**DETECTING COUNTERFEIT**

**PRODUCT USING BLOCKCHAIN**

A PROJECT REPORT

Submitted by

**ATHULYA A**

**CUAUMCS008**

In Partial Fulfilment of the Requirements for the Degree of  **MSC COMPUTER SCIENCE**



**CENTRE FOR COMPUTER SCIENCE AND**

**INFORMATION TECHNOLOGY (CCSIT) UNIVERSITY OF CALICUT**

**JULY 2022**

# DECLARATION

I **ATHULYA A,** hereby declare that the project work entitled **“DETECTING COUNTERFEIT PRODUCT USING BLOCKCHAIN”** is original work done by me, under the guidance of **Mrs. SONA R,** Assistant Professor**,** Centre for Computer Science and Information Technology, University of Calicut. This project report is submitted in partial fulfilment of the requirement for the award of the degree of MSc Computer Science during the period of study at University of Calicut.

Place: **CU CAMPUS** **ATHULYA A** Date: **CUAUMCS008**

# CERTIFICATE

*Certified that this is a bona fide record of the project work entitled* **“DETECTING**

**COUNTERFEIT PRODUCT USING BLOCKCHAIN”** *submitted by* **ATHULYA A (Reg No: CUAUMCS008)** *in partial fulfilment of the requirement for the award of the degree of* **MSC COMPUTER SCIENCE** *from University of* *Calicut.*

## Mrs. SONA R Mrs. SEEMA S

Assistant Professor Associate Coordinator

CCSIT, CU Campus CCSIT, CU Campus

**Place:** CU CAMPUS

**Date :**

Certified that the candidate was examined by us in the Project Viva Voce Examination

held on …..……………......................................................... and her Register Number is

……………………………………………………………………………………………

…….

## Examiners

……………………………….

……………………………….

# ACKNOWLEDGEMENT

Let me take this opportunity to thank all those who have been directly and indirectly involved in making this project a success. First of all, thankyou Universe for making this possible.

I express my gratitude to Mrs Seema S, Associate Coordinator, Centre for Computer Science and Information Technology, University of Calicut and Dr. C D Ravikumar,Chief Coordinator, Centre for Computer Science and Information Technology, Calicut University.

I am extremely grateful to Mrs. Sona R, Assistant Professor, Centre for Computer Science and Information Technology, University of Calicut, my Project-in-Charge for the encouragement, valuable suggestions and guidance rendered.

I express my sincere thanks to my family and friends who helped me in overcoming all the hurdles thus gearing me towards the fulfilment of the task that I had overtaken.

I also express my profound gratitude to all others who, in some way or another, helped me with this project till the end.

Place: **CU CAMPUS** **ATHULYA A** Date: **CUAUMCS008**

# INDEX

|  |  |
| --- | --- |
| **CONTENTS** | **PAGE NO.** |
| **1. INTRODUCTION** | 1 |
|  INTRODUCTION | 2 |
| **2. SYSTEM STUDY** | 3 |
|  PURPOSE | 4 |
|  SCOPE | 4 |
|  EXISTING SYSTEM | 4 |
|  PROPOSED SYSTEM | 5 |
| **3. SYSTEM ANALYSIS** | 6 |
|  PRELIMINARY INVESTIGATION | 7 |
|  FEASIBILITY STUDY | 7 |
| **4. SYSTEM SPECIFICATION** | 9 |
|  HARDWARE SPECIFICATION | 10 |
|  SOFTWARE SPECIFICATION | 11 |
|  MODULES | 12 |
| **5. SYSTEM DESIGN** | 14 |
|  INPUT DESIGN | 15 |
|  OUTPUT DESIGN | 16 |
|  DATA FLOW DIAGRAM | 16 |
|  DATABASE DESIGN | 22 |
| **6. SYSTEM TESTING AND IMPLEMENTATION** | 28 |
|  SYSTEM TESTING | 29 |
|  SYSTEM IMPLEMENTATION | 32 |
| **7. SYSTEM MAINTENANCE** | 33 |
|  SYSTEM MAINTENANCE | 34 |
| **8. CONCLUSION** | 35 |
|  CONCLUSION | 36 |
| **9. BIBLIOGRAPHY** | 37 |
|  BIBLIOGRAPHY | 38 |
| **10. APPENDIX** | 39 |
|  SCREENSHOTS | 40 |

# ABSTRACT

In recent years, blockchain has received increasing attention and numerous applications have emerged from this technology. A renowned Blockchain application is the cryptocurrency Bitcoin, that has not only been effectively solving the double-spending problem but also it can confirm the legitimacy of transactional records without relying on a centralized system to do so. Therefore, any application using Blockchain technology as the base architecture ensures that the contents of its data are tamper-proof. This paper uses the decentralized Blockchain technology approach to ensure that consumers do not fully rely on the merchants to determine if products are genuine. We describe a decentralized Blockchain system with products anti- counterfeiting, in that way manufacturers can use this system to provide genuine products without having to manage direct-operated stores, which can significantly reduce the cost of product quality assurance.

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **SL NO.** | **FIGURES** | **PAGE NO.** |
| 1 | Figure 1 (Level 0 Diagram) | 18 |
| 2 | Figure 2 (Level 1Diagram) | 19 |
| 3 | Figure 3 (Level 2 Diagram) | 20 |
| 4 | Figure 4 (Level 3 Diagram) | 21 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **SL NO.** | **TABLES** | **PAGE NO.** |
| 1 | Table 1 (login) | 24 |
| 2 | Table 2 (Manufacture) | 25 |
| 3 | Table 3 (user) | 26 |
| 4 | Table 4 (Category) | 27 |
| 5 | Table 5 (Complaint) | 27 |

**DETECTING COUNTERFEIT**

**PRODUCT USING BLOCKCHAIN**

**1.INTRODUCTION INTRODUCTION**

In recent years, Counterfeit goods play a vital role in product manufacturing industries. This Phenomenon affects the sales and profit of the companies. To ensure the identification of real products throughout the supply chain, a functional block chain technology used for preventing product counterfeiting. By using a block chain technology, consumers do not need to rely on the trusted third parties to know the source of the purchased product safely. Any application that uses block chain technology as a basic framework ensures that the data content is ‘tamper resistant’. In view of the fact that a block chain is the decentralized, distributed and digital ledger that stores transactional records known as blocks of the public in several databases known as chain across many networks. Therefore, any involved block cannot be changed in advance, without changing all subsequent block. In this paper, counterfeit products are detected using barcode reader, where a barcode of the product linked to a Block Chain Based Management (BCBM) system. So the proposed system may be used to store product details and unique code of that product as blocks in database. It collects the unique code from the customer and compares the code against entries in block chain database. If the code matches, it will give notification to the customer, otherwise it gets information from the customer about where they bought the product to detect counterfeit product manufacturer.

**2. SYSTEM STUDY**

# SYSTEM STUDY

## PURPOSE

The proposed concept is counterfeiting products. This project aims to detect product counterfeiting .Detecting counterfeit products and blockchain technology is used to detect fake products. Furthermore, the information is encoded into a QR code. Customers or users scan the QR code and then they can detect the fake product. Digital information of product can be stored in the form of blocks in blockchain technology

## SCOPE

Counterfeiting products are growing exponentially with the enormous amount online. So, there is a strong need to detecting counterfeit products and blockchain technology is used to detect fake products. Furthermore, the information is encoded into a QR code. Customers or users scan the QR code and then they can detect the fake product. Digital information of product can be stored in the form of blocks in blockchain technology. Future enhancement of current project includes efficient handling of products with major focus on quality and authenticity .Efficient tracing and tracking of products in our block chain.

* Enhancement of efficient tracking and tracing in our blockchain environment

* more transparency and less cost expenditure in authenticating the products

## EXISTING SYSTEM

Counterfeit goods play a vital role in product manufacturing industries. This Phenomenon affects the sales and profit of the companies. Current system does not provide any security in supply chain and does not provide any security in supply chain and does not detect product on counterfeit.

Limitations of existing system

* Not secure enough
* Prone to product counterfeit
* Involvement of third parties

## PROPOSED SYSTEM

we propose to implement a Blockchain architecture provided by Ethereum to record product ownership on the Blockchain. By using Blockchain’s untrace ability and transparency properties, and the assurance that each record cannot be forged on the Blockchain, consumers don’t need to fully rely on trusted third parties to safely know the source of the purchased product. counterfeit products are detected using QR code reader, where a QR code of the product linked to a Block Chain Based Management (BCBM) system. So the proposed system may be used to store product details and unique code of that product as blocks in database. It collects the unique code from the customer and compares the code against entries in block chain database. If the code matches, it will give notification to the customer, otherwise it gets information from the customer about where they bought the product to detect counterfeit product manufacturer.

**3. SYSTEM ANALYSIS**

# SYSTEM ANALYSIS

## PRELIMINARY INVESTIGATION

The main aim of preliminary analysis is to identify the problem. First, need for the new or enhanced system is established. Only after the recognition of need, for the processed system done then further analysis is possible.

Once the initial investigation is done and the need for new or improved system is established, all possible alternate solutions are chalked out. All those systems are known as

“candidate system”. All the candidate systems are then weighed and the best alternative of all those is selected as the solution system, which is termed as the “proposed system”. The proposed system is then evaluated of its feasibility. Feasibility for a system means whether it is practical and beneficial to build that system.

The system has been designed in such a way that it can be modified with very little effort when such a need arises in the future. The system has been found to work efficiently and effectively. The administrator controls the entry system. He can add new authorities and update the authorities.

## FEASIBILITY STUDY

The feasibility study is defined as the practical extent to which a project can be performed successfully. When the client approaches the organization for getting the desired product developed, it comes up with a rough idea about what all functions the software must perform and which all features are expected from the software. This feasibility study is focused on the goal of the organization. This study analyses whether the software product can be practically materialized in terms of implementation, the contribution of the project to organization, cost constraints and as per values and objectives of the organization. It explores technical aspects of the project and product such as usability, maintainability, productivity and integration ability.

Three key considerations involved in the feasibility analysis are:

1. Technical feasibility
2. Economic feasibility
3. Operational feasibility

### 1. TECHNICAL FEASIBILITY

Technical feasibility assesses the current resources (includes the hardware and software) and technology. Its centre on the existing computer system and to what extent it can support the proposed addition. Since the minimum requirements of the system like internet connection, a web server in the server system is met by an average user.

### 2. ECONOMICAL FEASIBILITY

The economic analysis is to determine the benefits and savings with the current system and the proposed system that are compared with costs. The system is economically feasible, as the organization possesses the hardware and software resources required for the functioning of the system. Any additional resources, if required, can also be easily acquired. The proposed system was developed with available resources. Since cost input for the software is almost nil the output of the software is always a profit. Hence Software is economically feasible.

### 3. OPERATIONAL FEASIBILITY

Operational feasibility assesses the extent to which the required software system performs a series of steps to solve business problems and user requirements. This is the longest phase in the development life cycle of a system. So, operational feasibility should be given much importance. The users of the application don’t need thorough training on the system. It has a user-friendly interface.

