “Not Initialized” is stored |
| Status | varchar(20) | Stores delivery status |

TABLE : **admin**

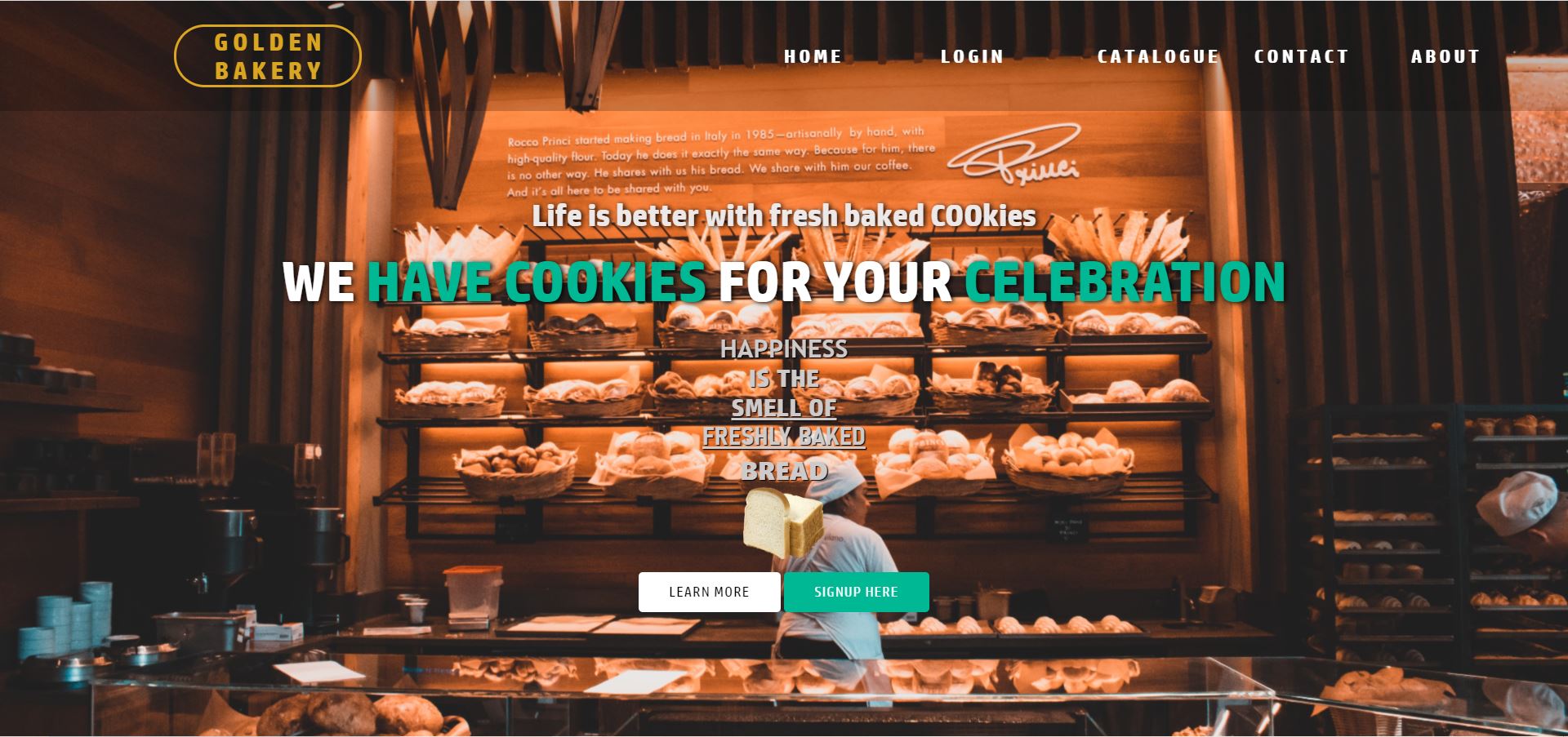
|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| Datetime | Timestamp | Auto Stores users input time. |
| dob | Date | Stores users D.O.B |
| Email | varchar(30) | Stores users email |
| contact | bigint(10) | Stores users mobile no. |
| aadhar | bigint(12) | Stores users aadhar no. |
| username | varchar(30) | Stores users username (Primary Key) |
| password | varchar(255) | Stores users password in hash format |
| gender | varchar(6) | Stores users gender |
| address | varchar(40) | Stores users address |

**6. INPUT AND OUTPUT**

**DESIGN**

**6.1 SCREENSHOTS**

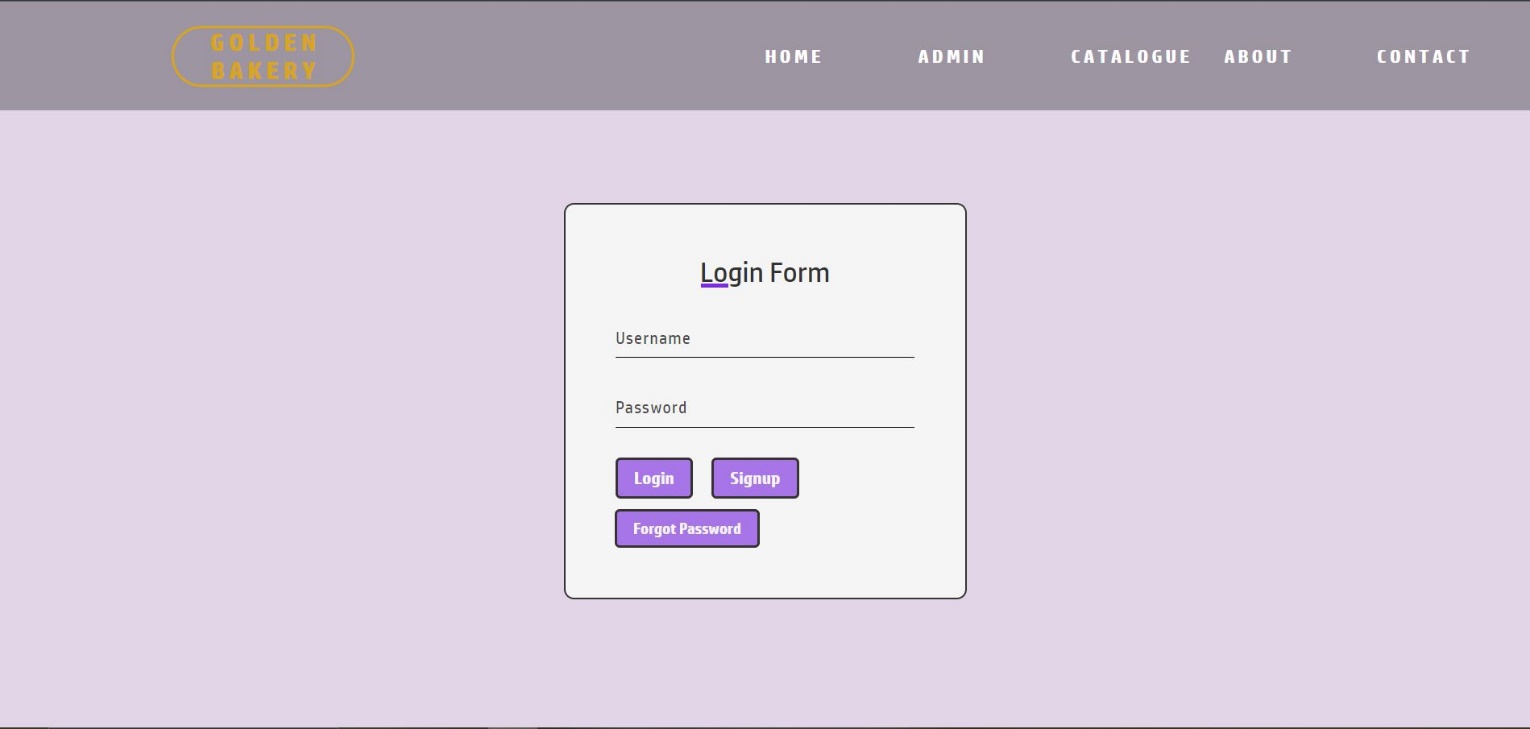
HOME FORM:



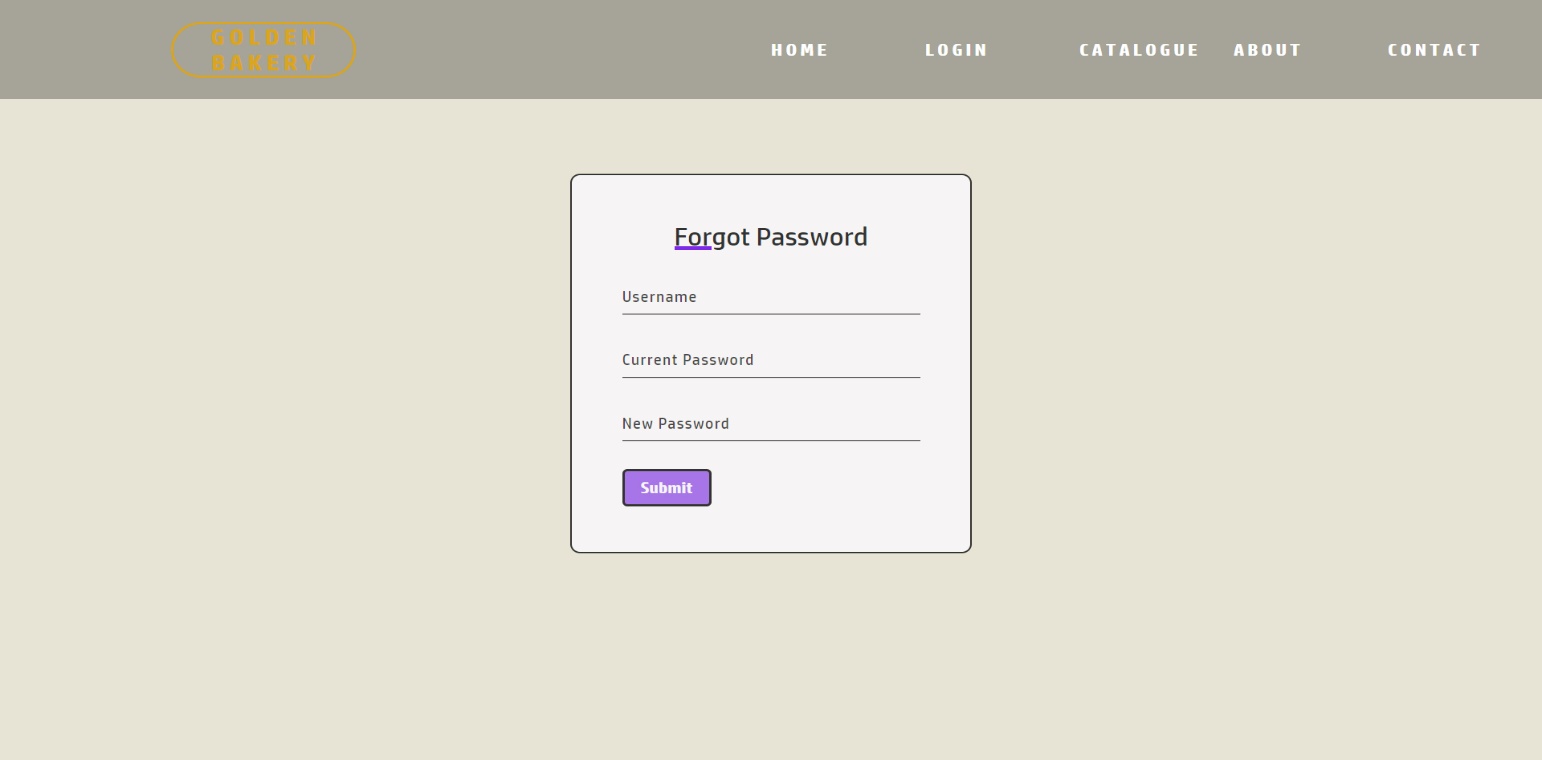
REGISTRATION FORM:



LOGIN FORM:



FORGET PASSWORD PAGE:



