ONLINE EXAMINATION SYSTEM

A

Major Project Report

Submitted

In partial fulfilment

For the award of the Degree of

## BACHELOR OF TECHNOLOGY

In Department of Computer science and Engineering

|  |  |  |
| --- | --- | --- |
| Submitted By: |  | Guided By: Submitted To: |
| Priyesh Soni  Rahul Badlani  Nitin Bagdi |  | Mr. Aaditya Maheshwari Mr. Gaurav Kumawat  (Project Incharge) (Head, Deptt. of I.T.) |



Department of Computer Science and Engineering

TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY

RAJASTHAN TECHNICAL UNIVERSITY



### CERTIFICATE

This is to certify that this project report “Retail Shopping Mart System” is the confide work of “Priyesh Soni, Rahul Badlani, Nitin Bagdi” who have carried out the project work under my supervision. I approve this project for submission of the Bachelor of Technology in the Department of Computer Science and Engineering, Techno India NJR Institute of Technology, affiliated to Rajasthan Technical University, Kota.

Mr. Gaurav Kumawat Mr. Pankaj Chittora

Assistant Professor & Head Assistant professor, Project In charge

Department of Computer Science Department of Computer Science

# ABSTRACT

1. Purpose
   1. Introduction

This Desktop application is basically a inhance about stock management of a retail shop. It handels all the basic needs required by retailer to manage stock and generate bills.

The main objective of this handheld application is to provide simplicity to the shop owner about inventory and data management very efficiently and effectively.

* 1. Scope

Talking about scope of this application. Shortlisting important:Few of them are:-

* + - This application provides the benefit to software holder for keeping record of all the products.

Further more whenever the user wants to handle the store to someone else it would be very easy for new member to keep record and understand the system.

1. Document overview

The remainder of this document is 7 chapters, the first providing introduction of the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

# ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide Mr. Aaditya Maheshwari(Project In charge), TINJRIT for their valuable guidance, encouragement and help for completing this work. Their useful suggestions for this whole work and co-operative behavior are sincerely acknowledged.

I would like to express my sincere thanks to Mr. Gaurav Kumawat, Head of Department, Dept. of CSE TINJRIT for giving me this opportunity to undertake this project.

I also wish to express my indebtedness to my parents as well as my family member whose blessings and support always helped me to face the challenges ahead.

At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Place: Udaipur  Date: |  |  |  |  | Priyesh Soni  Rahul Badlani  Nitin Bagdi |

# TABLE OF CONTENTS

Table of Contents

[Abstract iv](#_Toc65284)

[Acknowledgement v](#_Toc65285)

[Table of Contents vi](#_Toc65286)

CHAPTER – 1 Introduction .............................................................................................................. 1-5

Introduction ............................................................................................................................................. 2

1.1. Purpose ............................................................................................................................................. 2

1.2. Overall Description ........................................................................................................................... 4

CHAPTER – 2 Software Requirement Specification ...................................................................... 6-20

2.1. Purpose............................................................................................................................................. 7

2.2. Overall Description ........................................................................................................................... 9

2.2.1. Functional Requirements definitions .............................................................................................. 9

2.2.2. Use Cases ...................................................................................................................................... 9

2.2.3. Non – Functional Requirements ................................................................................................... 14

2.3. Requirement Specification .............................................................................................................. 15

2.3.1. External interface specifications ................................................................................................... 15

2.3.2. Functional Requirements ............................................................................................................. 15

2.4. Hardware Specification ................................................................................................................... 19

2.5. Software Specification .................................................................................................................... 19

2.6. Hardware and Software Requirements in detail ............................................................................... 20

CHAPTER – 3 System Analysis and Design .................................................................................. 21-39

3.1. Study & Weaknesses of current system ........................................................................................... 22

3.2 Requirements of new system ............................................................................................................ 23

3.3. Feasibility Study ............................................................................................................................. 23

3.4. Features of new System .................................................................................................................. 24

3.5. Data flow diagram .......................................................................................................................... 25

3.6. UML Modeling............................................................................................................................... 34

3.6.1. Activity Diagram ......................................................................................................................... 34

3.6.2. Class Diagram ............................................................................................................................. 35

3.6.3. Context Diagram .......................................................................................................................... 36

3.6.4. Sequence Diagram ....................................................................................................................... 37

CHAPTER - 4 Screen Shots .......................................................................................................... 46-61

5.1. Admin Master ................................................................................................................................. 47

5.2. User Master .................................................................................................................................... 54

CHAPTER – 5 Testing ................................................................................................................... 62-67

CHAPTER – 6 Conclusion and Future Enhancements ................................................................. 68-71

7.1. Limitations ..................................................................................................................................... 69

7.2. Future Enhancements ...................................................................................................................... 70

7.3. Conclusion ..................................................................................................................................... 71

CHAPTER – 8 Bibliography .............................................................................................................. 72

CHAPTER – I

INTRODUCTION

Introduction

1.1. Purpose

### 1.1.1. Introduction

This Desktop application is basically a inhance about stock management of a retail shop. It handels all the basic needs required by retailer to manage stock and generate bills.

The main objective of this handheld application is to provide simplicity to the shop owner about inventory and data management very efficiently and effectively.

### 1.1.2. Scope

Talking about scope of this application. Shortlisting important:Few of them are:-

* This application provides the benefit to software holder for keeping record of all the products.
* Further more whenever the user wants to handle the store to someone else it would be very easy for new member to keep record and understand the system.

### 1.1.3. Document overview

The remainder of this document is 8 chapters, the first providing introduction of the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

This document is meant for describing all the features and procedures that were followed while developing the system.

This document specially mentions the details of the project how it was developed, the primary requirement, as well as various features and functionalities of the project and the procedures followed in achieving these objectives.

Retail Shopping Mart Application depicts the cycle for the owner about inventory management, employee management and as well sell management. It is very essential for a shop owner to handle the repository of stock and their sale. It is very useful for keeping in record all the goods and that too with distributed system in categories of product. This desktop application generated using tkinter helps in generating bill, searching bill. With respect to employee management, through this application admin can add, delete employee.

With the effective use, any Shop Owner can use the “Retail Shopping Mart Application” for generating bills and managing inventory.

1.2. Overall description

Retail Shopping Mart System is designed for Shopkeepers. The application is basically a desktop application developed over python which helps the shop manager to generate the bill and keep record of the products. This application sends the bill to customer over email, so as to avoid keeping hard copies which gets misplaced many a times. The applications works on principal to simplify keeping record of customer.

### 1.2.1. Functional requirements definitions

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Non- Functional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

### 1.2.2. Use cases

This system will be used in Two User Modules which are Administrator and Employee. As all of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of all two User Modules have been described below.

1. User can do the following functions in the Administrator Module
   * Inventory Management:
2. Add Product
3. Search Product
4. Delete Data
5. Update Product
6. Display all products
7. Logout
   * Employee Management
8. Search Employee
9. Add Employee
10. Update Employee
11. Delete Employee
12. Logout
    * Invoice
13. Display Data
14. Delete Invoice
15. Search Invoice
16. Logout
    * Open Bill
17. User can do the following functions in the Employee Module
    * Search Bill
    * Generate Bill
    * Calculate Amount
      1. User characteristics

The user should be familiar with the Product related information like Category and subcategory Product is in.

The user should be familiar with handling of software.

The user should know bill number before looking for a bill.

* + 1. Constraints

The bill Generated would be saved as a CSV file.

The new upcoming bill will replace the CSV file holding the previous record.

The Inventory is based on local system database, not over cloud.

CHAPTER – II

## SOFTWARE REQUIREMENT SPECIFICATION

Software Requirement Specification

2.1. Purpose

### 2.1.1. Introduction

This Desktop application is basically a inhance about stock management of a retail shop. It handels all the basic needs required by retailer to manage stock and generate bills.

The main objective of this handheld application is to provide simplicity to the shop owner about inventory and data management very efficiently and effectively.

### 2.1.2. Scope

Talking about scope of this application. Shortlisting important:Few of them are:-

* This application provides the benefit to software holder for keeping record of all the products.

* Further more whenever the user wants to handle the store to someone else it would be very easy for new member to keep record and understand the system.

### 2.1.3. Glossary

Table 2.1

|  |  |
| --- | --- |
| Term | Definition |
| Admin | The only user who can manage inventory and employee details, etc. |
| Employee | Employees working under admin |
| Inventory | Record of products |
| Invoice | Bill generated by employees |
| Database | Storage of data. |
| Tkinter | Python Library |

### 2.1.5. Document overview

The remainder of this document is two chapters, the first providing a full description of the project for the owners of the Online Examination System. It lists all the functions performed by the system. The final chapter concerns details of each of the system functions and actions in full for the software developers’ assistance. These two sections are cross-referenced by topic; to increase understanding by both groups involved.

2.2. Overall description

Retail Shopping Mart Application depicts the cycle for the owner about inventory management, employee management and as well sell management. It is very essential for a shop owner to handle the repository of stock and their sale. It is very useful for keeping in record all the goods and that too with distributed system in categories of product. This desktop application generated using tkinter helps in generating bill, searching bill. With respect to employee management, through this application admin can add, delete employee.

This application uses database to save records such as:

* Inventory Management
* Employee Details
* Invoice Details

### 2.2.1. Functional requirements definitions

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

### 2.2.2. Use cases

This system will be used in two User Modules which are Administrator and Employee. As all of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of all two User Modules have been described below.