## 4. SYSTEM SPECIFICATION HARDWARE SPECIFICATION

The selection of hardware is very important in the existence and proper working of any of the software. When selecting hardware, the size and capacity requirements are also important. The hardware must suit all application developments**.**

* PROCESSOR : i3 or above.
* SYSTEM BUS : 32Bit or 64Bit
* RAM : 4 GB or Above
* HDD : 500 GB or Above
* MONITOR : 14” LCD or Above
* KEYBOARD : 108 Keys
* MOUSE : Any Type of mouse

: Android supported

* MOBILE

mobile phone

|  |  |
| --- | --- |
|  | Any 32-bit or 64-bit platform |
|  FRONT END | : Python  HTML  CSS |
|  BACK END | : MySQL Sever |
|  IDE | : Python 3.6 or above |
|  | PyCharm |

# SOFTWARE SPECIFICATION

One of the most difficult tasks is selecting software. Once the system requirement is found out then we have to determine whether a particular software package fits for those system requirements. This section summarizes the application requirement.

 OPERATING SYSTEM : Windows 10

# MODULES

## ADMIN

* login
* view manufacture
* view user
* view user complaint and send reply
* category management
* change password

## MANUFACTURE

* Registration
* Login
* View profile
* Product management
* Change password

|  |
| --- |
| **USER**     * Registration * Login * View profile * Scan QR * Send complaint and view reply * View product * Change password |

## 5. SYSTEM DESIGN

# SYSTEM DESIGN

The detailed design of the system selected in the study phase is accomplished in the design phase and the user-oriented performance specification is covered into a technical design specification. The principle activities performed by the design phase are general system design, input design and the design of the database.

## INPUT DESIGN

Input design is the process of converting the user-oriented input data is to make the automation is easy and free from errors. The design of handling input specifies how data are accepted for computer processing. Input design is art of overall system design that needs careful attention and if includes specifying the means by which actions are taken. A system user interacting through a work station must be able to tell the system whether to accept input produce a report or end processing.

The collection of input data is considered to be the most expensive part of the system design. Since the inputs have to be planned in such a manner so as to get the relevant information extreme care is taken to obtain the information if the data going into the system is incorrect then the processing and outputs will magnify these errors. Input design is a part of overall system design, which requires careful attention.

The major objective of the input design is to make the data entry easier, logical and error free. With this objective the screen for the system is developed. The input design requirement such user friendliness, consistent format and interactive dialogue boxes for giving the development of the project. The data entry operator needs to know the space allocated for each field, the field sequence, which must match with source document and the format in which the data is entered.

## OUTPUT DESIGN

A quality output is the one, which meets the requirements of the end user and presents the information early. In any system, the results of processing are given to the user through the output. In the output design it is determined how the information is to be displayed for immediate need.

Output design should improve the relationship of the system with user and help in decision making. The objective of the output design is to define the format of all printed documents and of the screens that will be produced by the system. The output has been designed as per the needs of the institution. The suggestions of the user are also taken into consideration while designing the layout and the fields that are to be included in the project.

Operations are required primarily to communicate about the results of the processing to the user and to the management. They also provide the hardcopy of these results for later consultations and rectification. In this system operations are of different from like messages, input boxes and reports. All recent reports can be generated by the system Menu will help the user to select the required outputs.

## DATA FLOW DIAGRAM

Data flow diagram is used is to define the flow of the system and the its resources such as information. As the name suggests DFD is a representation of the data flow within the system. It includes processes, data store and external interfaces to the system. External interfaces may be treated as external entities. DFD's can by explode to sub-processes. Any two entities data store must be connected to each other by some process. The process cannot be directly connected to each other. The following are the four major components of the DFD:

###  Process  Data store  Data flow  External Entity

It is also known as the bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to system, various processing carried out in these data and output data generated by the system. The main reason why this DFD technique is so popular is probably because of facts that DFD is 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 these systems. Starting with a set of high-level functions that a system performance of DFD model in hierarchically it represents various sub functions. The data flow diagramming technique also follows a simple set of intuitive concepts and rules.

### Process

Process shows the work of the system. Each process has one or more data inputs and produce one or more data outputs. Process is represented by rounded rectangles, circles or ovals in DFD's. Each process has a unique name and number. This name and number appear inside the rectangle that represents the process in a DFD.

### Data store

Data store is a repository of data. Processes can enter data into or retrieve the data from data store. Each data has a unique name.

### Data flow

Data flow show the passage of data in the system and represented by lines joining system components. An arrow indicates the direction of flow and the line is labeled by name of the data flow.

### External Entity

External Entities are outside the system but they either supply input data into the system or use other system output. They are represented by rectangles. They are entities on which the designer has control. They may be any natural person, corporation, partnership, sole proprietorship, association, organization, company, government agency etc.

**DFDs**

**LEVEL 0**

Admin

User

Manufacture

Detecting

counterfeit product

DB

Request

Request

Request

Request

Response

Response

Response

Response

*Figure*

*1*

*(*

*Level 0 Diagram)*

### LEVEL 1

Login

Change

password

View/Approve/Re

ject /manufacture

View user

View complaint

& send reply

Admin

Login

Category

management

Manufacture

User

Complaint

Category

*Figure 2 (Level 1 Diagram)*  **LEVEL 2**

User

Registration

Login

Change

password

View profile

Scan

QR

Send complaint

and view reply

View product

User

Login

Complaint

Product

*Figure 3 (Level 2 Diagram)*

**LEVEL**

**3**

Manufacture

Registration

Login

Change

password

View profile

Product

management

Manufacture

Login

Product

*Figure*

*4*

*(*

*Level*

*3*

*Diagram)*

## DATABASE DESIGN

Database design is one of the most important parts of the system design phase. In a database environment, common data are available and are used by several users. Instead of each program managing its own data, authorized users share data across the application with the database software managing the data as an entity. The primary objective of database design is fast response time to enquiries, more information at low cost, control of redundancy, clarity and ease of use, date and program independence, accuracy and integrity of the system, fast recovery and availability of powerful end-user languages. The theme behind a database is to handle information as an integrated whole, thus the main objective is to make the information as to access easy, quick, inexpensive and flexible for the users. Data directory specifies the major element in the system, and care should be taken while designing, in order to avoid unnecessary duplication of data. The entire package depends on, how the data are maintained in the system. Several tables are maintained in the system to store data that are required for the processing of various data as well as storing intermediate or final processed results.

Database design mainly aims at handling large volumes of information, involving the definitions for the structure of storage and provisions for the manipulation of information, providing safety of information despite system crashes due to unauthorized access.

Some conditions are satisfied in the database design stage:

* Control redundancy.
* Ease of use.
* Data independence.
* Accuracy and integrity.
* Recovery from failures.
* Security and privacy.
* Performance.

### NORMALISATION

It is a process of efficiency organizing data in database. Normalization has two main goals. One is to eliminating redundant data, ensuring data dependencies make sense. They reduce the amount of space a database consumes, and ensure that is logically stored.

* **First normal form**:

First normal form eliminates duplicate columns from the same table. It creates separate table for each group of related data and identify each row with a unique column or set of columns (primary key).

* **Second normal form**:

Second normal form meets all the requirements of the first normal form. It removes subset of data that apply to multiple rows of a table and place them in separate tables. Create relationship between these new tables and their predecessors through the use of foreign keys.

* **Third normal form**:

Third normal form meets all the requirements of the second normal form. It removes columns that are not dependent upon the primary key.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s.no | Field name | Data type | Length | Constraint |
| 1 | login\_id | int | 11 | Primary key, Not null,Auto increment |
| 2 | user\_name | varchar | 100 | Not null |
| 3 | password | varchar | 100 | Not null |
| 4 | User\_type | varchar | 100 | Not null |

**TABLES**

*Table 1 (Login)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s.no | Field Name | Data type | Length | Constraint |
| 1 | ins\_id | int | 11 | Primary key,Not null ,Auto increment |
| 2 | ins\_name | varchar | 200 | Not null |
| 3 | ins\_license | varchar | 200 | Not null |
| 4 | ins\_establishedyr | varchar | 10 | Not null |
| 5 | ins\_place | varchar | 100 | Not null |
| 6 | ins\_post | varchar | 100 | Not null |
| 7 | ins\_pin | varchar | 10 | Not null |
| 8 | ins\_district | varchar | 100 | Not null |
| 9 | ins\_ph | varchar | 20 | Not null |
| 10 | Ins\_email | varchar | 150 | Not null |
| 11 | Ins\_photo | varchar | 200 | Not null |
| 12 | status | varchar | 15 | Not null |

*Table 2 (Manufacture)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s.no | Field name | Data type | Length | constraint |
| 1 | id | Int | 11 | Primary key, Not null ,Auto increment |
| 2 | name | Varchar | 200 | Not null |
| 3 | place | Varchar | 100 | Not null |
| 4 | post | Varchar | 100 | Not null |
| 5 | pin | Varchar | 10 | Not null |
| 6 | district | Varchar | 100 | Not null |
| 7 | ph | Varchar | 20 | Not null |
| 8 | email | Varchar | 150 | Not null |
| 9 | photo | Varchar | 200 | Not null |

*Table 3 (User)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.no | Field name | Data type | Length | constraint |
| 1 | ctry\_id | int | 11 | Primary key, Not null ,Auto increment |
| 2 | Cat\_name | varchar | 50 | Not null |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s.no | Field name | Data type | Length | Constraint |
| 1 | Complaint\_id | int | 11 | Primary key, Not null, Auto increment |
| 2 | date | date |  | Not null |
| 3 | lid | int | 11 | Not null |
| 4 | complaint | varchar | 200 | Not null |
| 5 | reply | varchar | 200 | Not null |
| 6 | status | varchar | 200 | Not null |

*Table 4 (Category*

*Table 5 (Complaint)*

## *6.SYSTEM TESTING AND IMPLEMENTATION*

# SYSTEM TESTING

In general, testing is finding out how well software or a product works. Testing is a set of activity that can be planned and conducted systematically. Testing begins at the module level and work towards the integration of entire computers-based system. Nothing is complete without testing, as it is vital success of the system.

Software testing is a process of executing a program or application with the intent of finding the software bugs. Testing is for evaluating a system or its components to find whether it satisfies the specified requirements or not. Testing cannot show the absence of defects, it can also be stated as the process of validating and verifying that a software program or software or product.

## TESTING TECHNIQUES

* Unit testing
* Integration Testing
* Validation Testing
* System Testing
* Output Testing
* User Acceptance Testing

## UNIT TESTING

It is the first level of testing. Each module is tested individually and focus is given for finding errors limited to each individual module and correcting them. The different modules of the system are tested individually and corrected all errors. Each module is focused to work satisfactorily with regard to the expected output from the module.

## INTEGRATION TESTING

Integration testing is a systematic testing for construction the program structure while as the same time conducting tests to uncover Instead of testing the system as a whole, unit testing focuses on the module that make up the system. Each module is taken up individually and tested for correctness in coding and logic. Error resulting from interaction of module is initially avoided. It tests for the errors resulting from integration of modules. One specifies target of integration testing is the interface, whether errors associated with interfacing. The objective is to take until- tested modules and build a program structure that has been dictated by design.

## VALIDATION TESTING

Administrator has to specify the username and password. When the user enters username and password, checking it with the already registered username and password in the database will validate it. If they do not match, user is defined access there by providing a strong security.

