E-WASTE MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted in partial fulfilment of the
requirements for the award of the Degree of

BACHELOR OF SCIENCE

(INFORMATION TECHNOLOGY)

By

PRATIMA KAMBLE

SEAT NO.: 4020040

Under the esteemed guidance of

Prof. Bhupinder Singh

DEPARTMENT OF INFORMATION TECHNOLOGY

SIDDHARTH COLLEGE OF ARTS, SCIENCE, COMMERCE

(*Affiliated to university of Mumbai***)**

MUMBAI

MAHARASHTRA

YEAR: 2019-2020

PROFORMA FOR THE APPROVAL
OF PROJECT PROPOSAL

(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

PNR No.: 2017016400561611 Roll No.: 10

Name of the Student: PRATIMA P. KAMBLE

Title of the project: E-WASTE MANAGEMENT SYSTEM

Name of the Guide: Prof. BHUPINDER SINGH

Teaching experience of guide: 15 years

Is this your first submission? YES ☐

NO ☐

Signature of student

Date:_____

Signature of guide

Date: _____

Signature of coordinator

Date:_____

SIDDHARTH COLLEGE OF ARTS, SCIENCE, COMMERCE

(*Affiliated to university of Mumbai***)**

MUMBAI

MAHARASHTRA

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTTIFICATE

This is to certify that the project entitled "**E-WASTE MANAGEMENT SYSTEM**" is bonafied work of **PRATIMA P. KAMBLE** bearing **Seat no(10)** submitted in partial fulfilment of the requirement for the award of degree of **BACHELOR OF SCIENCE** in **INFORMATION TECHNOLOGY** from University of Mumbai.

Internal guide **Coordinator**

External Examiner

Date: **College seal**

ABSTRACT

This project aims to provide another way for the customer to giving the E-waste material. The E-waste Collection System is and Internet based application that can be accessed throughout the Net and can be accessed by anyone who has a net connection. It is an automatic system, where we will automate the selling of waste material and enquiries about which waste equipment are collection. After inserting the data to database, staff need not to worry about the orders received through the system and hence reduce the manual labour.

ACKNOWLEDGEMENT

We express sincere gratitude to the Prof. Bhupinder Singh and HOD of IT Department Mr. Javed Moosa and core faculty of Information Technology. SIDDHARTH COLLEGE OF ARTS, SCIENCE, COMMERCE is the constant source of inspiration and guidance to us. Their valuable knowledge and experience helped us to get through all the difficulties.

I would also thank my friends for giving me the opinions and various inputs in long discussion on the project which helped me shape the website keeping in mind the user friendly.

I would also like to thank everyone helped me in my project in some way or other which includes providing me with some information.

DECLARATION

I here by declare the project entitled, "**E-WASTE COLLECTION SYSTEM**" done by Pratima Kamble , has not been in any case duplicated to submit to any other university for the award of an degree. To the best of my knowledge other than me, no one has submitted to any other university.

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

PRATIMA P. KAMBLE

TABLE OF CONTENTS

CHAPTER 1 :INTRODUCTION

“E-waste Collection System” as a Website Management System which create Admin and user database. This module allows admin to login and view the total new customers and the customer logged in can also see the feedback and the their request status. It is used for the collection of the electronic waste material from the customer, local collector, so on. The customer can define the details about which type of waste having by logged in. E-waste Collection System is for developers and recycling it in the company. This module also assigns login rights to both user and admin.

E-waste Collection System covers all types of E-waste in integrative manner from all the waste source including E-waste from domestic/municipal and industrial sources.

1.1 BACKGROUND:-

"**E-waste**" is a popular, informal name for electronic products nearing the end of their "useful life. E-wastes are considered dangerous, as certain components of some electronic products contain materials that are hazardous, depending on their condition and density. The hazardous content of these materials pose a threat to human health and environment. Discarded computers, televisions, VCRs, stereos, copiers, fax machines, electric lamps, cell phones, audio equipment and batteries.

Electronic wastes contain toxic substances such as lead, mercury, cadmium, and lithium. These toxic materials can be released upon disposal, posing a threat to human health and the environment. Inconsistencies in worker safety and

environmental protection mean potential liability concerns for those sending electronics to recycling facilities.

1.2 OBJECTIVES:-

The major object of project is to build capacity of practitioners and decision makers to guide and handhold them to plan, design and implement online E-waste Collection System including policy, Collection, transportation and treatment in a city/ geographical area and country.
Encourage and promote the development and progress of E-waste Collection System towards achieving in the field of computer sciences and technology for theatre applications both for recycling and deployment of electronic waste.

Online E-waste Collection System amongst client for awareness and recycling of the waste material by using latest system. Simulation and offer aid for system for the benefit of manufactures and users. Help in the improvement of standards, terminology equipment's, methods and implementation practices in the field of E-waste Collection System.

1.3 PURPOSE, SCOPE & APLICABILITY :-

1.3.1 PURPOSE:-

The main purpose of E-waste Collection System is to provide another way for the customer to giving the E-waste material. The E-waste Collection System is and internet based website that can be accessed throughout the Net and can be accessed by anyone who has a internet connection. It is an automatic system, where we will automate the selling of waste electronic material and enquiries about which waste equipment are collection. After inserting the data to database, staff need not to worry about the orders received through the system and hence reduce the manual labour. One of the best features of the system is to deploy or recycling the electronic equipment from the customer house and the city.

The goals of system are:

- To provide anytime anyplace service for the customer.
- To reuse electronic waste material by recycling or deploy.
- To decrease the electronic waste material from household.
- To obtain static information about the problems effect by the e-waste material.
- To provide awareness about electrical and electronic material using for household.

1.3.2 SCOPE:-

The scope of the project is to provide approaches and strategies which have proved to be the suitable when assessing the e-waste system of the defined region. This collection will reduce the e-waste from the household, company, industries, city, etc. The environment pollution will reduce and the electronic waste will recycle or deploy. The fundamental aims of the **Basel Convention** are the control and reduction of trans boundary movements of hazardous and other waste including the prevention and minimization of their generation, the environmentally sound management of such waste and the active promotion of the transfer and use of technologies.

This technique could eliminate waste disposal costs, reduce raw material costs and provide income from a saleable waste. Waste can be recovered on-site, or at an off-site recovery facility, or through inter industry exchange. A number of physical and chemical techniques are available to reclaim a waste material such as reverse osmosis, electrolysis, condensation, electrolytic recovery, filtration, centrifugation etc. for example, a printed circuit board manufacturer can electrolytic recovery to reclaim metals from copper and tin-lead platin.

1.3.3 APPLICABILITY:-

We offer progressive end-to-end solutions keeping in mind to maintain the specific data of contact us form, feedback form, upcoming event.

Services They Provide: -

- Maintaining the user's information.
- Maintaining the waste information.
- To add/update/delete the user data.

- To add/update/delete the requested data.
- To add/update/delete the event data.
- Password reset mailing Feedback mail.

1.4 ORGANIZATION OF REPORT :-

The current chapter explains what the project is all about. Second chapter will explains the planning of the project and its development, its requirement its tasks and all the requirements, such as hardware and software requirements.

The planning of resources and application to be used in the project will be decided. Once the project functionality will be explained with required flow diagrams, process diagrams etc. The data structure of the project and functionality will be explained. Algorithm designs with input data, output data and logic and design will also be explained.

CHAPTER 2:SURVEY OF TECHNOLOGY

2.1 ABOUT ASP.NET:-

ASP.NET is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web applications for PC, as well as mobile devices.

ASP.NET works on top of the HTTP protocol, and uses the HTTP commands and policies to set a browser-to-server bilateral communication and cooperation.

ASP.NET is a part of Microsoft .Net platform. ASP.NET applications are compiled codes, written using the extensible and reusable components or objects present in .Net framework. These codes can use the entire hierarchy of classes in .Net framework.

The ASP.NET application codes can be written in any of the following languages:

- C#
- Visual Basic.Net
- Jscript
- J#

ASP.NET is used to produce interactive, data-driven web applications over the internet. It consists of a large number of controls such as text boxes, buttons, and labels for assembling, configuring, and manipulating code to create HTML pages.

2.2 ABOUT C#:-

C# is a general object-oriented programming (OOP) language for networking and Web development. C# is specified as a common language infrastructure (CLI) language.

In January 1999, Dutch software engineer Anders Hejlsberg formed a team to develop C# as a complement to Microsoft's NET framework.

Initially, C# was developed as C-Like Object Oriented Language (Cool). The actual name was changed to avert potential trademark issues. In January 2000, NET was released as C#. Its NET framework promotes multiple Web technologies. The term is sometimes spelled as C Sharp or C-Sharp.

2.3 ABOUT VISUAL STUDIO 2019:-

Visual Studio 2019 is an integrated, complete solution with development tools, cloud services and extensions that enables you and your team to create great applications and games for desktops, the web, Windows Store, Android and iOS.

Choose between different editions, depending on whether you work alone or in a small team (Professional edition) or in a complex project across departments and locations (Enterprise edition).

Visual Studio 2019 provides everything you need to help you deliver software in less time and even better quality.

With Visual Studio 2019, you'll get best-in-class tools and services for any developer, any app, and any platform. Whether you're using Visual Studio for the first time or you've been using it for years, there's a lot to like in this new version!

Here's a high-level recap of what's new:

- Develop: Stay focused and productive with improved performance, instant code clean up, and better search results.
- Collaborate: Enjoy natural collaboration through a Git-first workflow, real-time editing and debugging, and code reviews right in Visual Studio.

- Debug: Highlight and navigate to specific values, optimize memory use, and take automatic snapshots of your application's execution.

2.4 ABOUT MSSQL SERVER:-

MSSQL is a suite of database software published by Microsoft and used extensively within our enterprise.

Typically, it includes a relational database engine, which stores data in tables, columns and rows, Integration Services (SSIS), which is a data movement tool for importing, exporting and transforming data, Reporting Services (SSRS), which is used to create reports and serve reports to end users, and also Analysis Services (SSAS), which is a multidimensional database used to query data from the main database engine.

Microsoft SQL Server (MSSQL) is widely used in enterprise deployments. MSSQL is a scalable data platform which includes several ETL (Extract, Transform and Load) tools and reporting services where data can be added, modified and queried using a standardized structured query language (SQL).

MSSQL is an evolving data platform used for mission critical business and data solutions on premise, in the cloud and on hybrid platforms.

MSSQL server is an incredibly popular database solution used today, and one of its strongest advantages is its ease of use. MSSQL comes with many excellent tools which make database development a fast and agile process.

SQL Server management studio allows any approved user to manage and maintain the databases, run SQL queries, perform backups and analyze performance charts. MSSQL integrates with Visual Studio to give your DevOps team a powerful, familiar platform to create and manage custom applications which seamlessly integrate with MSSQL Server.

There are many products which make up the SQL Server database platform, but there are 4 key services built into MSSQL which define it and make it a popular choice as a database management system (DBMS).

These options are available to install when deploying the MSSQL instance. The

latest releases of MSSQL are not just compatible with Windows; more recently, Microsoft has offered SQL for Linux (Red hat and SUSE),as well as Docker container platforms.

CHAPTER 3

REQUIREMENTS & ANALYSIS

3.1 PROBLEM DEFINITION :-

- The existing system has one ways of waste collection system.
- The waste collection system is offline which is done by the government, like the waste material with all the mixture with solid and liquid waste. The waste is send to the garbage factories for the deployment of the waste.

3.2 REQUIREMENT SPECIFICATION :-

The waste collection system is now online waste collection website. The public get the information about the e-waste material and aware about the waste. We will collect the household electronic and electric equipment from the public and which will recycle or deploy waste. The recycling waste will be used in the other equipment's, and industries can use the recycling equipment's for new material, etc**.**

The goals of the system are:

- To provide anytime anyplace service for the customer.
- To reuse electronic waste material by recycling or deploy.
- To decrease the electronic waste material from household.
- To obtain statistic information about the problems effect by e-waste material.

3.3 PLANNING & SCHEDULING :-

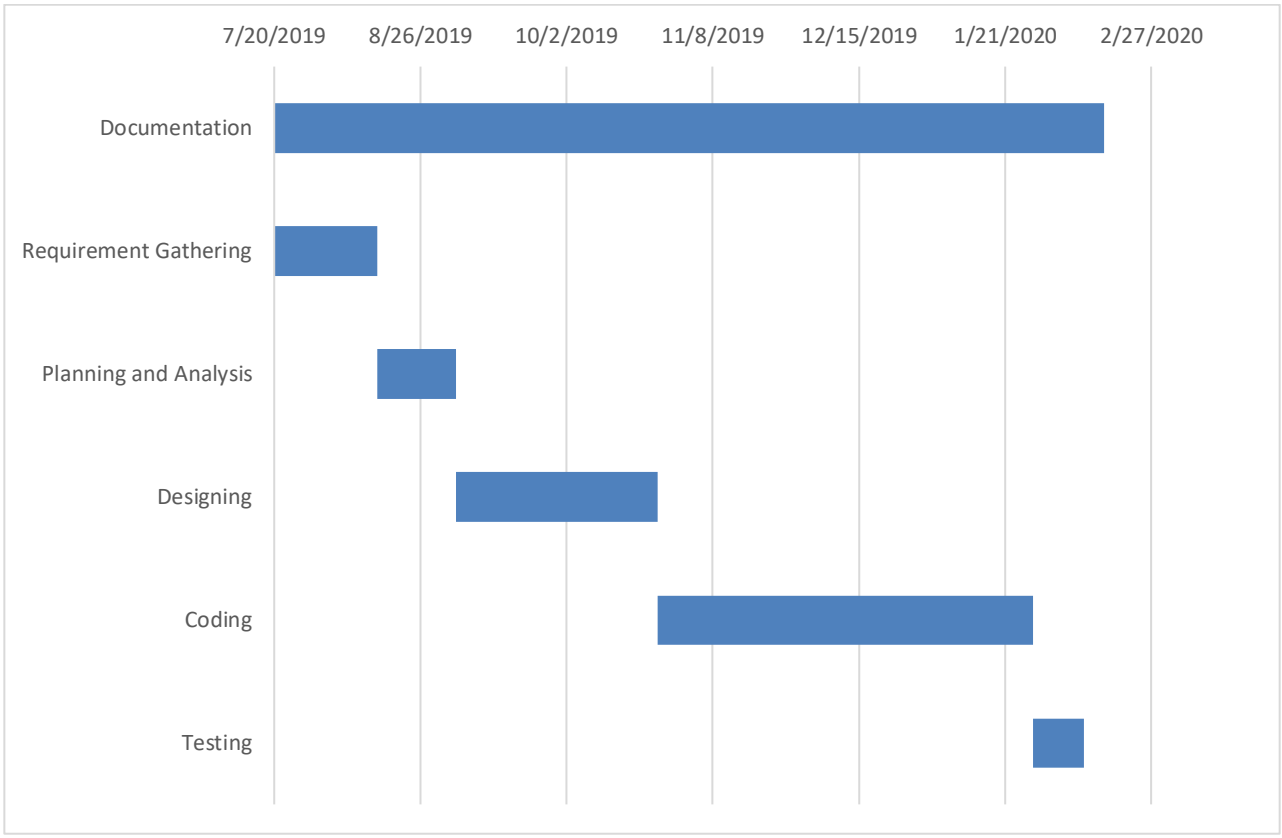
3.3.1 GANTT CHART :-

Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project. Modern Gantt charts also show the dependency (i.e., precedence network) relationships between activities.