* User can do the following functions in the Administrator Module
  + Inventory Management

Add Product

Search Product

Delete Data

Update Product

Display all products

Logout

* + Employee Management

Search Employee

Add Employee

Update Employee

Delete Employee

Logout

* + Invoice

Display Data

Delete Invoice

Search Invoice

Logout

* User can do the following functions in the Employee Module
  + Search Bill
  + Generate Bill
  + Calculate Amount
  + Save Bill

2.2.2a. Use Case: User Login or Signup

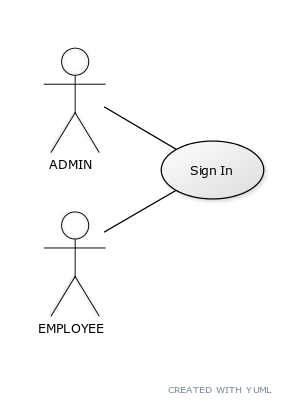


Fig. 2.1 LOGIN ADMIN/EMPLOYEE

Brief Description:

Either the admin or employee logins.

Initial step-by-step description:

For this use case to be initiated the user must be on the Login Page.

1. For employee to login.
2. Enter Username
3. Enter Password
4. Username and Password matches database, login success.
5. For Admin to login.
6. Enter Username
7. Enter Password
8. Username and Password matches database
9. Designation matches to “ADMIN”.
10. Login Success.

2.2.2b. Use Case: AFTER ADMIN LOGINS SUCCESSFULLY.

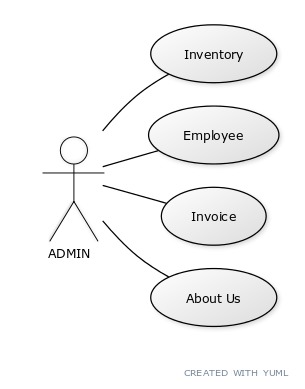


Fig. 2.2 Admin Inventory/Employee/Invoice/About Us:

The Admin chooses Inventory/Employee/Invoice/About Us.

Initial step-by-step description:

For this use case to be initiated, the admin must login successful and should be designated at ADMIN.

* The Admin selects” Inventory/Employee/Invoice/About Us”.
* The application returns the corresponding page.
* In Inventory, admin can add product , update product, delete product and search product.
* In Employee, admin can add employee , update employee, delete employee and search employee.
* In Invoice, admin can search and delete invoice.
* The system adds the data to the Database.
* Admin can either logout at end or can redirect to other mode.

2.2.2c. Use Case: AFTER EMPLOYEE LOGINS SUCCESSFULLY.

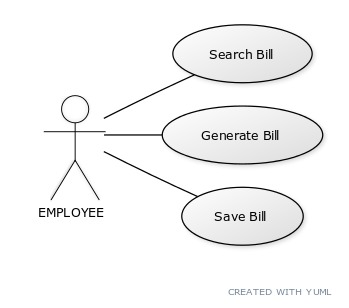


Fig. 2.3 Employee can search bill, generate Bill, Save Bill:

Initial step-by-step description:

For this use case to be initiated, the employee must login successful.

* The Employee selects” search bill, generate Bill, Save Bill”.
* In Search, employee search the invoice/bill.
* In generate, employee select product and generate bill.
* In save, employee saves the bill over local csv file.

### 2.2.3. Non-functional requirements

There are requirements that are not functional in nature. Specifically, these are the constraints the system must work within.

The system must have SQL so as to save data.

2.3. Requirement specifications

2.3.1. External interface specifications

None

### 2.3.2. Functional Requirements

Table 2.1 Access Home Page:

|  |  |
| --- | --- |
| Use Case Name: | Access Home Page |
| Priority | Essential |
| Trigger | Menu selection |
| Precondition | User is on the home page. |
| Basic Path | 1. The user must run application. |
| Alternate Path | N/A |
| Postcondition | The User is on the Home Page |
| Other |  |

Table 2.2ADMIN Login:

|  |  |
| --- | --- |
| Use Case Name: | ADMIN Login |
| Priority | Essential |
| Trigger | Selects |
| Precondition | The User is on the Home Page |
| Basic Path | 1. The User selects the “ADMIN” button. 2. User gets directed to login page. 3. User has to enter username and password. 4. User has to click on login button. 5. If designation matches to “ADMIN”, login succeed. 6. The user gets directed to ADMIN portal. |
| Postcondition | The user is on the ADMIN page. |
| Other |  |

Table 2.3 EMPLOYEE Login:

|  |  |
| --- | --- |
| Use Case Name: | EMPLOYEE Login |
| Priority | Essential |
| Trigger | Selects |
| Precondition | The User is on the Home Page |
| Basic Path | The User selects the “EMPLOYEE” button.  User gets directed to login page.  User has to enter username and password.  User has to click on login button.  If credentials matches, login succeed.  The user gets directed to EMPLOYEE portal. |
| Postcondition | The user is on the ADMIN page. |
| Other |  |

2.4. Hardware Specification

Client Side:

* Processor: intel.
* RAM : 512 MB
* Hard Disk : 20GB

2.5. Software Specification

Client Side:

* Jupyter Notebook
* Pycharm
* Windows XP/Vista/Windows 7

Data Base Server:

* DB Browser(SQLite)

2.6. Hardware and Software Requirements in detail

Hardware Requirements:

* Processor: intel.
* RAM : 512 MB
* Hard Disk : 20GB
* Disk space : 2GB

CHAPTER – III

## SYSTEM ANALYSIS AND DESIGN

System Analysis and Design

3.1. Features of the RETAIL SHOPPING MART System.

The new system has been designed as per the user requirements so as to fulfil almost all them.

* + Quick Invoice Generation

Invoice generation speeds up in this desktop application. This application provides simplification of products by dividing them into categories and further sub category.

The items kept in store are already added to the database making the management go fast. The inventory handheld by the owner/Admin. The inventory also tells the admin the Cost Price and Selling Price to calculate the income generated.

* + Employee Management

The employee details are added in the database with all the necessary details such as Aadhar Number and phone number. The employees are too managed by the admin.

It is a process that helps your workers to perform their best and to achieve our business goals. This is most important in MART application.

* + Easy to Store and Retrieve Information

The new system makes it easy to store and retrieve information as required and does not involve storing information in separate sheets or papers. It thus saves data management problems of keeping record of stock, employee and etc.

* + Cost Effective

One of the main reasons of the new system is its cost effectiveness. It saves the amount spend on stationary as well as overall cost of conducting an examination which also involves paying supervisors, paper checkers, question paper printers etc. It doesn’t require internet connection too.

3.5. Data Flow Diagram (DFD)

The DFD (also known as bubble chart) is a simple graphical formalism that can be used to represent a system in terms of the input data into the system, various processes carried on these data, and the output data generated by the system.

The main reason why the DFD technique is so popular is because the fact that the DFD is a very simple formalism – it is simple to understand and use. A DFD model uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among the functions. Starting with a set of high-level functions that a system performs, a DFD model hierarchy represents various sub-functions.

