INTRODUCTION

1. INTRODUCTION

In today's dynamic world, the seamless movement of goods has become essential, with individuals and businesses alike relying on the timely and secure delivery of items ranging from imported furniture and electronic devices to gifts and business goods. However, the existing transportation systems often employ manual methods, leading to a lack of efficient tracking mechanisms for shipped articles. Once items are dispatched, customers are left in the dark regarding the whereabouts of their shipments.
Recognizing this gap in the industry, E CONSIGNMENT emerges as a transformative online software solution tailored for cargo management professionals. This innovative system addresses the pressing need for automation in cargo activities, offering real-time tracking capabilities for dispatched articles. E-consignment not only streamlines the receiving and dispatching processes but also ensures that customers are kept informed with continuous tracking updates through service providers.

E CONSIGNMENT stands out as a comprehensive online software solution specifically crafted to meet the demands of cargo management professionals. By incorporating automation into the cargo workflow, the system goes beyond mere transaction facilitation. With a primary focus on real-time tracking of dispatched articles, E CONSIGNMENT provides a robust platform for cargo management, enabling professionals to efficiently manage the entire logistics process from source to destination. The software's capability to furnish continuous updates to service providers ensures a transparent and reliable communication channel, revolutionizing the way cargo activities are conducted in the modern era.

SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

2.1 Existing System

The current courier service system operates within a centralized framework where individuals or businesses connect with courier service providers through established platforms to send or receive packages. These platforms act as intermediaries, managing the logistics of package delivery and charging fees for their services. Users, including both senders and recipients, rely on these platforms to initiate and track shipments, with the expectation of safe and timely deliveries. However, a significant drawback of the existing courier service system is the lack of real-time tracking capabilities. Unlike more advanced systems, the current framework often fails to provide users with comprehensive, up-to-the-minute tracking information throughout the entire transit process. This absence of detailed tracking features introduces uncertainties for senders and recipients, who may be left in the dark about the exact location and status of their packages.

This deficiency becomes particularly problematic in scenarios where precise tracking is crucial, such as time-sensitive deliveries or shipments of valuable items. Without real-time tracking, users are unable to monitor the progress of their shipments in a granular manner, leading to increased anxiety and potential disruptions. Moreover, the limited tracking functionality hampers the system's ability to address issues promptly, such as delays, losses, or damages during transit. The lack of real-time insights makes it challenging to pinpoint when and where a problem occurred, hindering effective problem resolution for both users and courier service providers.

2.2 Proposed System

The proposed E-Consignment system seeks to transform the conventional courier service model by introducing a decentralized online software solution that addresses the limitations of the existing system. By leveraging cutting-edge technology, E-Consignment enhances the efficiency, transparency, and security of cargo activities. Unlike the centralized approach of the current system, E-Consignment incorporates real-time tracking features, providing users with instant and accurate visibility into the status and location of their shipments. The system's automation streamlines the cargo process, eliminating manual interventions, reducing errors, and ensuring timely deliveries. Utilizing blockchain technology, E-Consignment establishes a secure and transparent platform, maintaining an immutable record of all transactions. This approach not only overcomes the lack of real-time tracking in the existing system but also offers increased flexibility and global reach, allowing for a broader range of cargo projects to be funded and managed. In essence, E-Consignment revolutionizes cargo management, creating a reliable, transparent, and user-friendly environment for professionals in the logistics industry.

2.3 Module Description

This project has 2 modules:

ADMIN

- Login
- Port office management
- Clerk management
- View customers
- View and track consignment
- View delivered consignment
- Change Password

CLERK

- Login
- View Profile

- Add consignment
- Update status
- Tracking consignment
- Add Profit
- View delivered consignment
- View consignment report
- View delivered bill report
- Change password

2.4 Sprint

Sprint 1

Module	Task	Pending Task of Any	Hours For Completion	Expected Date of Completion	Actual Date of Completion	Reason For Deviation
Admin	Login		2 hours	17/09/2023	17/09/2023	
	Portoffice Management		4 hours	21/09/2023	21/09/2023	
	Clerk Management		5 hours	01/10/2023	03/10/2023	
	View Consignment		5 hours	03/10/2023	03/10/2023	
	View Customer		4 hours	15/10/2023	16/10/2023	
	View Delivered		5 hours	16/10/2023	17/10/2023	
	Change Password		3 hours	17/10/2023	22/10/2023	
	Logout		2 hours	17/10/2023	22/10/2023	

Sprint 2

Module	Task	Pending Task of Any	Hours For Completion	Expected Date of Completion	Actual Date of Completion	Reason For Deviation
	Manage Profile		2 hours	22/10/2023	24/10/2023	
	Customer Management		6 hours	24/10/2023	24/10/2023	
	Consignment Management		8 hours	25/10/2023	26/10/2023	
	Billing		3 hours	26/10/2023	26/10/2023	
	Update Status		5 hours	27/10/2023	27/10/2023	
	Report		5 hours	04/11/2023	04/11/2023	

2.5 User Stories

In the E-Consignment web application, there are two primary modules: Admin and Clerk. The Admin module focuses on efficient management of Port offices and Clerks within the system. The Admin is empowered with functionalities to add, edit, or remove Port offices, as well as oversee user accounts for Clerks. Additionally, the Admin seeks real-time updates on transactions within the Port offices, ensuring a comprehensive understanding of consignment movements.

The Clerk module, on the other hand, caters to the operational aspects of the courier service platform. Clerks can securely log in to the E-Consignment system and play a pivotal role in customer management by adding new customers to the database. This simplifies the parcel sending process, requiring the Clerk to generate accurate and transparent bills for each transaction. The Clerk is also responsible for updating consignment statuses upon their arrival at the assigned port office, enabling customers to conveniently track their parcels. Furthermore, the E-Consignment system provides the advantage of automatic report generation on the Clerk's web page, eliminating the need for manual reporting and ensuring a streamlined, digital approach to record-keeping.

FEASIBILITY STUDY

3. FEASIBILITY STUDY

An analysis of the ability to complete a project successfully, taking into account legal, economic, technological, scheduling, and other factors is considered a feasibility study. Rather than just diving into a project and hoping for the best, feasibility study allows project managers to investigate the possible negative and positive outcomes of a project before investing too much money and time.

3.1 Economic Feasibility

The economic analysis is done to determine the benefits and savings that are expected from the candidate system and compare them with costs. Thus, coming to a conclusion on whether the system is economically feasible or not. This system is cost effective as well as time effective, thereby making it economically feasible. This study presents tangible and intangible benefits from the project by comparing the developments and operational costs. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility.

3.2 Technical Feasibility

The technical requirements for the system are economic and it does not use additional software. That is whether the system can be implemented using the existing technologies or not. This application is developed using python, whose development kits are easily available and free of cost, thus making our system technically feasible.

3.3 Operational Feasibility

The system working is quite easy to use and learn due to its simple but attractive interface. Users require no special training for operating the system. Technical performance includes issues such as whether the system can be organized so that it always delivers information at the right place and on time using internet services. Acceptance revolves around the current system and its personnel.

3.4 Behavioral Feasibility

This analysis involves how it will work when it is installed and the assessment of the political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to facilitate change. The new proposed system is very much useful to the users and therefore it will accept a broad audience.

3.5 Software Feasibility

Even though this application is developed in a very high software environment, it is also supported by many other environments with minimal changes. The system is fully feasible to be executed on any kind of operating systems and browsers.

3.6 Hardware Feasibility

Software can be developed with the existing resources. But the existing resources may or may not be used to produce hardware. If no hardware is newly bought for a project, then software is said to achieve hardware feasibility. The system is hardware-wise feasible because it needs absolutely no new hardware.

SOFTWARE ENGINEERING PARADIGM

4. SOFTWARE ENGINEERING PARADIGM

The software engineering paradigm which is also referred to as a software process model or Software Development Life Cycle (SDLC) model is the development strategy that encompasses the process, methods and tools. SDLC describes the period of time that starts with the software system being conceptualized.