## SYSTEM TESTING

System testing was performed to verify that all system elements have been properly integrated and perform allocated function. Security testing was done to check the security mechanisms built into the system, which will protect it from improper penetration, performance testing was done to test the runtime performance of the software. For user acceptance testing the system was given to the end user to use.

## OUTPUT TESTING

After performing the validation testing, the next step is the output testing of the enhanced system. No system could be useful if it does not produce the required output in the required format. The outputs generated or the displayed by the system are tested by asking the users about the format required by them.

## USER ACCEPTANCE TESTING

User acceptance testing comprises a completed and successful end-to-end system test with review of the results by one or more users with specific knowledge. Users may apply a variety of validation techniques. For example: generate a report from the new system and compare the results with the same report from the current system, data inspection and others.

Prior to this testing the system was delivered to the clients along with the steps for implementation. In user acceptance testing the system was run in the client environment by system users. The users were allowed to test the system and raise any issues for a specific period after the system is assumed to be complete and error free.

# SYSTEM IMPLEMENTATION

A software implementation method is integrating software-based service systematically. This is the phase in the software life cycle where the actual software is implemented. The result of this phase consists of source code, together with documentation to make the code more readable. Implementation is the stage of a project where the theoretical design is turned into a working system. If the implementation is not carefully planned and controlled, it may cause confusion. Implementation is done only after testing is done. The most crucial stage is achieving a new successful system and giving confidence about the new system for the user that it will work according to the requirements. It involves careful planning, investigation of the current system and its constraints on implementation. The more complex the system is, the more effort is needed to implement the system. In other words, implementation is the process of bringing a developed system into operational use and turning over it to the user implementation activities extend from planning through the old system to the new one.

The implementation involves the following formalities:

* Careful planning.
* Investigation of the systems and constraints.
* Design the methods to achieve the changes.
* Training the staffs in the changed phase.
* Evaluation of the changes over method.

## 7. SYSTEM MAINTENANCE

# SYSTEM MAINTANANCE

A process of modifying a software system or component after delivery to correct faults, to improve performance is known as software maintenance. A common perception of maintenance is that it merely involves fixing defects. However, one study indicated that the majority, over 80%, of the maintenance effort is used for non-corrective actions. Software maintenance is a very broad activity that includes error correction, enhancements of capabilities, deletion of obsolete capabilities, and optimization. Because change is inevitable, mechanisms must be developed for evaluation, controlling and making modifications. So, any work done to change the software after it is in operation is considered to be maintenance work. The purpose is to preserve the value of software over the time. The value can be enhanced by expanding the customer base, meeting additional requirements, becoming easier to use, more efficient and employing newer technology. Maintenance may span for 20 years, whereas development may be 1-2 years.

Corrective maintenance is done to repair the faults or defects found in day –to – day system functions, that is software design, logic and coding errors. Adaptive maintenance is the implementation of changes in a part of the system which has been affected by changes that occurred in some other part of the system. The objective of perspective maintenance should be to prevent failures and optimize the software. Minor adaptive changes should be handled by normal maintenance process. Major adaptive changes should be carried out as a separate development project.

## 8. CONCLUSION

# CONCLUSION

This project proposes a fully functional anti-product forgery system .This project helps users of our system no longer need to be concerned about the possibility of acquiring a counterfeited product. Our system can effectively lower the threshold of the anti-counterfeiting of branded goods and provide the companies with limited financial resources an as well an easier approach to provide consumers with the confidence that they will not purchase counterfeited goods.

## 9. BIBLIOGRAPHY

**BIBLIOGRAPHY**

### WEBSITES

* [**www.fullstackpython.com**](http://www.fullstackpython.com/)
* [**http://realpython.com/**](http://realpython.com/)
* [**http://www.mysql.com/** ](http://www.mysql.com/) **http://www.w3schools.c om/**
* [https://www.hyperledger.org](https://www.hyperledger.org/)

[/](https://www.hyperledger.org/)

* <https://www.tzero.com/>
* [https://litecoin.info/index.ph p/Main](https://litecoin.info/index.php/Main)

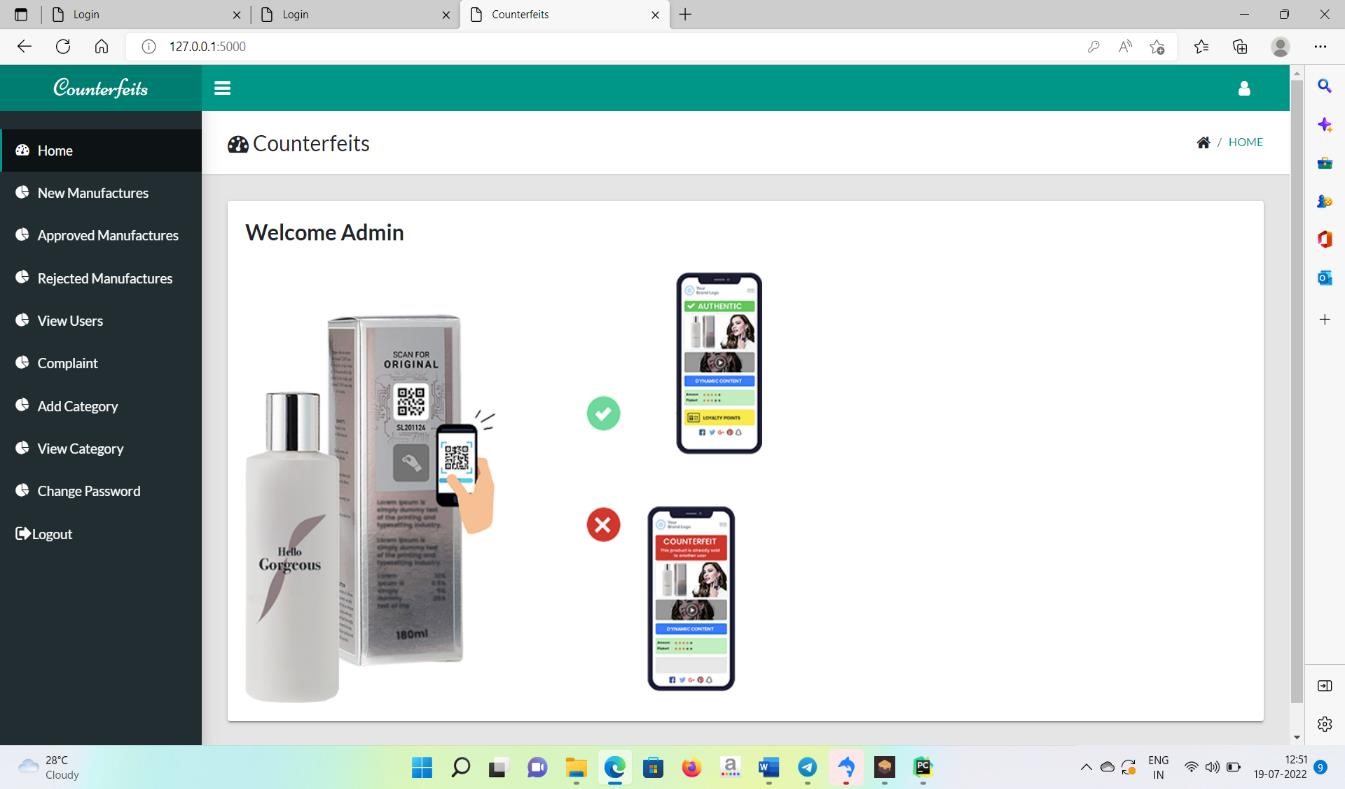
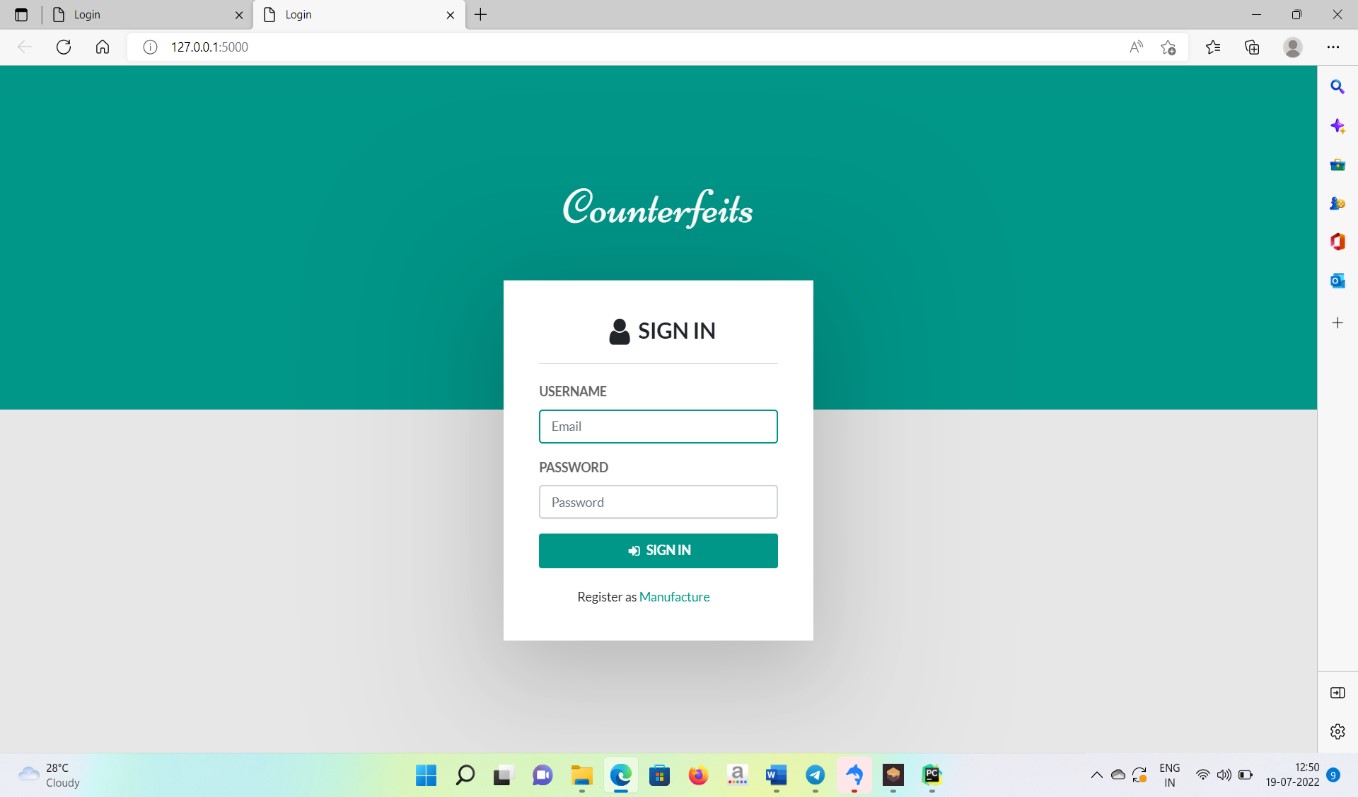
## 10. APPENDIX