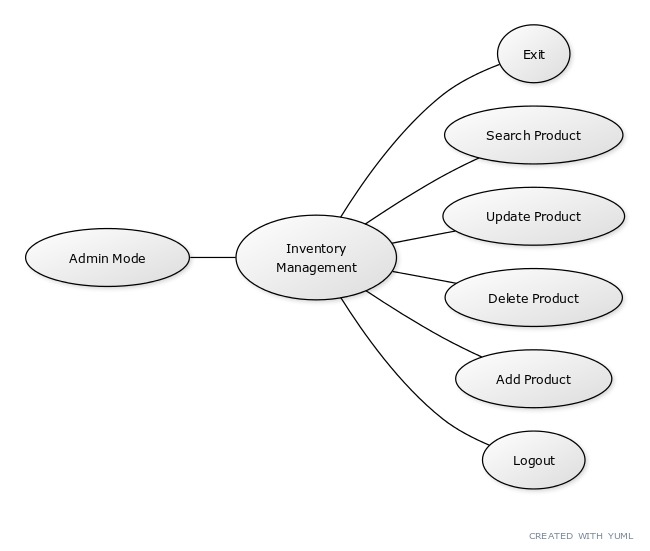


Fig. 3.1 Inventory Management in Admin Portal

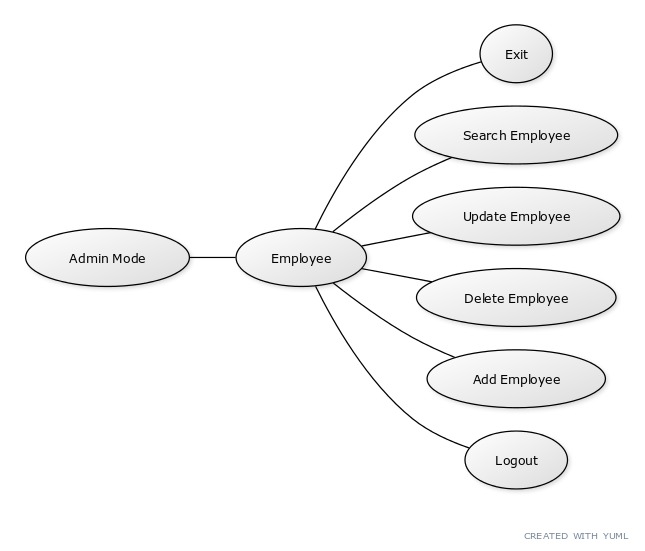


Fig. 3.2 Employee Management in Admin Portal