4.1 Agile Model

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. At the end of the iteration, a working product is displayed to the customer and important stakeholders. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

4.2 Scrum

Scrum is an agile framework for managing knowledge work, with an emphasis on software development. It is designed for teams of three to nine members, who break their work into actions that can be completed within time boxed iterations, called "sprints", no longer than one month and most commonly two weeks, then track progress and re-plan in 15-minute stand-up meetings, called daily scrums. Scrum is an iterative and incremental framework for managing product development. It defines "a flexible, holistic product development strategy where a development team works as a unit to reach a common goal", challenges assumptions of the "traditional, sequential approach to product development, and enables teams to self organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication among all team members and disciplines involved.

SYSTEM REQUIREMENT SPECIFICATION

5. SYSTEM REQUIREMENT SPECIFICATION

5.1 Software Requirements

- Operating System: Windows 11
- Front End: HTML, CSS, JS
- Back End: MySQL
- Language Used: Python
- IDE: PyCharm
- Web browser: Internet Explorer/Google Chrome/Firefox
- Frame work: Django

5.2 Hardware Requirements

- Processor: Intel core i3 or above
- Ram: 8 GB
- Storage: 512 GB Hard Disk

SYSTEM DESIGN

6. SYSTEM DESIGN

System design is the first in the development phase for many engineered products or systems. It may define the process of applying various techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

6.1 Database Design

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data.

In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system. The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must: Determine the relationships between the different data elements and superimpose a logical structure upon the data on the basis of these relationships.

Normalization

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy i.e., repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insertion, updation, deletion anomalies.

Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this normal forms or rules for structuring relations are used.

Insertion Anomaly: Inability to add data to the database due to absence of other data.

Deletion Anomaly: Unintended loss of data due to deletion of other data.

Update Anomaly: Data inconsistency resulting from data redundancy and partial update.

Normal Forms: These are the rules for structuring relations that eliminate anomalies.

1. First Normal Form (1NF)

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. This means simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

2. Second Normal Form (2NF)

A relation is said to be in second Normal form if it is in first normal form and it should satisfy any one of the following rules.

- Primary key is a not a composite primary key
- No non key attributes are present
- Every non key attribute is fully functionally dependent on a full set of primary keys.

3. Third Normal Form (3NF)

A relation is said to be in third normal form if there exist no transitive dependencies.

Transitive Dependency:

If two non-key attributes depend on each other as well on the primary key then they are said to be transitively dependent. the above normalization principle was applied to decompose the data in multiple tables thereby making the data to be maintained in a consistent state.

6.2 Tables

LOGIN

TITLE	TYPE	WIDTH	CONSTRAINT
Login id	Bigint	20	Primary key
Username	Varchar	100	Not null
Password	Varchar	100	Not null
Type	Varchar	100	Not null

Port Office

TITLE	TYPE	WIDTH	CONSTRAINT
ID	Bigint	20	Primary key
Building number	Varchar	100	Not null
Building name	Varchar	100	Not null
Licence number	Varchar	100	Not null
Email id	Varchar	100	Not null
Mobile No	Varchar	100	Not null
Place	Varchar	100	Not null
City	Varchar	100	Not null
State	Varchar	100	Not null
Pin code	Varchar	100	Not null

CLERK

TITLE	TYPE	WIDTH	CONSTRAINT
ID	Bigint	5	Primary key

Name	varchar	100	Not null
Gender	varchar	100	Not null
Joined Date	varchar	100	Not null
House name	varchar	100	Not null
House number	varchar	100	Not null
Place	varchar	100	Not null
City	varchar	100	Not null
State	varchar	100	Not null
Pin	varchar	100	Not null
Email	varchar	100	Not null
Phone Number	varchar	100	Not null
Port office	varchar	100	Foreign key
Login ID	Bigint	20	Foreign key

CONSIGNMENT

TITLE	**TYPE**	**WIDTH**	**CONSTRAINT**
ID	Bigint	20	Primary key
Consignment number	varchar	100	Not null
Customer id	Bigint	20	Foreign key
Destination	varchar	100	Not null
House name	varchar	100	Not null
House number	varchar	100	Not null
Place	varchar	100	Not null
City	varchar	100	Not null
State	varchar	100	Not null
Pin code	varchar	100	Not null
Parcel narration	varchar	100	
Status	varchar	100	Not null

TRACK

TITLE	TYPE	WIDTH	CONSTRAINT
Track id	Big int	5	Primary key
Clerk id	Big int	20	Not null
Consignment	varchar	20	Foreign key
Port office	varchar	20	Foreign key
Date	date		Not null
Time	varchar	100	Not null
Status	varchar	20	Not null

CUSTOMER

TITLE	TYPE	WIDTH	CONSTRAINT
ID	Bigint	20	Primary key
Customer name	varchar	100	Not null
Phone number	varchar	100	Not null
House name	varchar	100	Not null
Place	varchar	100	Not null
City	varchar	100	Not null
State	varchar	100	Not null
Pin	varchar	100	Not null
Clerk id	Big int	20	Not null