• Gantt charts can be used to show current schedule status using percent-complete shadings and a vertical "TODAY" line.

• Although now regarded as a common charting technique, Gantt charts were considered revolutionary when first introduced. This chart is also used in information technology to represent data that have been collected.

• Gantt charts can be used for scheduling generic resources as well as for their use in project management. They can also be used for scheduling adoption processes and employee rostering.

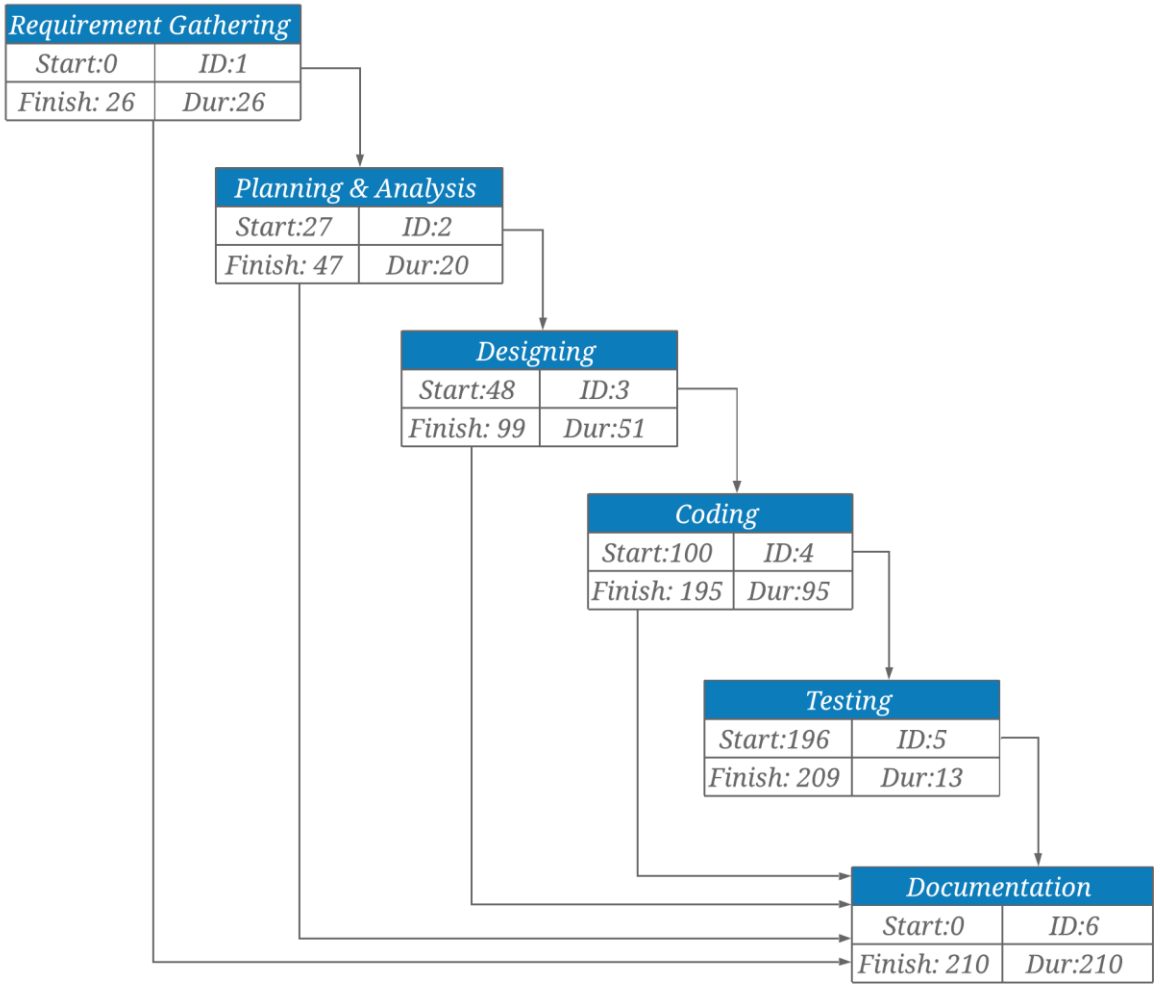


Activity	Documentation	Requirement Gathering	Planning & Analysis	Designing	Coding	Testing
Start Date	20-07-2019	20-07-2019	15-08-2019	04-09-2019	25-10-2019	28-01-2020
Duration	210 days	26 days	20 days	51 days	95 days	13 days

3.3.2 PERT CHART :-

A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project. PERT stands for *Program Evaluation Review Technique*, a methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program. A similar methodology, the *Critical Path Method* (CP chart M) was developed for project management in the private sector at about the same time.

The PERT chart is sometimes preferred over the Gantt , another popular project management charting method, because it clearly illustrates task dependencies. On the other hand, the PERT chart can be much more difficult to interpret, especially on complex projects. Frequently, project managers use both techniques.


Requirement Gathering
Start:0
ID:1
Finish: 26
Dur:26
Planning & Analysis
Start:27
ID:2
Finish: 47
Dur:20
Designing
Start:48
ID:3
Finish: 99
Dur:51
Coding
Start:100
ID:4
Finish: 195
Dur:95
Testing
Start:196
ID:5
Finish: 209
Dur:13
Documentation
Start:0
ID:6
Finish: 210
Dur:210

3.4 HARDWARE & SOFTWARE REQUIREMENT :-

3.4.1 HARDWARE REQUIREMENT :

Processor : Intel Core i3 or above

Memory (RAM): Minimum of 2GB RAM

Hard Disk: Minimum of 512GB

3.4.2 SOFTWARE REQUIREMENT:

Operating System: Window 7 or above

Frontend: Microsoft Visual Studio

Backend: MSSQL

3.5 PRELIMINARY PRODUCT DESCRIPTION :-

Preliminary investigation helps in clarification, understanding and evaluation of the project request. Preliminary investigation basically refers to the collection of information that guides the management of an organization to evaluate the merits and demerits of the adoption request and make informed judgement of about the feasibility of their project aim.

For this report, we conducted the preliminary investigation by using:

1. The document provided by the local common people.
2. The requirements of the user searching in the website.

3.6 CONCEPTUAL MODELS:-

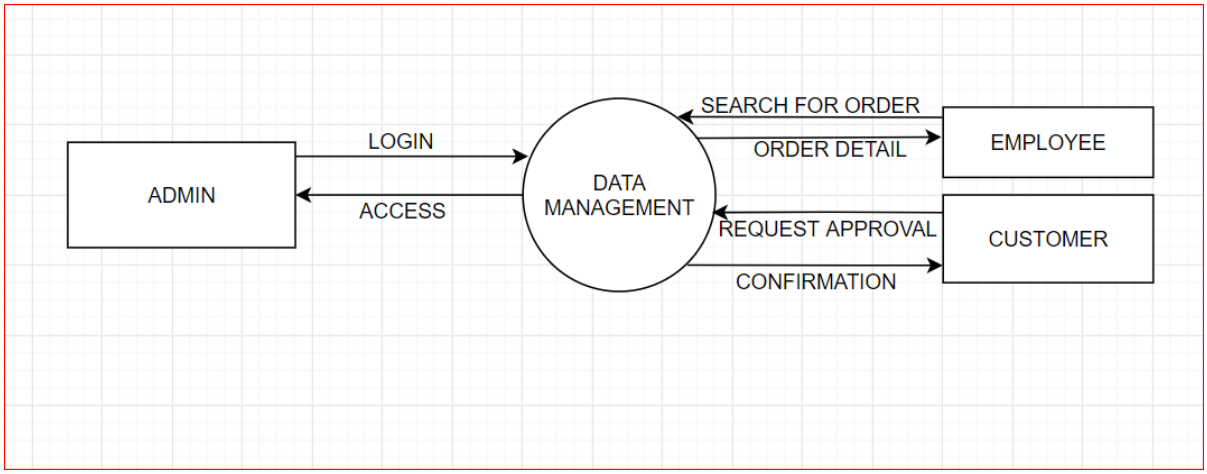
To understand the problem domain and produce a model of the system, which describes operations that can be performed on the system, and the allowable sequences of those operations.

3.6.1 DATA FLOW DIAGRAMS :-

A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates, its focus is on the flow of information, where data comes from, where it goes and how it gets stored.

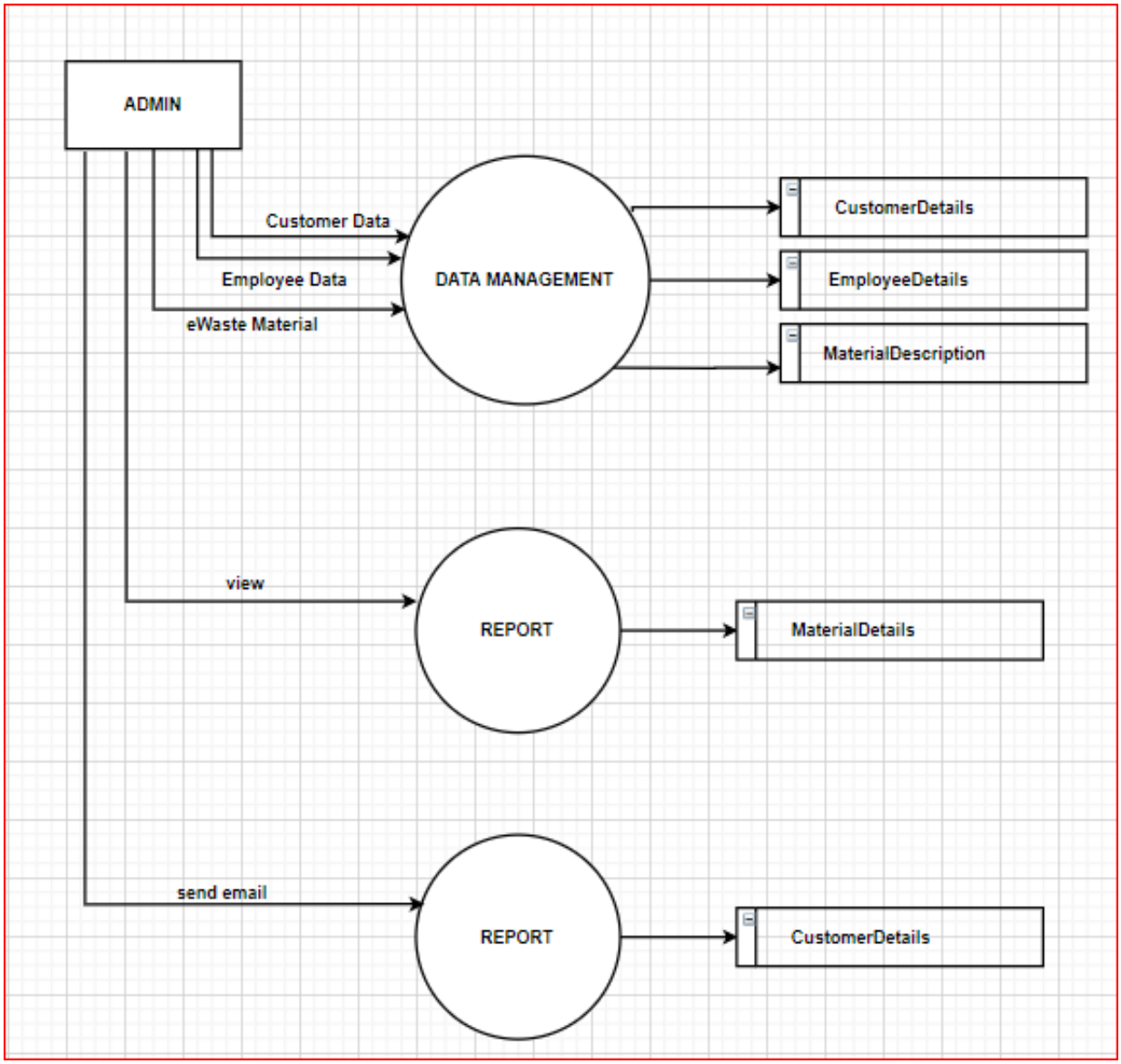
Level 0 Diagram :

- Shows all the processes that comprise the overall system.
- Shows how information moves from and to each process.
- Adds data stores.