**SCREENSHOTS**

**LOGIN:**

**ADMIN**

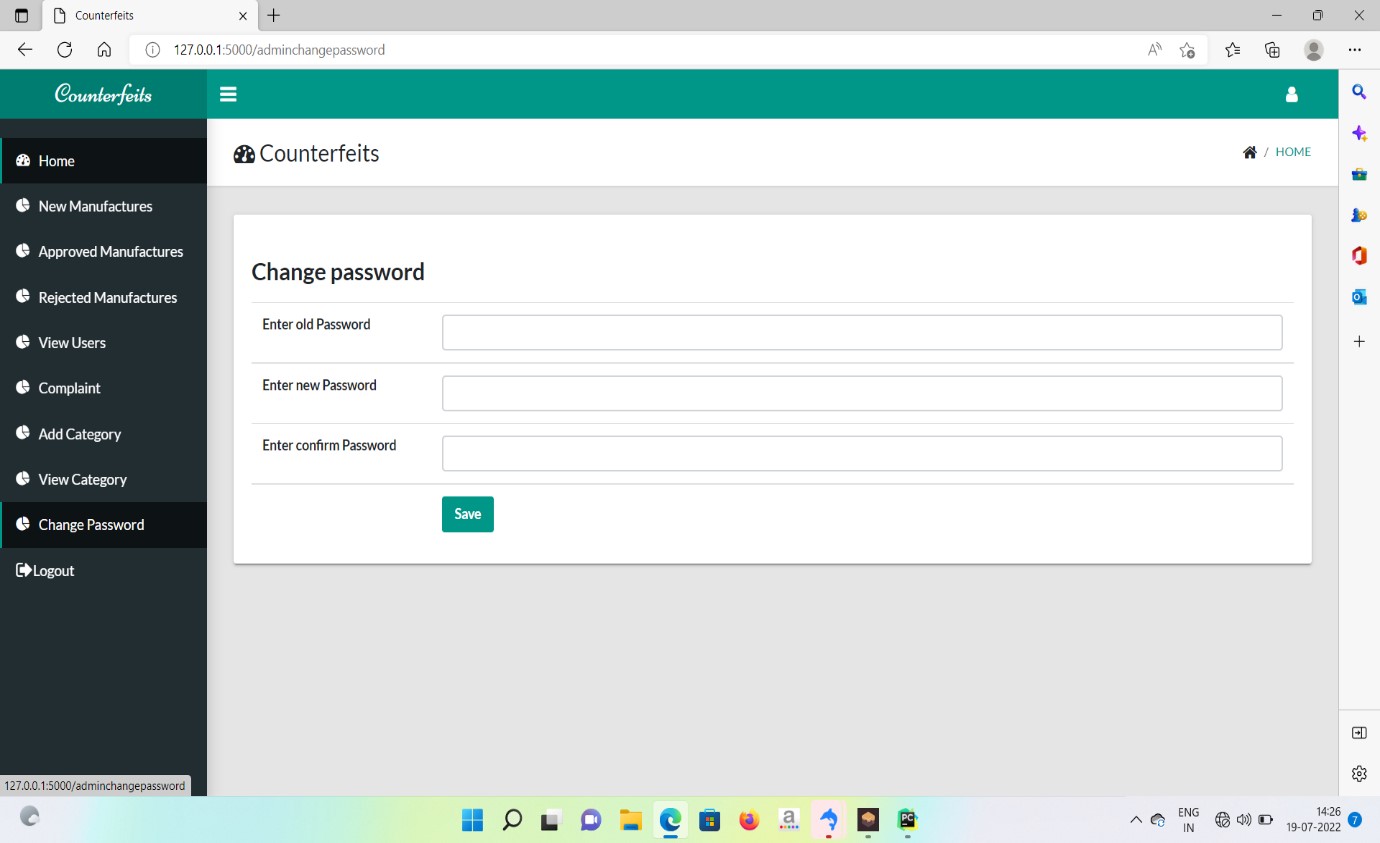
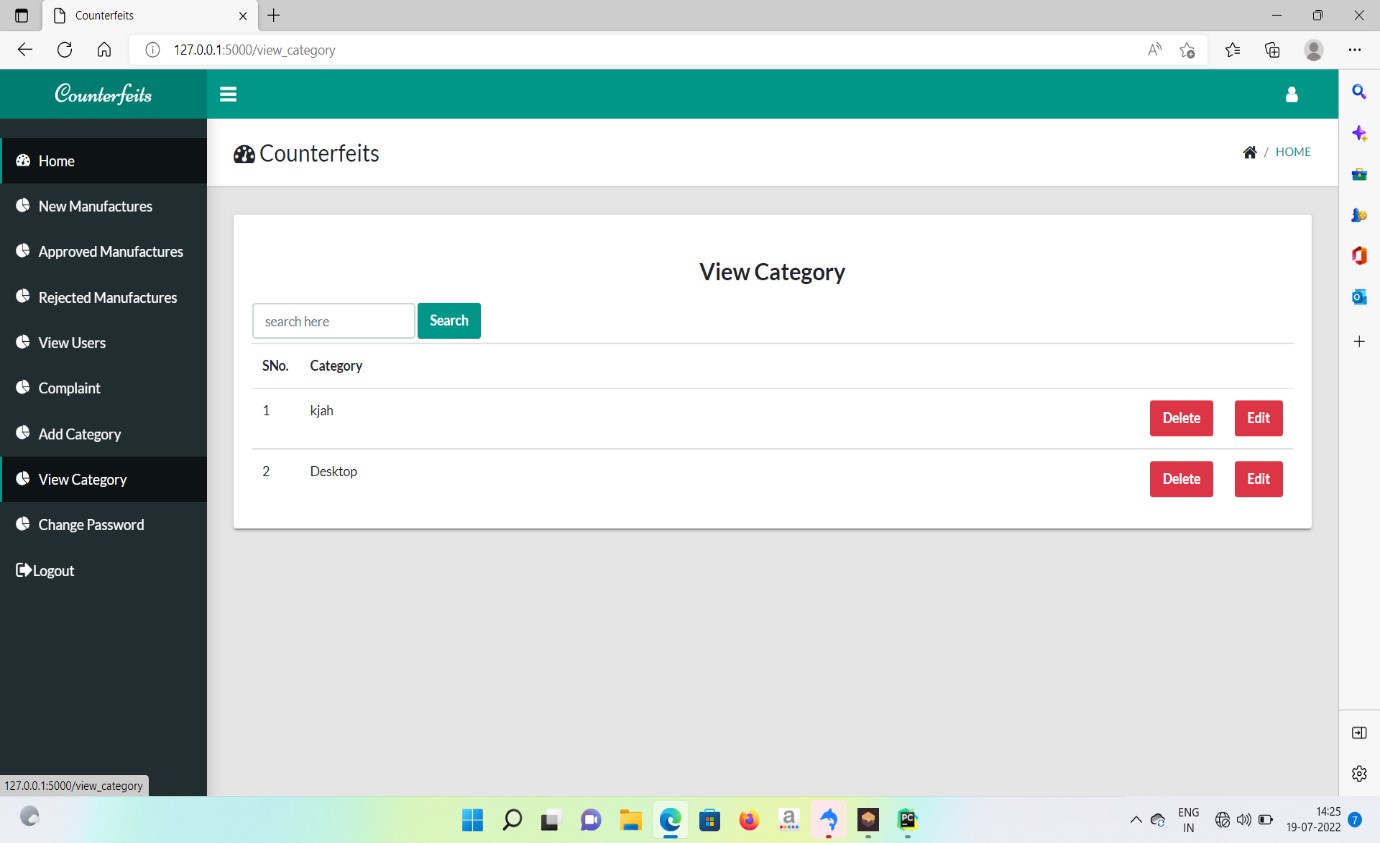