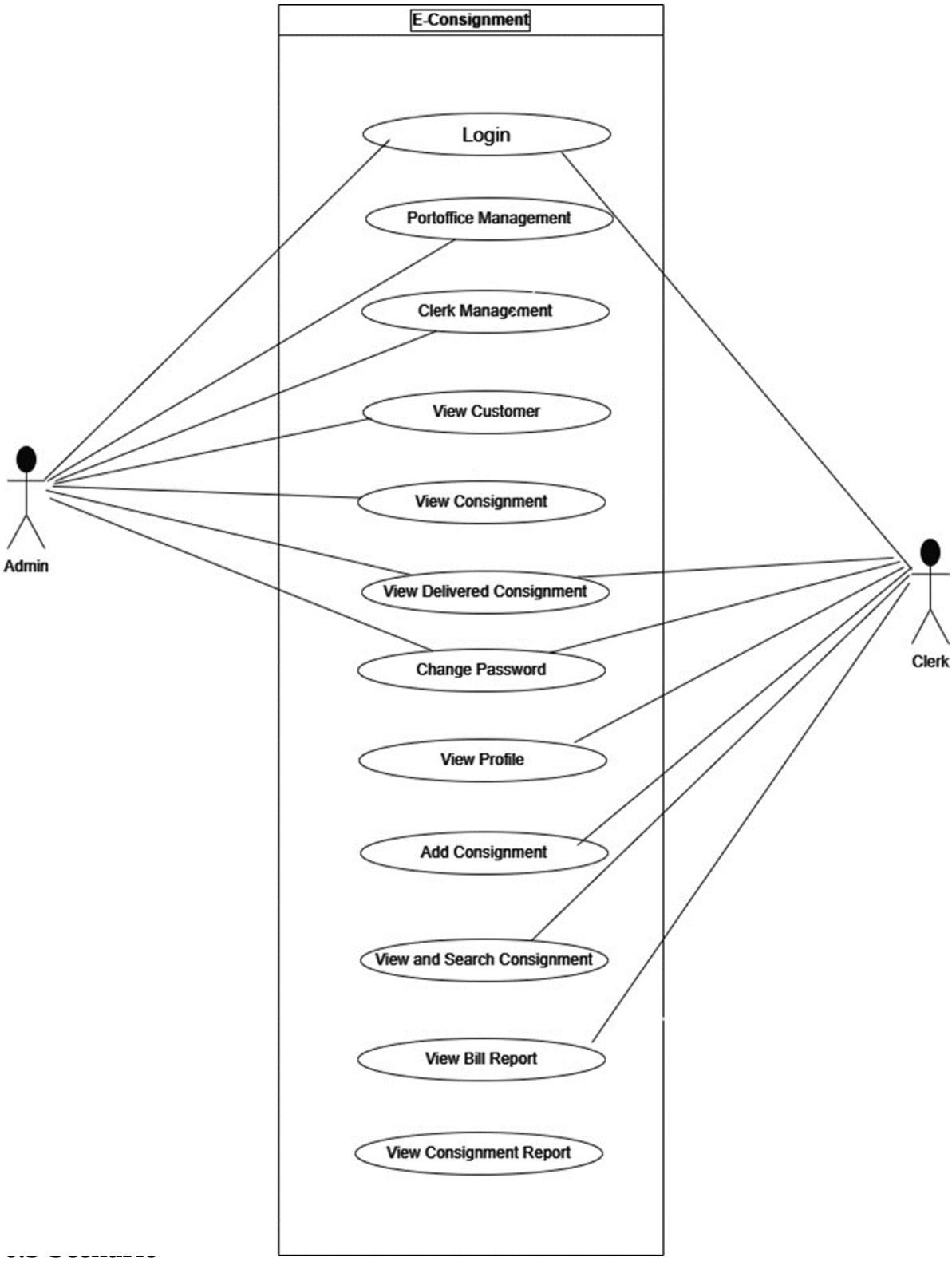
BILLING

TITLE	TYPE	WIDTH	CONSTRAINT
ID	Bigint	20	Primary key
Consignment id	Bigint	20	Foreign key
Amount	varchar	100	Not null
Date	date		Not null

6.3 UML Design

The Unified Modelling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of the software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing object oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

6.4 Use Case Diagram



ADMIN

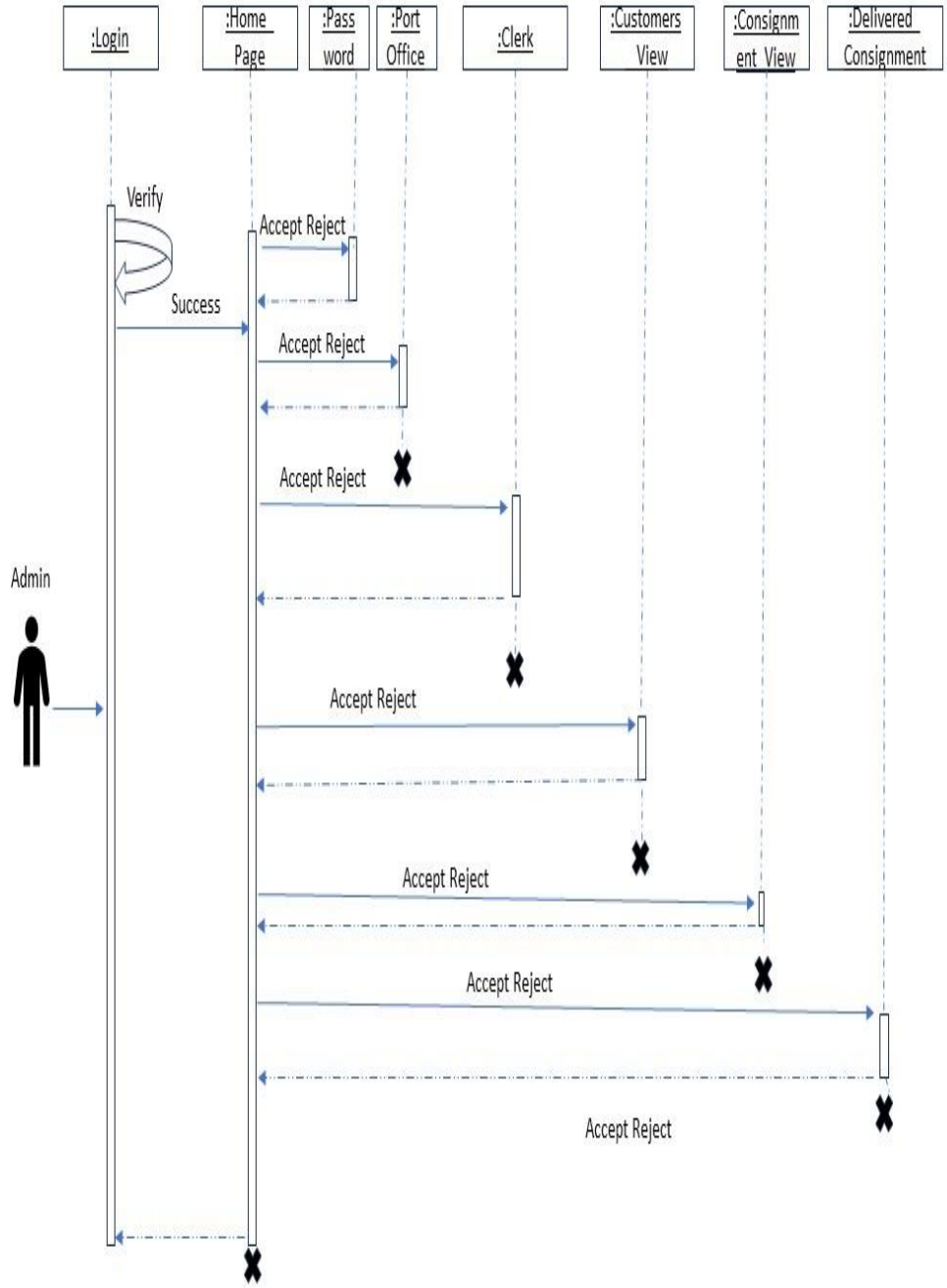
- Can login
- Can add, view, edit, delete Port office
- Can add, view, edit, delete Clerk
- Can view Customers
- Can View Consignments
- Can change password

CLERK

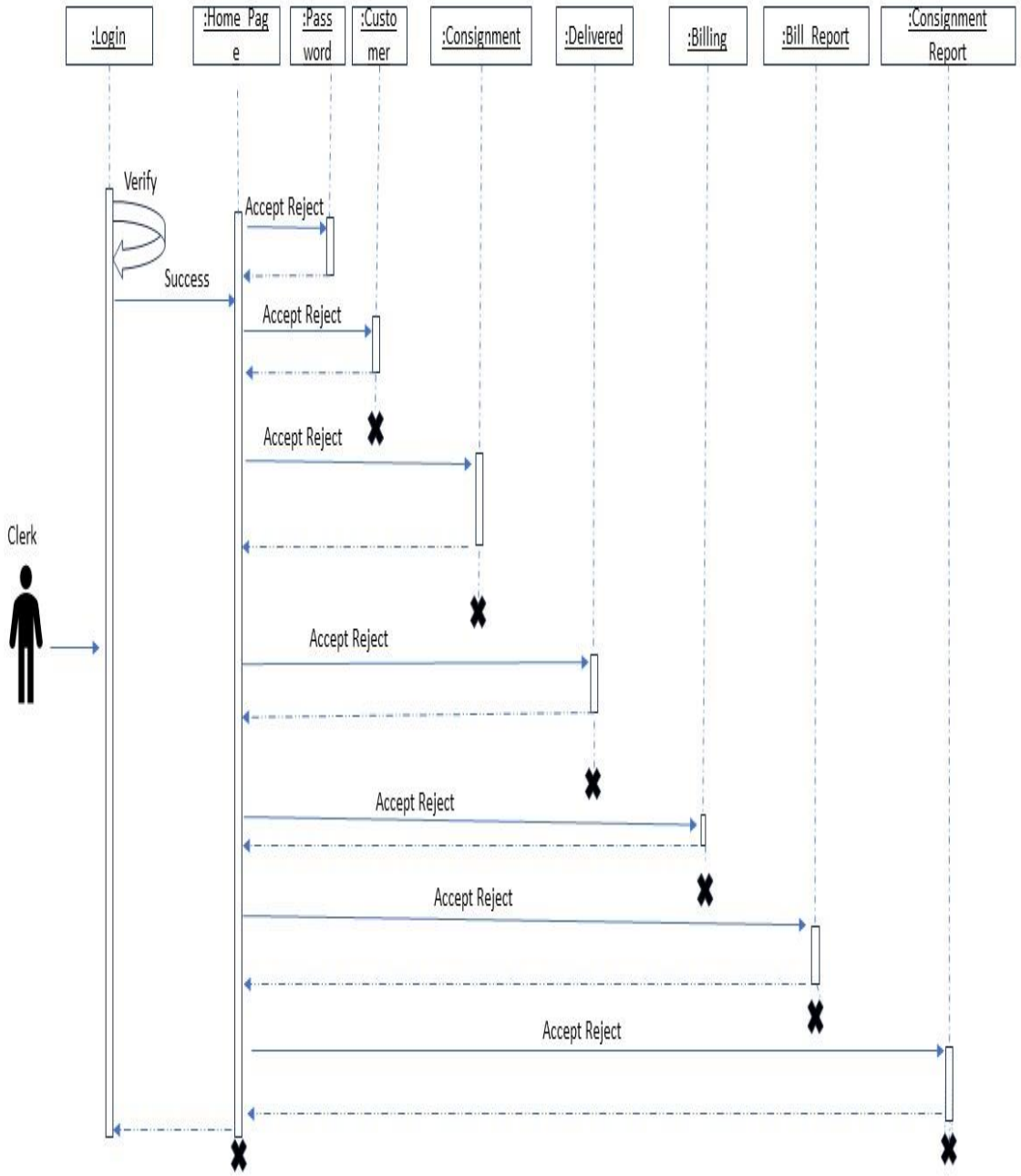
- Can login View Loan Details
- Can add, view, edit, delete Customer
- Can add, view, edit, delete, track Consignment
- Can bill Consignment
- Can update status of arrived parcel
- Can view Consignment report
- Can view bill report
- Can change password
- Can manage Expenses