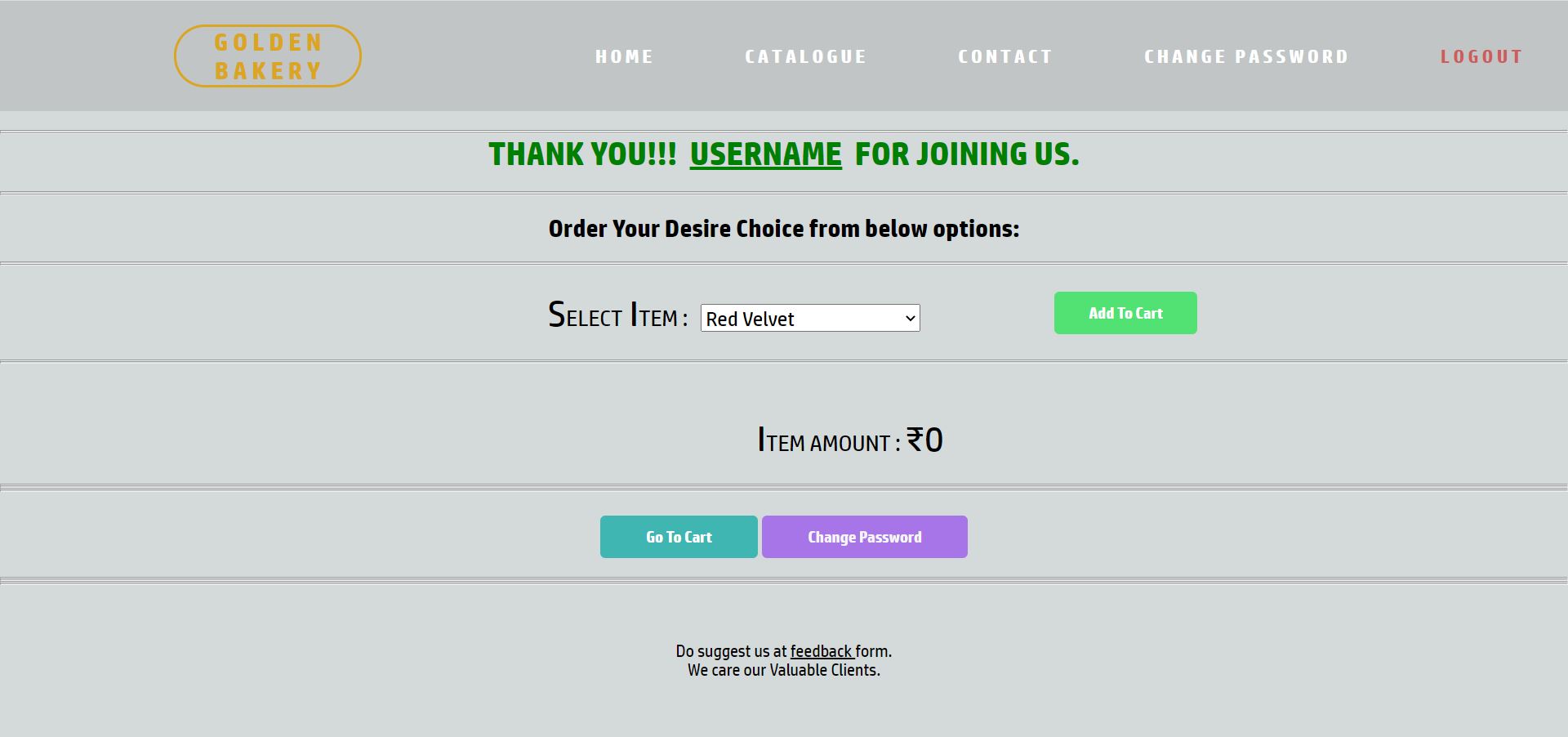
CATALOGUE PAGE:



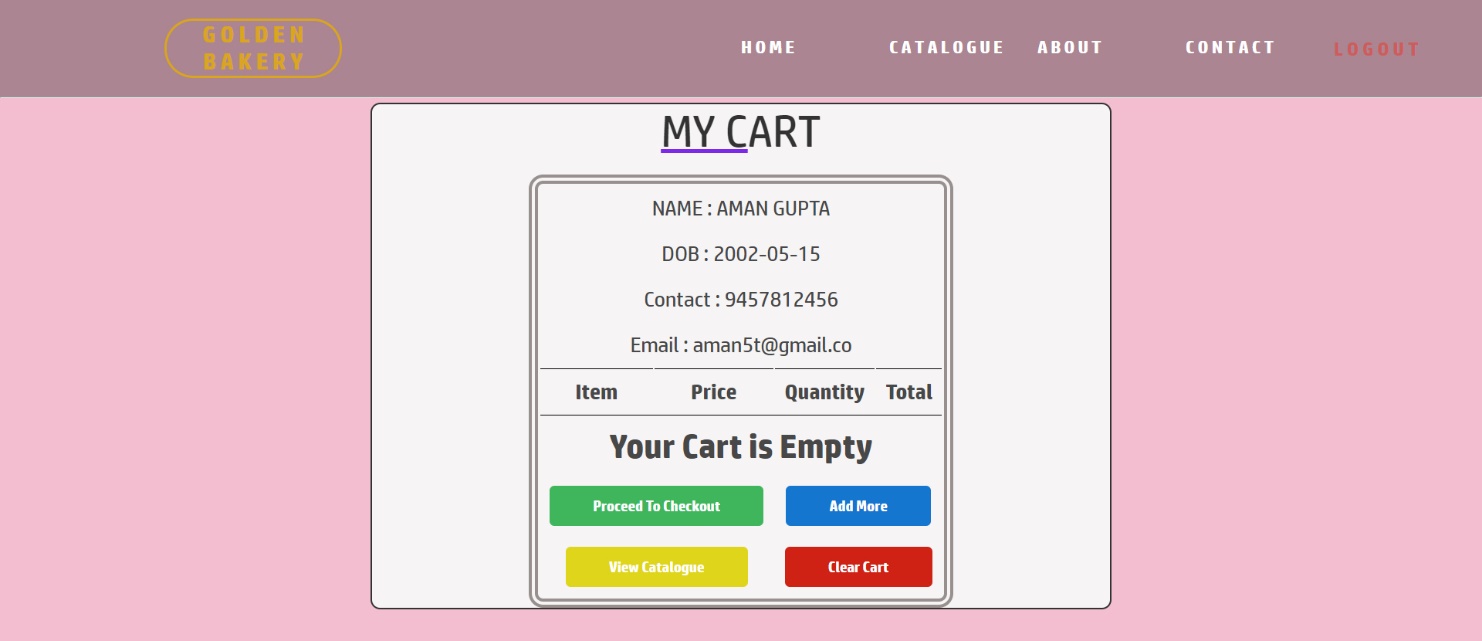
CATALOGUE FOR LOGGED IN USERS:



USERPAGE:



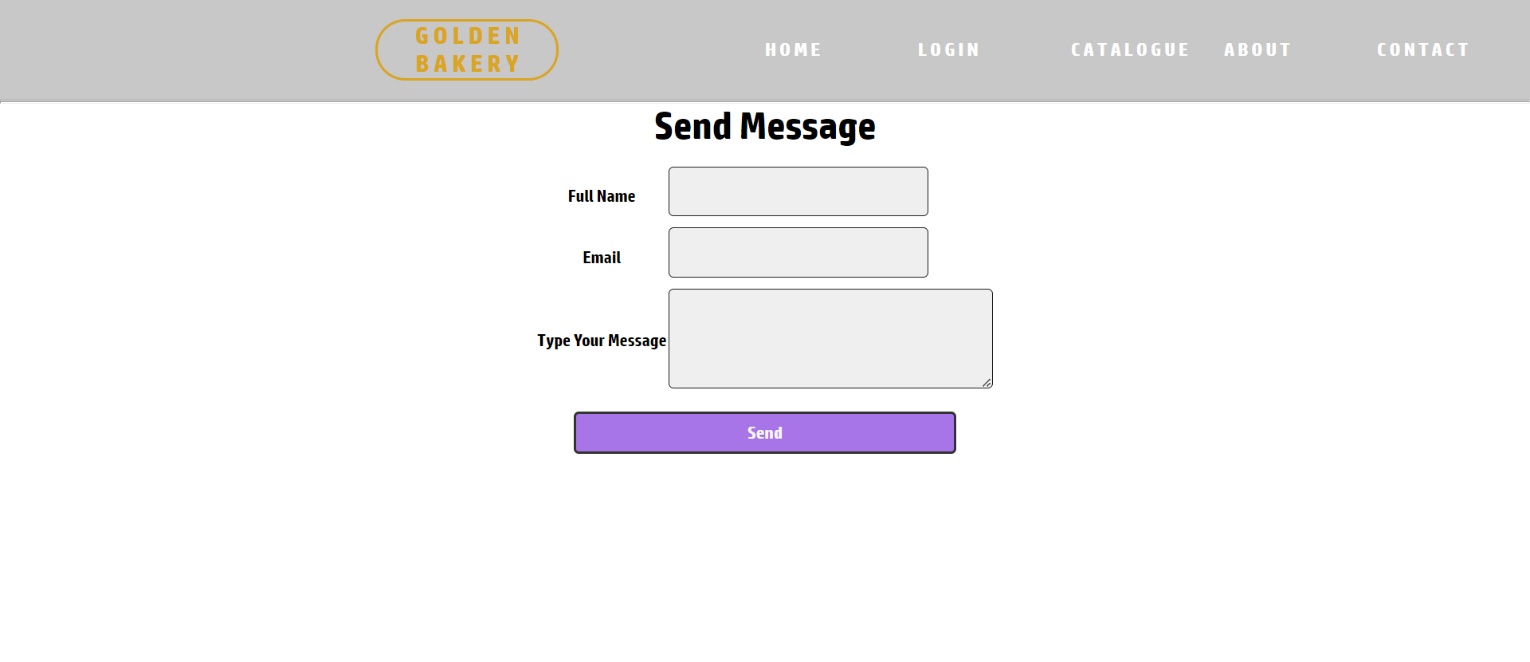
CART PAGE:



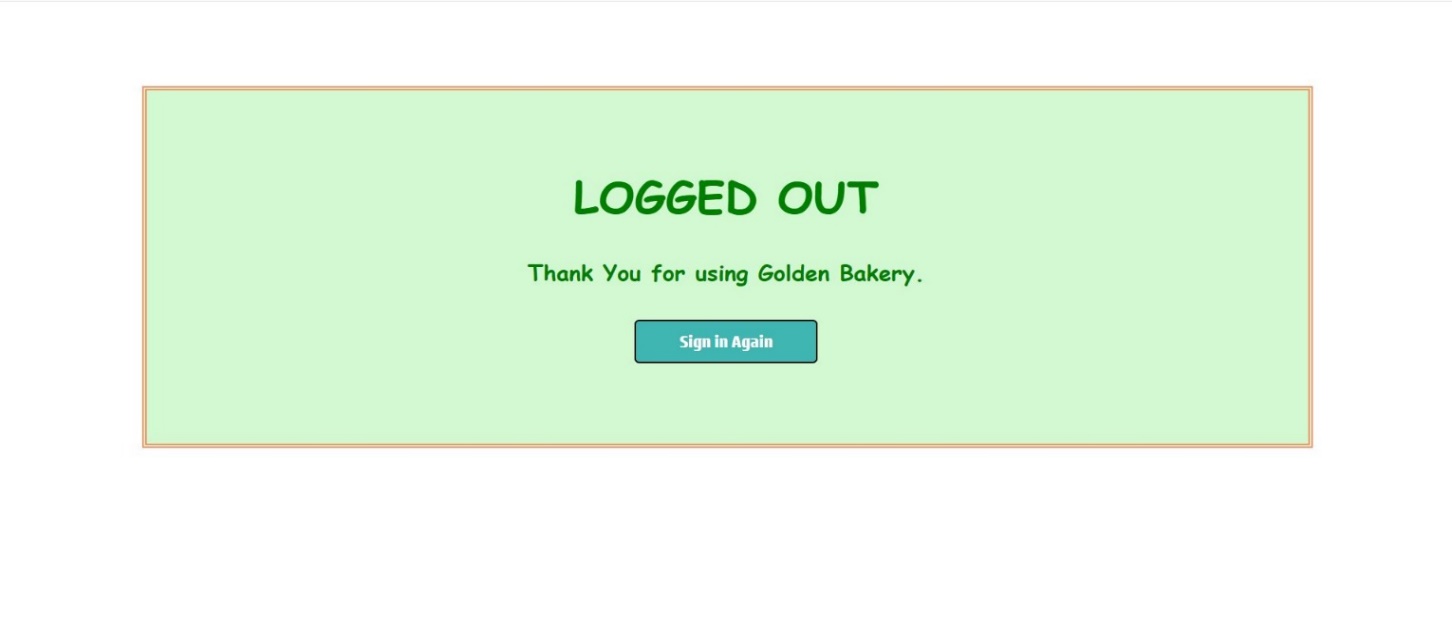
ORDER PAGE



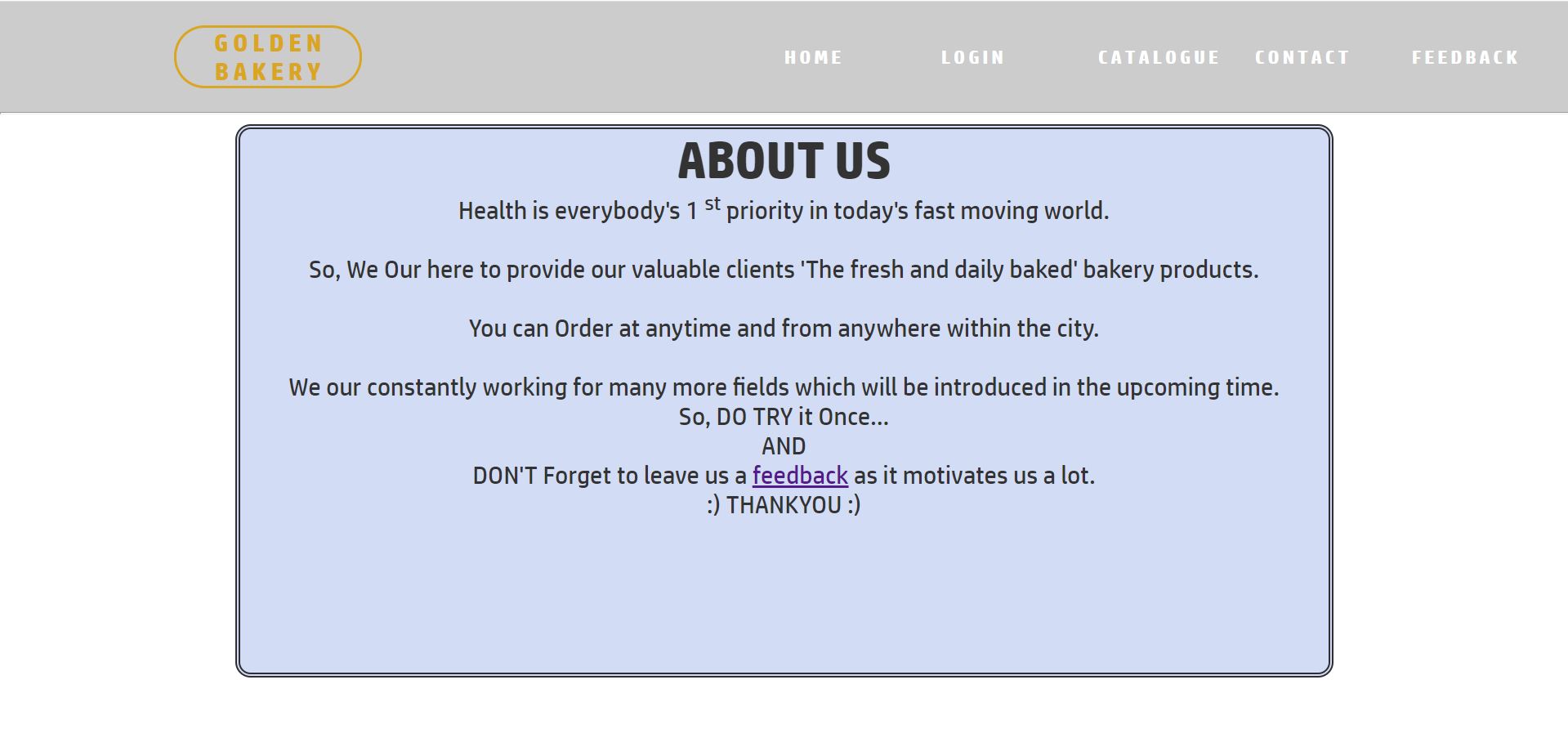
FEEDBACK PAGE



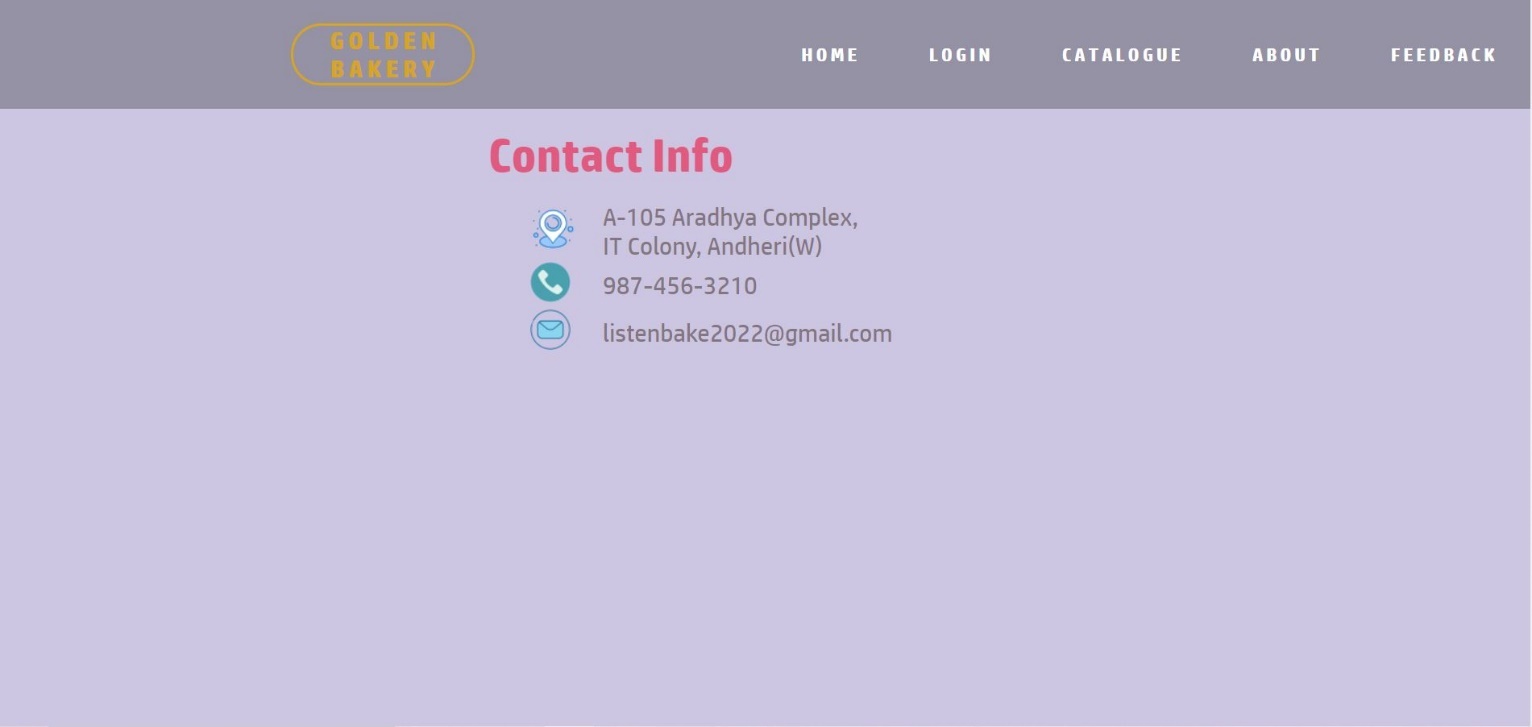
THANK YOU PAGE:



ABOUT PAGE



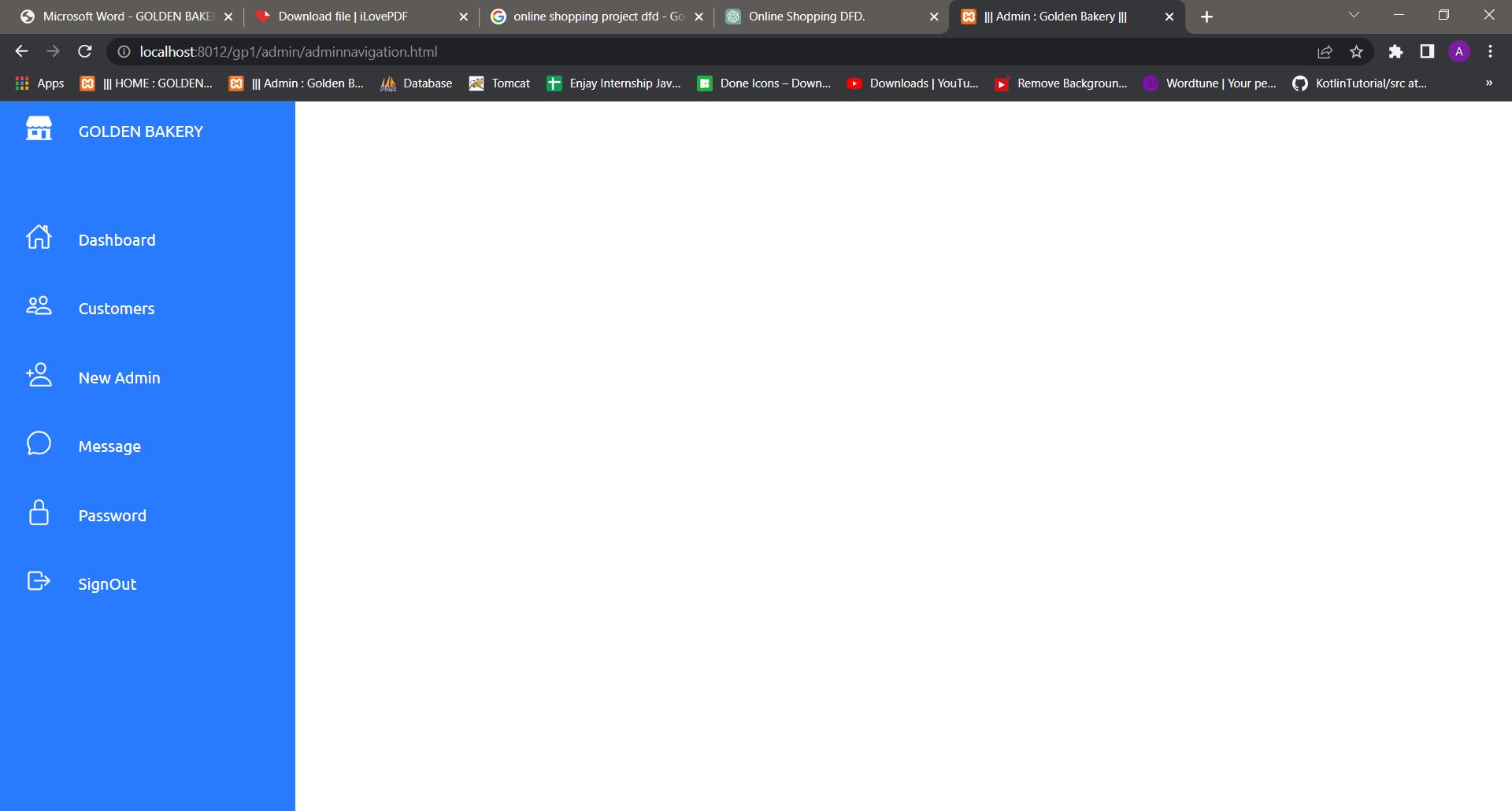
CONTACT PAGE:



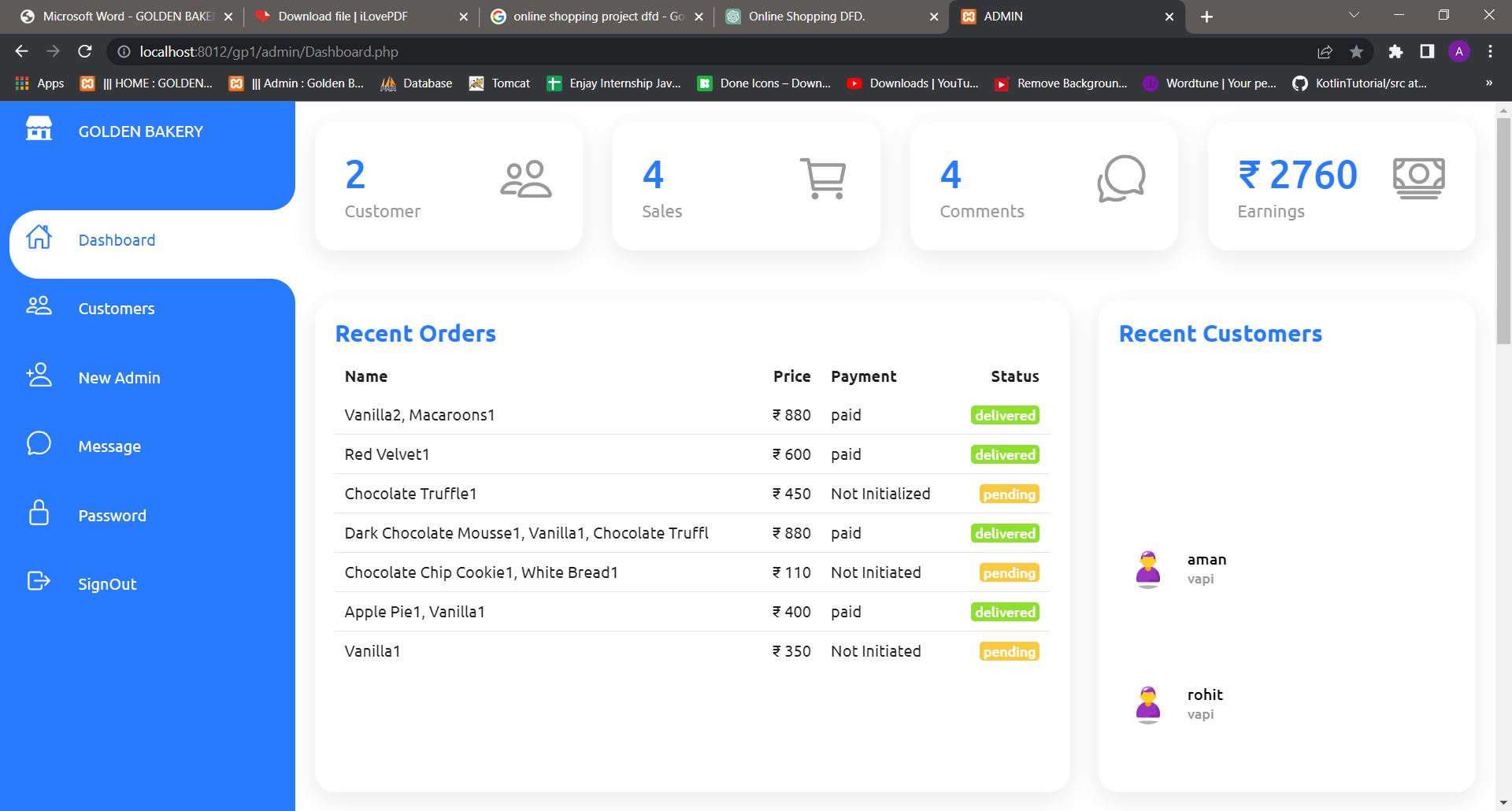
ADMIN LOGIN FORM:



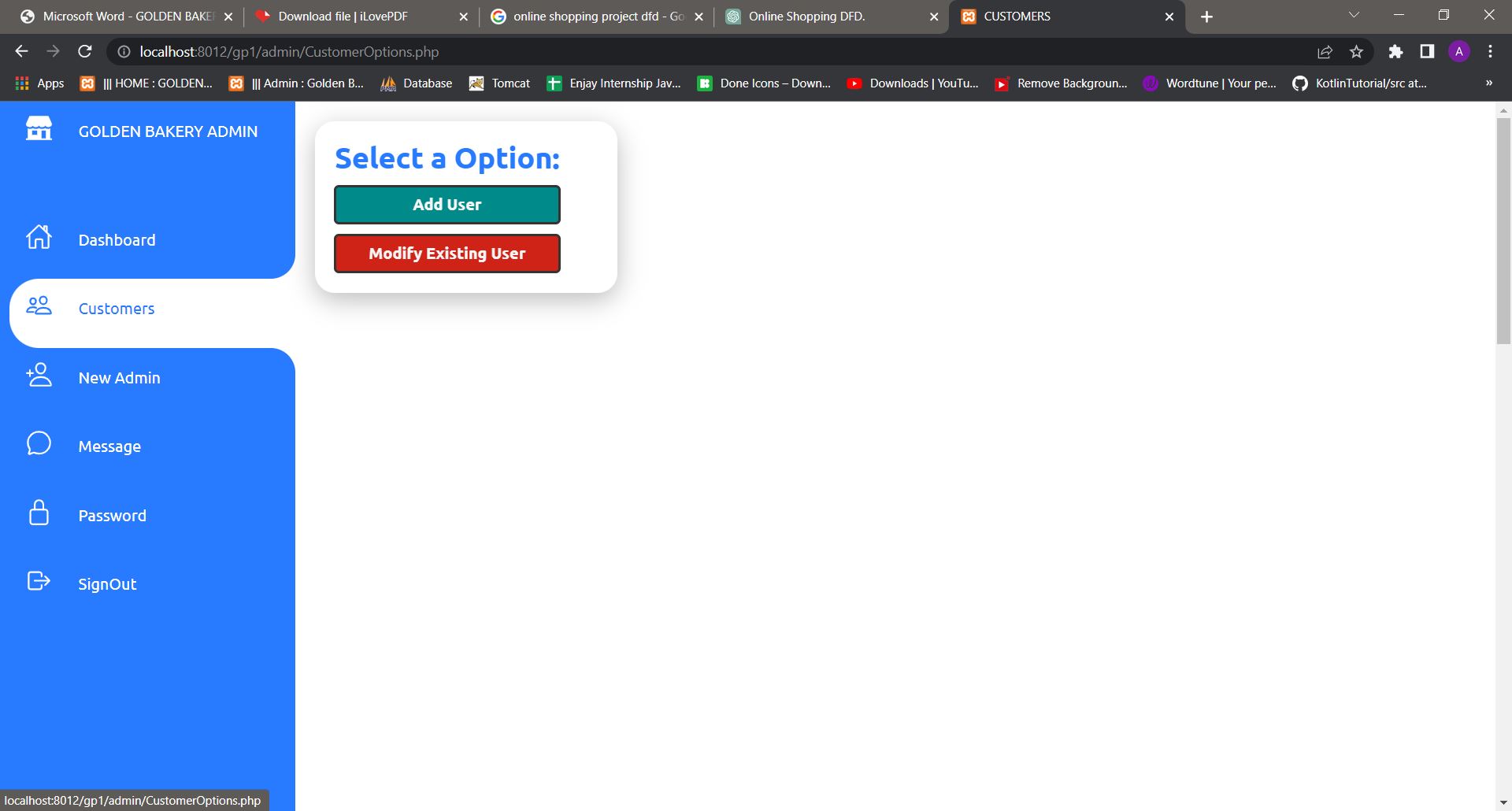
ADMIN NAVIGATION PAGE:



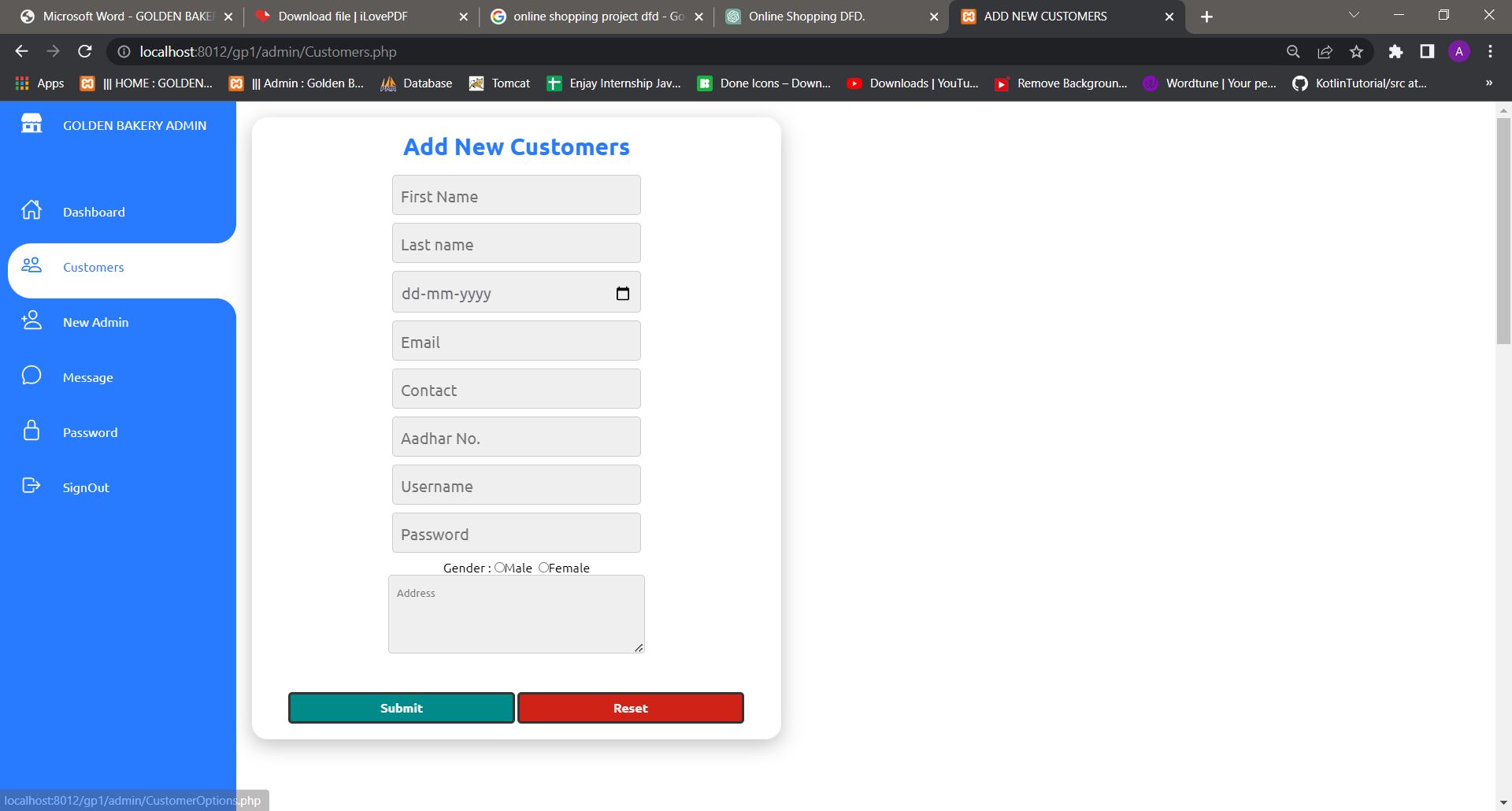
ADMIN DASHBOARD:



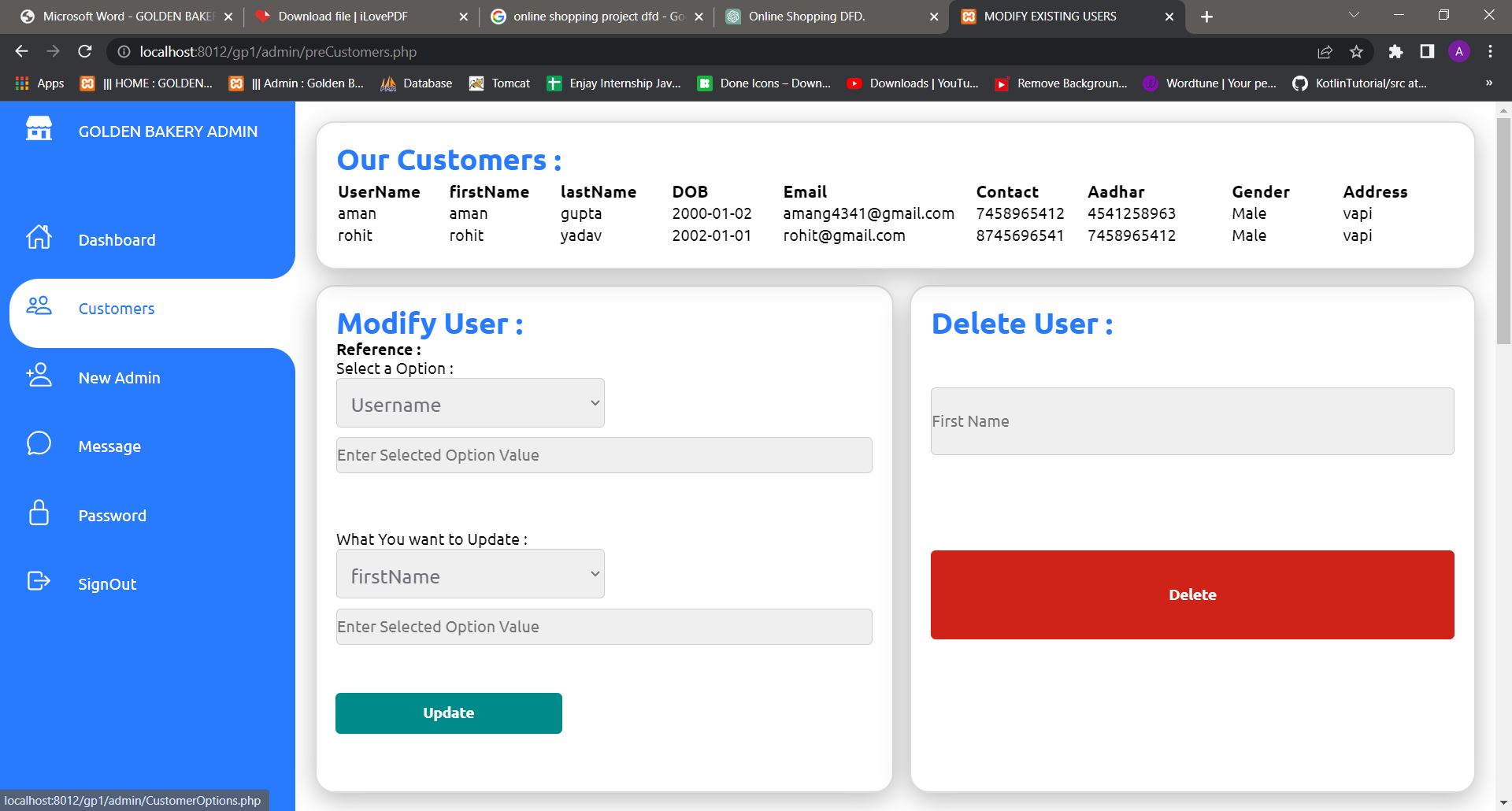
CUSTOMERS PAGE:



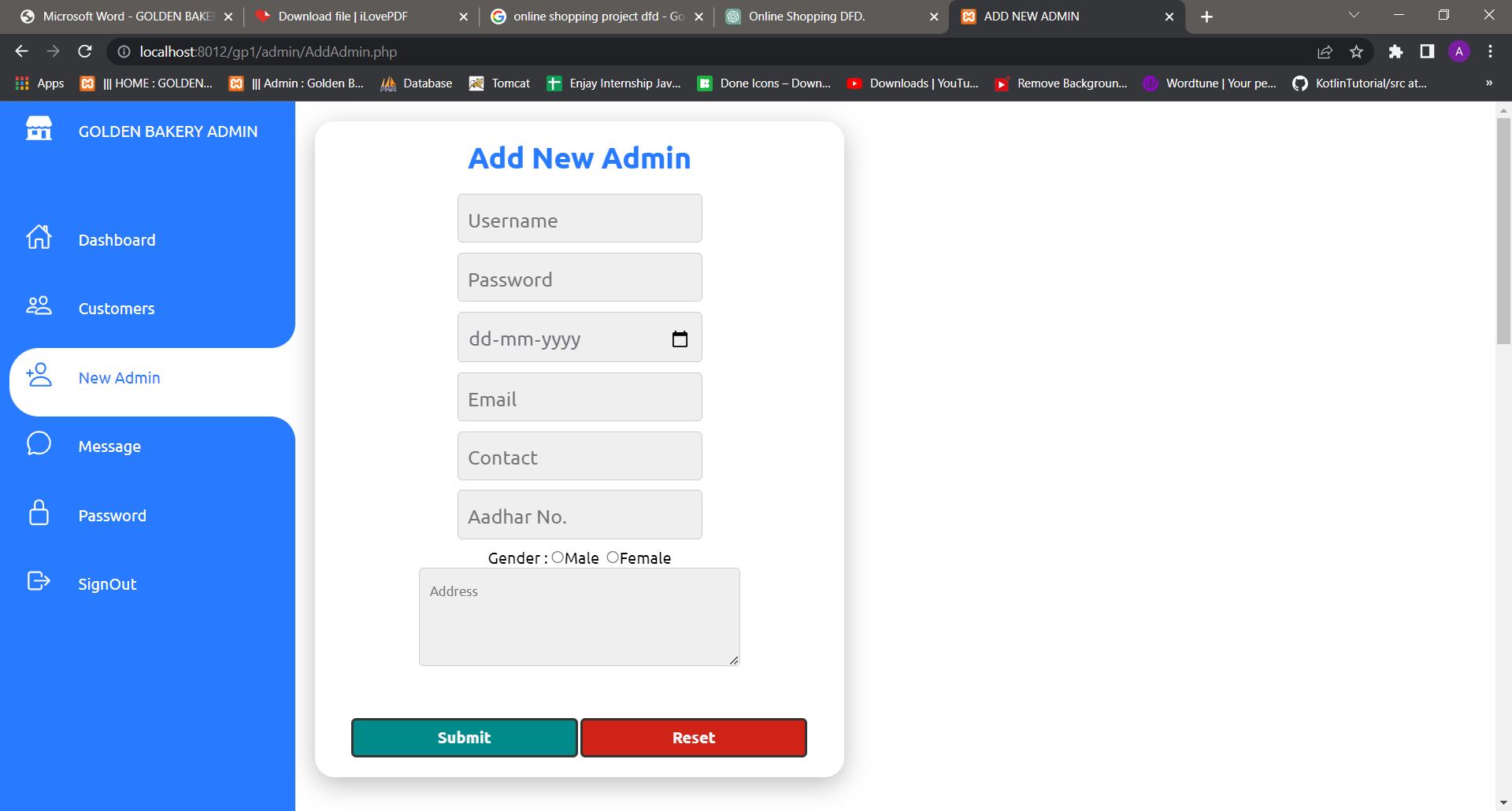
ADD NEW CUSTOMER:



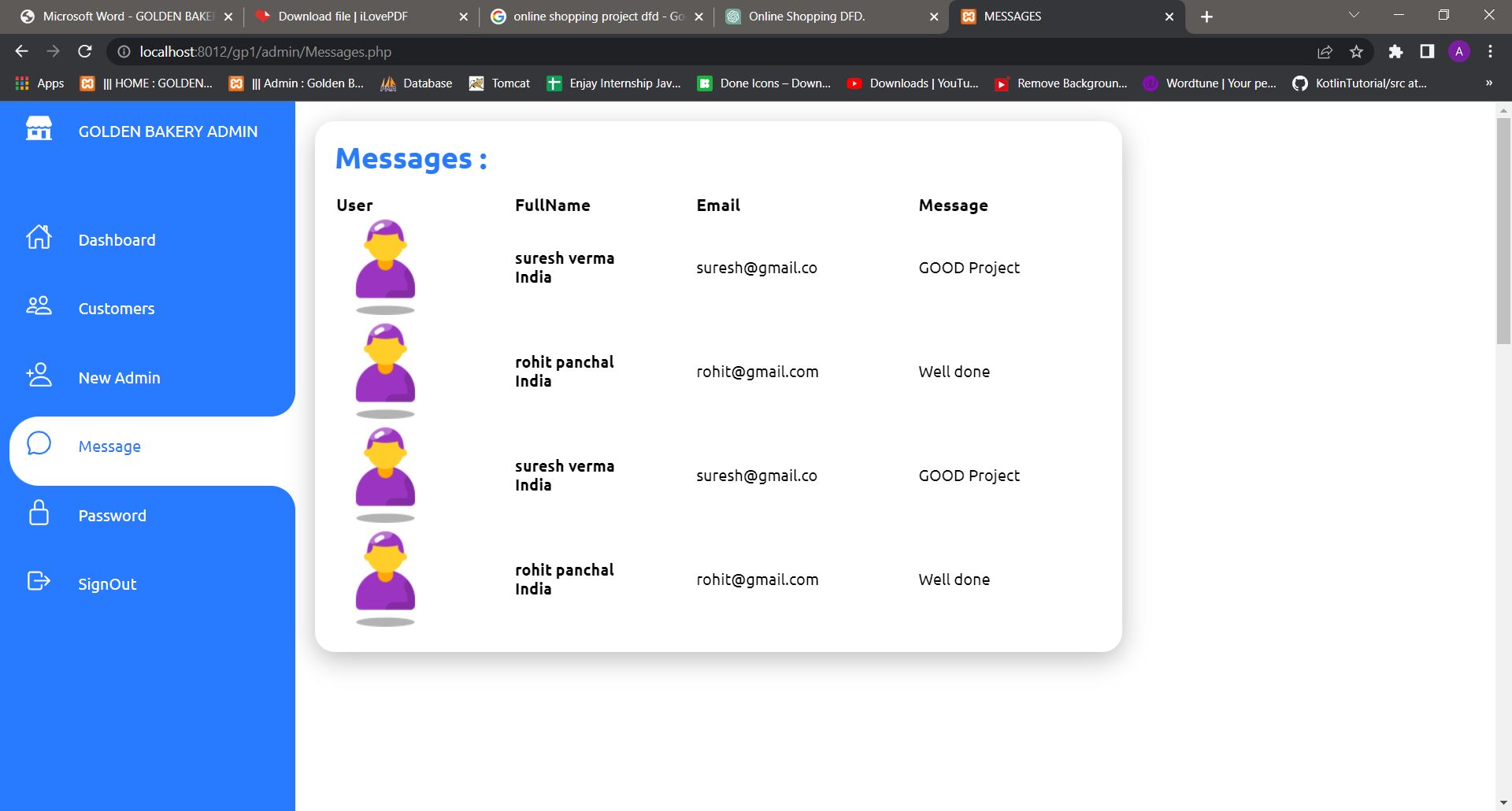
MODIFY EXISTING CUSTOMERS:



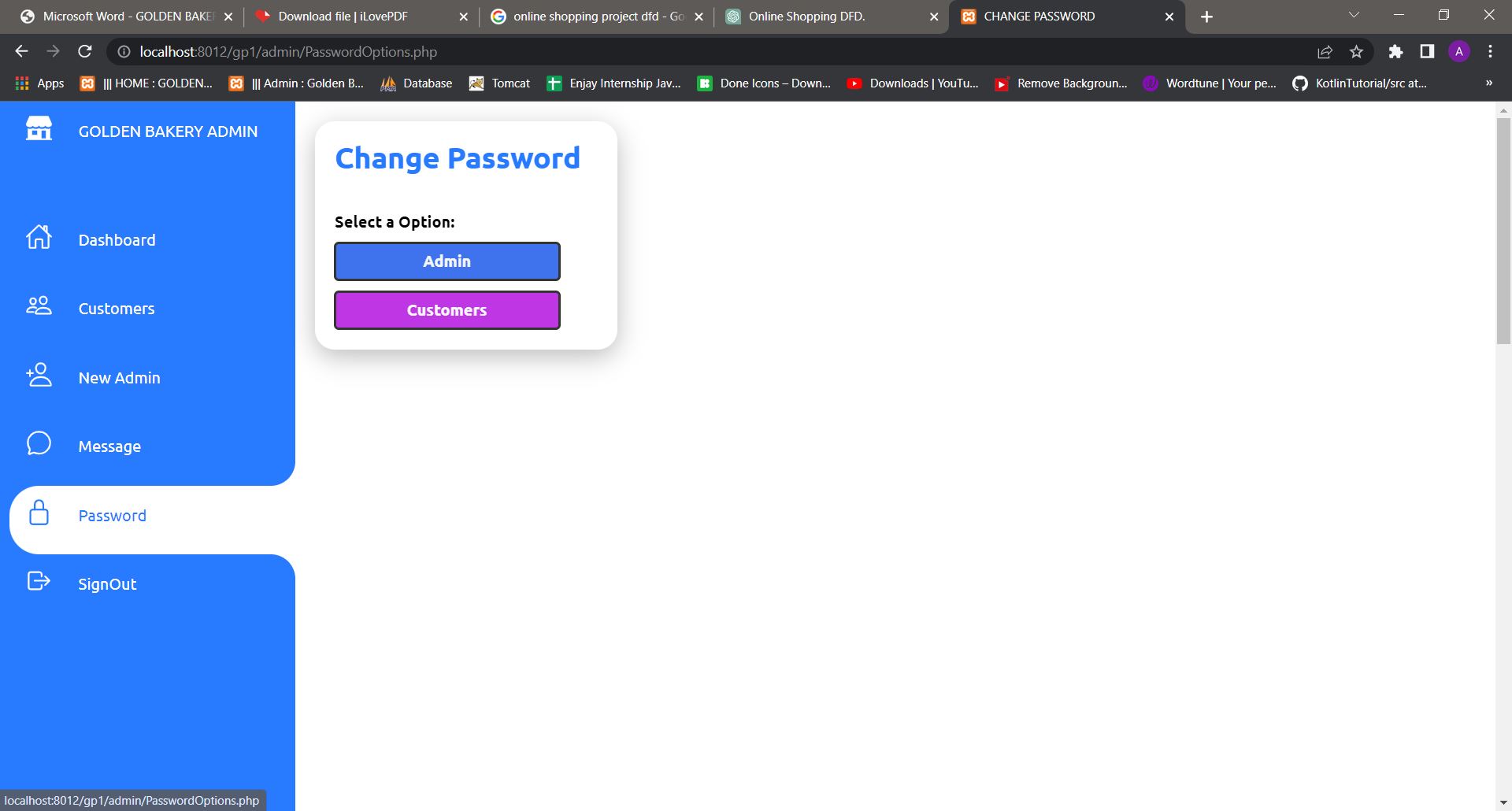
NEW ADMIN :