Level 1 Diagram :

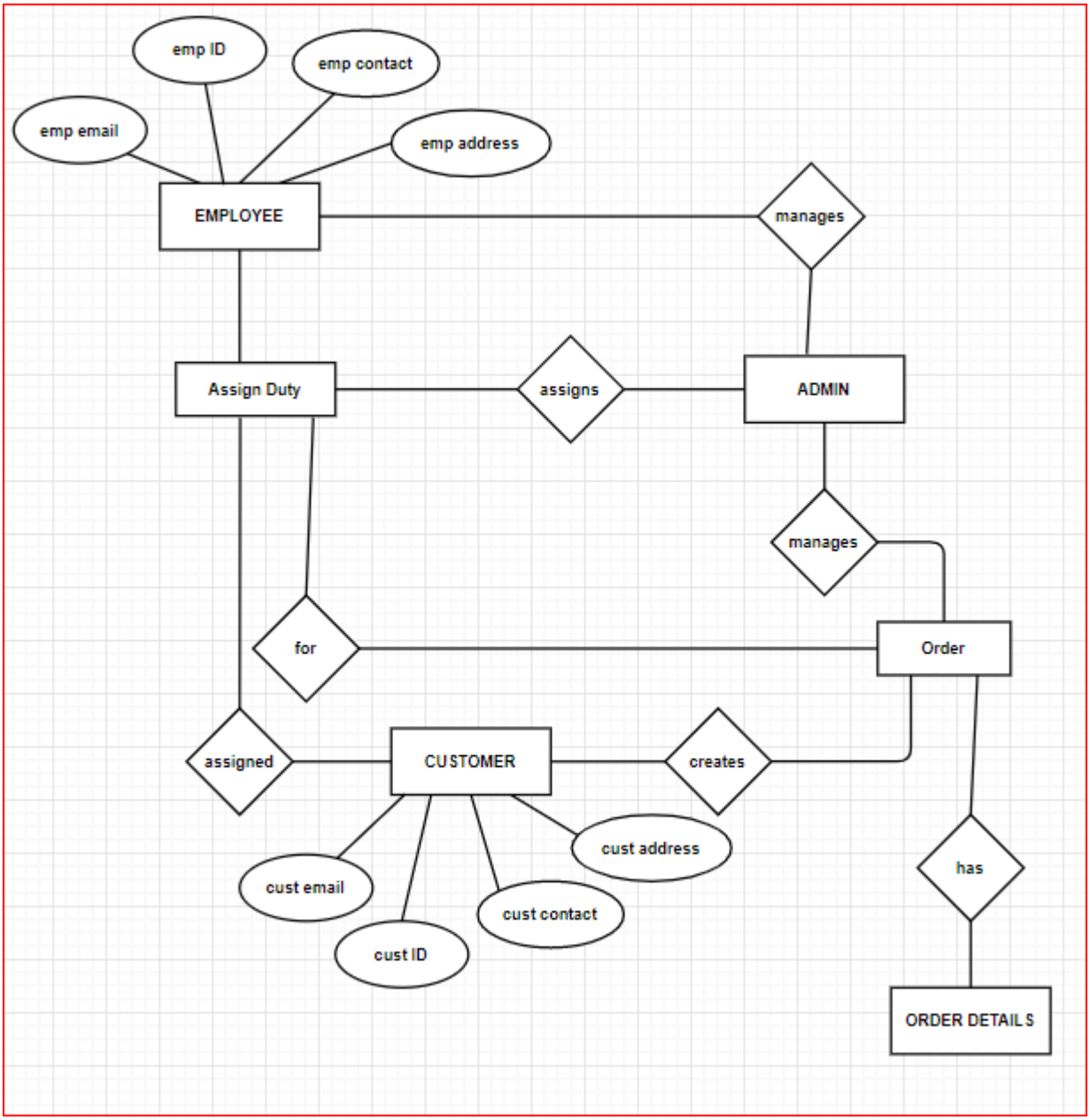
- Shows all the processes that comprise a single process on the level 0 diagram.
- Shows how information moves from and to each of these processes.
- Shows in more detail the content of higher level process.
- Level 1 diagrams may not be needed for all level 0 processes.



3.6.2 ER DIAGRAM :-

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

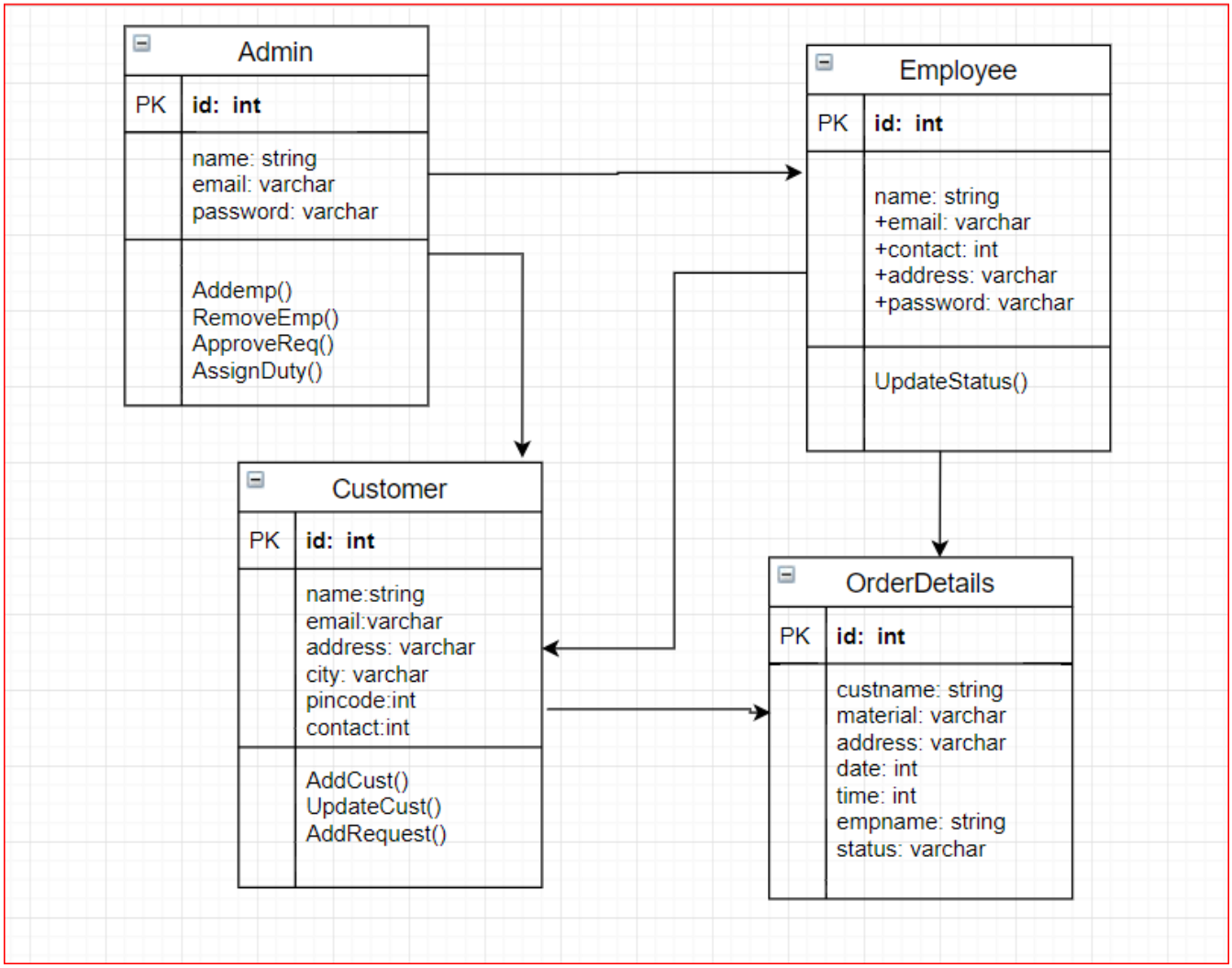
ER diagrams are related to data structure diagrams (DSDs), which focus on the relationships of elements within entities instead of relationships between entities themselves. ER diagrams also are often used in conjunction with data flow diagrams (DFDs), which map out the flow of information for processes or systems.


emp ID
emp contact
emp email
emp address
EMPLOYEE
manages
Assign Duty
assigns
ADMIN
manages
for
Order
assigned
CUSTOMER
creates
cust address
has
cust email
cust contact
cust ID
ORDER DETAILS

3.6.3 CLASS DIAGRAM :-

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The purpose of class diagram is to model the static view of an application.



CHAPTER 4
SYSTEM DESIGN

4.1 BASIC MODULE:-

LOGIN:

A login generally required the user to enter two pieces of information, first a user name and then a password. This information is enter into a login window on a GUI(Graphical User Interface) or on the command line in a console depending on the system.

A user name, also referred to as an account name, is string(i.e. sequence of characters) that uniquely identifies a user. User name can be same as or related to real names of user, or they can be completely arbitrary.

A password is likewise a string, but it differs from a user in that it is intended to be kept a secret that is known only to its user and, perhaps to the system administrator.

ADMIN LOGIN:

Admin page has following module like Change password, view request, view details of new registered user, View order and Approve. Admin can view the order which are put up for collection. Admin can see all the order details. Admin can see total number of orders which are completed or pending. Admin can assign duty to an employee. He/she can view all the detail of the user or customer.

USER LOGIN:

A login, logging in or logging on is the entering of the identifier information into a system by a user in order to access that system (computer or website).

NEW USER:

This module is for the users who do not have their account. Here user is allowed to create an account to login. The account creation is done by filling the registration form with user details such as name, phone, email, address etc.

4.2 DATA DESIGN:-

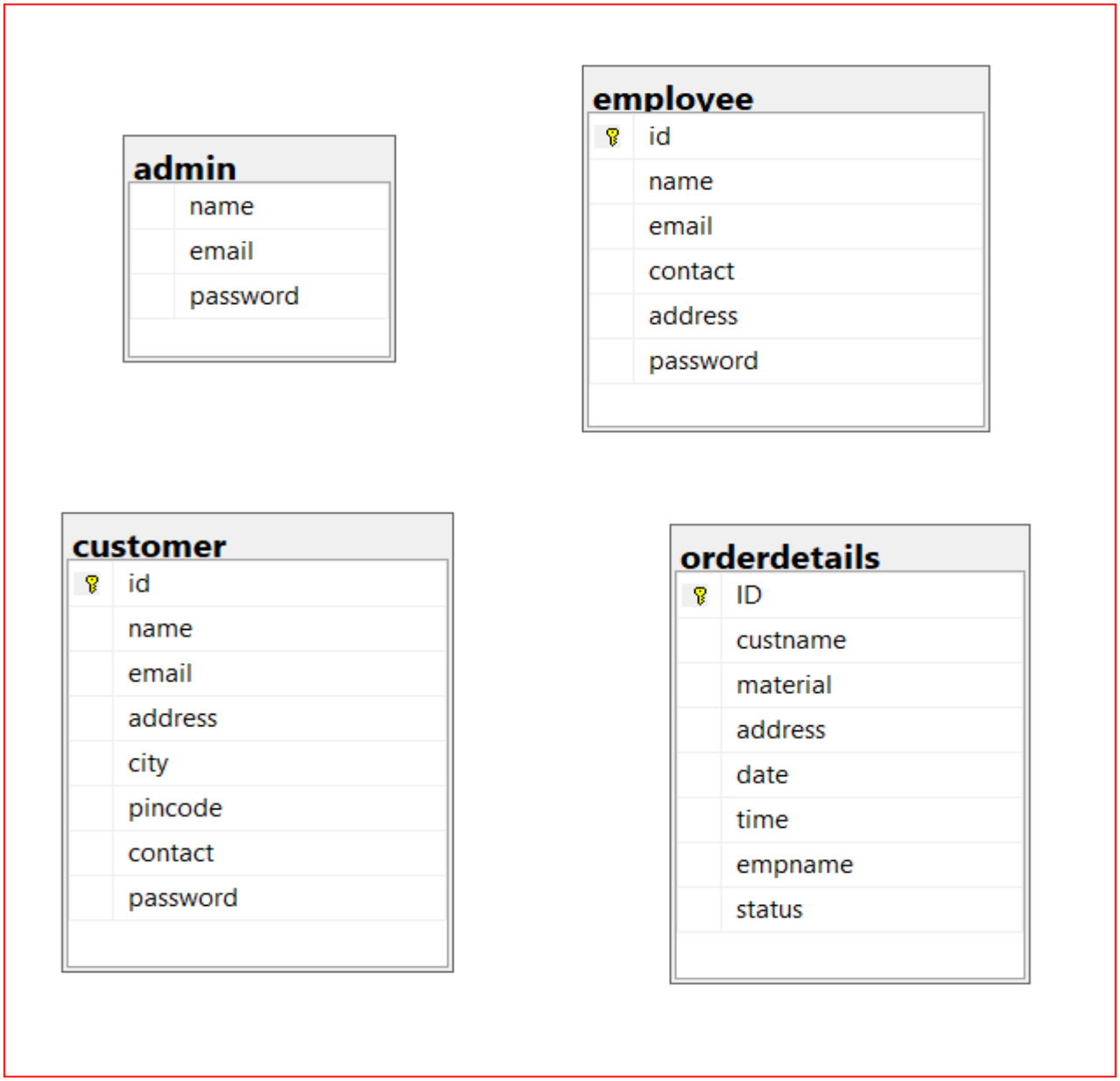
Database design is the organization of data according to a database model. The designer determine what data must be stored and how the data elements interrelate. With this information they can being to fit the data to the database model. Database design involves classifying data and identifying interrelationship.

4.2.1 SCHEMA DESIGN :-

The **database schema** of a database system is its structure described in a formal_language supported by the database management system (DBMS). The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language. The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modeled in the database.

"A database schema specifies, based on the database administrator's knowledge of possible applications, the facts that can enter the database, or those of interest to the possible end-users." The notion of a database schema plays the same role as the notion of theory in predicate calculus. A model of this "theory" closely corresponds to a database, which can be seen at any instant of time as a mathematical object. Thus a schema can contain formulas representing integrity constraints specifically for an application and the constraints specifically for a type of database, all expressed in the same

database language. In a relational database, the schema defines the tables, fields, relationships, views, indexes, packages, procedures, functions, queues, triggers, types, sequences, materialized views, synonyms, database links, directories, XML schemas, and other elements.



4.2.2 DATA INTEGRITY & CONSTRAINTS :

- Customer Table

Column name	Data type	Constraints
Id	Int	Unique, primary key
Name	Varchar	Not null
Email	Varchar	Not null
Address	Varchar	Not null
City	Varchar	Not null
Pincode	Varchar	Not null
Contact	Int	Not null
Password	Varchar	Not null

- Admin Table

Column name	Data type	Constraints
Name	Varchar	Not null
Email	Varchar	Not null
Password	Varchar	Not null

- Employee table

Column name	Data type	Constraints
Id	Int	Unique, primary key
Name	Varchar	Not null
Email	Varchar	Not null
Contact	Varchar	Not null
Address	Varchar	Not null
Password	Varchar	Not null

- Order details

Column name	Data type	Constraints
Id	Int	Unique, primary key
Custname	Varchar	Not null
Material	Varchar	Not null
Address	Varchar	Not null
Date	Int	Not null
Time	Int	Not null
Empname	Varchar	Not null
Status	Varchar	Not null

4.3 PROCEDURAL DESIGN :-

Component level design also called procedural designoccurs after data, architectural and interface designs have been established.

Data, architectural and interface design must be translated into operational software. To accomplish this design must be represented at a level of abstraction that is close to code.

Component-level design establishes:-

- The algorithmic detail required to manipulate data structures.
- Effect communication between software components via their interfaces, and
- Implement the processing algorithms allocated to each components.