6.5 Sequence Diagram

ADMIN



USER


:Login
:Home Pag e
:Pass word
:Custo mer
:Consignment
:Delivered
:Billing
:Bill Report
:Consignment Report
Clerk
Verify
Success
Accept Reject
Accept Reject
Accept Reject
Accept Reject
Accept Reject
Accept Reject
Accept Reject
Accept Reject

SYSTEM DEVELOPMENT

7. SYSTEM DEVELOPMENT

System development is a series of operations to manipulate data to produce output from a computer system. The principal activities performed during the development phase can be divided into two major related sequences.

- External system development
- Internal system development

The major external system activities are:

- Implementation
- Planning
- Equipment acquisition
- Installation

7.1 Coding

The purpose of code is to facilitate the identification and retrieval of items of information. A code is an ordered collection of symbols designed to provide unique identification of an entity or an attribute. Code also shows interrelationship among different items.

Python

Python is a widely used high-level programming language for general purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasises code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.

Libraries

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom

numbers, arithmetic with arbitrary precision decimals, manipulating regular expressions, and unit testing.

Django

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.

MYSQL Client

Licence GPL Platforms OS Independent Python versions Python 2.7 and 3.4+ PyPI. https://pypi.org/project/mysqlclient/ MySQL client is a fork of MySQL python. It adds Python 3 support and fixes many bugs. It is the MySQL library that is recommended by the Django documentation

SYSTEM TESTING AND IMPLEMENTATION

8. SYSTEM TESTING AND IMPLEMENTATION

Testing is vital to the success of the system. It makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved in this project. It is the stage of implementation, which ensures that the system works accurately and effectively before the live operation commences. It is a confirmation that all are correct and an opportunity to show users that the system must be tested and show that the system will operate successfully and produce expected results under expected conditions.

8.1 Types of Testing

Different types of testing are:

1. Unit testing
2. Integration testing
3. System testing
4. Validation testing
5. User acceptance testing

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases. All modules were tested individually as soon as they were completed and were checked for their correct functionality. Unit testing deals with testing a unit as a whole.

Integration Testing

Integration testing is a systematic technique for testing to overcome the errors associated within the interface. In this System all the modules such as login, registration, admin in web, user in web, and the final hardware and software are combined and then the entire program is tested as a whole. Thus, in the integration testing step all the errors in the

implementation of the system are corrected. The objective is to take unit tested modules and to combine them and test it as a whole.

System Testing

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it doesn't produce the required data in the specific format. The output displayed or generated by the system under consideration is tested by, asking the user about the format displayed. The output format on the screen is found to be correct as the format was designed in the system phase according to the user needs. Hence the output testing doesn't result in any correction in the system.

User Acceptance Testing

User Acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user Acceptance by constantly keeping in touch with prospective systems at the time of developing and making changes wherever required.

8.2 Implementation

Implementation is the stage of a project, when theoretical design is turned into a working system. The most crucial stage is achieving a successful system and confidence that the new system will work effectively. It involves careful planning, investigation of the manual system and a new system. Implementation means converting a new or revised system design into an operational one. The implementation includes all those activities that take place to convert from the old system to the new one.

There are several activities involved while implementing a project:

1. Careful planning.
2. Investigating the current system and its constraints on implementation.
3. Design of methods to achieve the changeover.
4. Training of the staff in the changeover procedure and evaluation of change over method

SYSTEM MAINTENANCE

9. SYSTEM MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited. There are three types of maintenance:

- Corrective (Fixing bugs/errors)
- Adaptive (Updates due to environment changes)
- Perfective (Enhancements, requirements changes)

Maintenance is an enigma of the system development. The definition of the software maintenance can be given describing four activities that are undertaken after the program is released for use. The maintenance activity occurs since it is unreasonable to assume that software testing will uncover all in a large system. The second activity that contributes to the definition of maintenance occurs since rapid changes are encountered in every aspect of computing. The third activity involves recommendation for new capabilities, modification to the existing functions and general enhancements when the software is used. The fourth maintenance activity occurs when software is changed to improve future maintainability or reliability.

FUTURE ENHANCEMENT

10. FUTURE ENHANCEMENT

The proposed system exhibits certain drawbacks. Firstly, there is a notable deficiency in providing customers with comprehensive parcel tracking, leading to ambiguity regarding the location of their shipments. Additionally, the absence of a global container booking system restricts our capacity to facilitate smooth transactions on an international scale. Furthermore, the current system lacks the convenience of QR code functionality for efficient parcel management. In response to these limitations, forthcoming enhancements are designed to address these issues, ensuring a more inclusive and user-friendly courier service experience.

CONCLUSION

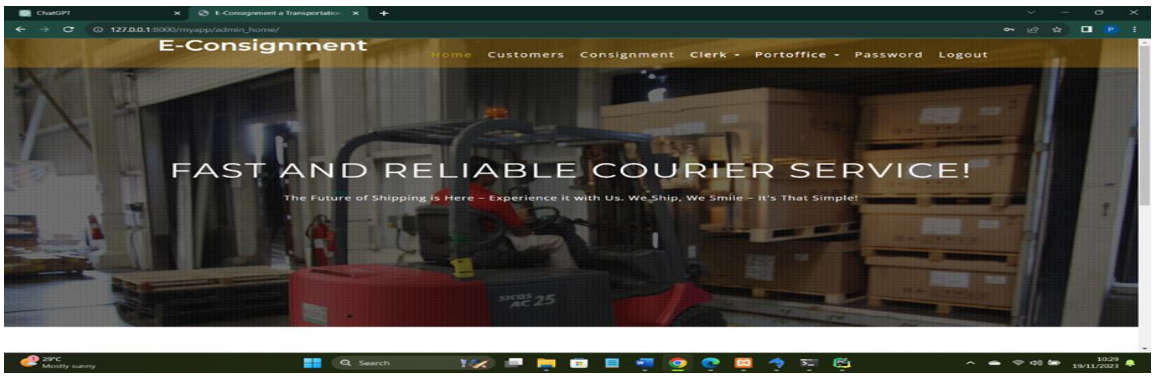
11. CONCLUSION

E CONSIGNMENT website serves as a comprehensive solution for computerized courier management, encompassing key functionalities such as customer management, billing, and tracking. The computerized courier management system ensures operational efficiency and accuracy in handling shipments. The billing feature streamlines financial processes, and the tracking system keeps users informed about the real-time status of their parcels. The platform excels in delivering a reliable, user-friendly, and technologically sophisticated solution for courier services.