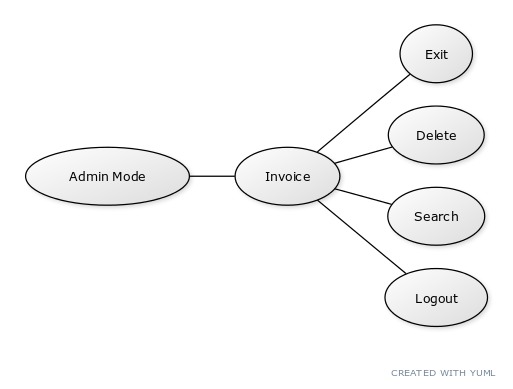


Fig. 3.3Invoice Management

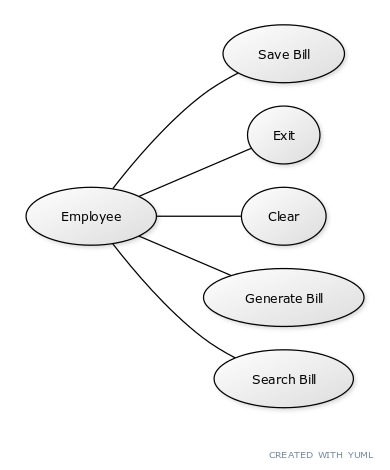
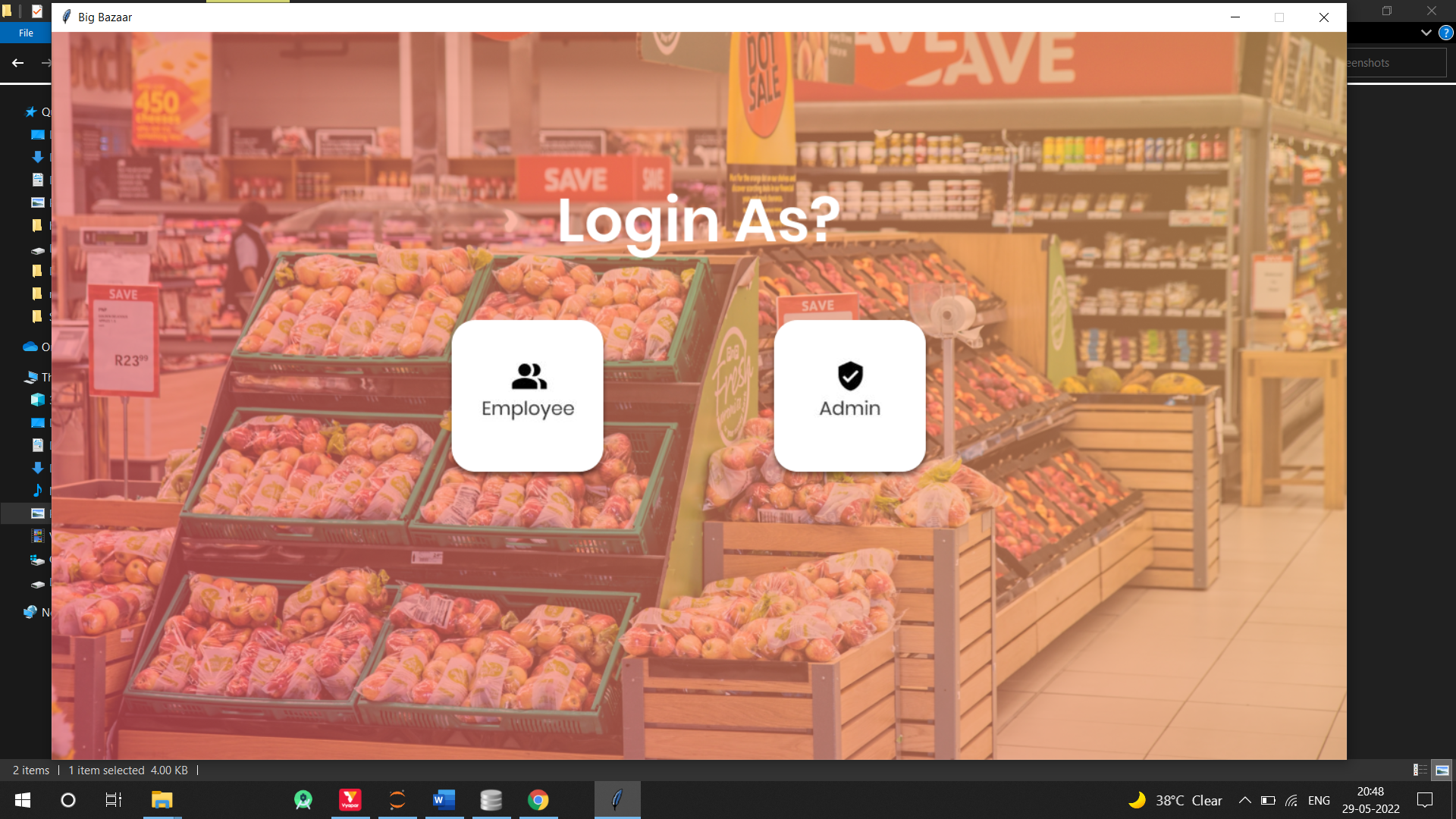