4.3.1 LOGICAL DIAGRAMS:-

CONTROL FLOW DIAGRAM :

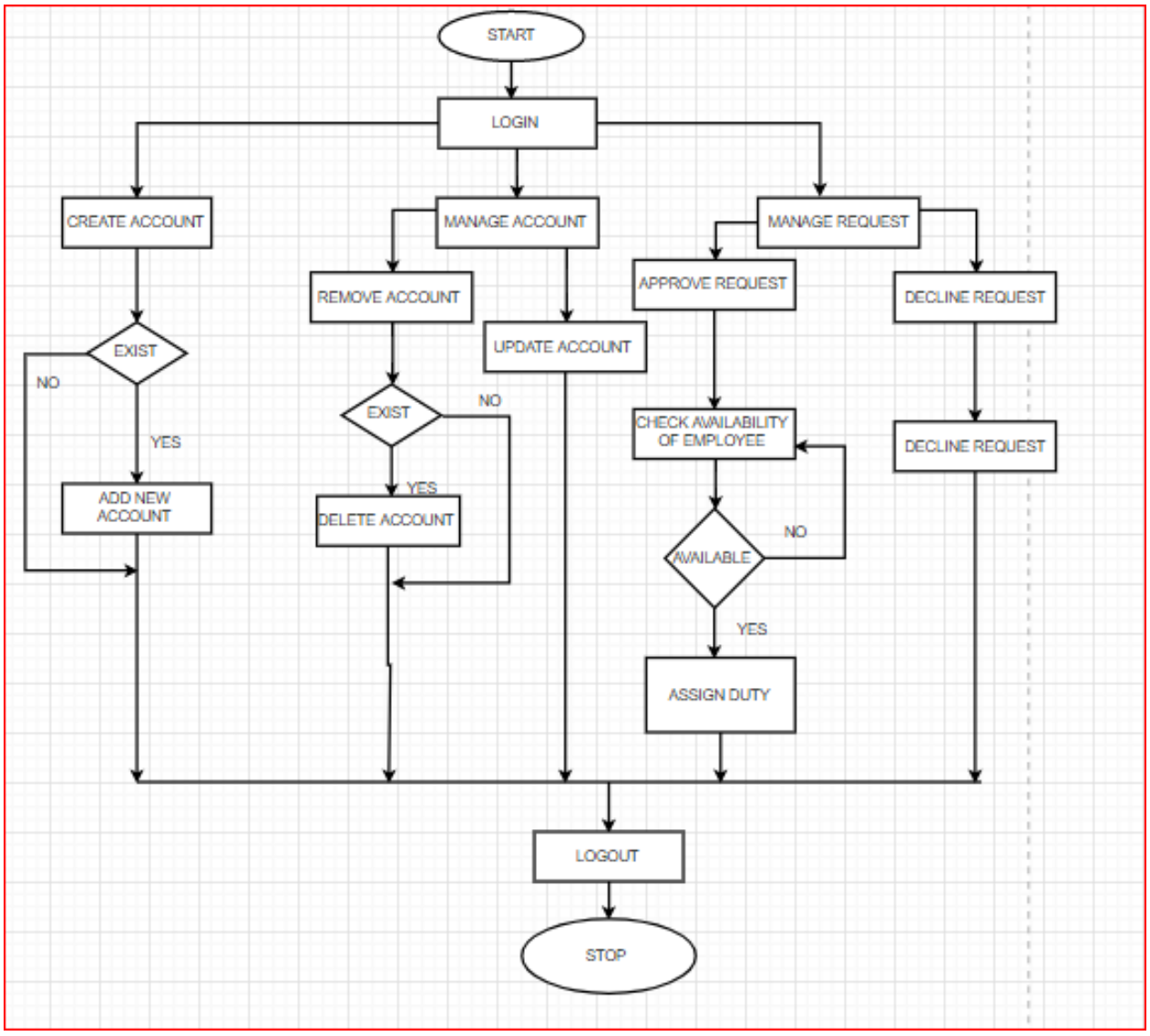
A control-flow diagram can consist of a subdivision to show sequential steps, with if-then-else conditions, repetition, and/or case conditions. Suitably annotated geometrical figures are used to represent operations, data, or equipment, and arrows are used to indicate the sequential flow from one to another.

There are several types of control-flow diagrams, for example:

- Change-control-flow diagram, used in project management
- Configuration-decision control-flow diagram, used in configuration management
- Process-control-flow diagram, used in process management
- Quality-control-flow diagram, used in quality control.

In software and systems development, control-flow diagrams can be used in control-flow analysis, data-flow analysis, algorithm analysis, and simulation.

Control and data are most applicable for real time and data-driven systems. These flow analyses transform logic and data requirements text into graphic flows which are easier to analyze than the text. PERT, state transition, and transaction diagrams are examples of control-flow diagrams.



ALGORITHM:

STEP 1 : Start.

STEP 2 : Login.

STEP 3 : If admin wants to create account click on Create Account.

STEP 4 : If account is existed press YES if not press NO and click on Add new account.

STEP 5 : If admin want to manage account click on Manage account. In manage account admin can update or delete account.

STEP 6 : If admin want to remove account press YES and click on delete account and if not press NO.

STEP 7 : If admin want to update press on update account.

STEP 8 : Logout.

STEP 9 : Stop.

ACTIVITY DIAGRAM:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc. The basic purposes of

activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

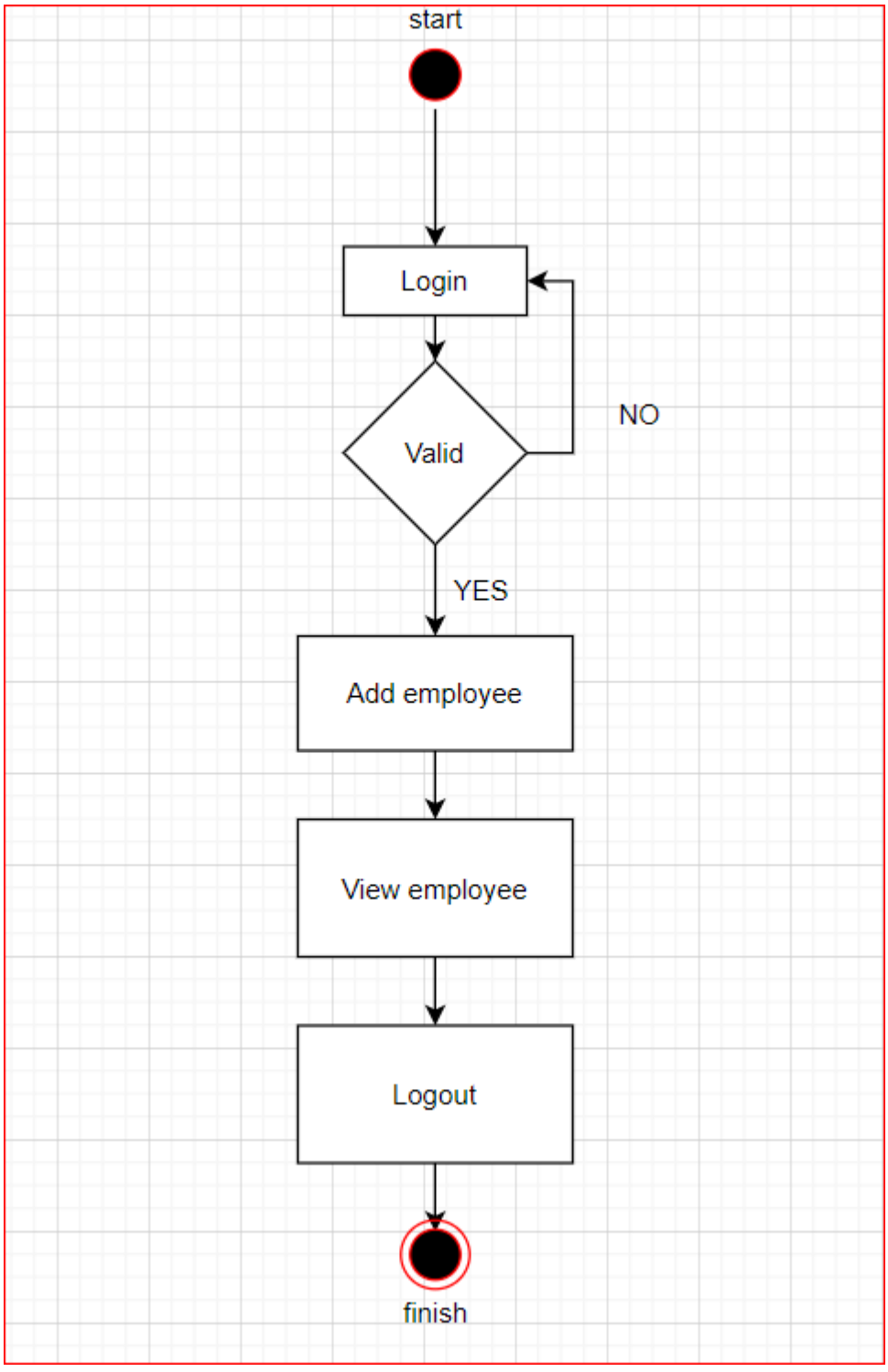
It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

The purpose of an activity diagram can be described as −

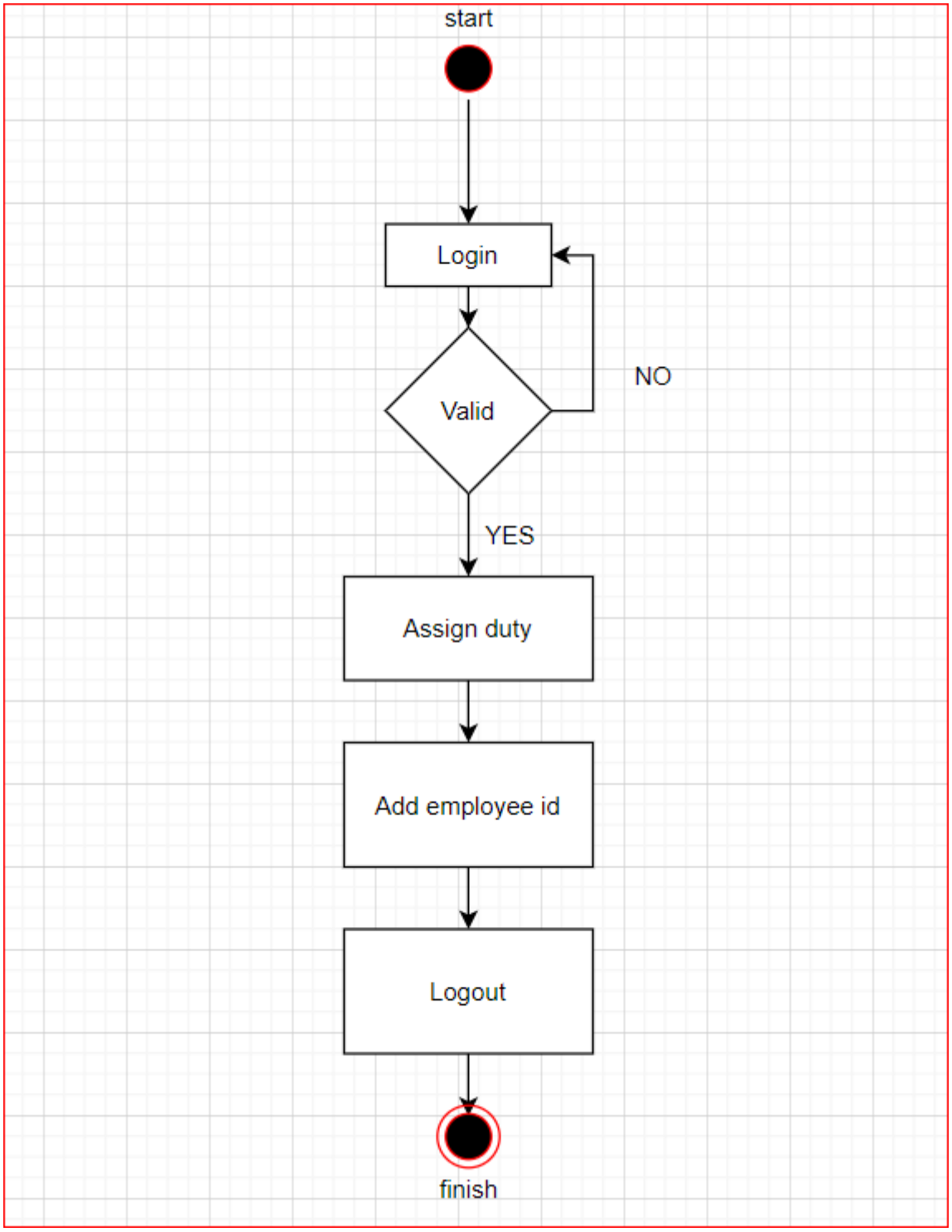
- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system.

➢ ADMIN:

- MANAGE EMPLOYEE:

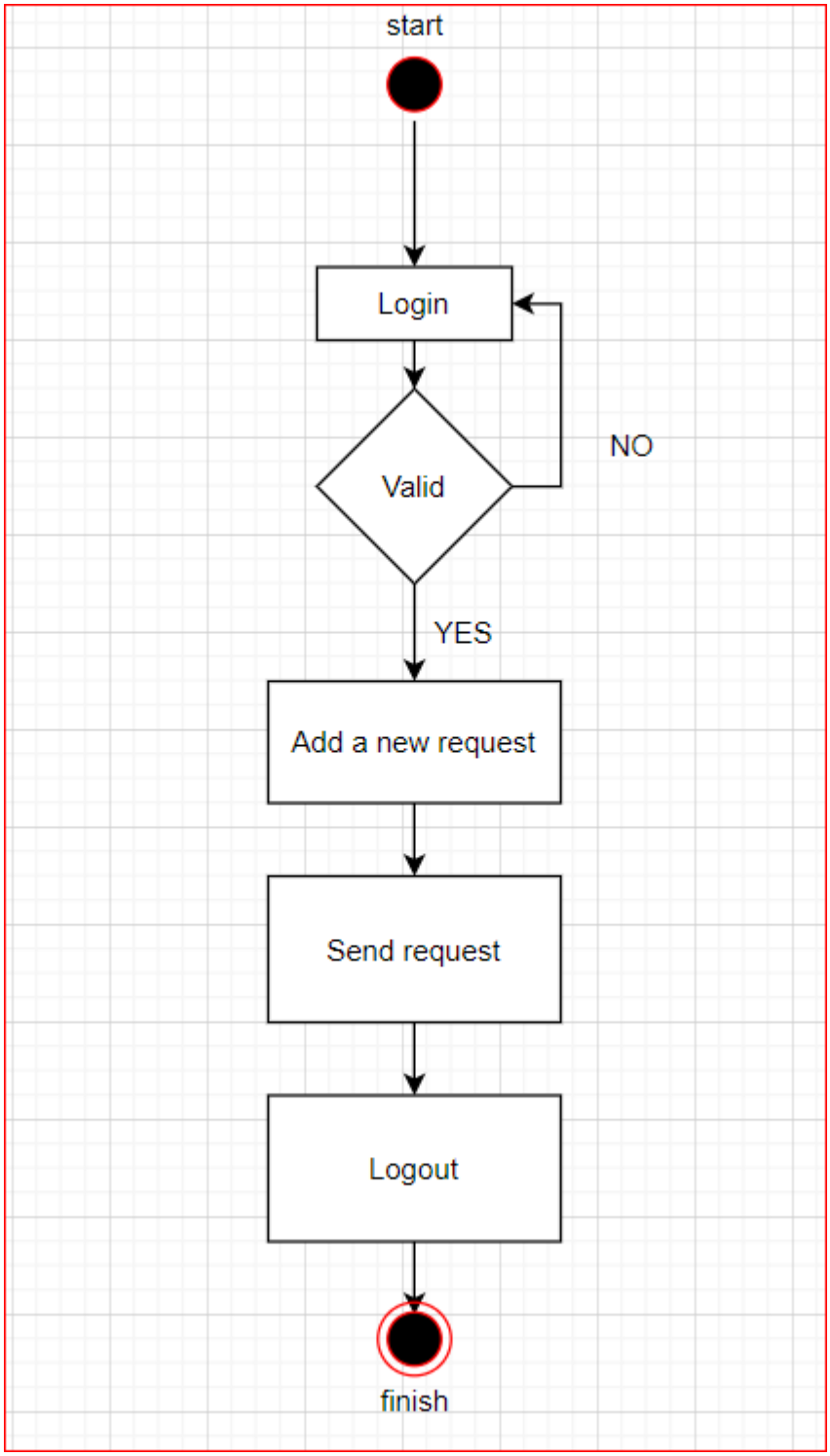


- ASSIGN DUTY:

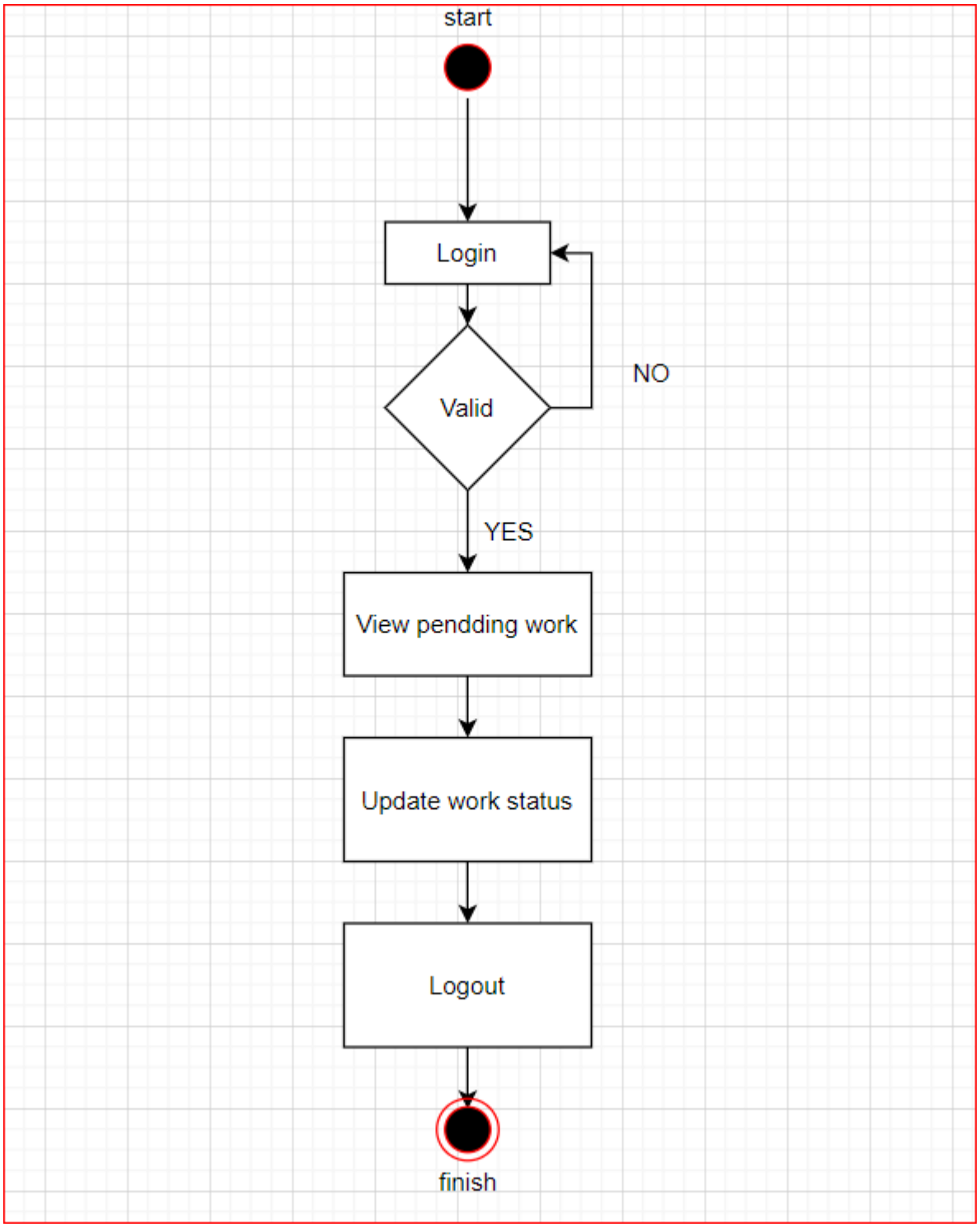

start
Login
Valid
NO
YES
Assign duty
Add employee id
Logout
finish

- CUSTOMER

3.4.3 ADD REQUEST

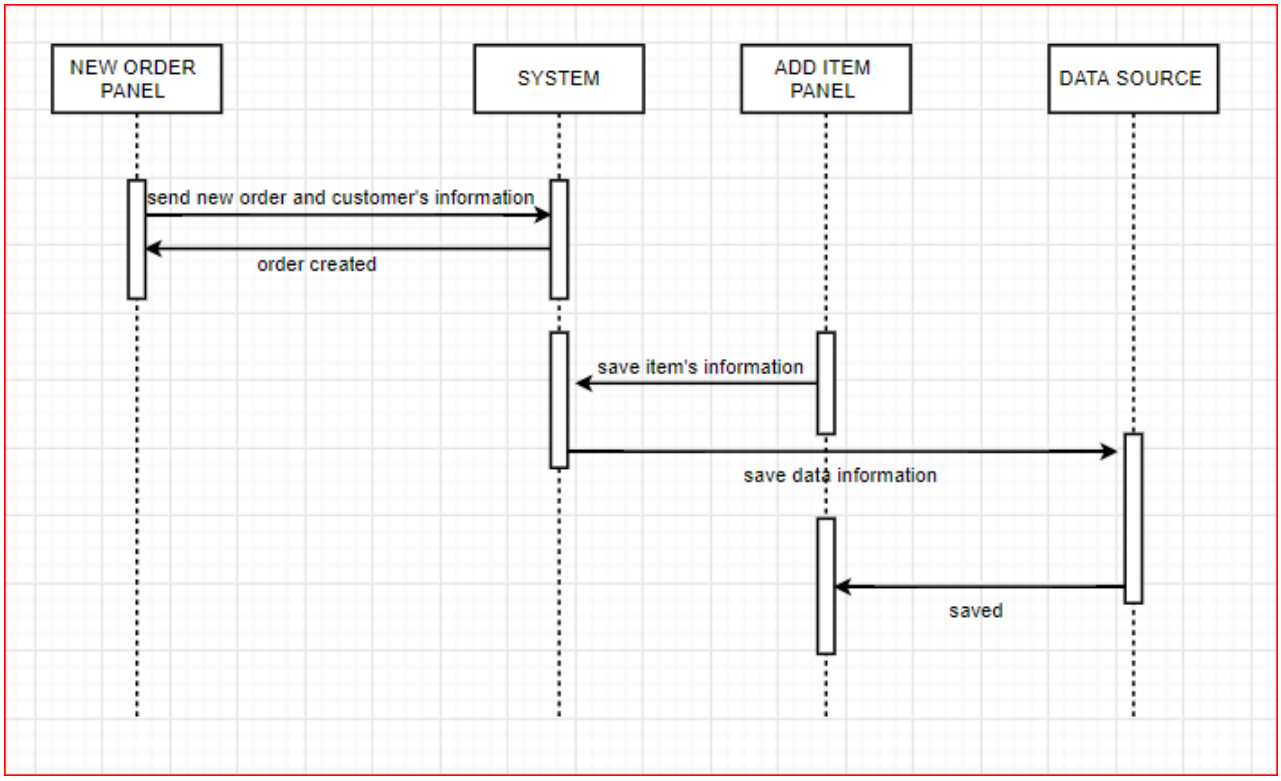


- EMPLOYEE


start
Login
Valid
NO
YES
View pendding work
Update work status
Logout
finish

SEQUENCE DIAGRAM:

A **sequence diagram** is defined by the **UML** Reference **Manual** as "a **diagram** that shows object interactions arranged in time **sequence**. In particular, it shows the objects participating in an interaction and the **sequence** of messages exchanged" . It represents objects as vertical lines and messages as arrows with labels.



USE CASE DIAGRAM:

A use case is a software and system engineering term that describes how a user uses a system to accomplish a particular goal. A use case acts as a software modelling technique that defines the features to be implemented and the resolution of any errors that may be encountered.

Three basic elements make up a use case:

• **Actors:** Actors are the type of users that interact with the system.

• **System:** Use cases capture functional requirements that specify the intended behaviour of the system.

• **Goals:** Use cases are typically initiated by a user to fulfil goals describing the activities and variants involved in attaining the goal.

The steps in designing use cases are:

• Identify the users of the system.

• For each category of users, create a user profile. This includes all roles played by the users relevant to the system.

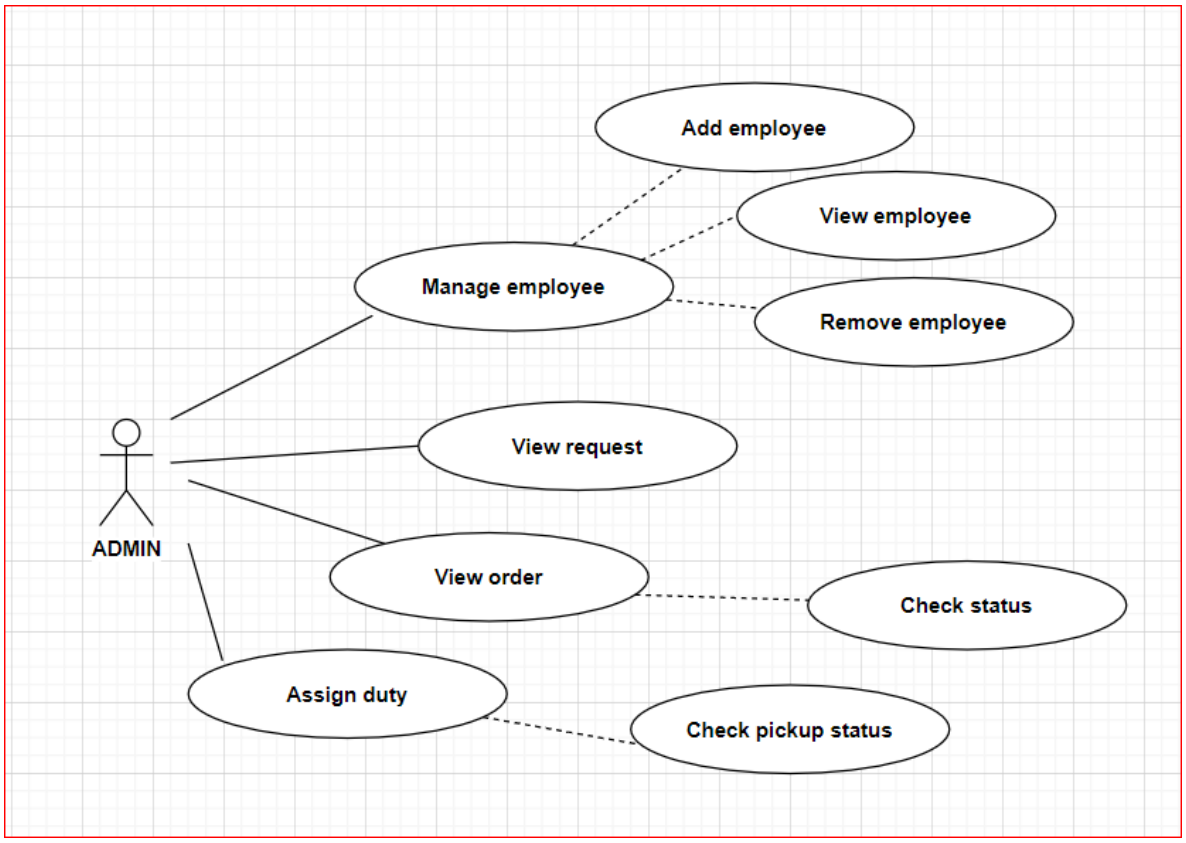
• Identify significant goals associated with each role to support the system. The system's value proposition identifies the significant role.

• Create use cases for every goal associated with a use case template and maintain the same abstraction level throughout the use case. Higher level use case steps are treated as goals for the lower level.

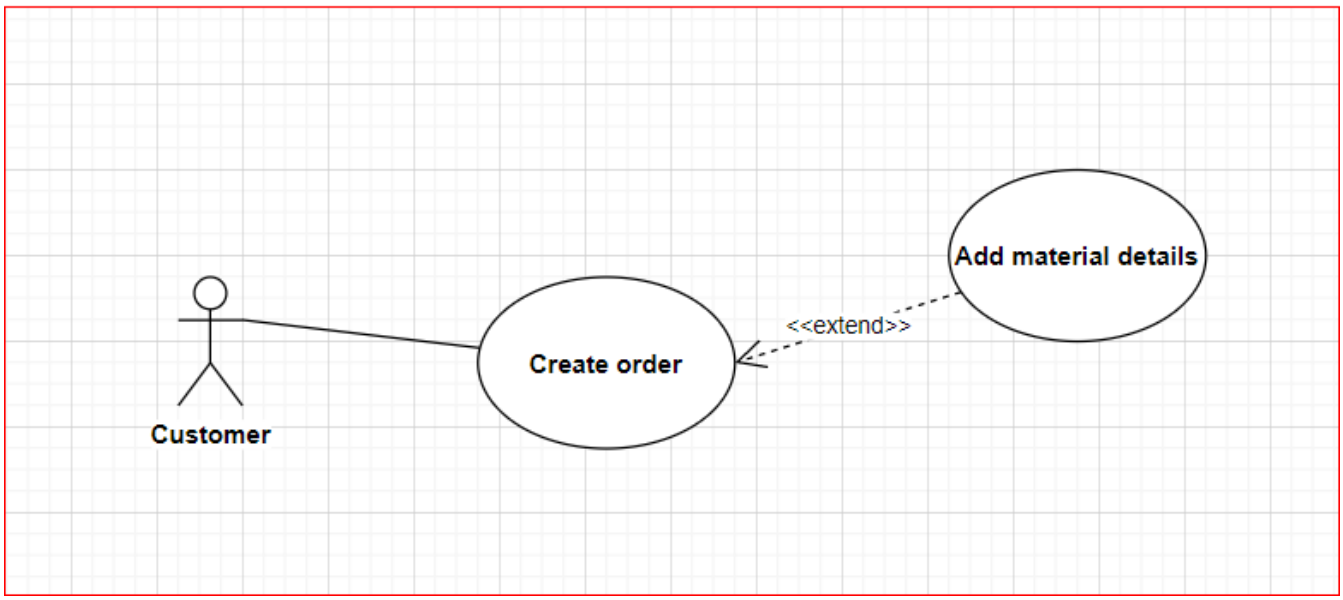
• Structure the use cases.