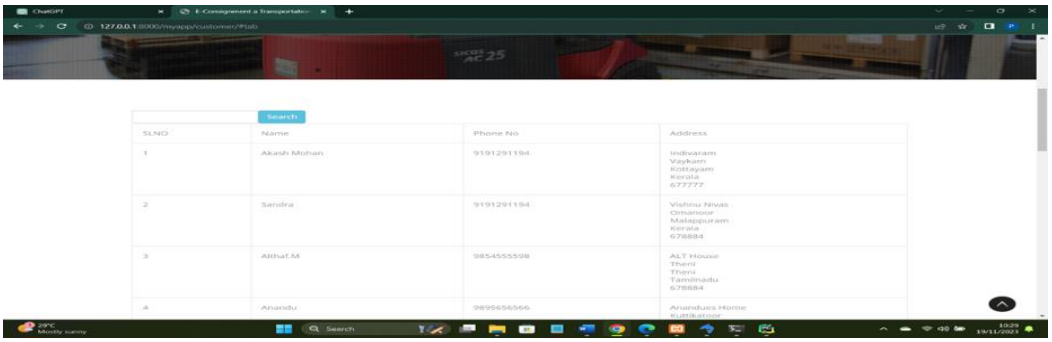
APPENDIX

Admin Login

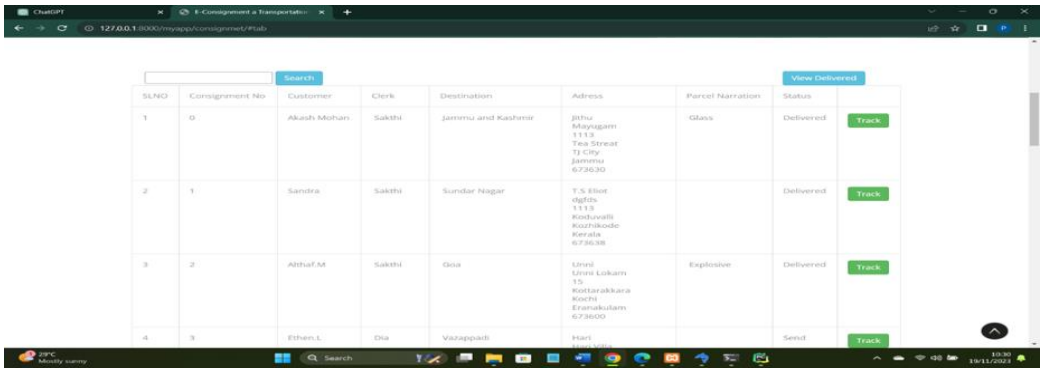
Admin Home page



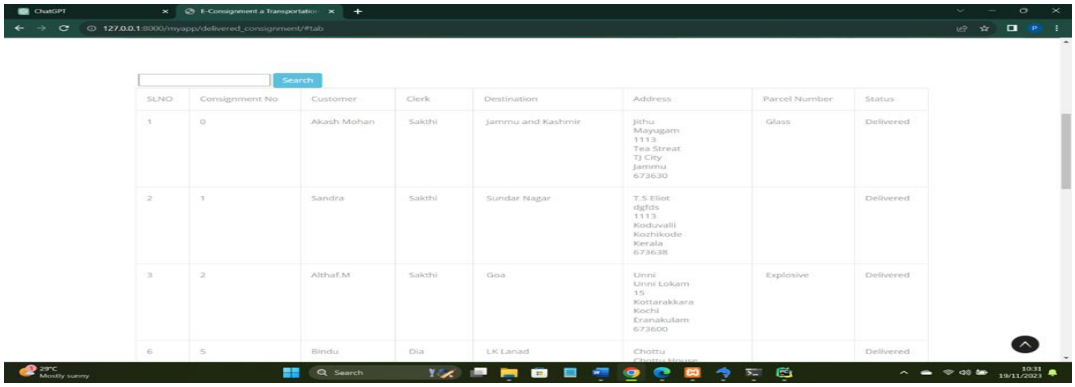
Admin View Customers



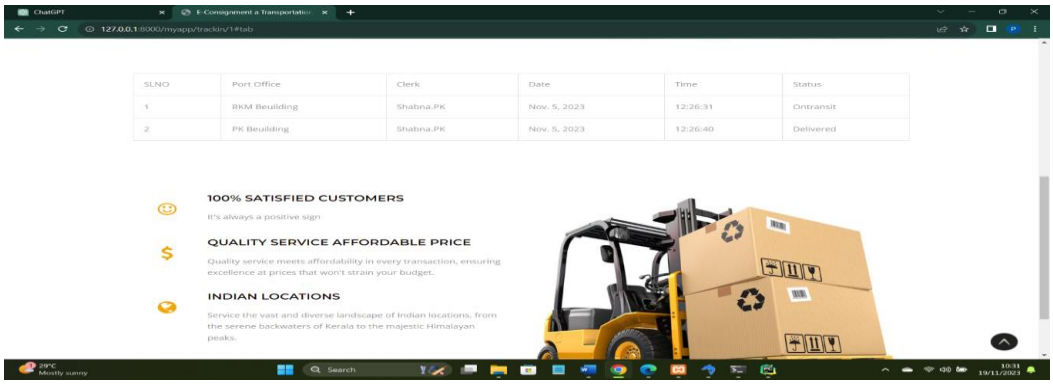
Admin View Consignment



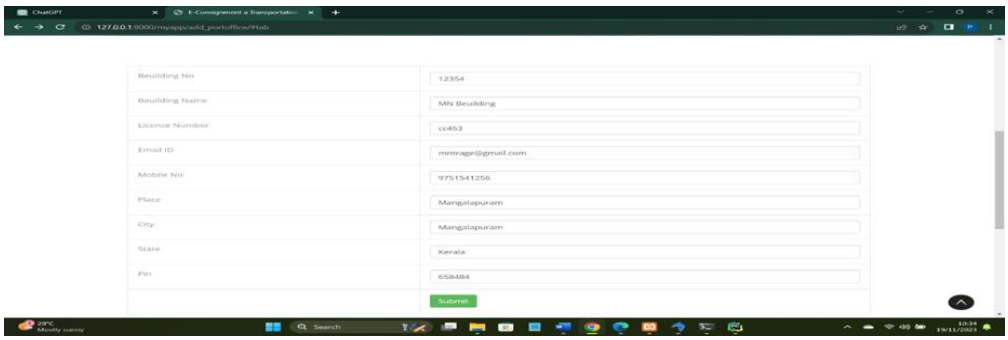
Admin view Delivered Consignment



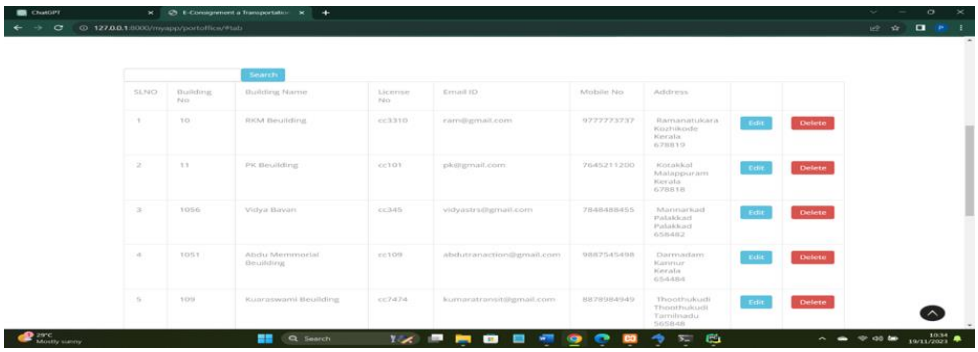
Admin View Track Consignment



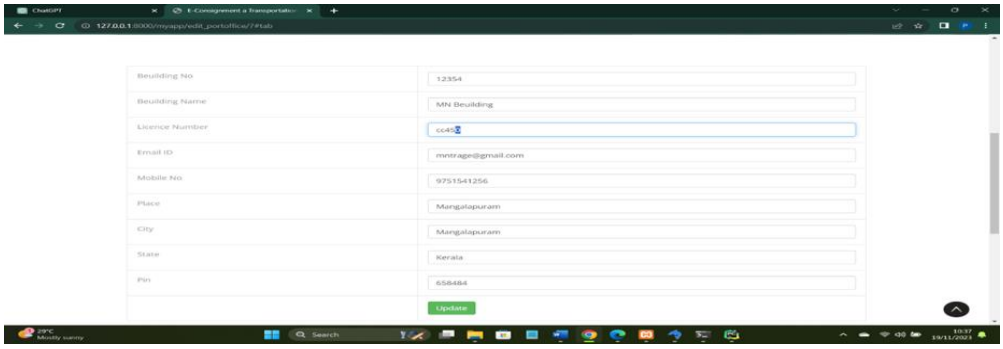
Admin View Add Port Office



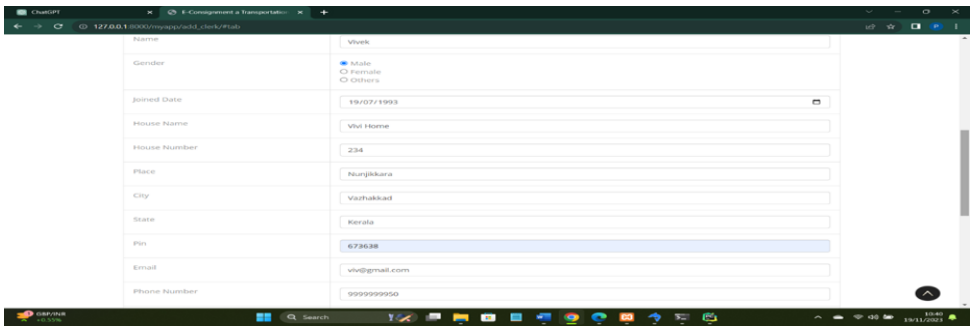
Admin View Port Office