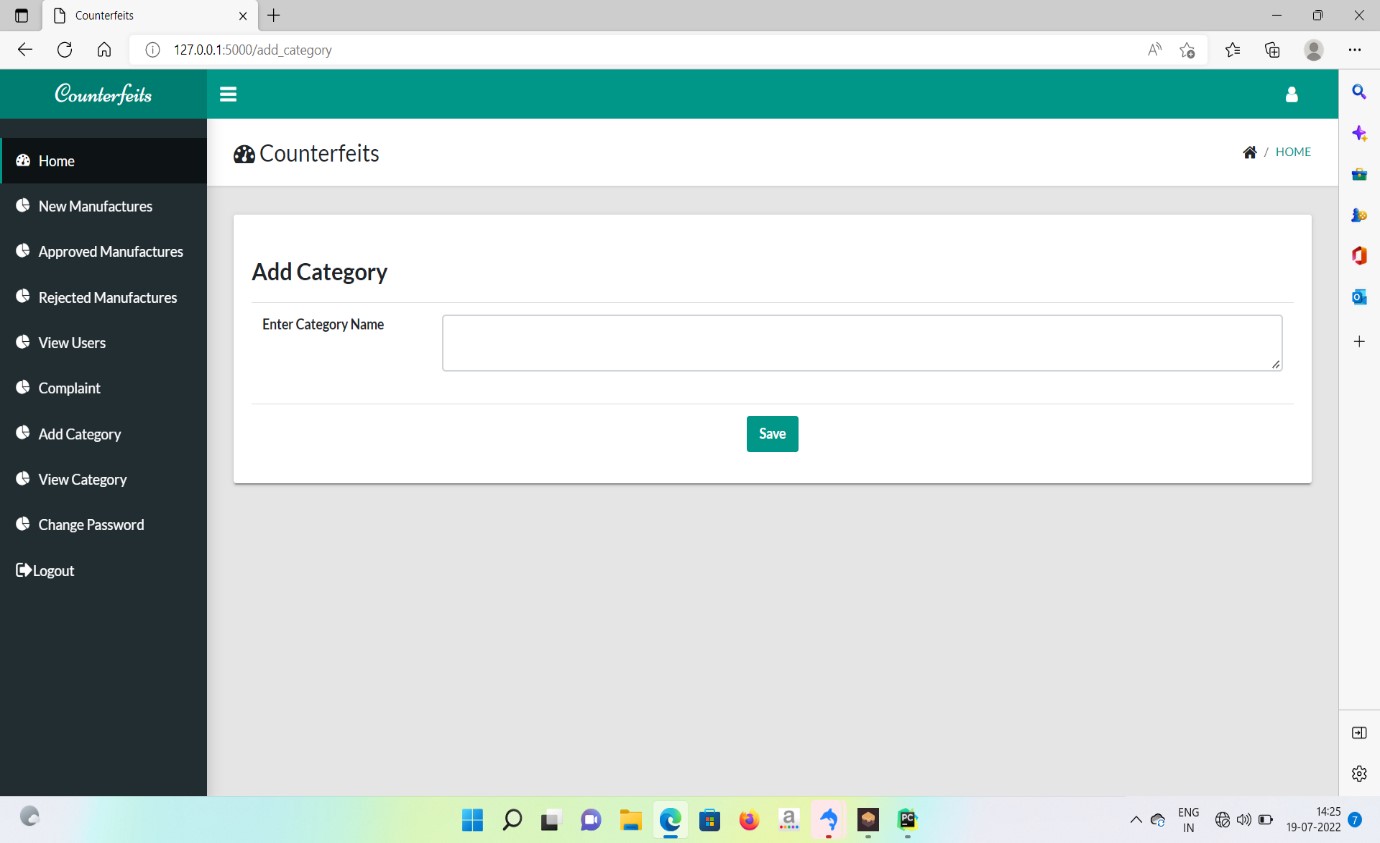
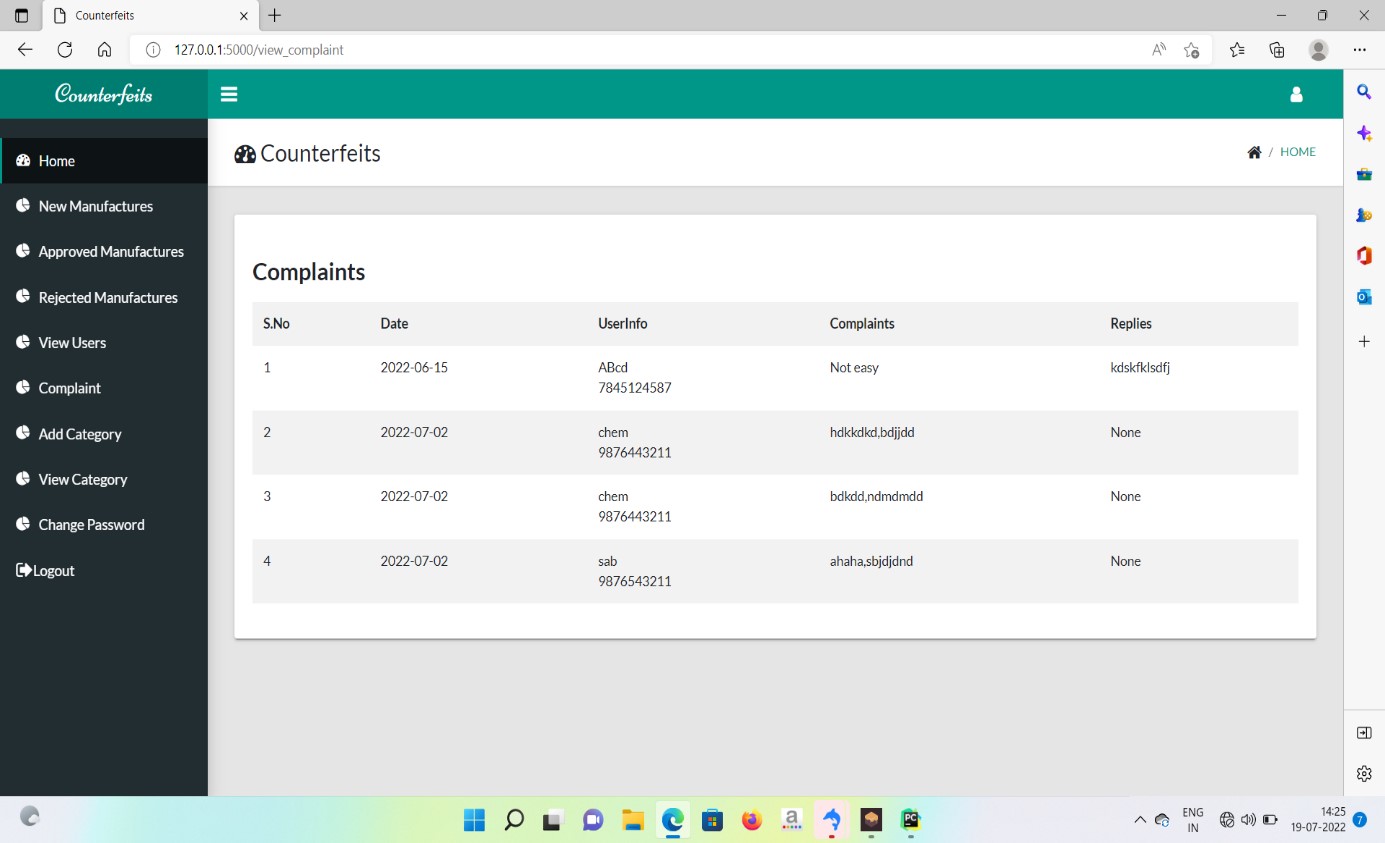
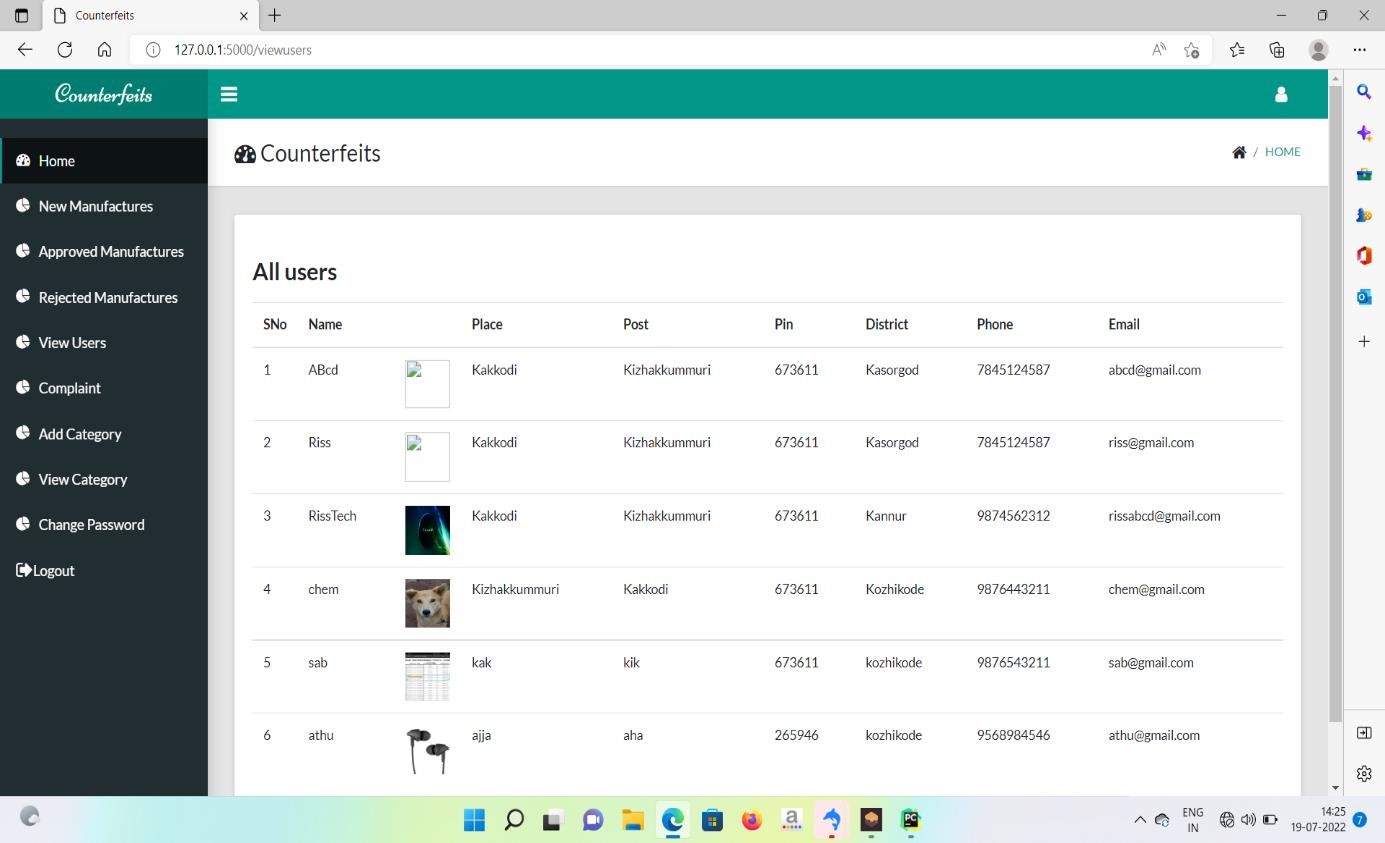