• Review and validate the users.

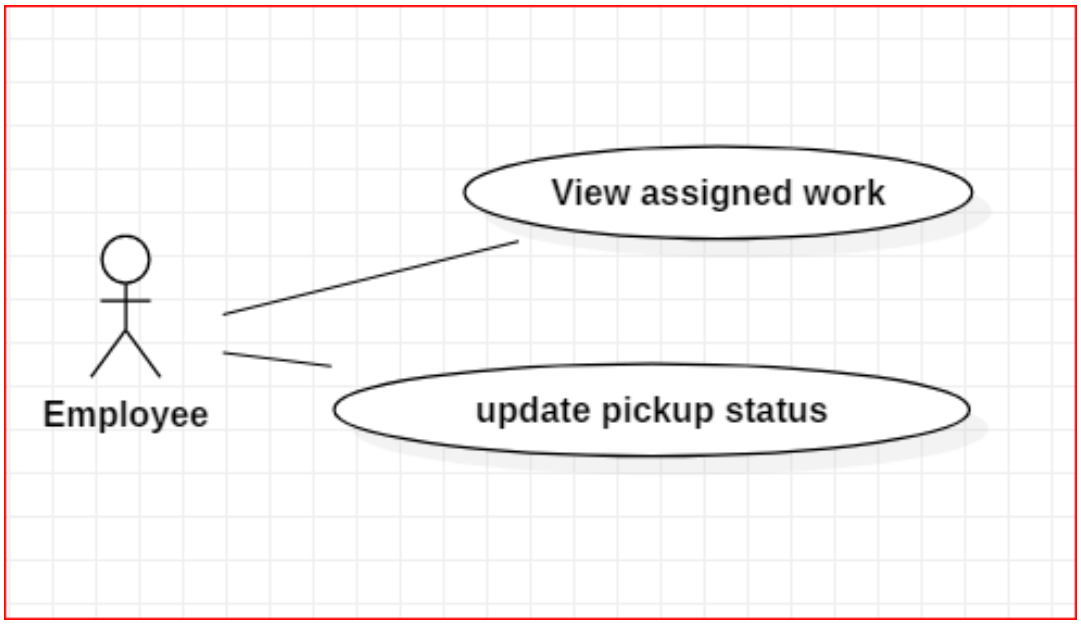
1. ADMIN:


Add employee
View employee
Manage employee
Remove employee
View request
ADMIN
View order
Check status
Assign duty
Check pickup status

2. CUSTOMER:

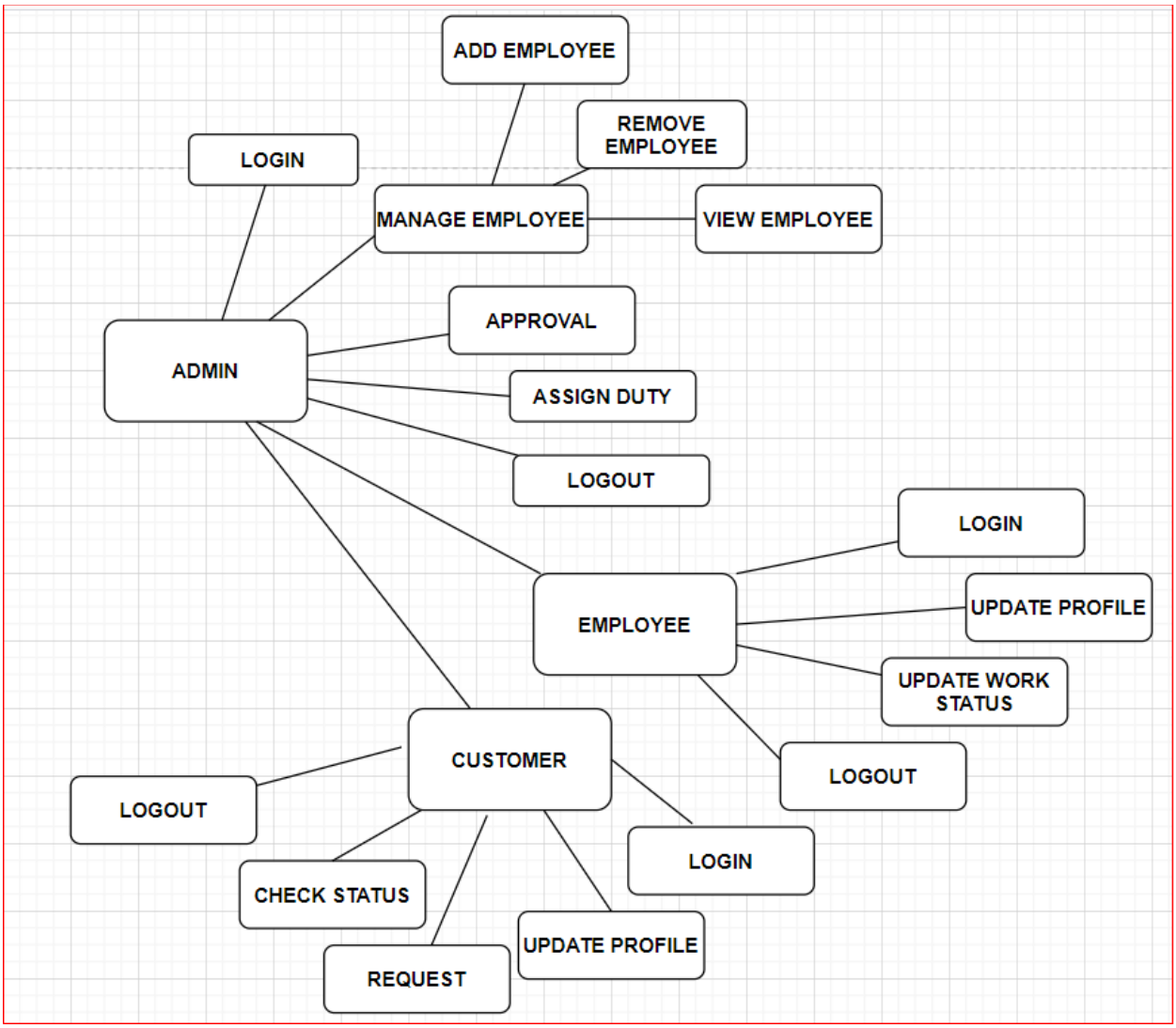


3. EMPLOYEE:



4.3.2 DATA STRUCTURES :-

TREE DIAGRAM:



4.3.3 ALGORITHM DESIGN :-

USER LOGIN:-

1. User visit the website.
2. User login into the website.
3. Enter email and password.
4. User clicks on login.

NEW USER REGISTRATION:-

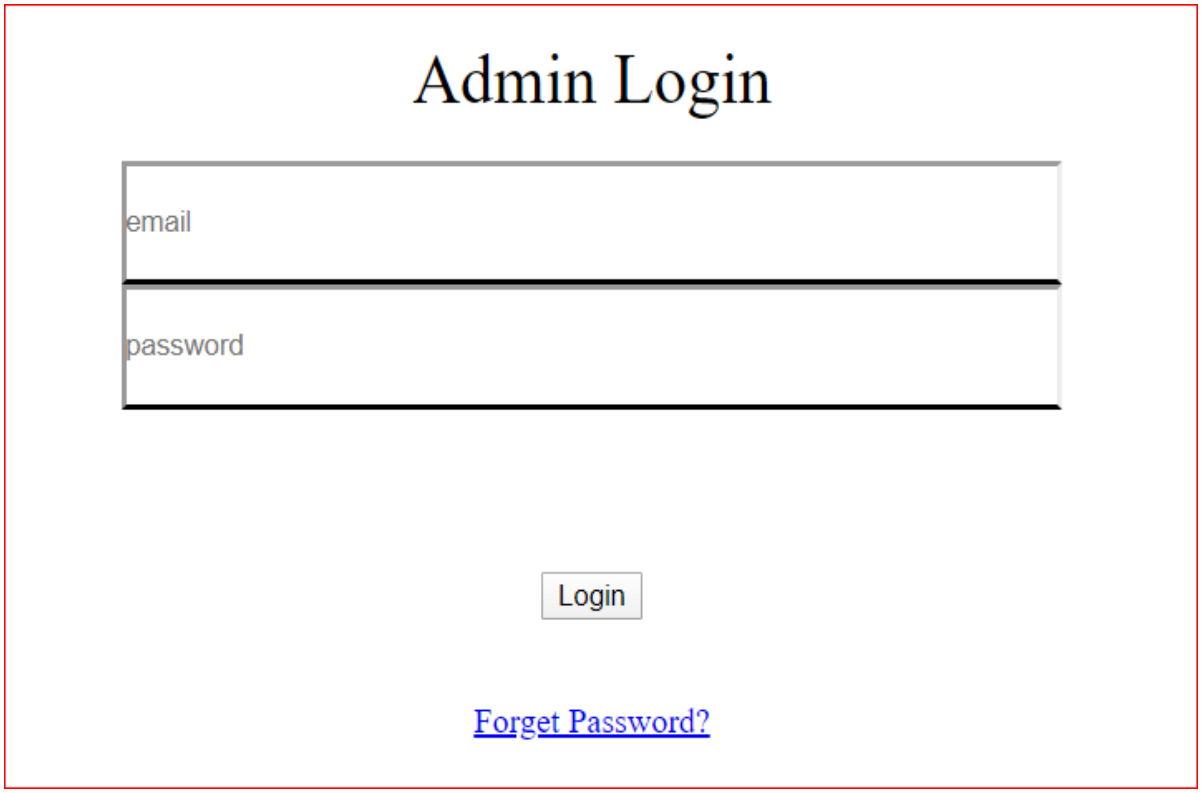
1. If user don't have Login ID and Password then customer create a new account.
2. Enter the following details in Register form:-
    - Name
    - Email
    - Address
    - City
    - Pincode
    - Contact
    - Password
3. After the filling details click on Submit.
4. After registration user gets message of registration.

ADMIN LOGIN:-

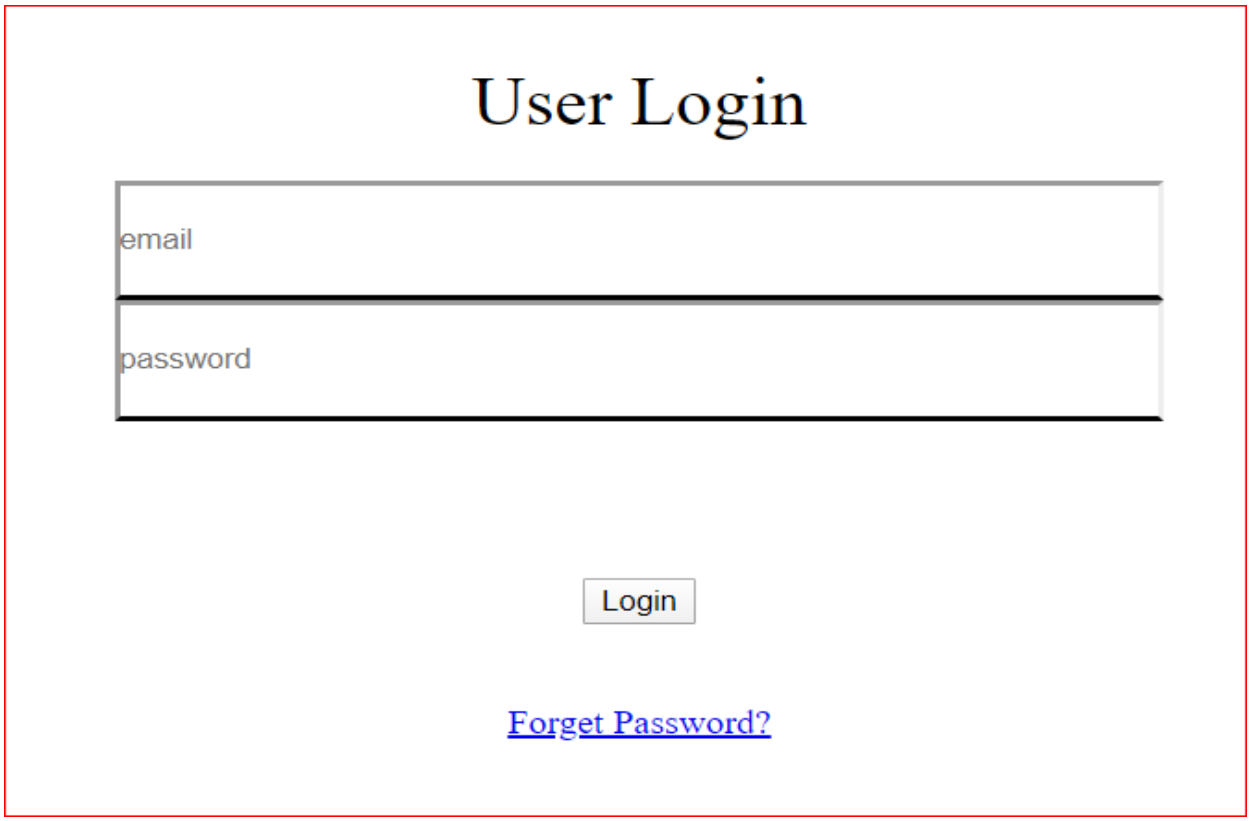
1. Admin go to website of eWaste Collection.
2. Go to login page.
3. Enter email and Password.
4. Click on Login button.
5. After login admin going to the home page.