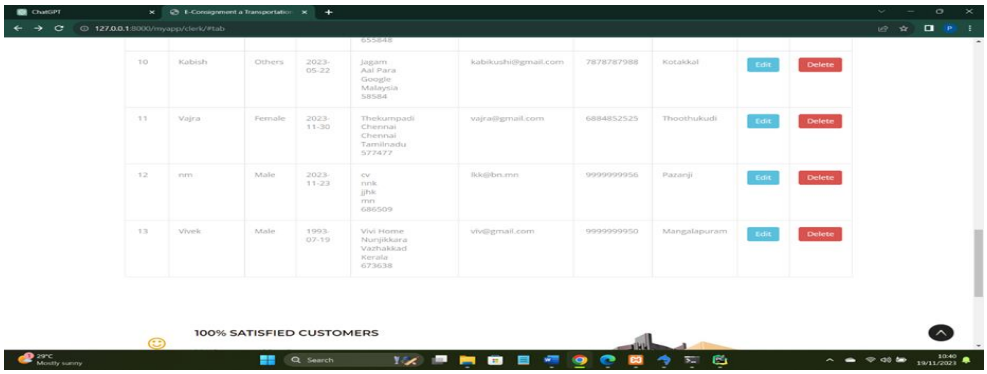
Admin View Edit Port Office



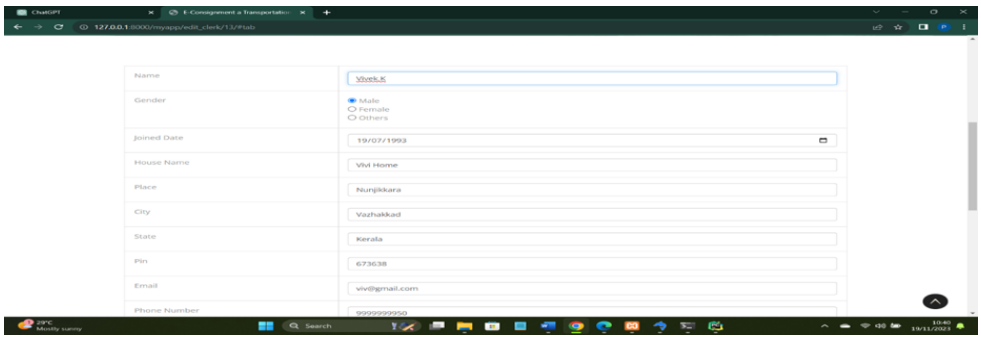
Admin View Add Clerk



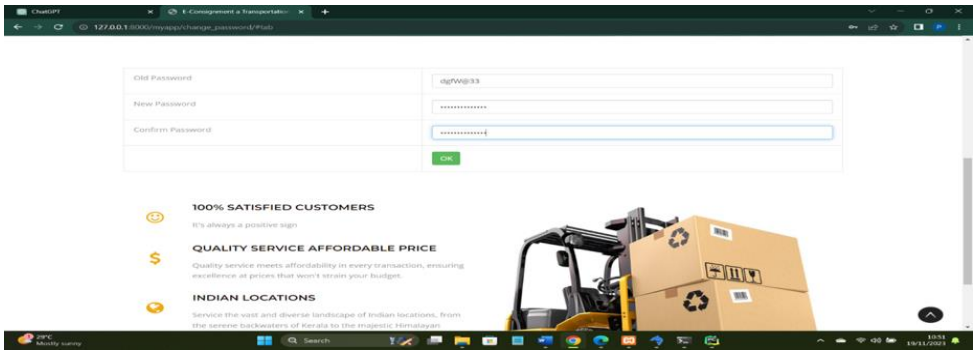
Admin View Clerk



Admin View Edit Clerk

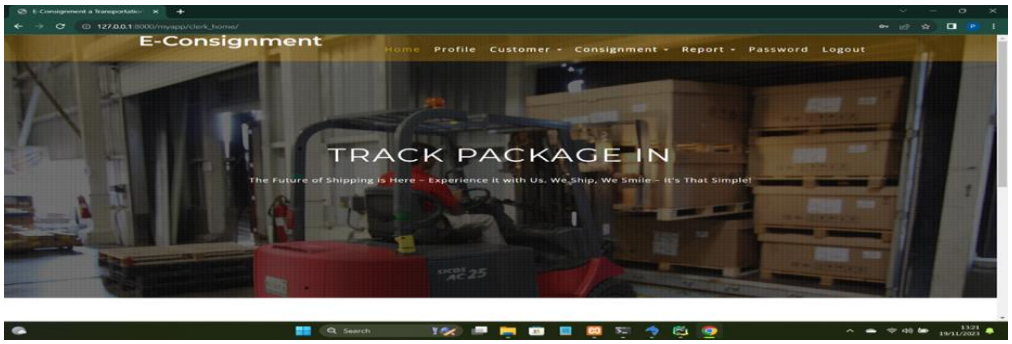


Admin View Change Password

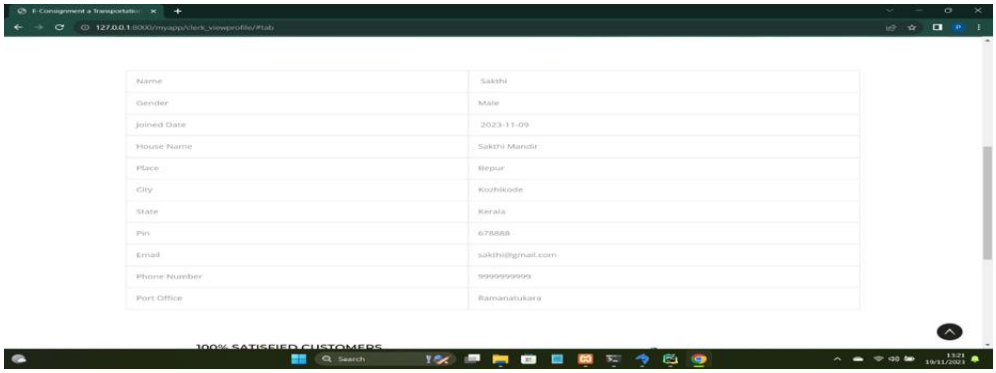


Clerk Login

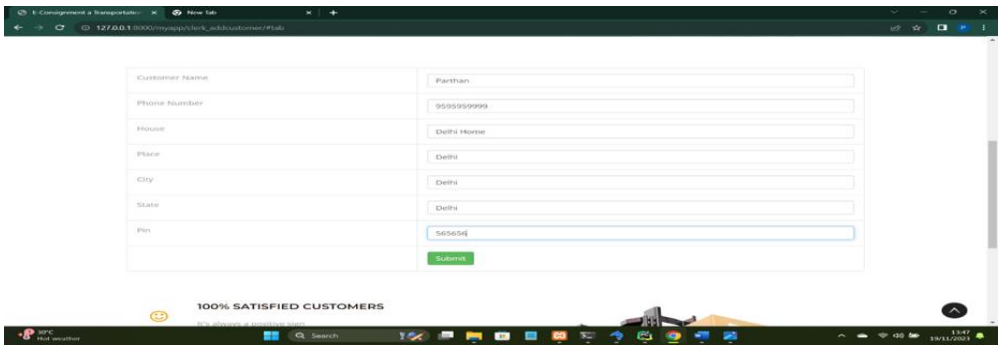
Clerk View Homepage



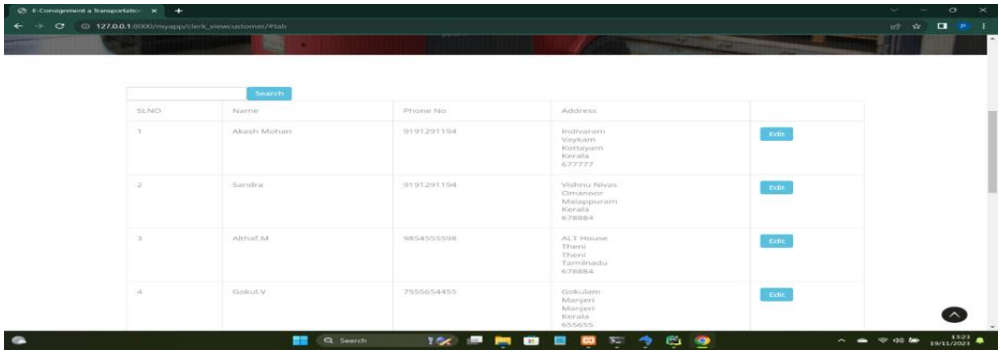
Clerk View Profile



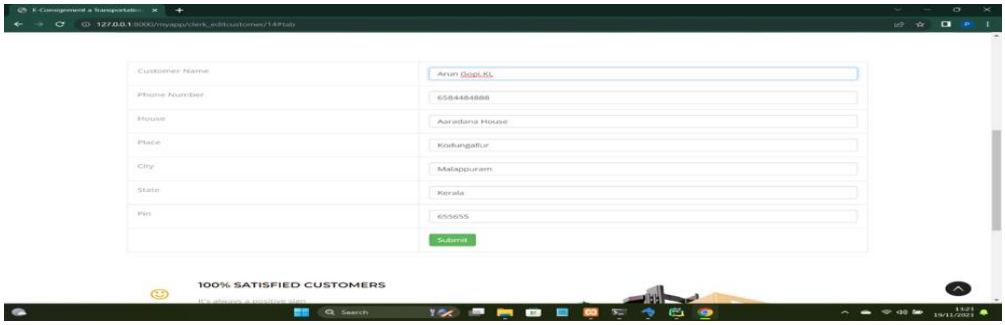
Clerk View Add Customer