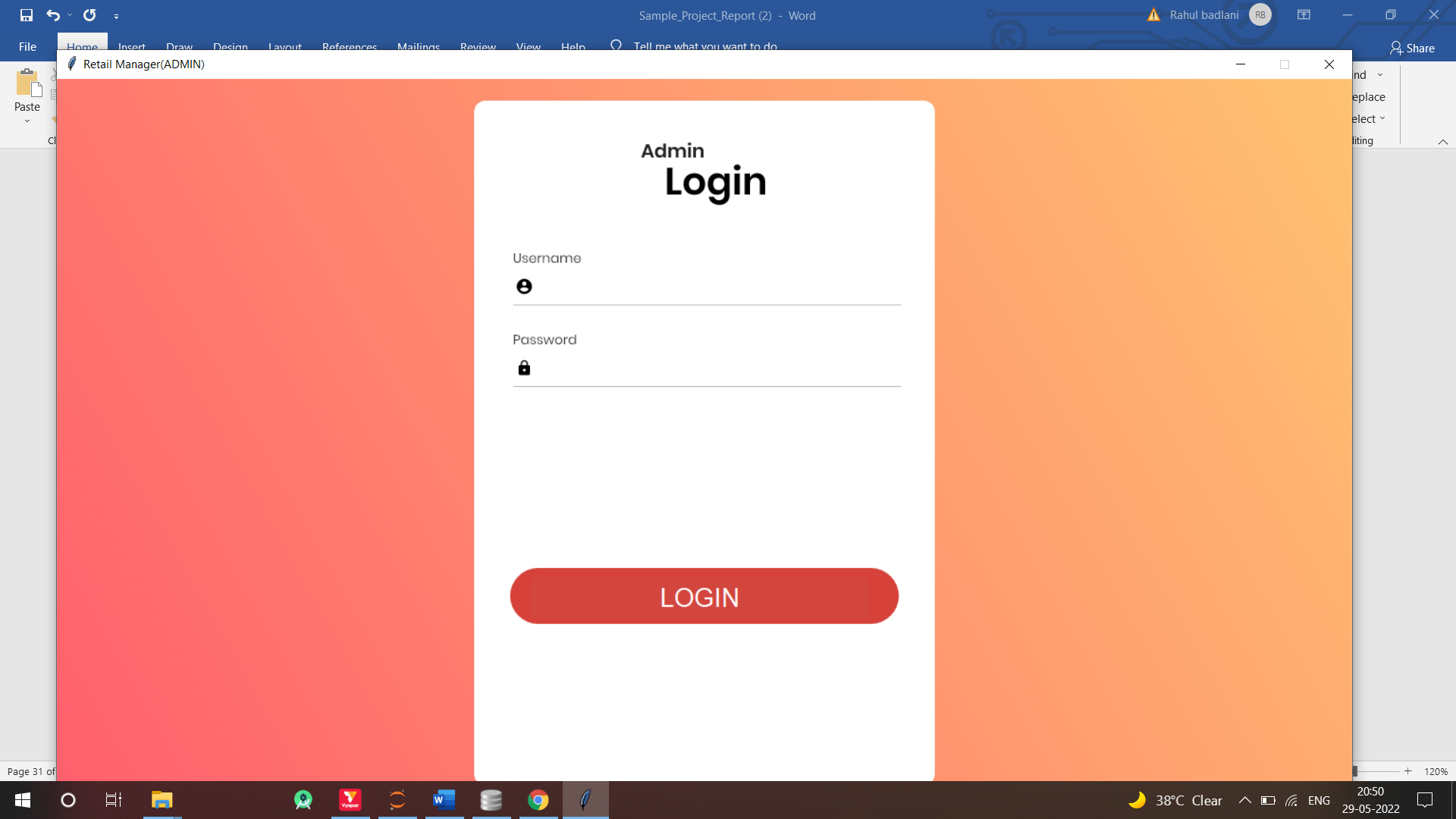
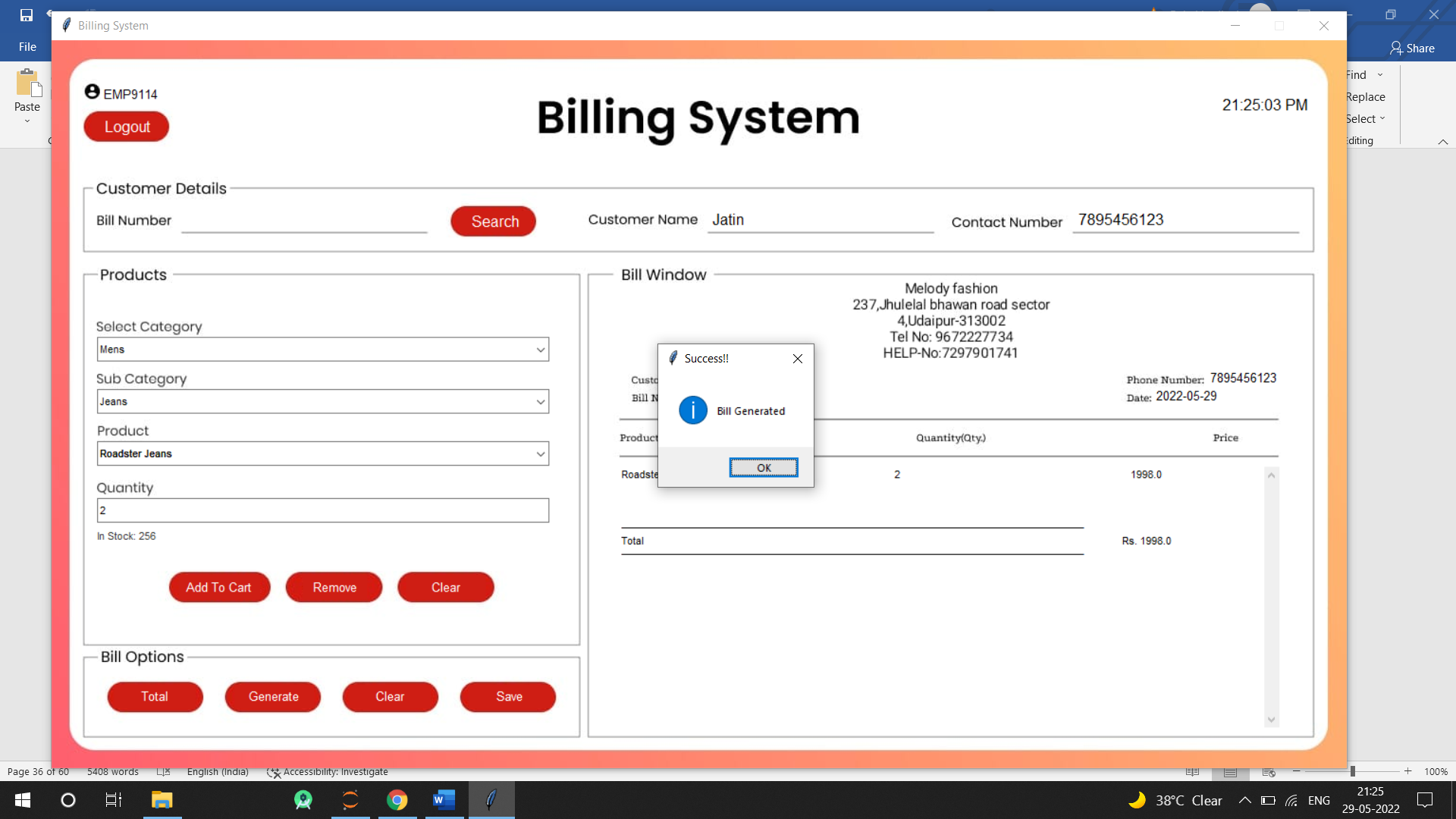
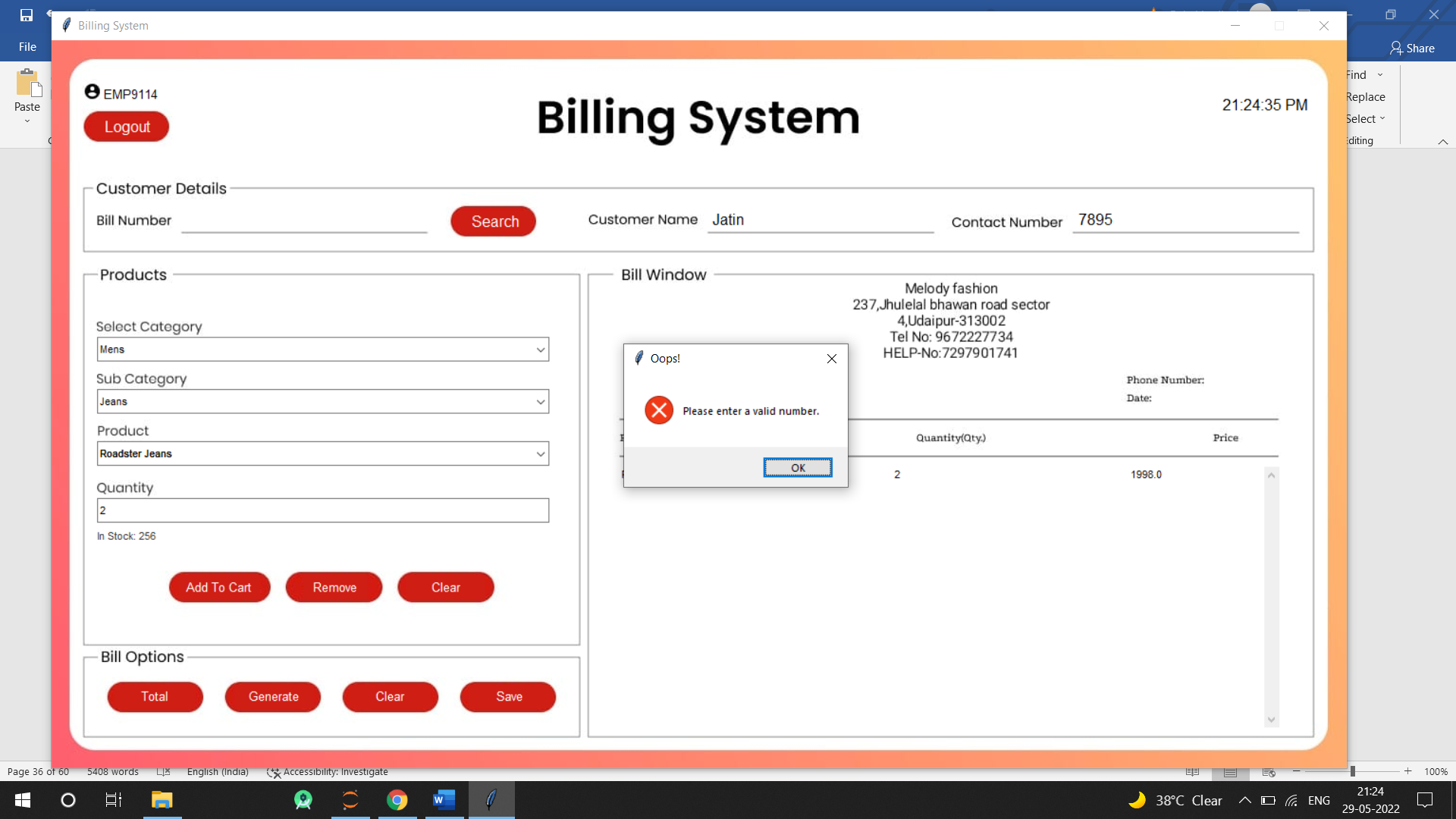
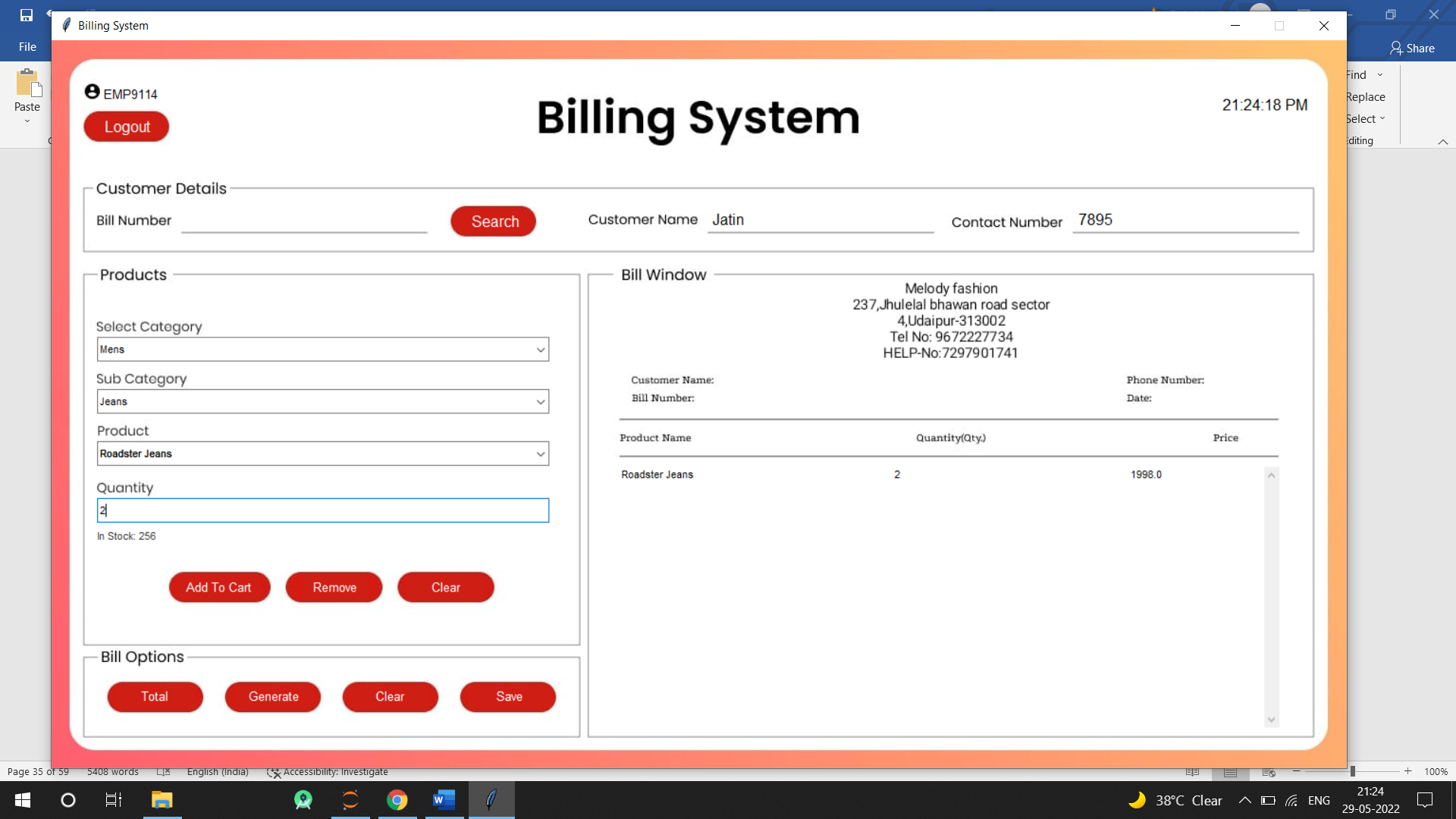