4.4 USER INTERFACE DESIGN :-

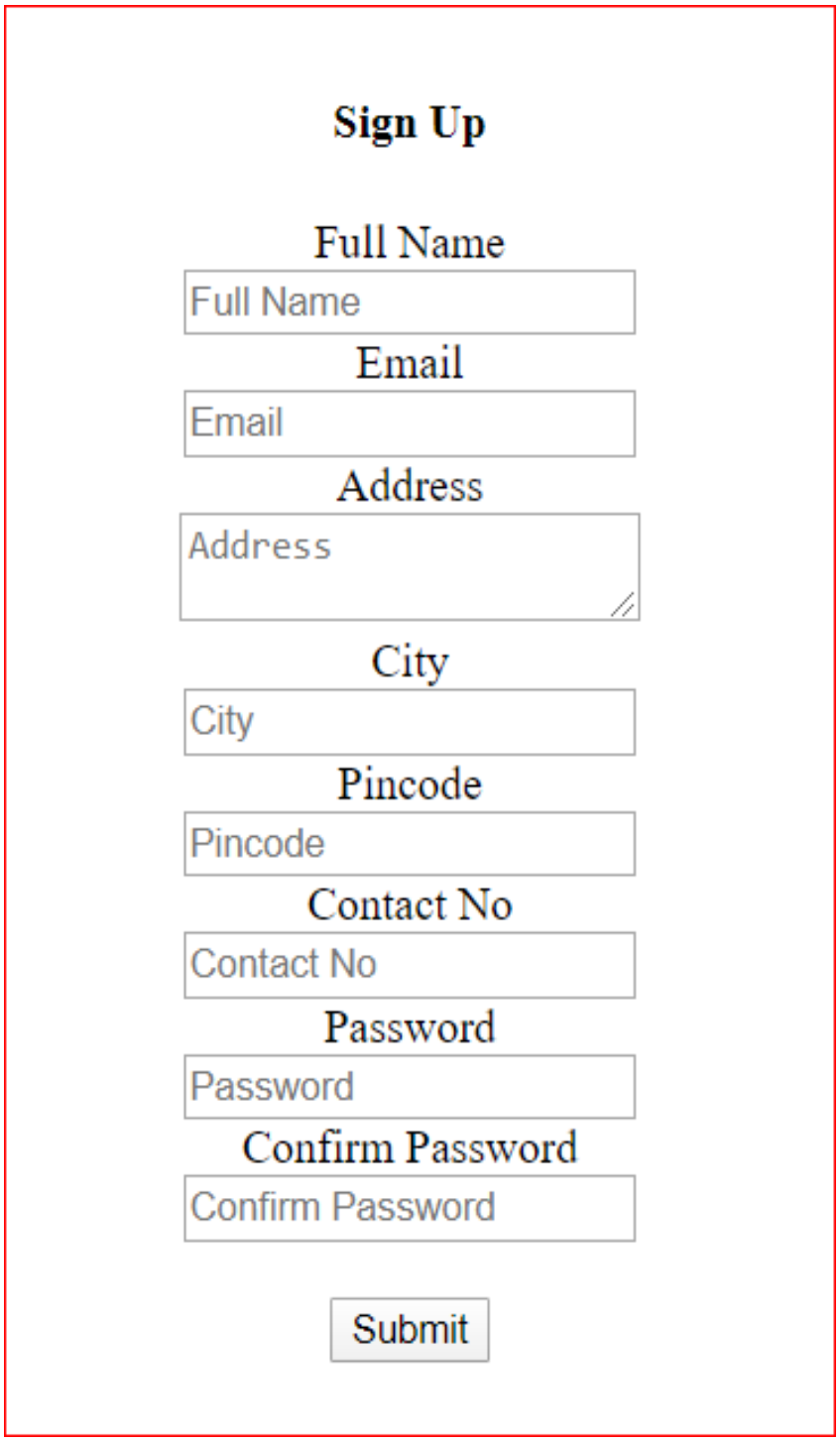
ADMIN LOGIN:



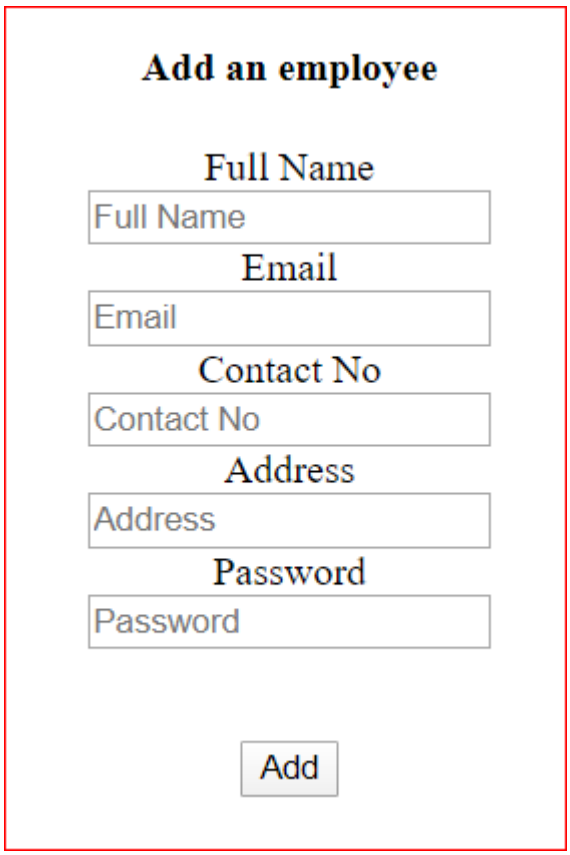
USER LOGIN:



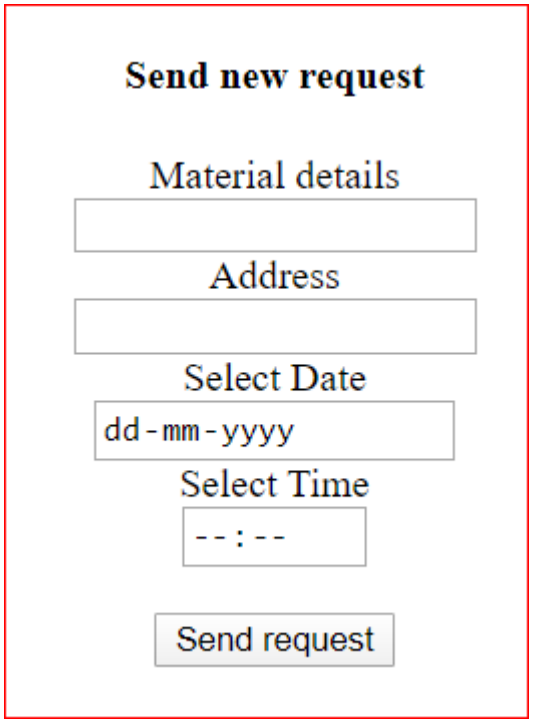
NEW USER REGISTRATION:



ADD EMPLOYEE :



NEW REQUEST:


Send new request
Material details
Address
Select Date
dd-mm-yyyy
Select Time
--:--
Send request

4.5 SECURITY ISSUES :-

AUTHENTICATION :-

Authentication is the process of determining whether someone or something is in fact who or what it declares itself to be Authentication technology provides access control for systems by checking to see if users credentials match the credentials in a database of authorized users or in a data authentication server.

Users are usually identified with a user ID and authentication is accomplished when the user provides a credential for example a password that matches with that user ID. Most of the users are most familiar with using a password which ,as a piece of information that should be known only to the user is called a knowledge authentication factor.

4.6 TEST CASES DESIGN :-

LOGIN PAGE TEST CASE :

TEST CASE ID	TEST CASE	EXPECTED RESULT
TC1	Test if registered user is able to login successfully.	User must be logged in to the web page.
TC2	Test if unregistered users is not able to login to the site	Proper error must be displayed and prompt to enter login again
TC3	Test with empty username and empty password and check if login fails	Proper error message must be displayed and prompt to enter login again
TC4	Check of the password is masked on the screen i.e., password must be in bullets or asterisks	The password field should display the characters in asterisks or bullets such that the password is not visible on the screen.

CHAPTER 5

IMPLEMENTATION & TESTING

5.1 IMPLEMENTATION APPROACH :-

- eWaste Management System is used for maintain one to one relationship with the user or customer.
- It helps to get a rid from the eWaste materials of the households or a company.
- The customer does not requires consulting any consultancy and not require any kind of payment to them.
- This software provides free service to customer to get a rid from the electronic waste material which are toxic for their health as per their requirements.
- eWaste Management System is a website where customer can send a request to admin for a service and can view their status.
- Admin can approve the customer's request and assign that duty to an employee.
- The entire project is developed using ASP.Net Visual Studio 2019 and MSSQL Server.
- ASP.Net C# is used to designing of the webpage and MSSQL is used to create a Database.
- All the modules in the project have proper validations so that the user only input appropriate input values in the input fields.
- The MSSQL is backend tool where the database resides.
- The database has required tables that are needed to store requirement information received from the Admin and user. Hence the design of the project is user friendly and easy to understand and use.
- ASP.Net is reliable, fast, easy to use and widely known. ASP.Net gives you full control of your development and can easily be used on any project, big or small.

Master Page 4 days 04/09/2019 to 07/09/2019

After detailed study about master page I tried to build one which took me about four days to build a one master page which includes all the modules of my project.

Home Page 5 days 08/09/2019 to 12/09/2019

The GUI (Graphical User Interface) which is the home page of website got completed within five days.

Login Page 8 days 13/09/2019 to 20/09/2019

Login pages contributed over 8 days in my work as each for required two days to construct.

Database 17 days 21/09/2019 to 07/10/2019

Database is an important aspect of any project which was completed building in 17 days furthermore the tables of database took 6 to 7 days to get build.

Design 51 days 08/10/2019 to 26/11/2019

Design was done within fifty-one days which includes detailed designing of project and also enhancing the design in better way.

Other Details 36 days 30/11/2019 to 06/01/2020

Other details include the error solving of each module, showing the progress to professors and also making all the necessary steps to finish the project.

Problem solving 25 days 07/01/2020 to 22/01/2020

All the errors which where occurred were noted and solved in the phase. Testing and Problem Solving are co-related and has worked out simultaneously.

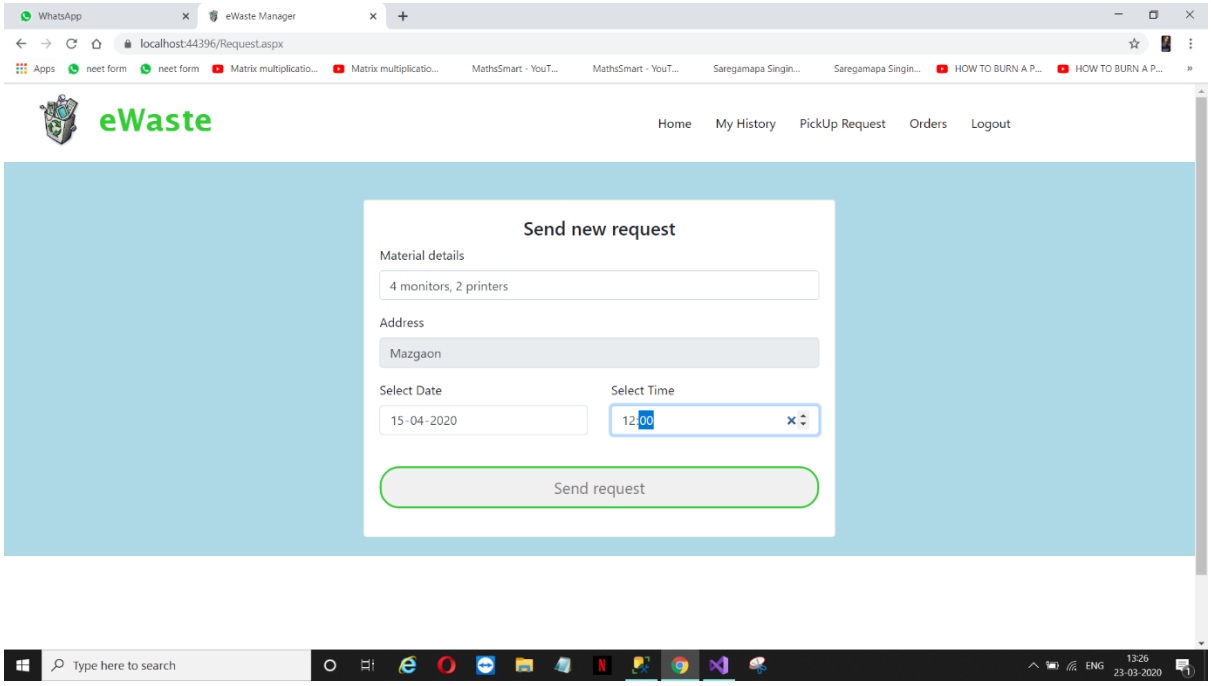
Testing 13 days 28/01/2020 to 10/02/2020

Testing includes detailed testing of the software which include Module testing, Integrated testing and also Beta testing to improve the working functionalities of the project.

5.2 CODING DETAILS & CODE EFFICIENCY:-

5.2.1 CODING DETAILS:

1. SEND REQUEST:



This page will allow the customer to send a request to admin by filling the form and successfully completing the phase of sending request.