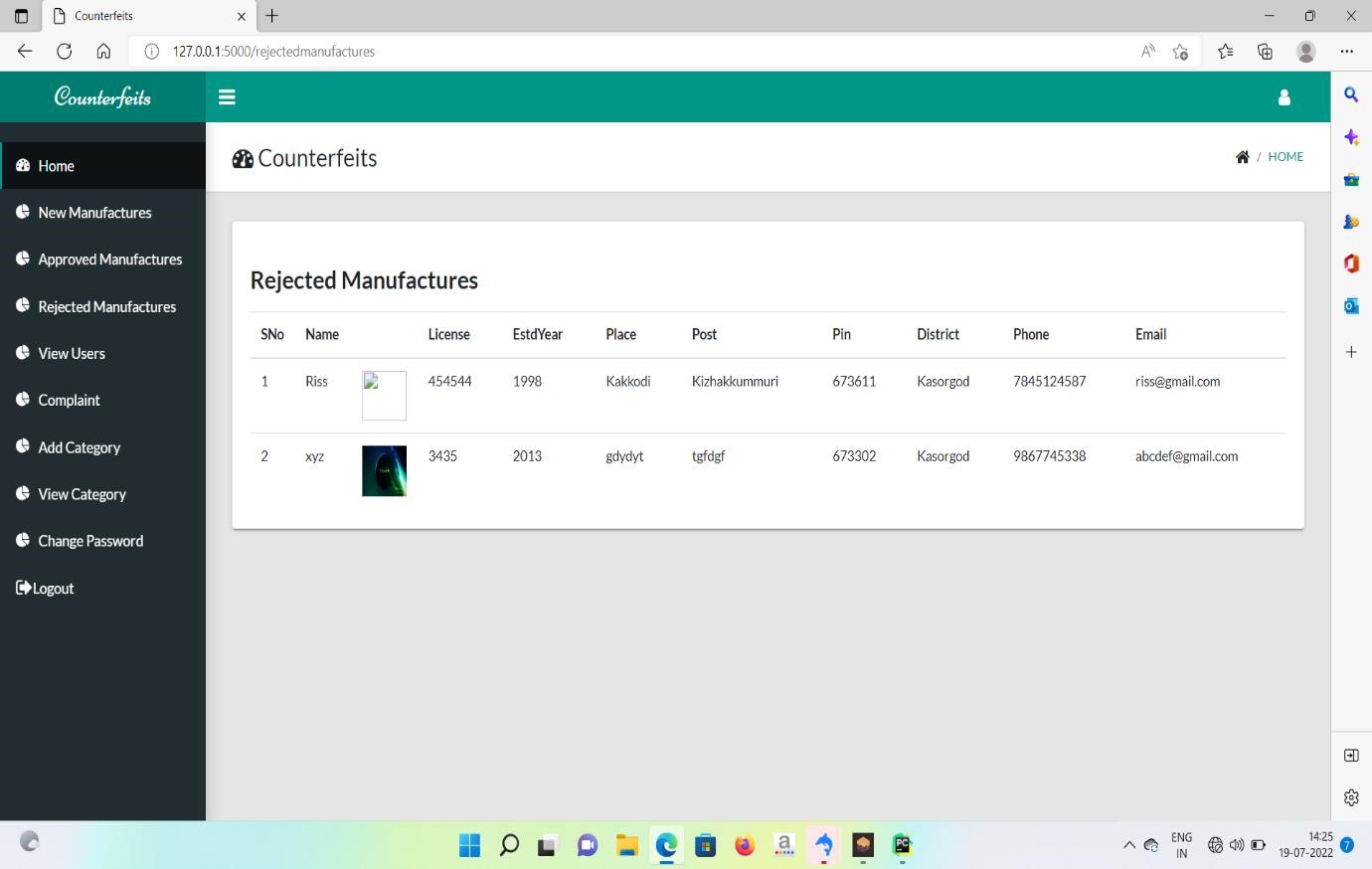
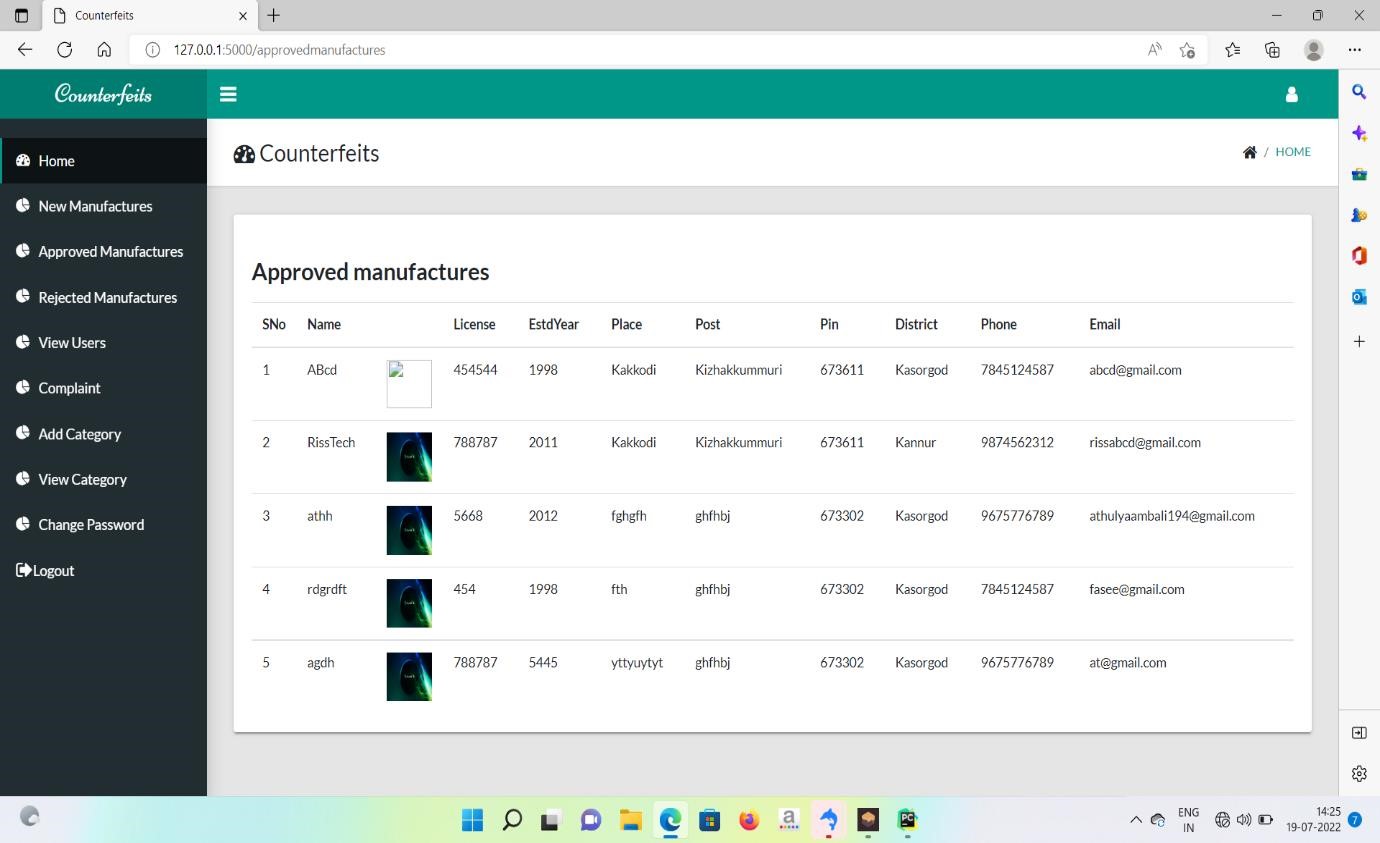
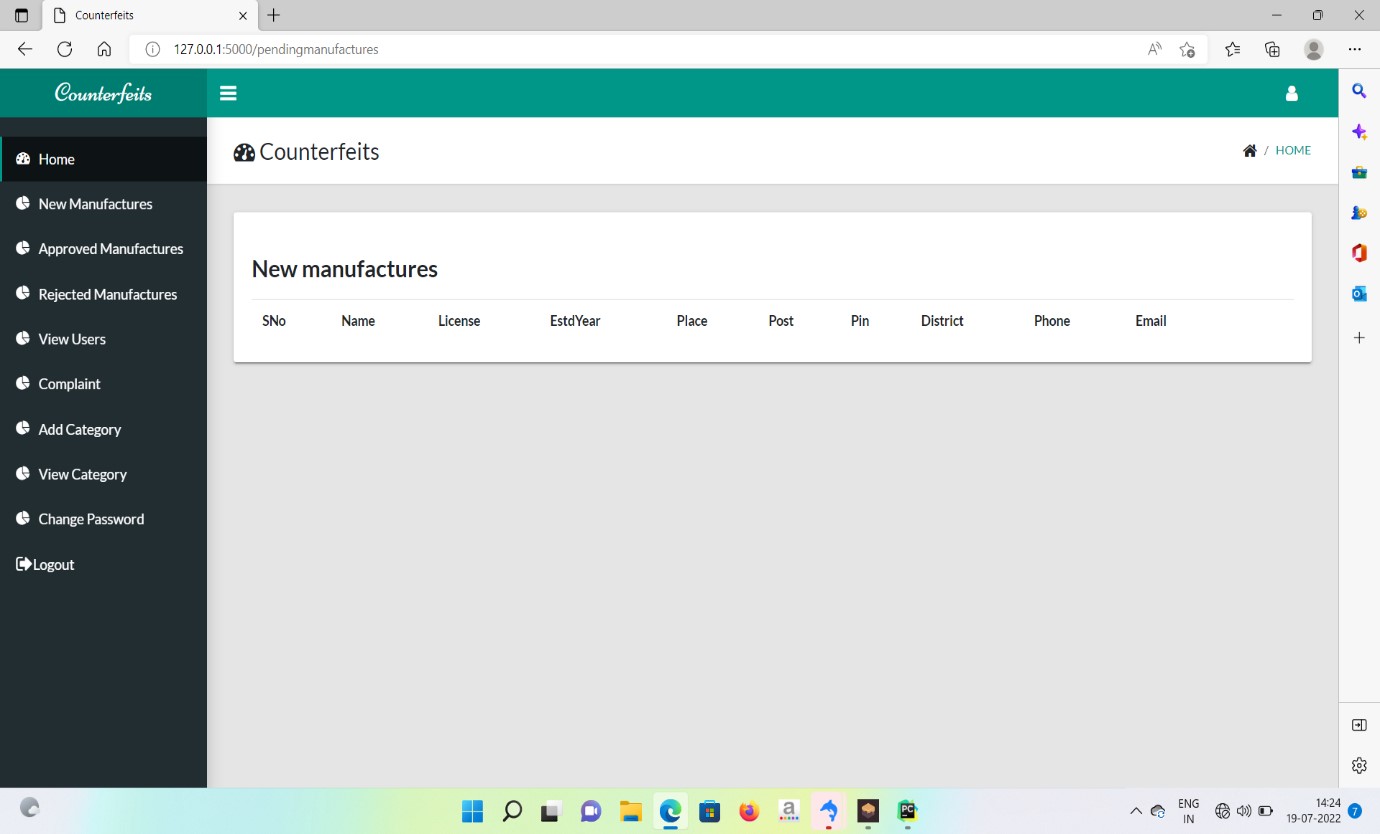
**HOME PAGE:**