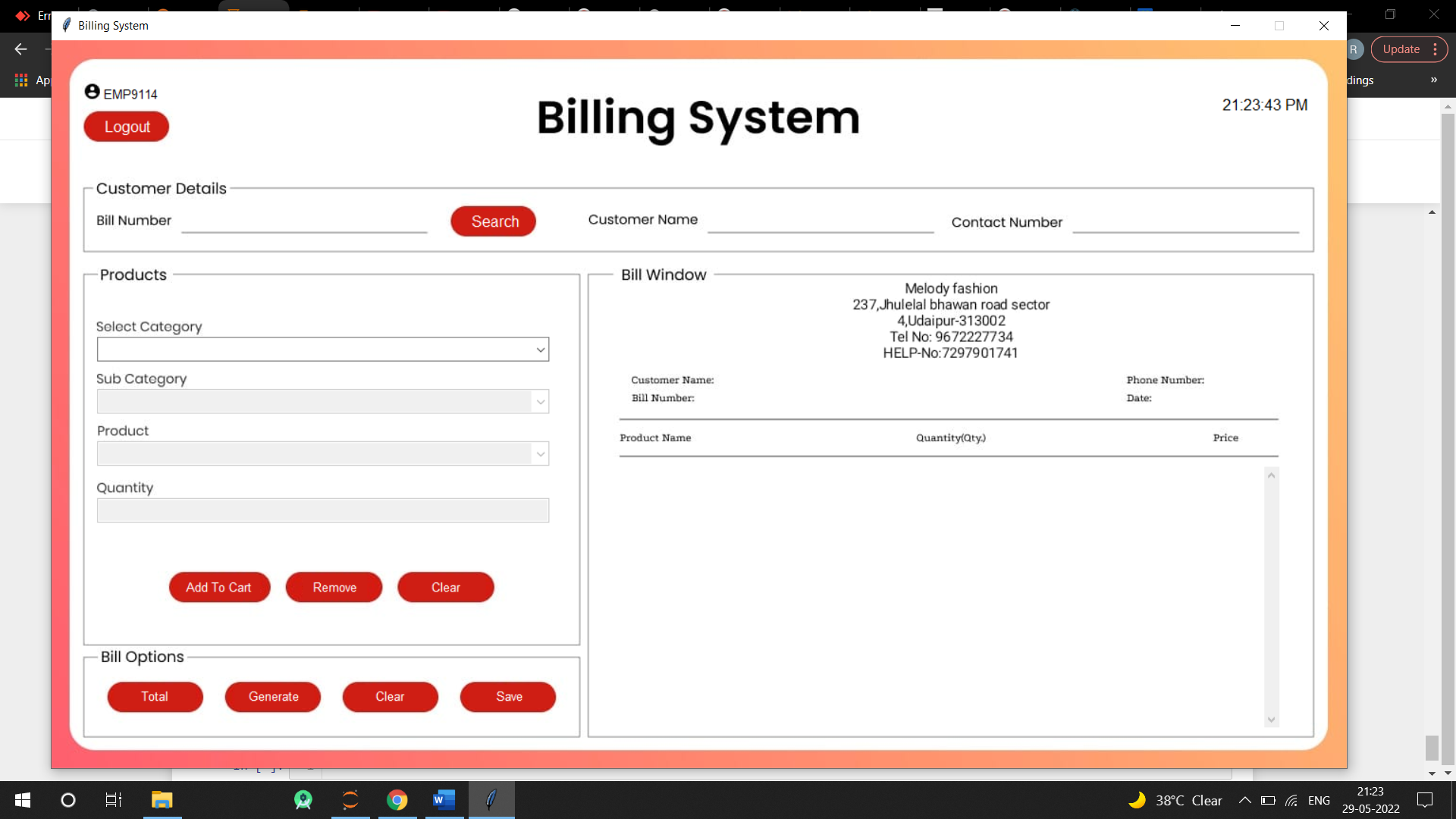
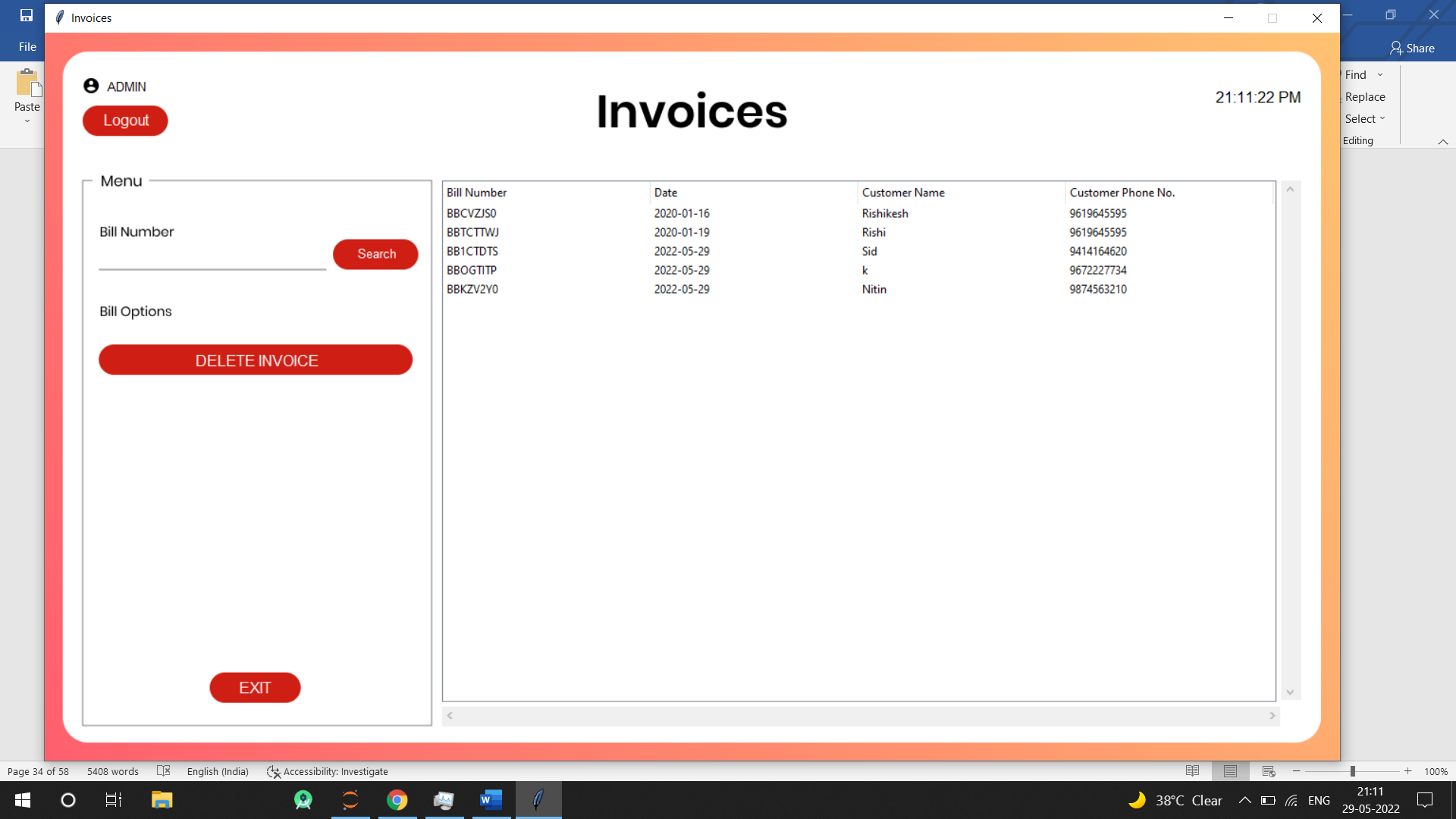
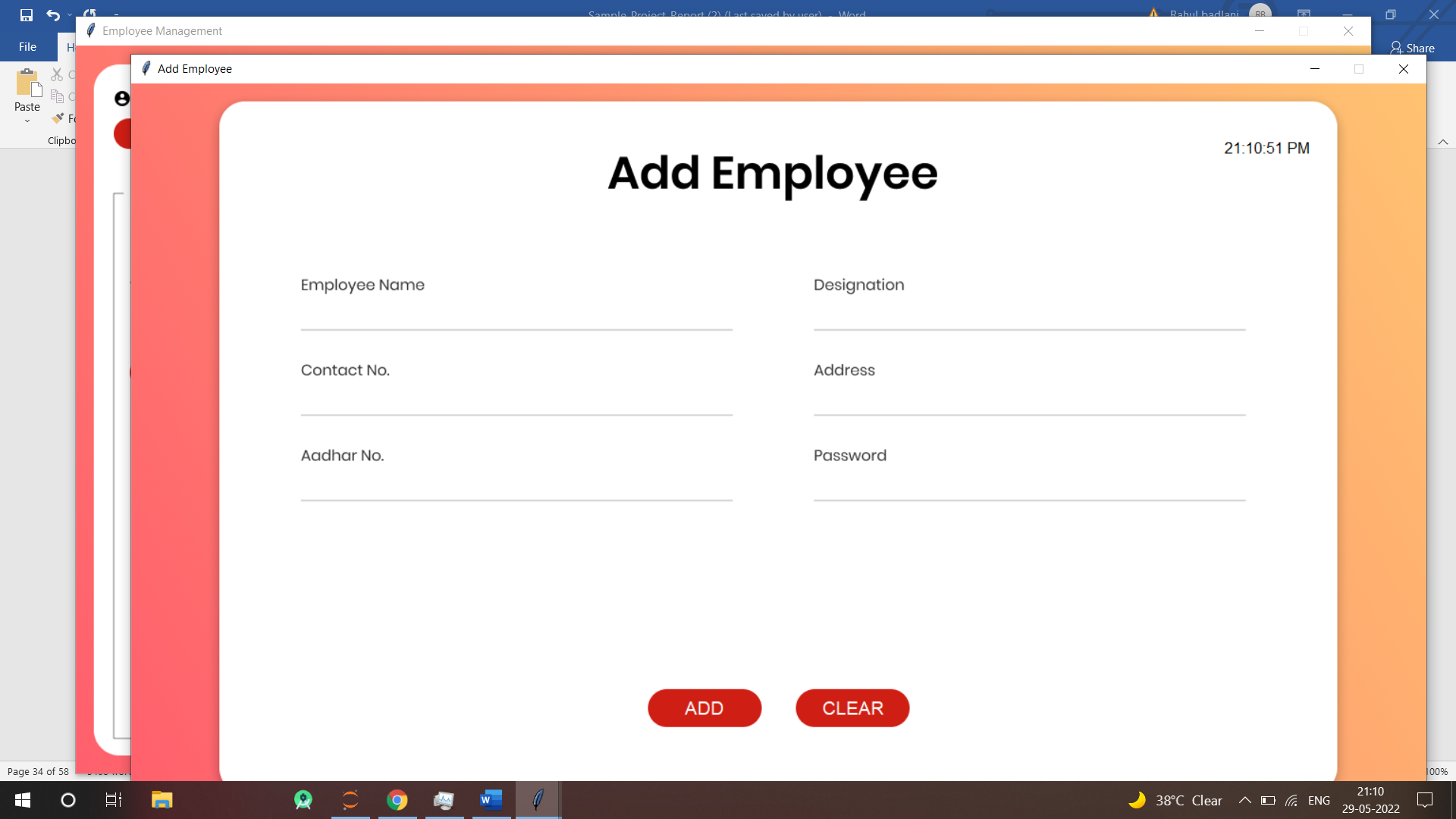
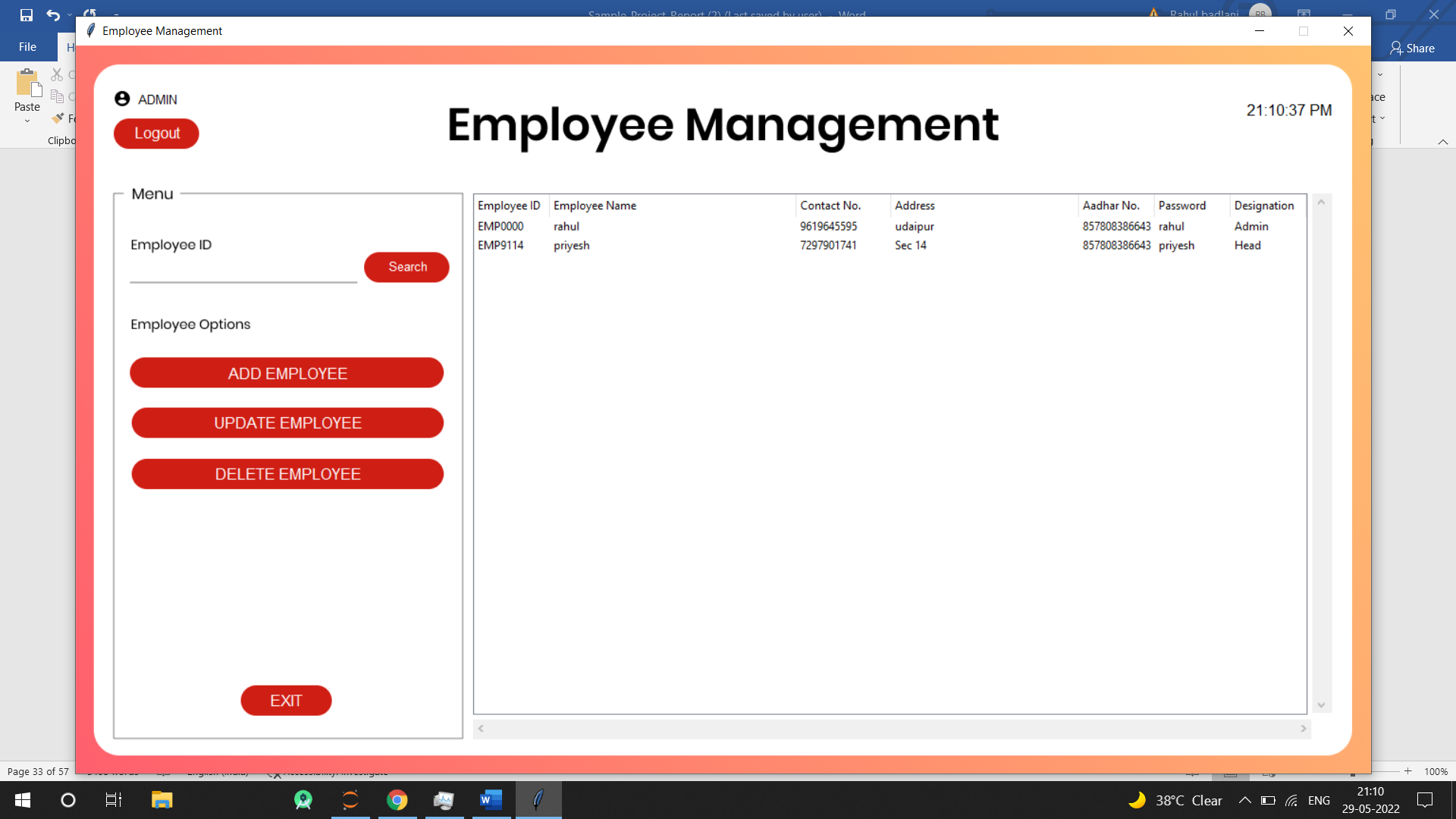