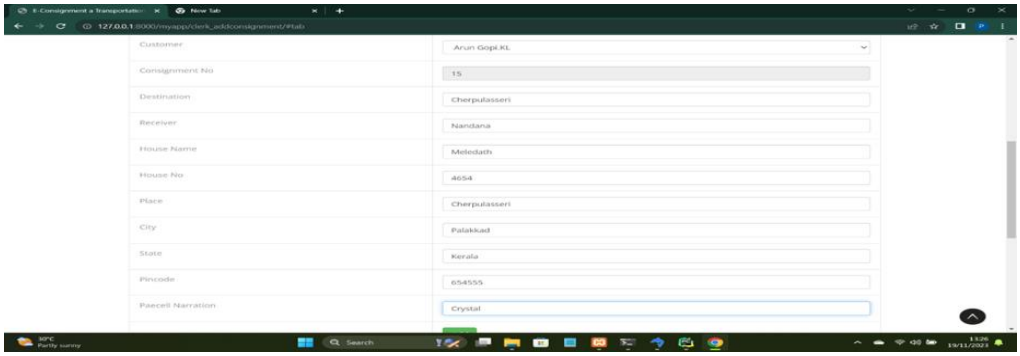
Clerk View Customer



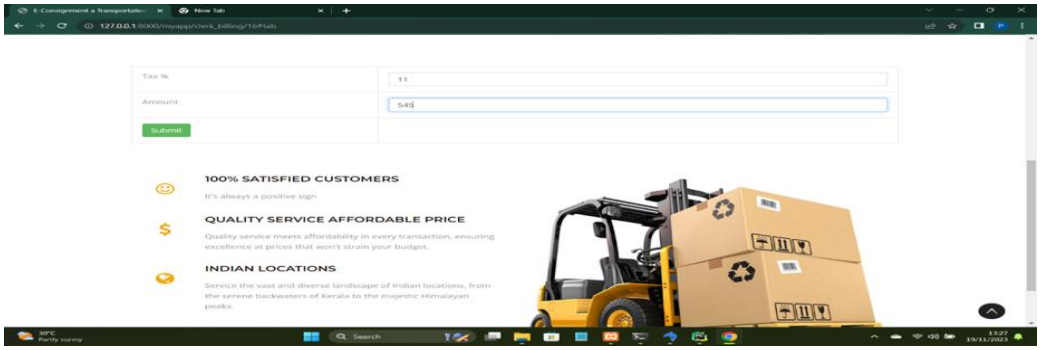
Clerk View Edit Customer



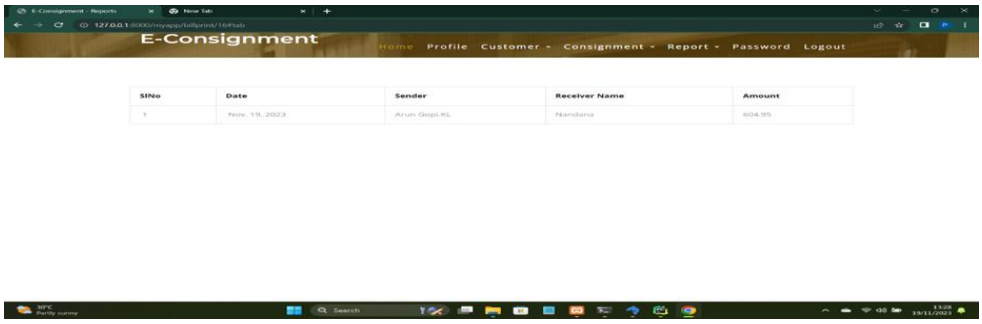
Clerk View Add Consignment



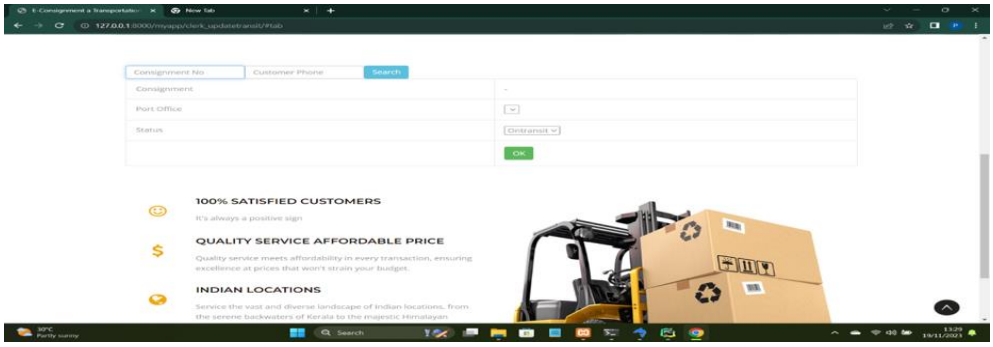
Clerk View Billing



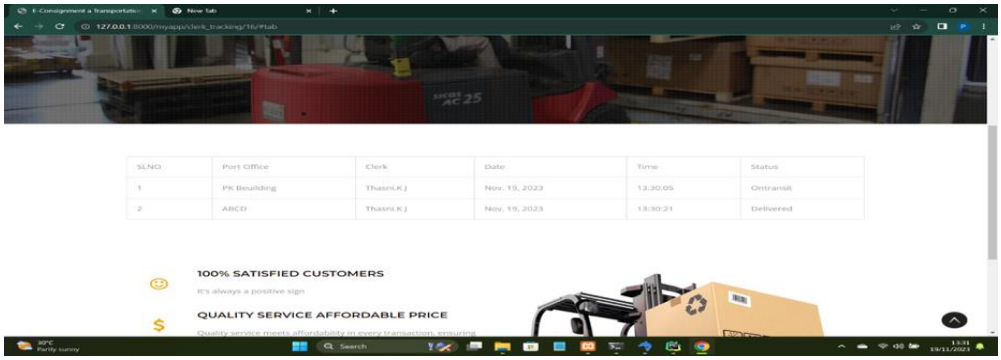
Clerk View Bill



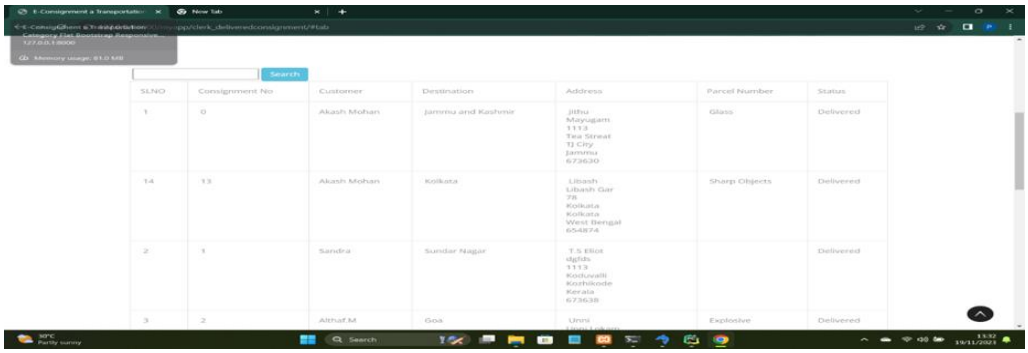
Clerk View Update Status



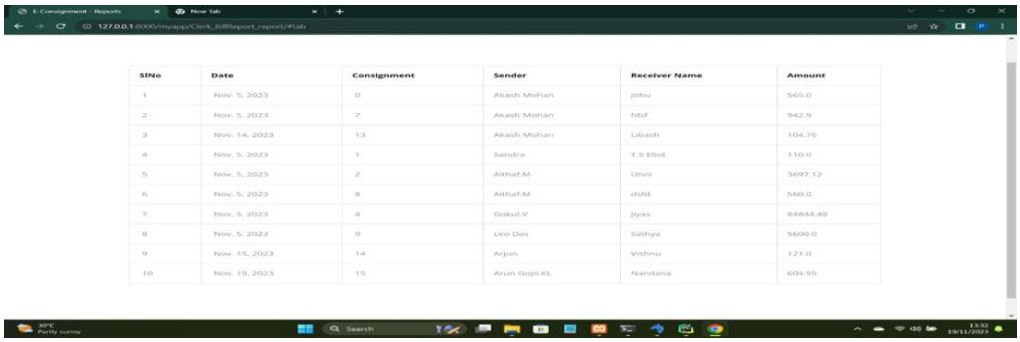
Clerk View Tracking



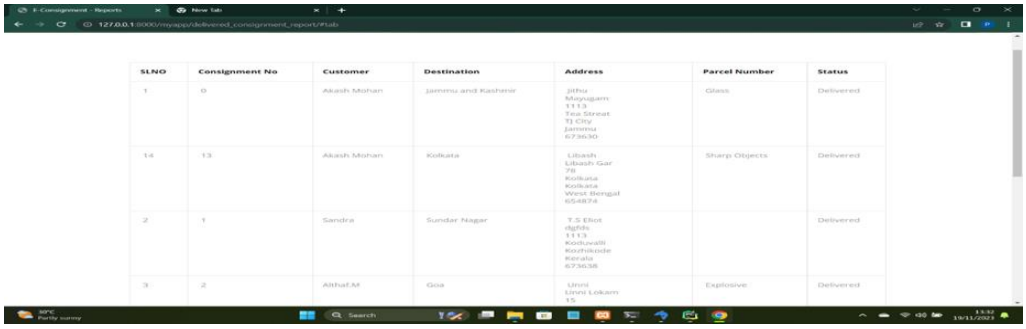
Clerk View Delivered



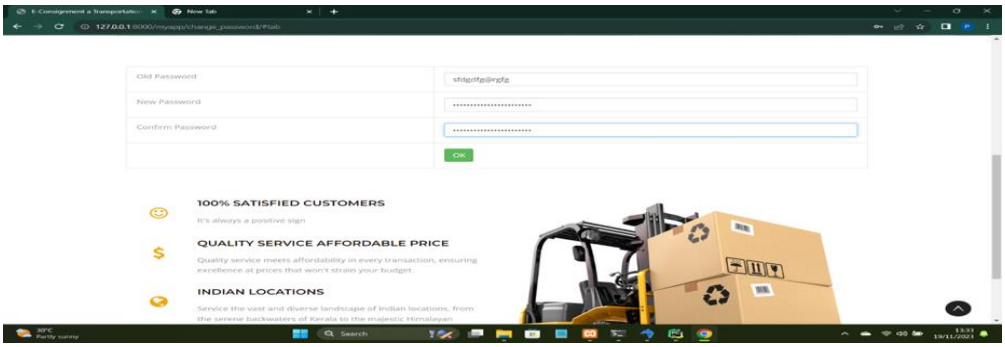
Clerk View Bill Report



Clerk View Consignment Report



Clerk View Change Password



BIBLIOGRAPHY

BIBLIOGRAPHY

Websites

[1] https://www.tutorialspoint.com/flask/index.html

[2] https://sqlyogkb.webyog.com

[3] https://docs.ipfs.io/how-to/

[4] https://www.tutorialspoint.com/ethereum/ethereum_smart_contracts.html [5] https://trufflesuite.com/ganache/

Books

[1] ThinkPython:AnIntroductiontoSoftwareDesign by Allen B.Downey

[2] Building Rest Apps With Flask by Kunal Relan

[3] Smart Contract Development with Solidity and Ethereum-Akhil Mittal

[4]