**ADMIN PAGE**

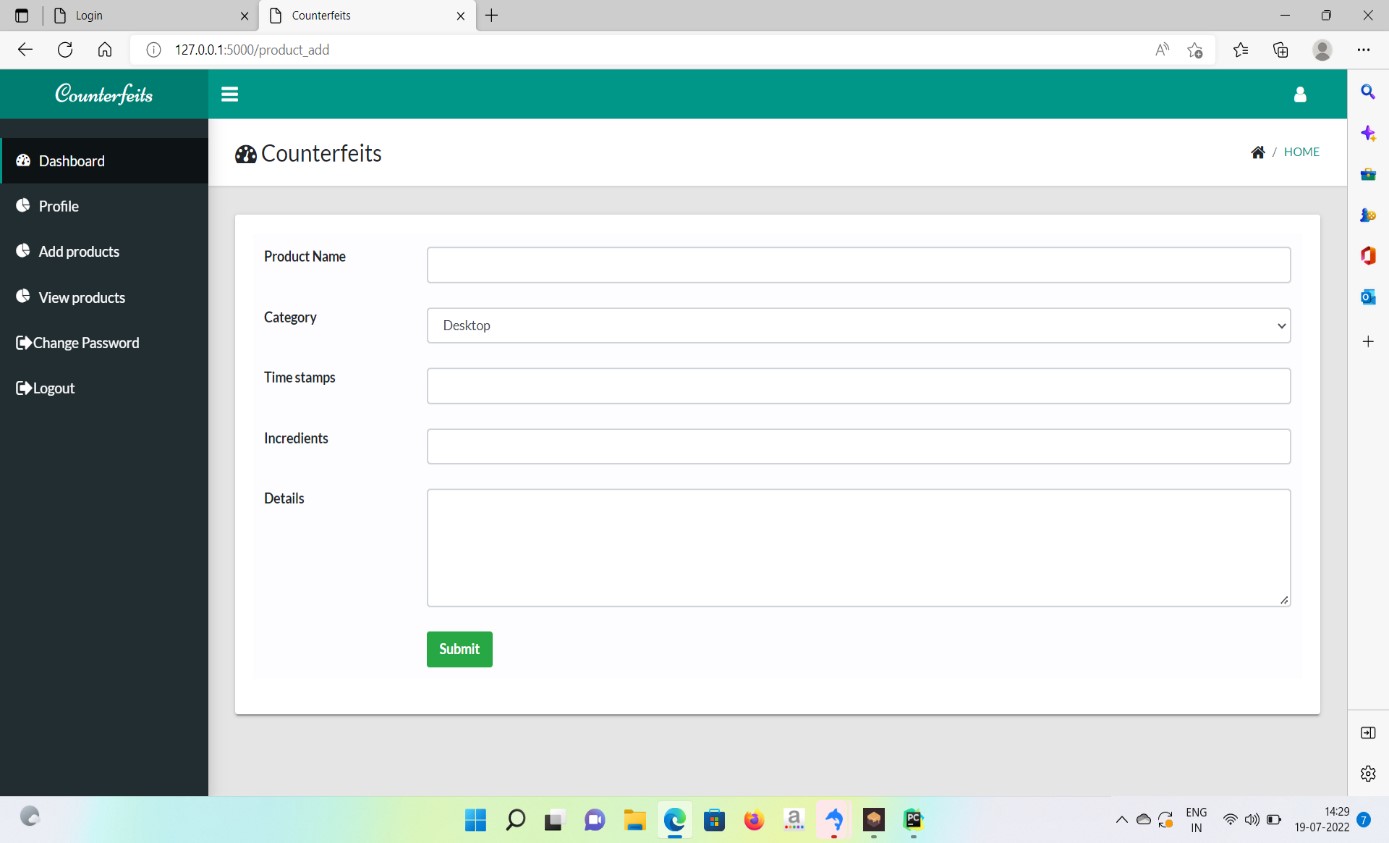
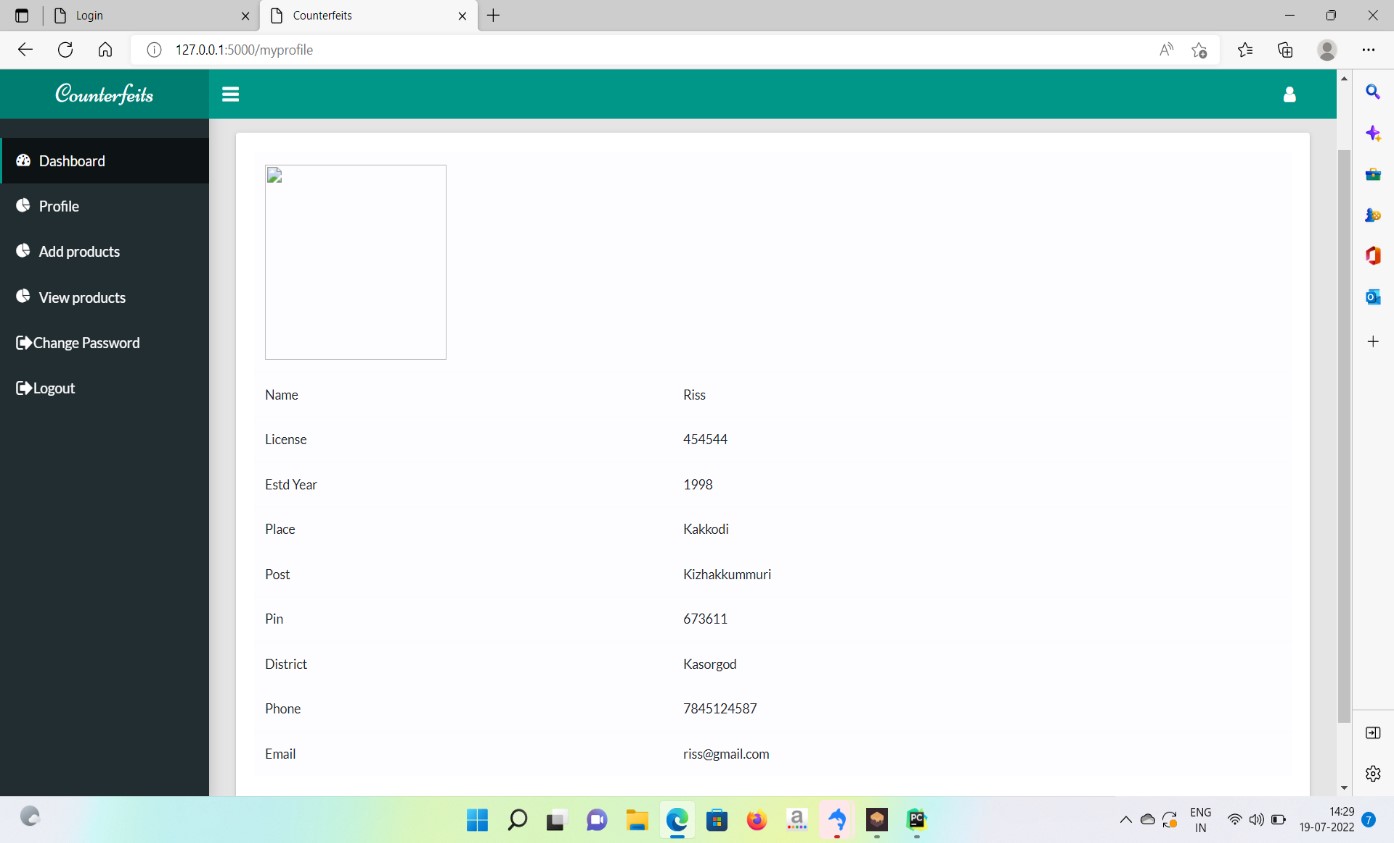
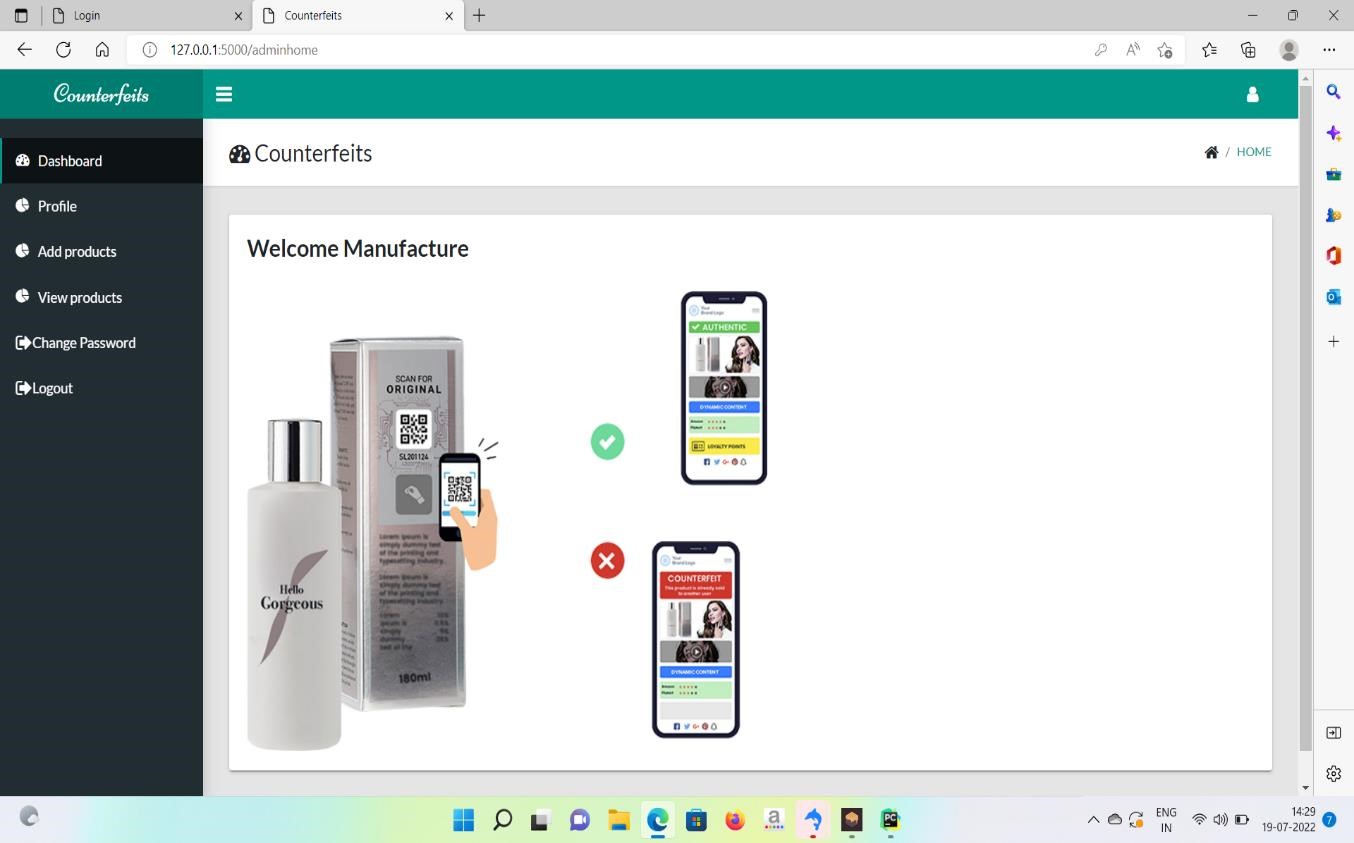
**SHOTS**