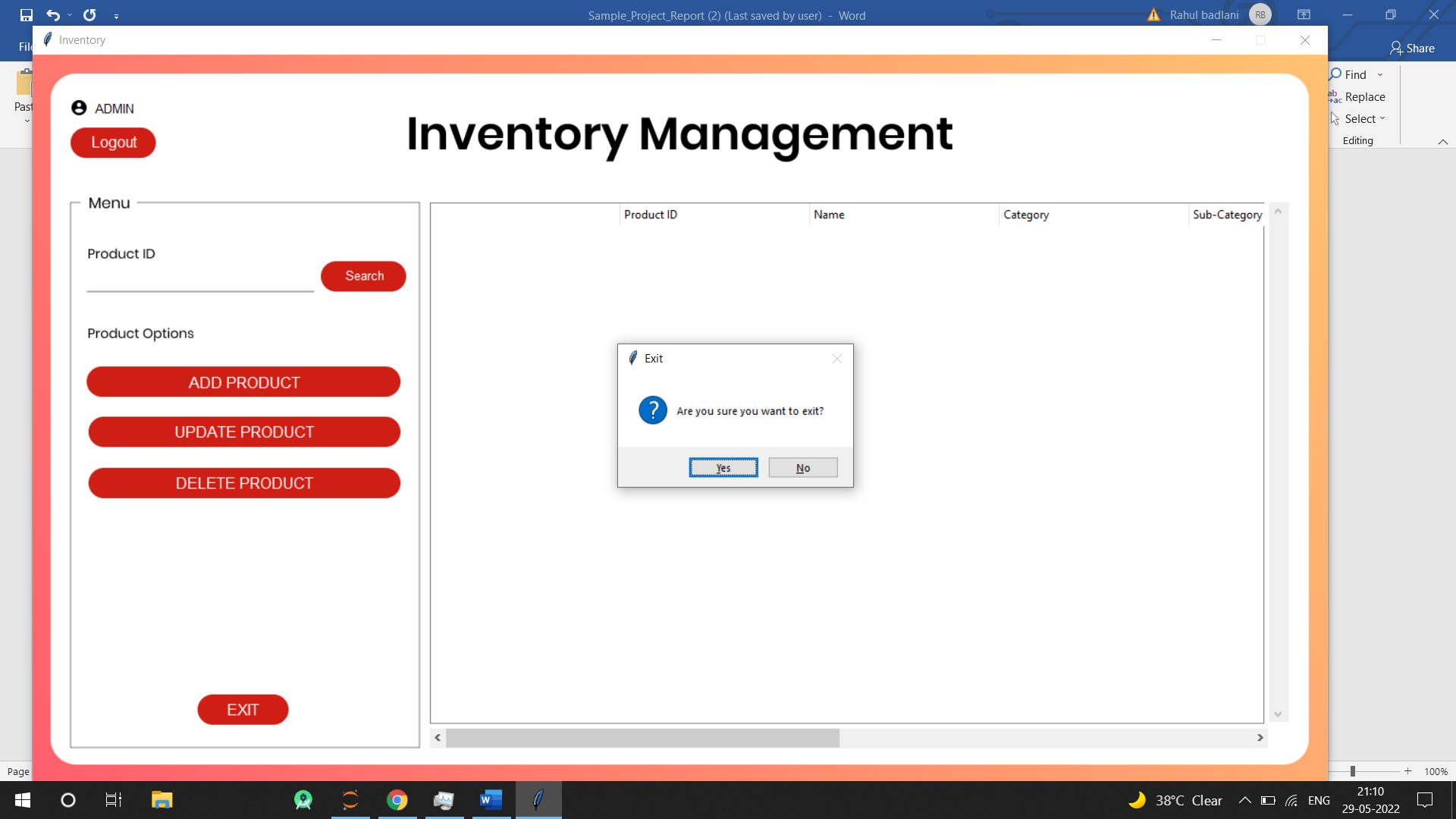
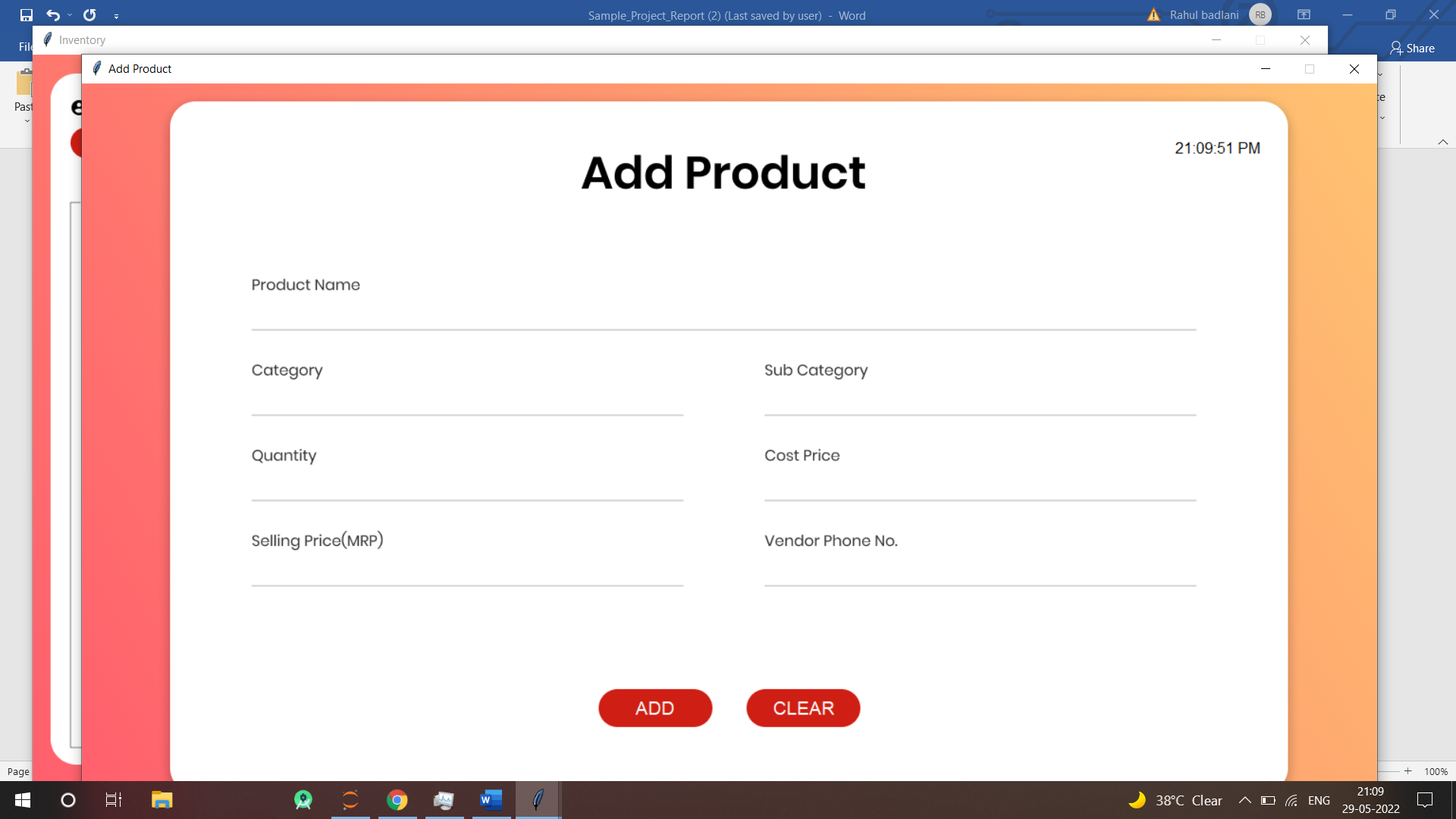
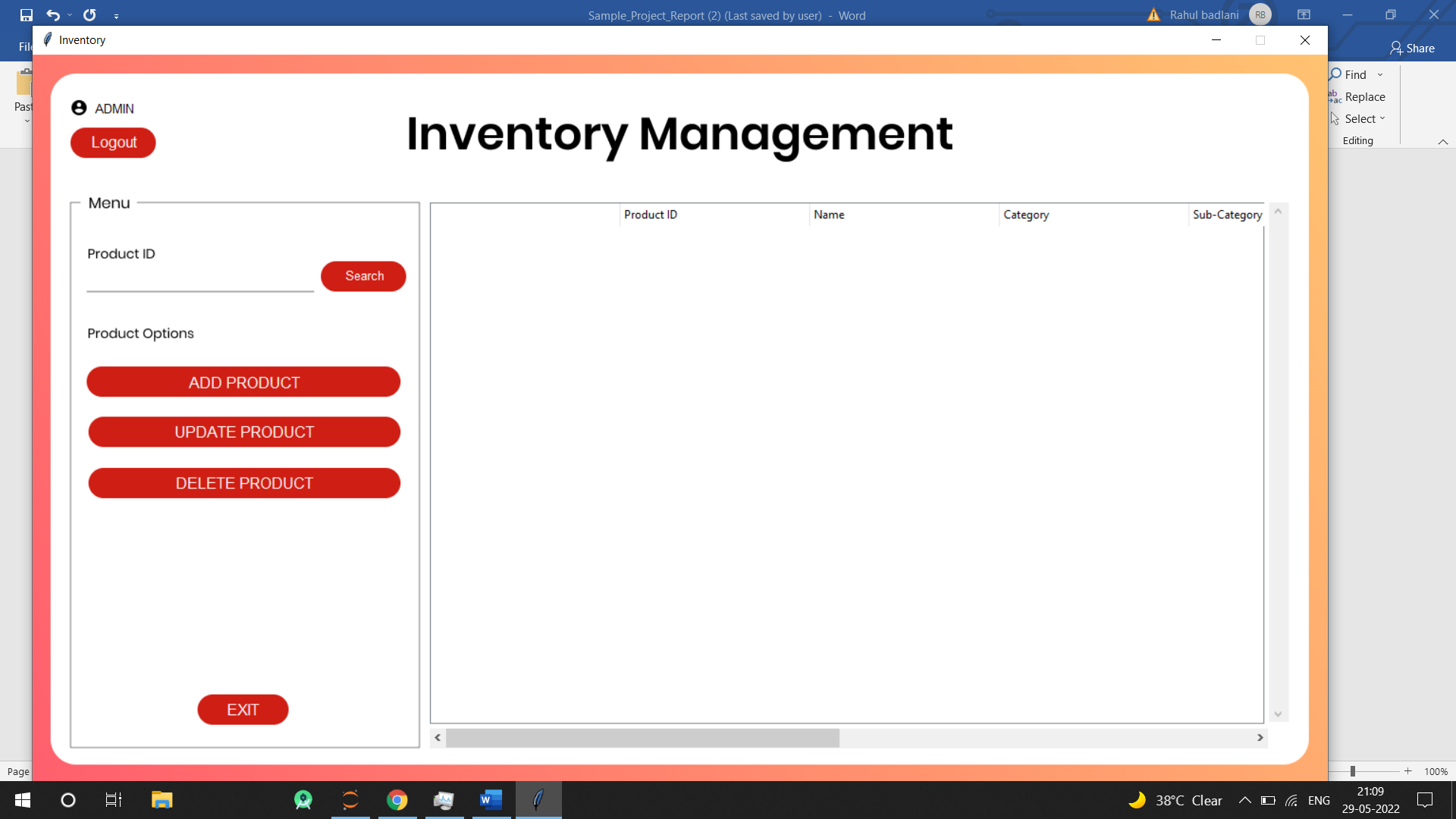
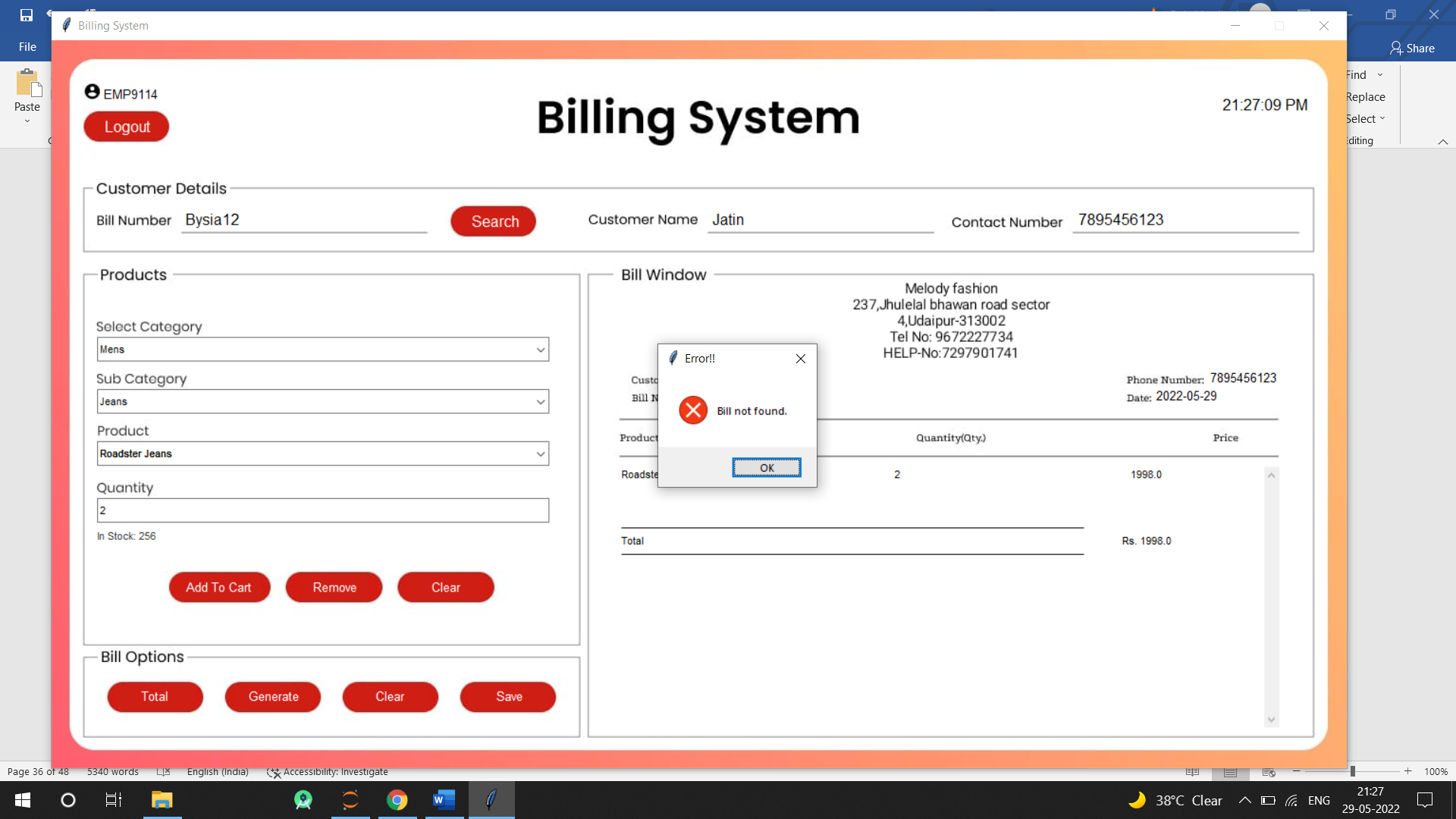