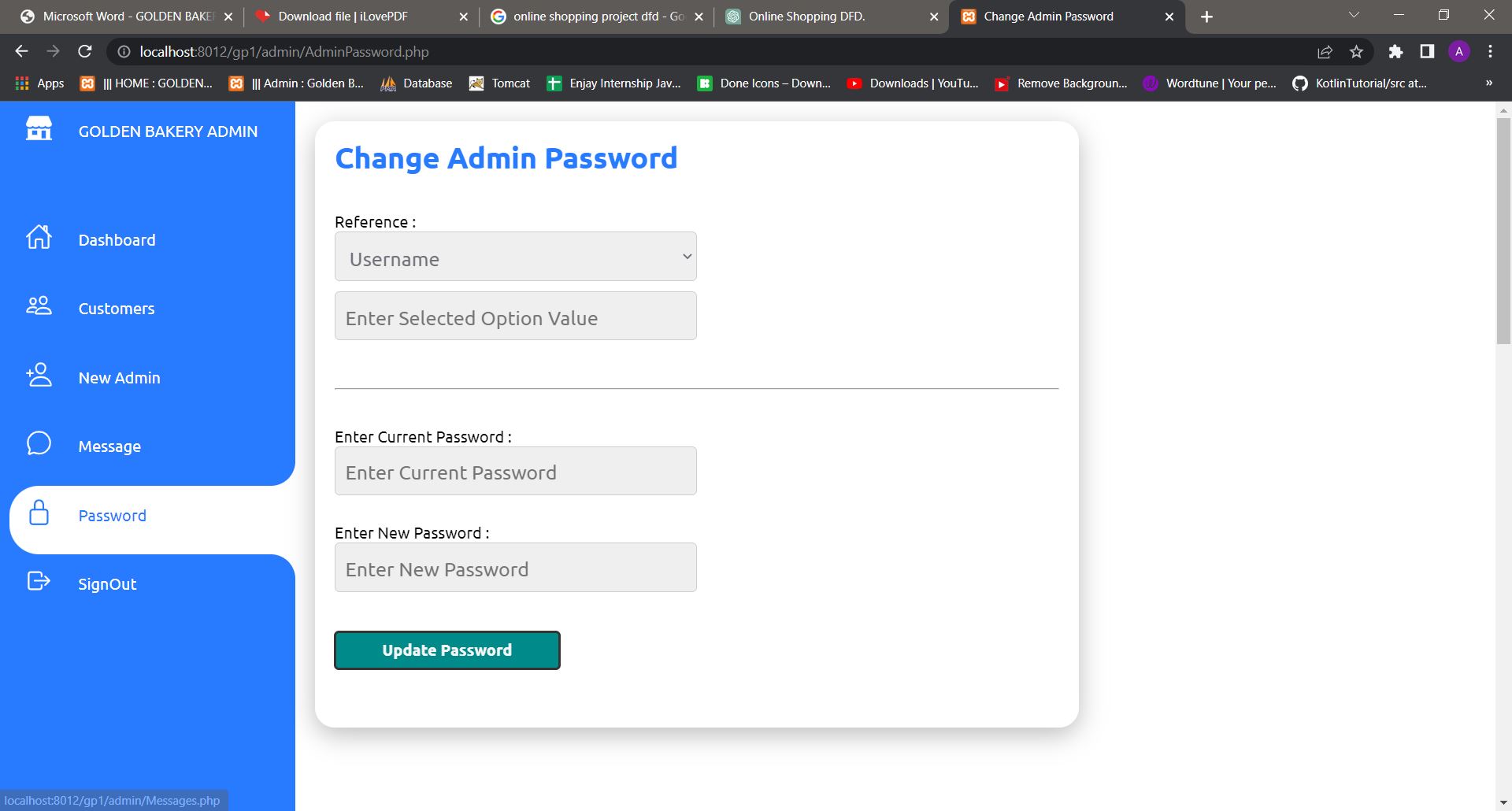
MESSAGES:



PASSWORD:



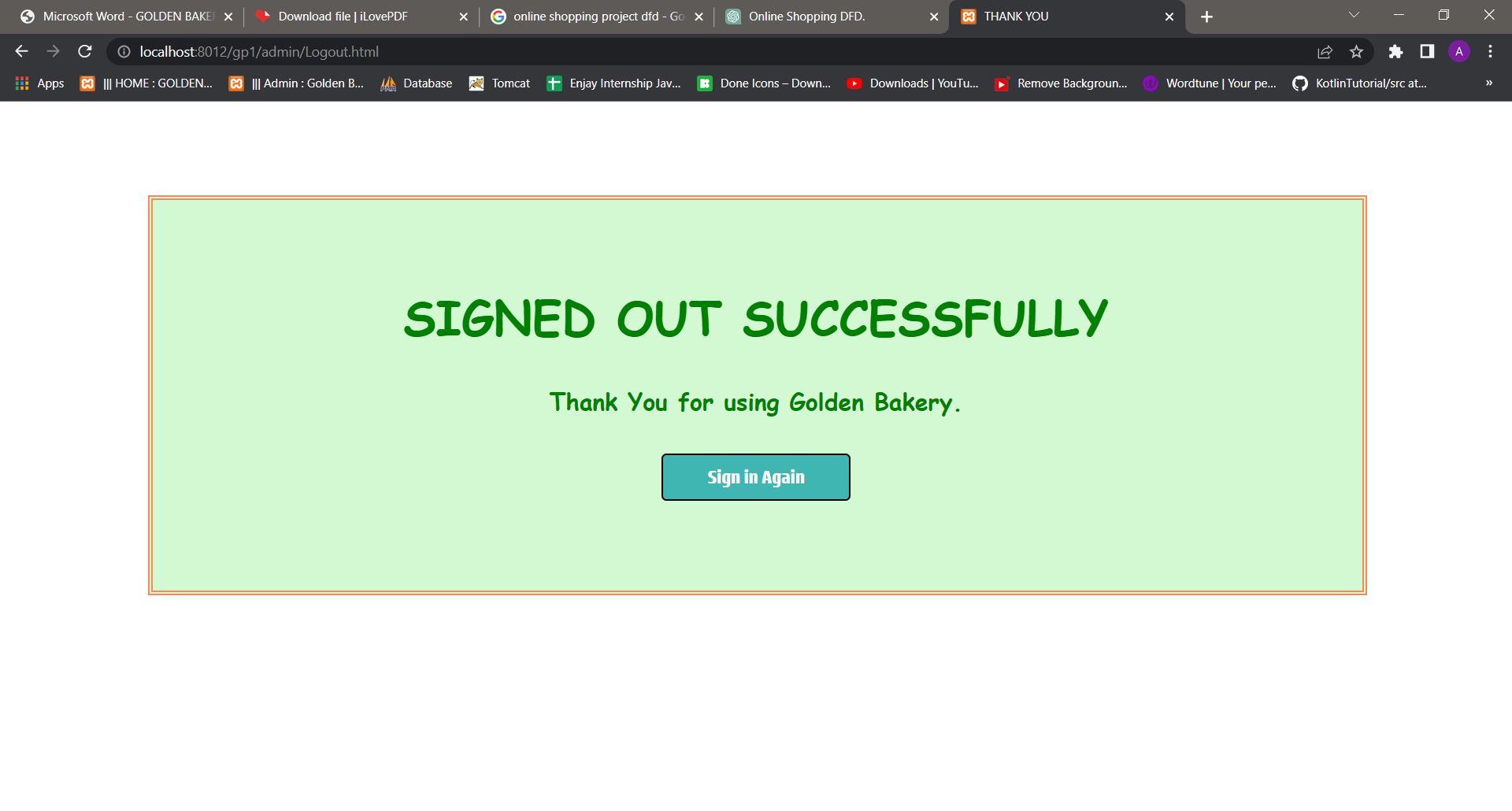
CHANGE ADMIN PASSWORD :



CHANGE CUSTOMER PASSWORD :

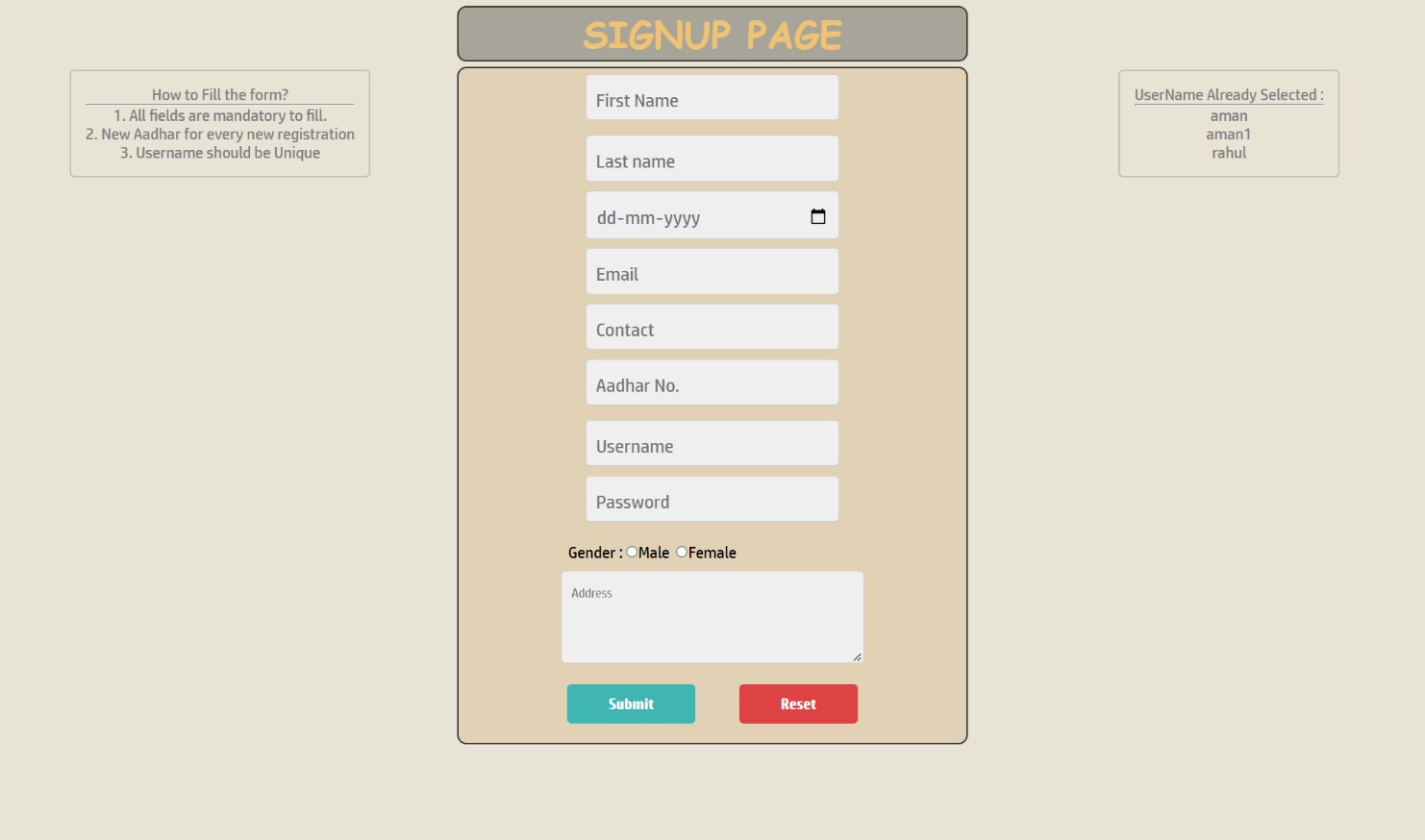


SIGNOUT:

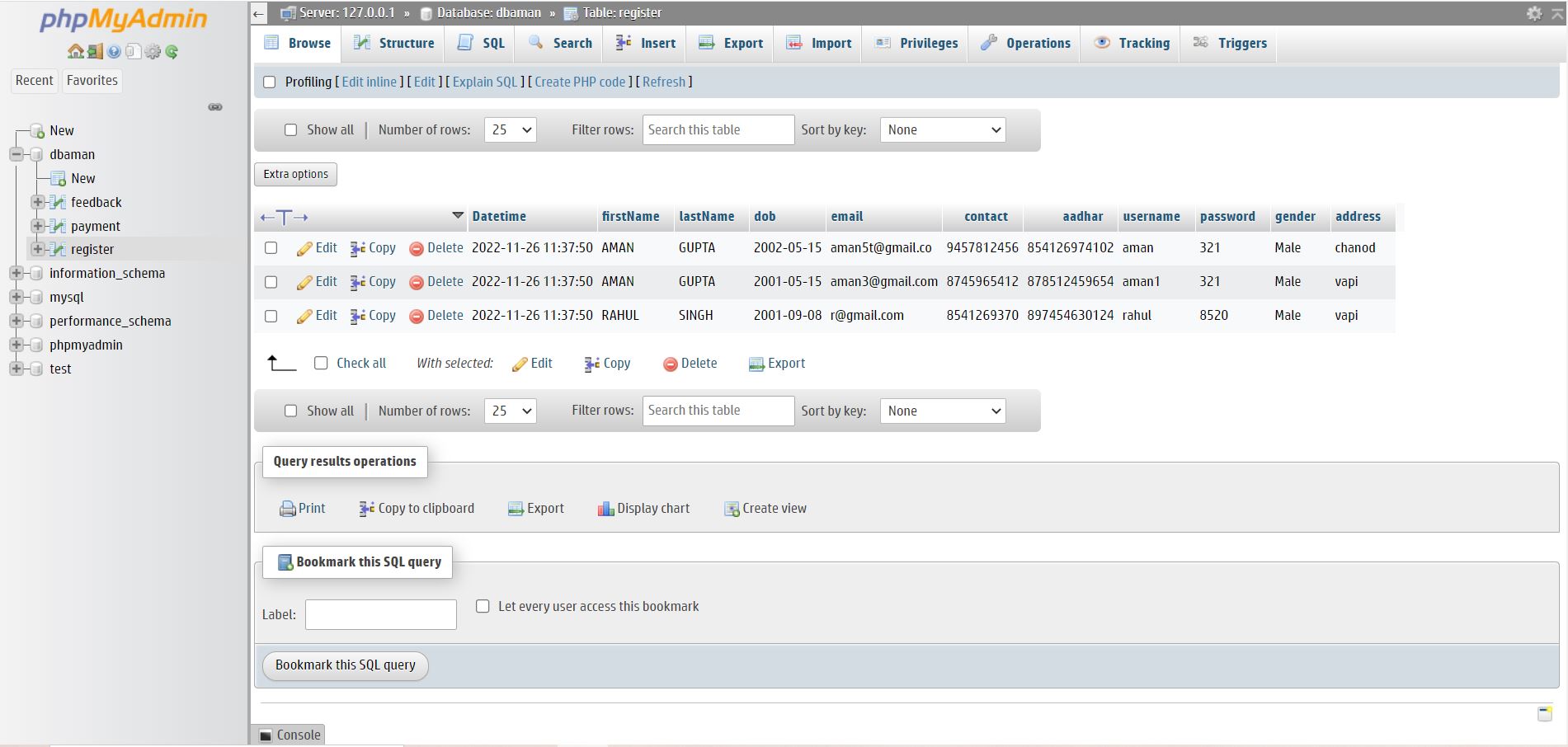


**7. REPORTS**

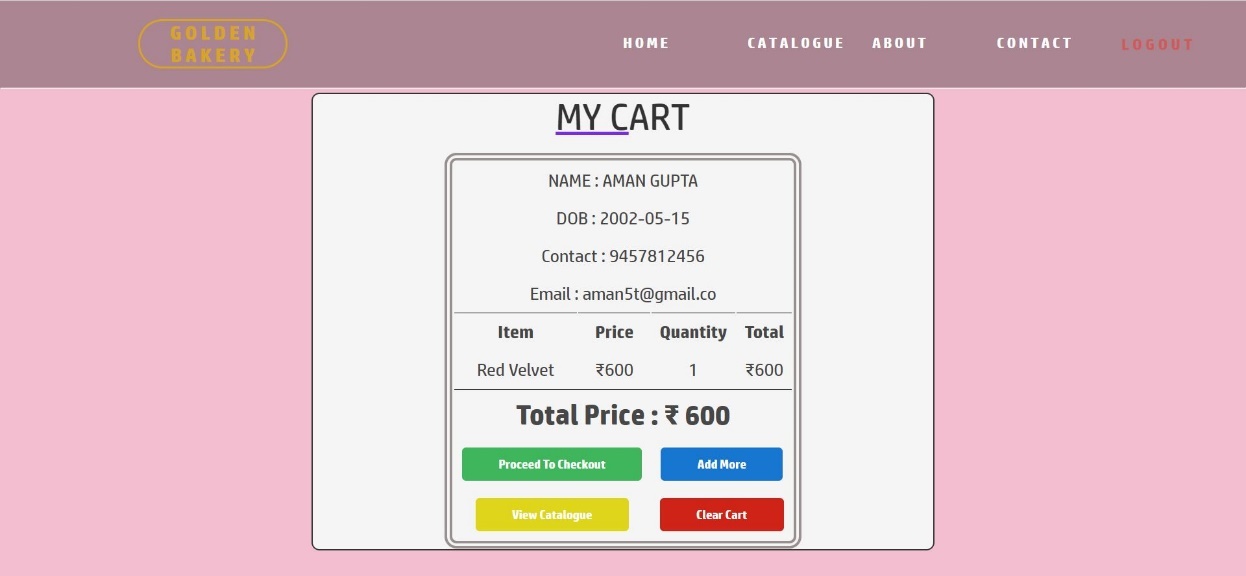
REGISTRATION PAGE:



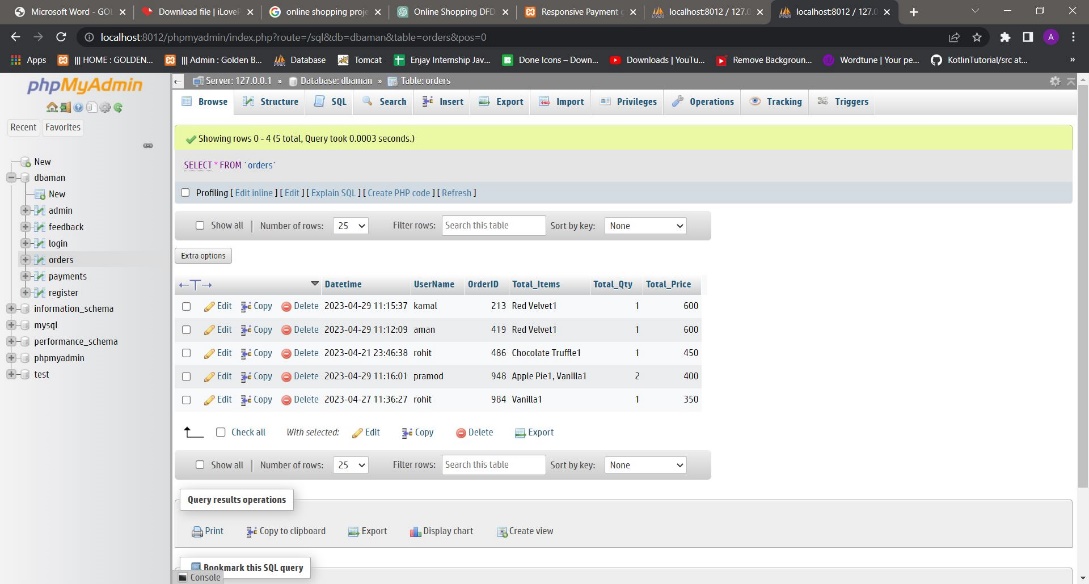
DATA REPORT:



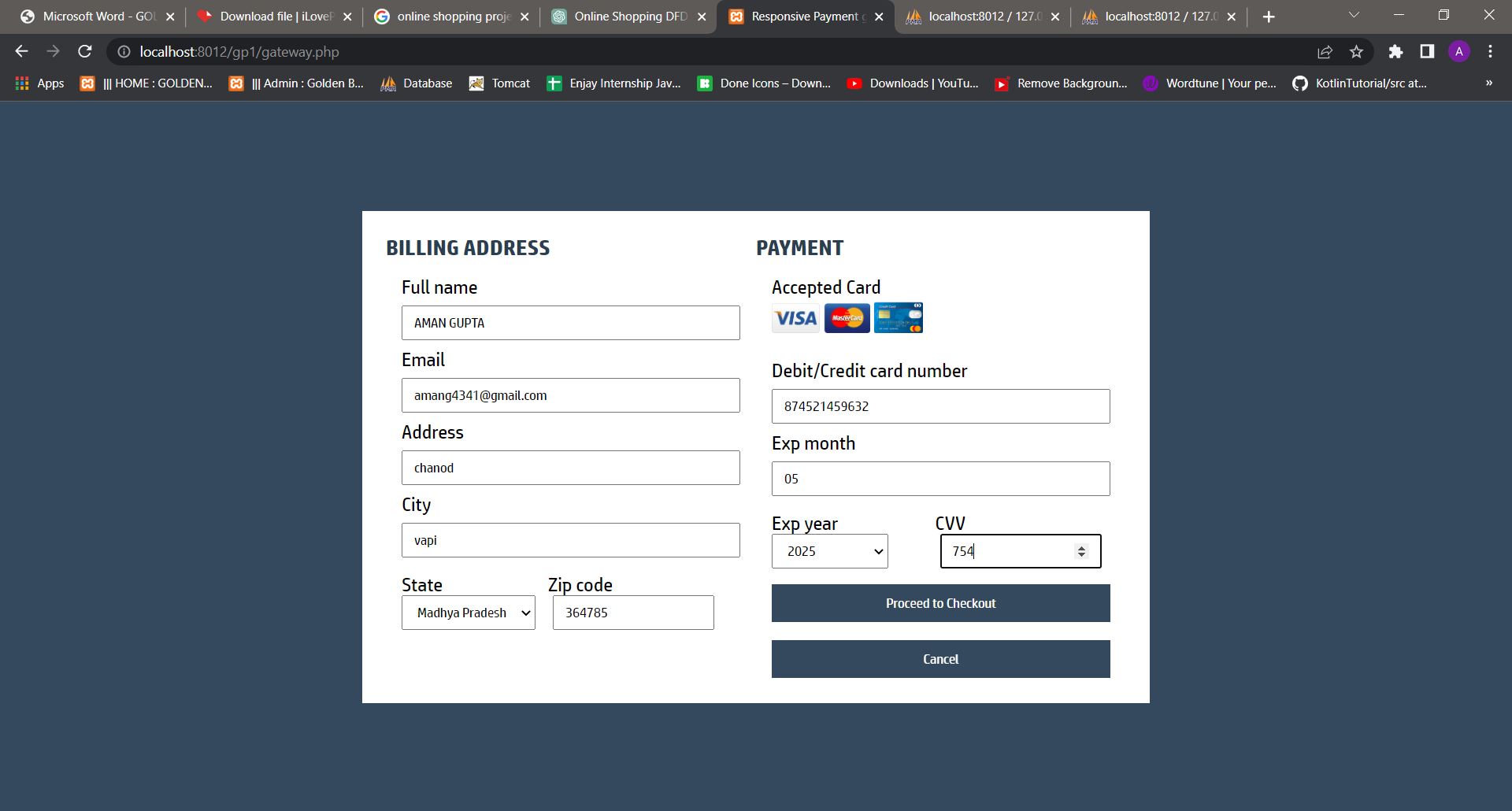
CART PAGE:



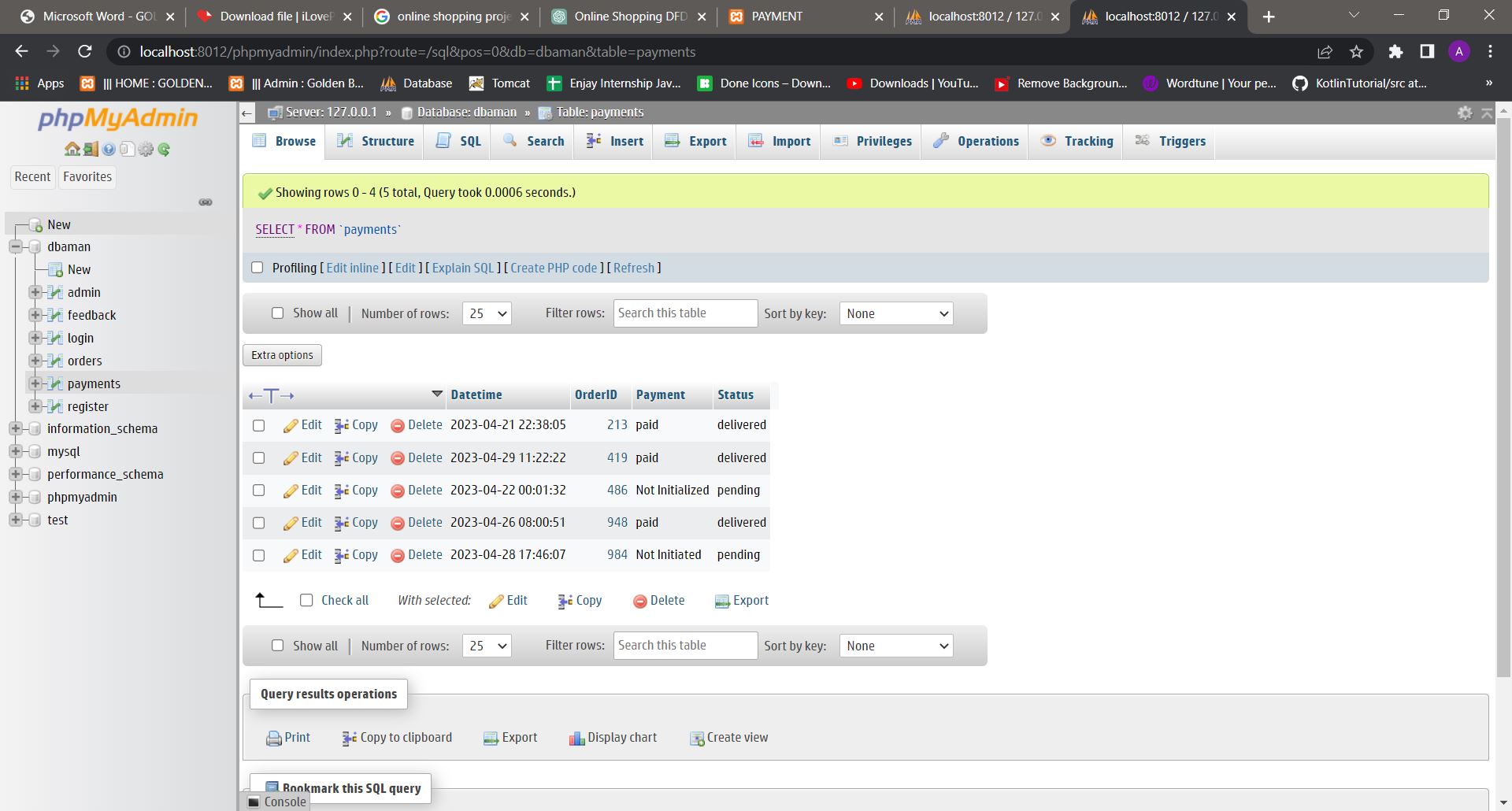
DATA REPORTS:



PAYMENT PAGE



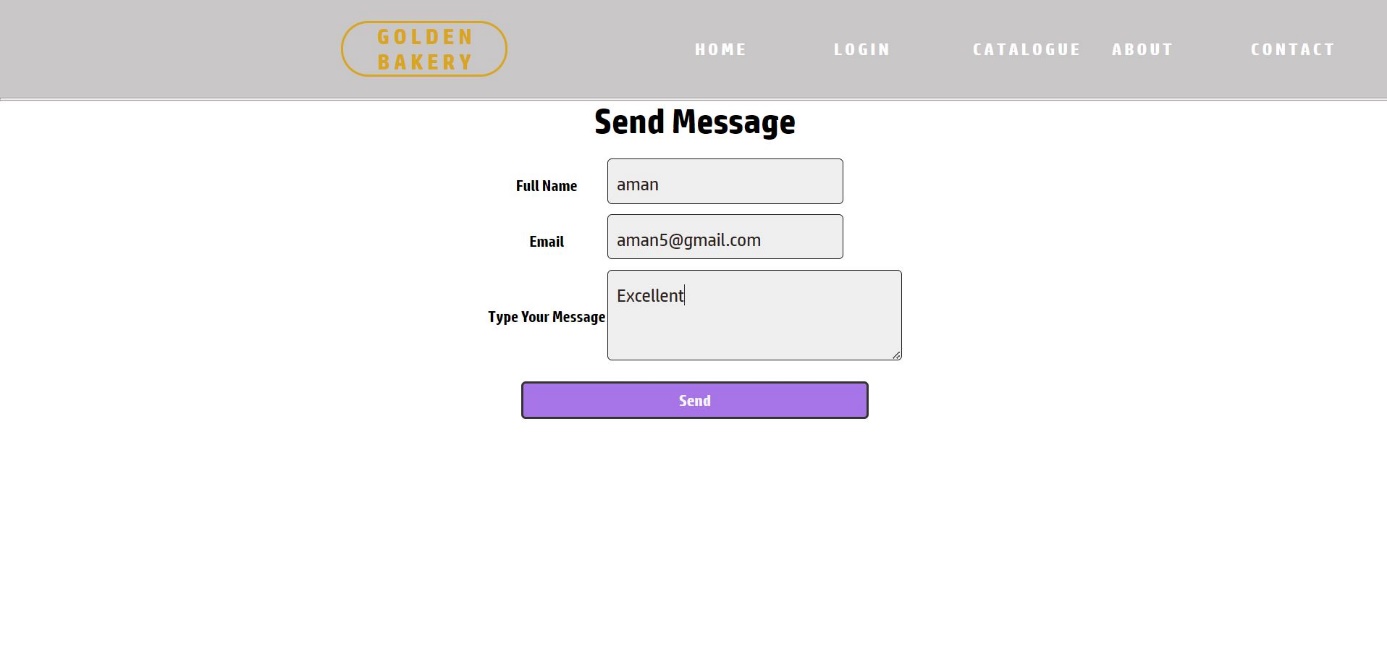
DATA REPORTS:



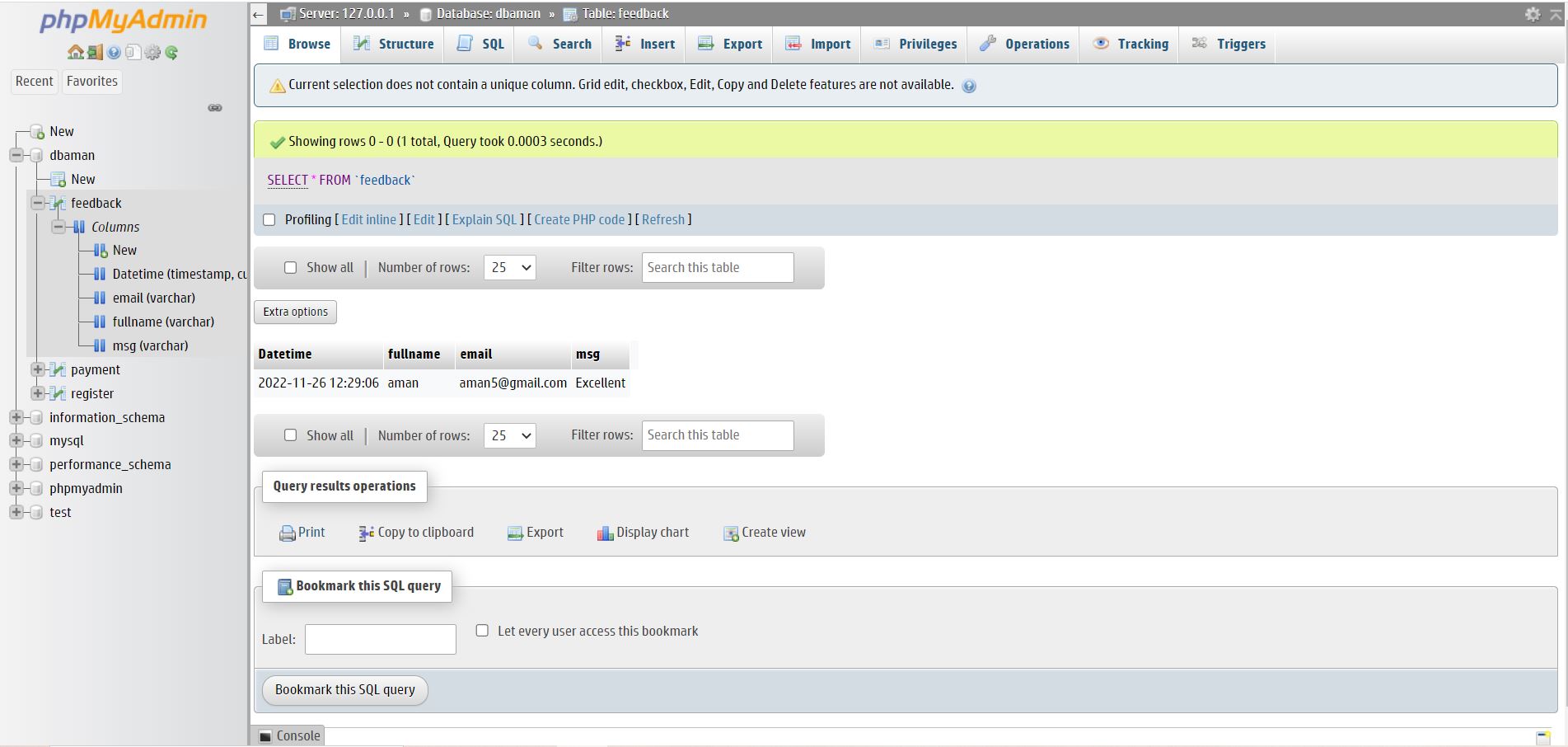
ORDER PAGE:



FEEDBACK:



DATA FEEDBACK:



1. **ADVANTAGES**

**AND**

**LIMITATIONS**

ADVANTAGES :

The system successfully meets the following requirements:

* Stores and maintains customer information.
* Forgotten passwords can be recovered through valid processes
* Easy calculation of selected items for billing.
* Can update user information.
* Shows the already selected username for new unique entry.
* Easy view of catalogue(bread, cake, pastries, cookies).
* Can add or remove the selected items.
* User could feeds us in feedback form.
* Easy to modify data of existing users through admin page (i.e. Add new users, update users and delete users).
* Even new admin could also be updated in the database by existing admin.
* Admin can see the users feedback on dedicated messages page.
* The admin dashboard is well designed which briefs each customer.

VALIDATION OF DATA:

* Invalid input is prompted using proper message boxes.
* Login details of a registered user is maintained in database each time when they logins.
* Password are stored in hash format.
* Incorrect username and passwords are not accessed.
* Forgotten password can be recovered through valid processes.

LIMITATIONS :

* Copyright violation.
* Bakery Catalogues can’t be updated through admin page
* Limited items in catalogue.
* It requires further development as per the human requirements, specially in the field of server side.

**9. FUTURE ENHANCEMENT**

The program is user-friendly. Proper used of well reserved words gives user ease of understanding.

Areas to work on under this project are:

1. We can add cookie functionality to prevent the need for logging in every time.
2. To enhance the smooth flow of the program and prevent program errors, a session should be added to prevent the user from repeatedly moving back and forth.
3. Enabling access ‘Catalogues’ to the catalog database and integrating it into the admin control.
4. To enhance the robustness and ease of use of the admin page, a delete button could be added to facilitate the deletion of users, messages, and catalogues.
5. The payment page could be enhanced by integrating real-world payment systems such as “instamoj”, in order to offer a more authentic payment experience.
6. Additional authentication methods could be implemented during the login process, such as one-time password (OTP) verification after entering the password, as well as OTP authentication when resetting a forgotten password, integrated into the system.

Above listed enhancements could be added in the future to enhance the capability of this program. The future holds a lot to offer to the development and refinement of this project. As proper documentation exists the whole system flow is traceable.

We may conclude that this Web program created will definitely find a good market in the bakery shops.

**10. BIBLIOGRAPHY**

**WEB REFERENCE :**

* [www.google.com](https://www.google.com)
* [www.youtube.com](https://www.youtube.com)
* [www.stackoverflow.com](https://www.stackoverflow.com)
* [www.javatpoint.com](https://www.javatpoint.com)
* [www.w3schools.com](https://www.w3schools.com)