**:**

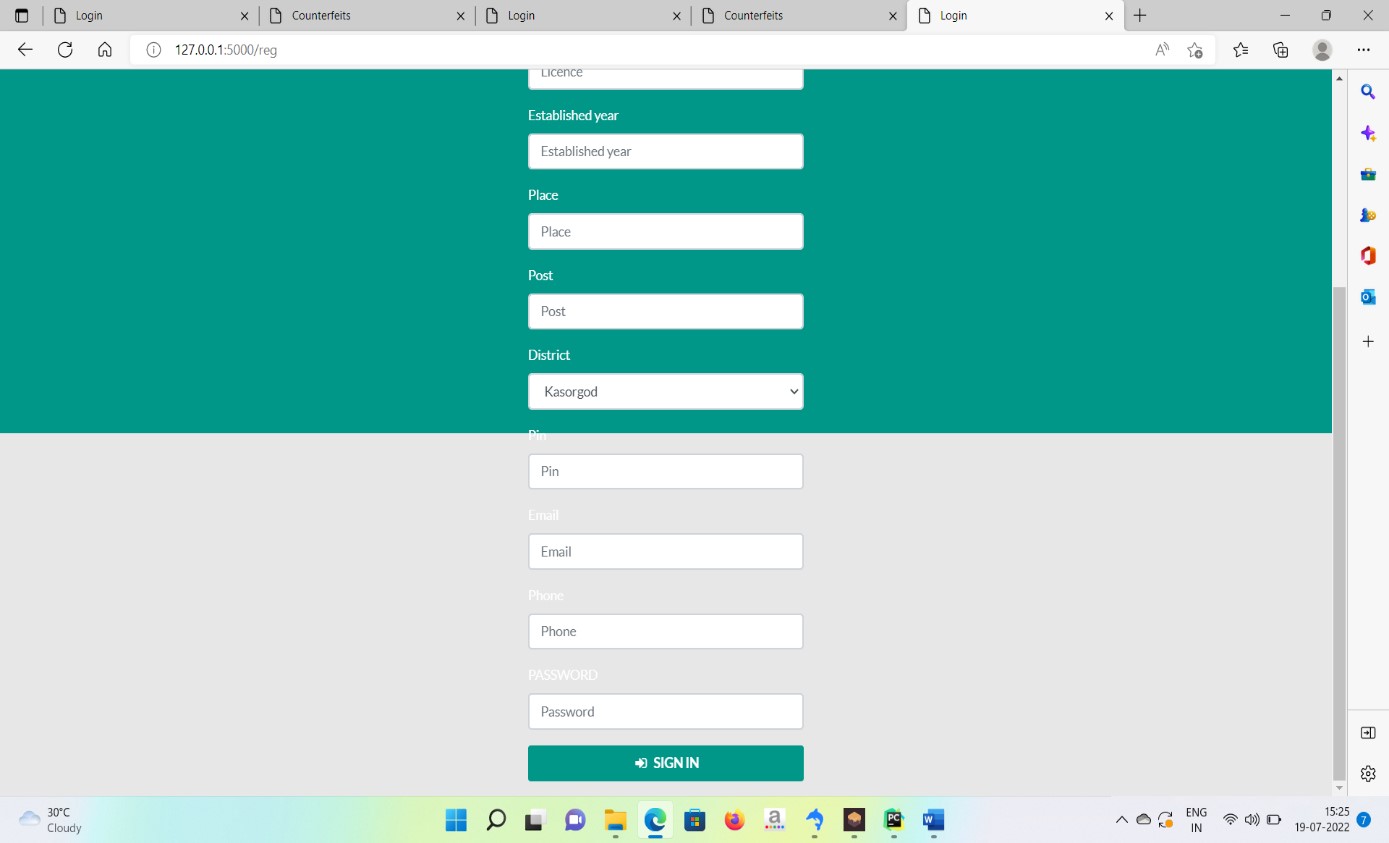
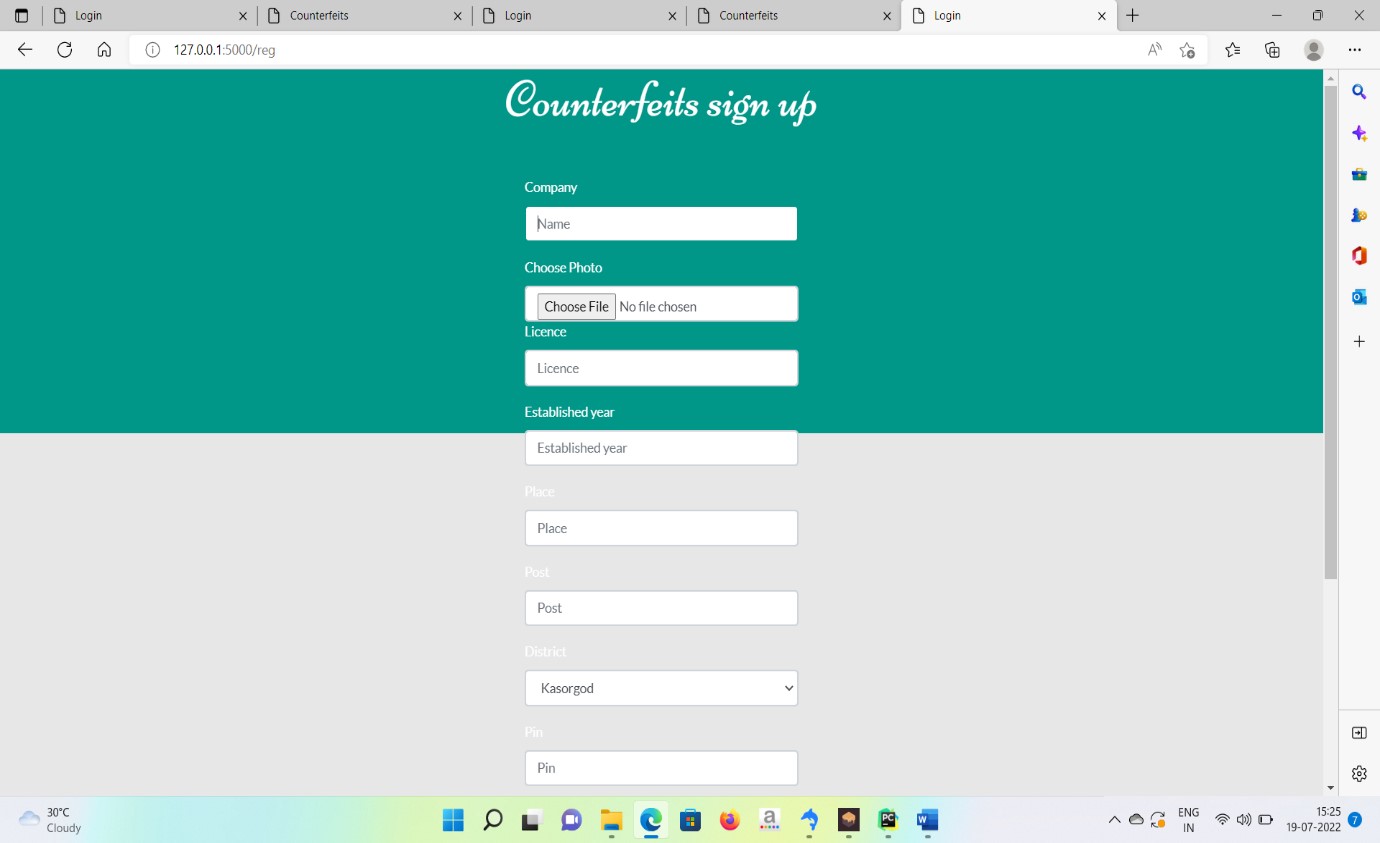
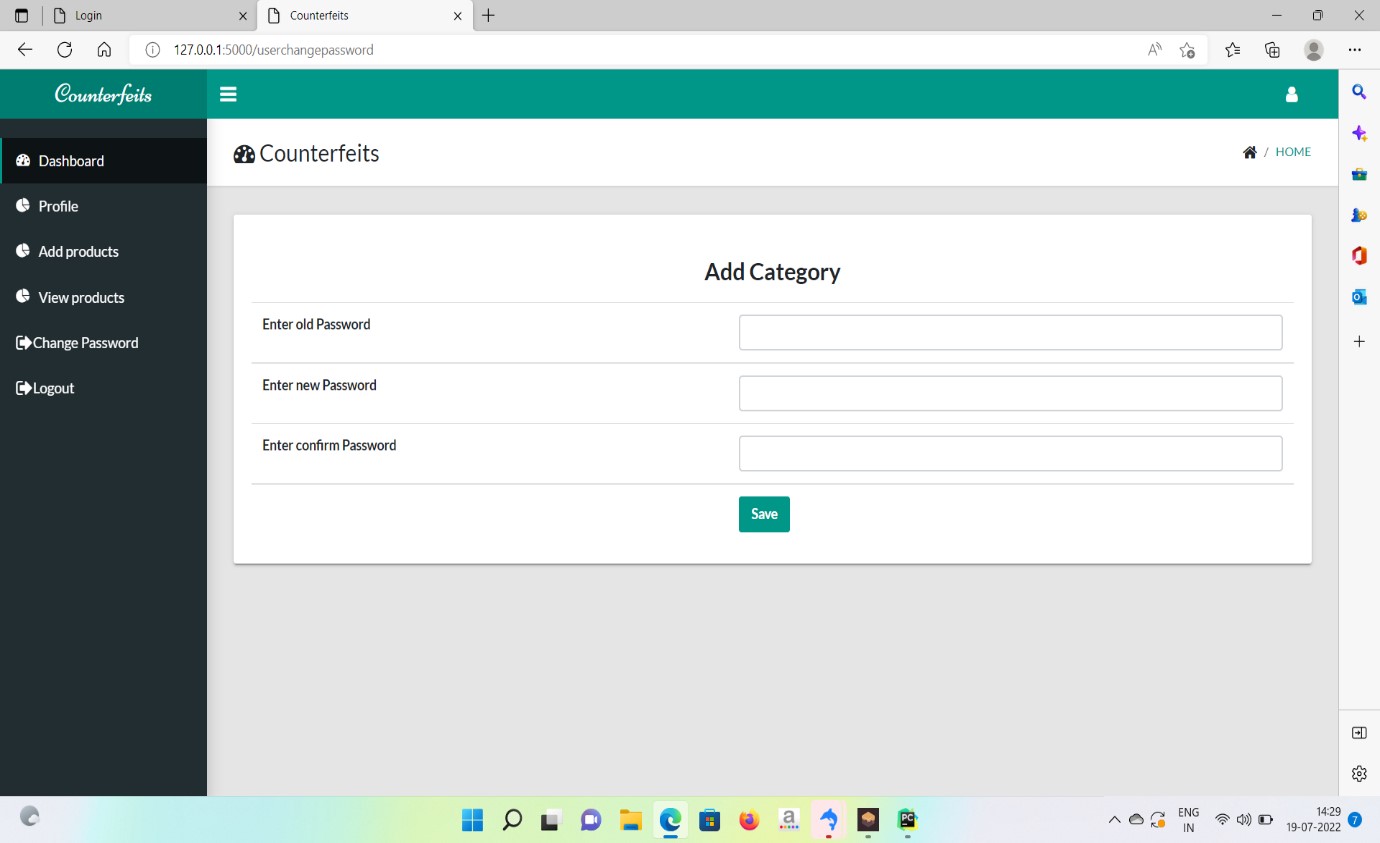


**MANUFACTURE HOME PAGE:**

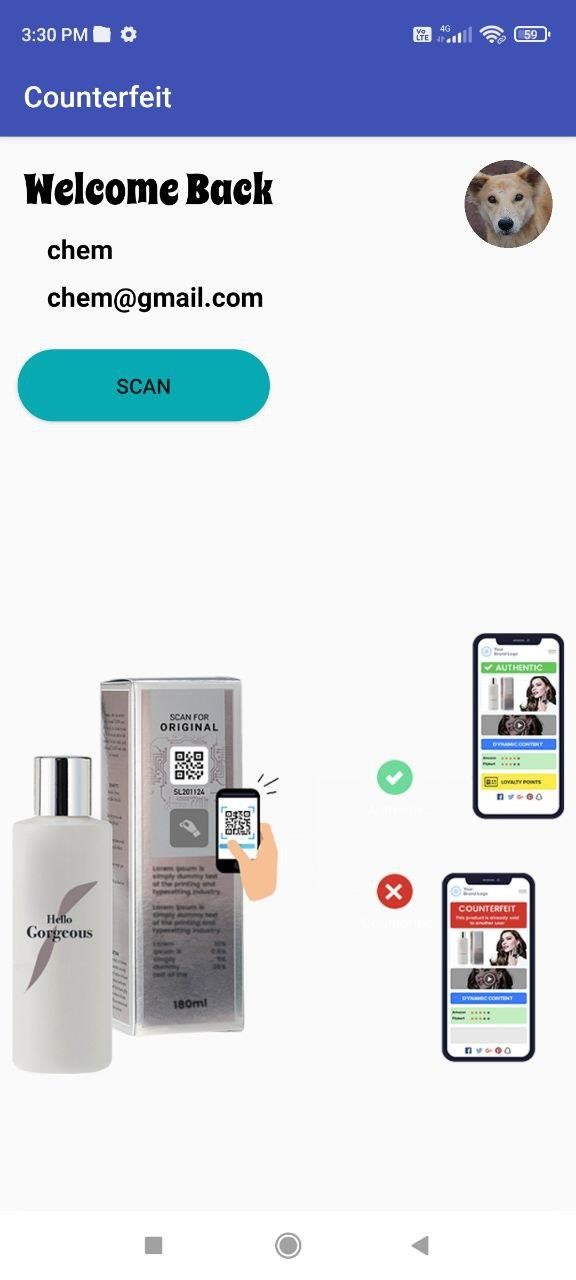
**MANUFACTURE PAGE SHOTS:**



**MANUFACTURE REGISTER:**



**USER HOME PAGE:**



**USER PAGE SHOTS:**