Fig. 3.4 Employee Mode

CHAPTER – IV

SCREEN SHOTS



CHAPTER – VI

## TESTING

Testing

Testing Methodology

Companies rely on software more than ever to provide and manage information with strategic and operational importance and to provide key decision support. Rising customer expectations for fault-free, requirements-exact software have increased awareness of the importance of software testing as a critical activity.

We begin the testing process by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process very fies that the application meets the requirements specified in the system requirements document and is bug free. At the end of each testing day, we prepare a summary of completed and failed tests. Applications are not allowed to launch until all identified problems are fixed. A report is prepared at the end of testing to show exactly what was tested and to list the final outcomes.

Our software testing methodology is applied in three distinct phases: unit testing, system testing, and acceptance were testing.

Unit Testing: The programmers conduct unit testing during the development phase. Programmers can test their specific functionality individually or with other units. However, unit testing is designed to test small pieces of functionality rather than the system as a whole. This allows the programmers to conduct the first round of testing to eliminate bugs before they reach the testing staff. In unit testing the analyst tests the programs making up a system.

For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided.

For example, a hotel information system consists of modules to handle reservations; guest checking and checkout; restaurant, room service and miscellaneous charges; convention activities; and accounts receivable billing. For each, it provides the ability to enter, modify or retrieve data and respond to different types of inquiries or print reports. The test cases needed for unit testing should exercise each condition and option.

Unit testing can be performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing a short program is used to execute the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

System Testing: The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together.

The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results.

The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results.

A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by persons other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

The system is tested as a complete, integrated system. System testing first occurs in the development environment but eventually is conducted in the production environment. Functionality and performance testing are designed to catch bugs in the system, unexpected results, or other ways in which the system does not meet the stated requirements.

The testers create detailed scenarios to test the strength and limits of the system, trying to break it if possible. Editorial reviews not only correct typographical and grammatical errors, but also improve the system’s overall usability by ensuring that on-screen language is clear and helpful to users. Accessibility reviews ensure that the system is accessible to users with disabilities.

System testing consists of the following five steps:

1. Program testing
2. String testing
3. System testing
4. System documentation
5. User acceptance testing

Program Testing

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. It is the responsibility of a programmer to have an error free program. At

The time of testing the system, there exists two types of errors that should be checked. These errors are syntax and logic.

A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields out of range items, and invalid combinations.

Since the logical errors are not detected by compiler, the programmer must examine the output carefully to detect them. When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of the instructions, must be traced to determine the problem. The process is facilitated by breaking the program down into selfcontained portions, each of which can be checked at certain key points.

String Testing

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each part of the system is tested against the entire module with both test and live data before the whole system is ready to be tested.

System Testing

System testing is designed to uncover weaknesses that were not found in earlier tests. This includes forced system failure and validation of total system as it will be implemented by its user in the operational environment. Under this testing, generally we

Take low volumes of transactions based on live data. This volume is increased until the maximum level for each transaction type is reached.

The total system is also tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency.

All this is done with the old system still in operation. When we see that the proposed system is successful in the test, the old system is discontinued.

System Documentation

All design and test documentation should be well prepared and kept in the library for future reference. The library is the central location for maintenance of the new system.

User Acceptance Testing

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of important data is maintained. Performance of an acceptance test is actually the user's show. User motivation is very important for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the system's tolerance, performance range, error rate and accuracy.

CHAPTER – VII

## CONCLUSION AND FUTURE ENHANCEMENTS

7.1. Limitations

The new system has been designed to meet almost all of the user requirements but it too has certain limitations some of which can be enhanced in the future enhancements or updates

7.1.1. Supports only Multiple choice and no Theory

The existing system supports only multiple choice questions and these can be used only for the logic testing of the student whereas paper based tests can provide a overall result of the users knowledge for that topic. This will do injustice to users who are sometimes aware of answers but not sure completely. It supports only two values True and False. If the selected answer is true the user gets marks, If not, loses.

7.1.2 Exam Time Cannot be Modified

The existing system allows the exam time to be set only once at the time of system set up which cannot be changed as per desired and all the exams will be conducted for the preset time only. If the user wants to increase or decrease the time for exams the system will have to be set up again. This is also one of the proposed enhancements.

7.1.3 No of Questions in an Exam cannot be changed.

The system will have a preset no of questions that can be attempted in an exam by the user which cannot be changes in the existing system. If a user wants that an exam has to have more or less questions than the system has to be set up again which is again not feasible. The proposed system enhancements will have a field in the faculty module where the faculty can set no questions and time allowed for a selected subject.

7.1.4 No Facility for the Disabled

The existing system has no options available by which a physically disabled user can use the system. If the user is blind or deaf the system has no sound response pr Braille response facility. This makes it very difficult to set this system as a standard for all examination procedures as it does not support all time of users that a standard system is supposed to.

7.2 Future Enhancements

Enhancements are the perquisite for development of a system. Every existing system has proposed enhancements which make it better and easier to use and more secure. The enhancements that have been proposed for this system are listed here.

7.2.1 Including Image Support.

The existing system has no image support due to which the images cannot be used. It will be one of the priorities in the proposed enhancements to enable Image Support in the system.

7.2.2 Facility to Set Exam Time.

The new system will allow the faculty to set the exam time for a particular subject. It will increase the efficiency of the system greatly as it will enable each subject to be tested for a different duration as desired and required.

7.2.3 Facility to Set No of Questions in Exam

The new system will allow the faculty to set no of questions that can be asked in a subject. It will improve the flexibility of the exams by allowing the faculty to change the no of questions that can be asked in a subject.

7.2.4 Flags

The new system will allow the users to set flags for an answered question. Flags like Passed. Doubtful, Answered etc will help a user to return to those questions to review them and change them accordingly.

7.3 Conclusion

The development of software includes so many people like user system developer, user of system and the management, It is important to identify the system requirements by properly collecting required data to interact with supplier and customer of the system.

Proper design builds upon this foundation to give a blue print, which is actually implemented by the developers.

On realizing the importance of systematic documentation all the processes are implemented using a software engineering approach. Working in a live environment enables one to appreciate the intricacies involved in the System Development Life Cycle (SDLC).

We have gained a lot of practical knowledge from this project, which we think, shall make us stand in a good state in the future.

CHAPTER – VIII

BIBLIOGRAPHY

Bibliography

 List of useful Websites

* www.msdn.microsoft.com
* www.w3schools.com
* www.webdevelopersnotes.com

 List of useful Books

* ASP.NET 2.0 Website Programming : Problem design solution by Marco Bellinaso
* OReilly(E-Book) : Programming in .NET  PL/SQL Programming : Evan Bayross