Request.aspx :

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="Request.aspx.cs" Inherits="eWasteProject.Request" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<style>

.btn1 {

color: gray;

```
            border-style: solid;
            border-color: limegreen;
            padding: 10px 30px 10px 30px;
            border-radius: 25px;
            width: 150px;
        }

            .btn1:hover {
                color: white;
                background-color: limegreen;
                text-decoration-line: none;
            }
    </style>

</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
        <div class="container-fluid" style="background-color: lightblue">
        <div class="row">
            <div class="col-md-5 mx-auto">
                <br />
                <br />
                <div class="card">
                    <div class="card-body">
                        <div class="row">
                            <div class="col">
                                <center>
                                <h4>Send new request</h4>
                            </center>
```

```
                </div>
            </div>
            <div class="row">
                <div class="col-md-12">
                    <label>Material details</label>
                    <div class="form-group">
                        <asp:TextBox CssClass="form-control" ID="TextBox1"
runat="server"></asp:TextBox>

                    </div>
                </div>
            </div>
            <div class="row">
                <div class="col-md-12">
                    <label>Address</label>
                    <div class="form-group">
                        <asp:TextBox CssClass="form-control" ID="TextBox2"
runat="server" ReadOnly="true"></asp:TextBox>
                    </div>
                </div>
            </div>

            <div class="row">
                <div class="col-md-6">
                    <label>Select Date</label>
                    <div class="form-group">
                        <asp:TextBox class="form-control" ID="TextBox3"
runat="server" TextMode="Date"></asp:TextBox>
```

```
                    </div>
                </div>
                <div class="col-md-6">
                    <label>Select Time</label>
                    <div class="form-group">
                        <asp:TextBox class="form-control" ID="TextBox4"
runat="server" TextMode="Time"></asp:TextBox>

                    </div>
                </div>
            </div>
            <br />
            <div class="row">
                <div class="col">
                    <div class="form-group">
                        <asp:Button class="btn1 btn-block btn-lg" ID="Button1"
runat="server" Text="Send request" OnClick="Button1_Click" />
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>

    <br>
</asp:Content>
```

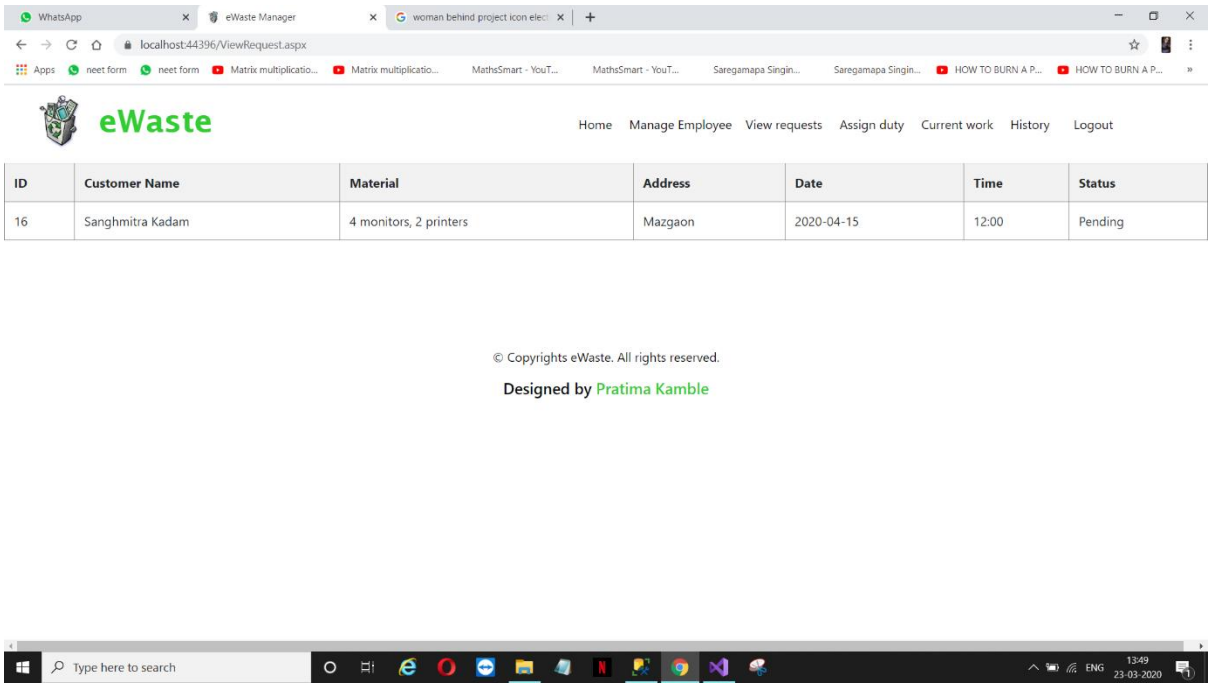
Request.aspx.cs :

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace eWasteProject
{
    public partial class Request : System.Web.UI.Page
    {
        string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;
        protected void Page_Load(object sender, EventArgs e)
        {
            if ((Session["role"].Equals("") || Session["role"].Equals("employee")))
            {
                Response.Redirect("Signin.aspx");
            }
            try
            {
                SqlConnection con = new SqlConnection(strcon);
                if (con.State == ConnectionState.Closed)
                {
                    con.Open();
                }
```

```
            SqlCommand cmd = new SqlCommand("Select * from customer where
name='" + Session["customer"].ToString() + "'", con);
            SqlDataAdapter da = new SqlDataAdapter(cmd);
            DataTable dt = new DataTable();
            da.Fill(dt);
            TextBox2.Text = dt.Rows[0]["address"].ToString();
        }
        catch (Exception ex)
        {
        }
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        try
        {
            SqlConnection con = new SqlConnection(strcon);
            if (con.State == ConnectionState.Closed)
            {
                con.Open();
            }

            SqlCommand cmd = new SqlCommand("INSERT INTO orderdetails
(custname,material,address,date,time,status) values
(@custname,@material,@address,@date,@time,@status)", con);

            cmd.Parameters.AddWithValue("@custname",
Session["customer"].ToString());
            cmd.Parameters.AddWithValue("@material", TextBox1.Text.Trim());
```

```
            cmd.Parameters.AddWithValue("@address", TextBox2.Text.Trim());
            cmd.Parameters.AddWithValue("@date", TextBox3.Text.Trim());
            cmd.Parameters.AddWithValue("@time", TextBox4.Text.Trim());
            cmd.Parameters.AddWithValue("@status", "Pending");

            cmd.ExecuteNonQuery();
            con.Close();
            Response.Write("<script>alert('Request has send');</script>");
            clear();
        }
        catch (Exception ex)
        {
            Response.Write("<script>alert('" + ex.Message + "');</script>");
        }

    }
    void clear()
    {
        TextBox1.Text = "";
        TextBox3.Text = "";
        TextBox4.Text = "";
    }
  }
}
```

2. VIEW REQUEST:



This page allows admin to view the requests send by the customers and approve it.

ViewRequest.aspx :

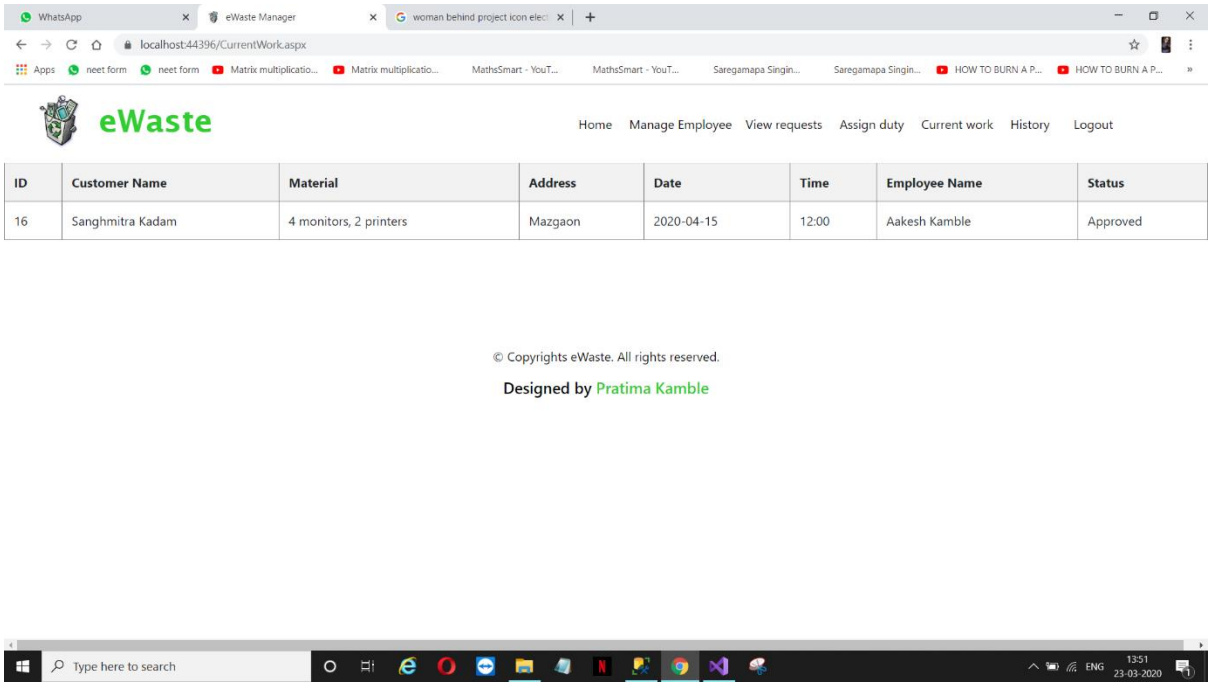
```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master"
AutoEventWireup="true" CodeBehind="ViewRequest.aspx.cs"
Inherits="eWasteProject.ViewRequest" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
   <center>
      <div class="conatiner">
         <div class="row">
            <div class="col">
               <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:ewasteConnectionString %>"
SelectCommand="SELECT * FROM [orderdetails] WHERE (([status] = @status))">
                  <SelectParameters>
                     <asp:Parameter DefaultValue="Pending" Name="status"
Type="String" />

                  </SelectParameters>
               </asp:SqlDataSource>
               <asp:GridView ID="GridView1" cssclass="table table-striped"
runat="server" AutoGenerateColumns="False" DataKeyNames="ID"
DataSourceID="SqlDataSource1">
                  <Columns>
                     <asp:BoundField DataField="ID" HeaderText="ID"
InsertVisible="False" ReadOnly="True" SortExpression="ID" />
                     <asp:BoundField DataField="custname" HeaderText="Customer
Name" SortExpression="custname" />
```

```
                    <asp:BoundField DataField="material" HeaderText="Material"
SortExpression="material" />
                    <asp:BoundField DataField="address" HeaderText="Address"
SortExpression="address" />
                    <asp:BoundField DataField="date" HeaderText="Date"
SortExpression="date" />
                    <asp:BoundField DataField="time" HeaderText="Time"
SortExpression="time" />
                    <asp:BoundField DataField="status" HeaderText="Status"
SortExpression="status" />
                </Columns>
            </asp:GridView>
        </div>
    </div>
  </div>
 </center>

</asp:Content>
```

ViewRequest.aspx.cs :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace eWasteProject
{
   public partial class ViewRequest : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         if ((Session["role"].Equals("") || Session["role"].Equals("customer") ||
Session["role"].Equals("employee")))
         {
            Response.Redirect("AdminLogin.aspx");
         }
         GridView1.DataBind();
      }
   }
}
```

3. APPROVE REQUEST:



This page allows employee to approve the requests send by the customers will they complete the order.

UpdateWork.aspx :

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master"
AutoEventWireup="true" CodeBehind="UpdateWork.aspx.cs"
Inherits="eWasteProject.UpdateWork" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
    <style>
        .btn1 {
            color: gray;
            border-style: solid;
            border-color: limegreen;
            padding: 10px 30px 10px 30px;
            border-radius: 25px;
            width: 150px;
        }
```

```
        .btn1:hover {
            color: white;
            background-color: limegreen;
            text-decoration-line: none;
        }
    </style>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
    <div class="container-fluid" style="background-color: lightblue">
      <div class="row">
        <div class="col-md-5 mx-auto">
            <br />
            <br />
            <div class="card">
              <div class="card-body">
                <div class="row">
                  <div class="col">
                    <center>
                  <h4>Update Status</h4>
                </center>
                  </div>
                </div>
                <br />
                <center>
                <div class="row">
                  <div class="col-md-12">
                    <div class="form-group">
```

```
                        <span>Order ID: </span><asp:TextBox ID="TextBox1"
runat="server" Width="84px"></asp:TextBox>
                        <asp:Button class="btn" ID="Button2" runat="server"
Text="Search" OnClick="Button2_Click"  />
                    </div>
                </div>
            </div>
            </center>
            <div class="row">
                <div class="col-md-12">
                    <label>Customer Name</label>
                    <div class="form-group">
                        <asp:TextBox CssClass="form-control" ID="TextBox2"
runat="server" ReadOnly="true"></asp:TextBox>
                    </div>
                </div>
            </div>
            <div class="row">
                <div class="col-md-12">
                    <label>Material</label>
                    <div class="form-group">
                     <asp:TextBox CssClass="form-control" ID="TextBox3"
runat="server" ReadOnly="true"></asp:TextBox>
                    </div>
                </div>
            </div>
            <div class="row">
                <div class="col-md-12">
                    <label>Address</label>
```

```
                    <div class="form-group">
                        <asp:TextBox class="form-control" ID="TextBox4"
runat="server" ReadOnly="true"></asp:TextBox>
                    </div>
                </div>
            </div>
            <div class="row">
                <div class="col-md-6">
                    <label>Date</label>
                    <div class="form-group">
                        <asp:TextBox class="form-control" ID="TextBox5"
runat="server" ReadOnly="true"></asp:TextBox>
                    </div>
                </div>
                <div class="col-md-6">
                    <label>Time</label>
                    <div class="form-group">
                        <asp:TextBox class="form-control" ID="TextBox6"
runat="server" ReadOnly="true"></asp:TextBox>
                    </div>
                </div>
            </div>
            <div class="row">
                <div class="col">
                    <div class="form-group">
                        <asp:Button class="btn1 btn-block btn-lg" ID="Button1"
runat="server" Text="Successful" OnClick="Button1_Click"  />
                    </div>
                </div>
```

```
                    </div>
                </div>
            </div>
            <br>
        </div>
    </div>
</div>
</asp:Content>
```

UpdateWork.aspx.cs :

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace eWasteProject
{
    public partial class UpdateWork : System.Web.UI.Page
    {
        string strcon = ConfigurationManager.ConnectionStrings["con"].ConnectionString;
        protected void Page_Load(object sender, EventArgs e)
        {
            if ((Session["role"].Equals("") || Session["role"].Equals("customer")))
            {
                Response.Redirect("EmpLogin.aspx");
            }
        }
        protected void Button2_Click(object sender, EventArgs e)
        {
            string st = "Approved";
            try
            {
                SqlConnection con = new SqlConnection(strcon);
```

```
            if (con.State == ConnectionState.Closed)
            {
               con.Open();
            }

            SqlCommand cmd = new SqlCommand("select * from orderdetails where
ID='" + TextBox1.Text.Trim() + "' AND empname='" + Session["empname"].ToString()
+ "' AND status='" + st + "'", con);
            SqlDataAdapter da = new SqlDataAdapter(cmd);
            DataTable dt = new DataTable();
            da.Fill(dt);
            TextBox2.Text = dt.Rows[0]["custname"].ToString();
            TextBox3.Text = dt.Rows[0]["material"].ToString();
            TextBox4.Text = dt.Rows[0]["address"].ToString();
            TextBox5.Text = dt.Rows[0]["date"].ToString();
            TextBox6.Text = dt.Rows[0]["time"].ToString();

        }
        catch (Exception ex)
        {
            Response.Write("<script>alert('Choose Correct Order ID');</script>");
        }

    }
    protected void Button1_Click(object sender, EventArgs e)
    {

        try
        {
```

```
            SqlConnection con = new SqlConnection(strcon);
            if (con.State == ConnectionState.Closed)
            {
               con.Open();
            }
            SqlCommand cmd = new SqlCommand("update orderdetails set
status=@status where ID='" + TextBox1.Text.Trim() + "'", con);



            cmd.Parameters.AddWithValue("@status", "Successful");


            cmd.ExecuteNonQuery();
            con.Close();
            Response.Write("<script>alert('Updated Successfully');</script>");
        }
        catch (Exception ex)
        {
            Response.Write("<script>alert('" + ex.Message + "');</script>");
        }

    }


}
  }
```

5.2.2 CODE EFFICIENCY:

Code efficiency is a broad term used to depict the reliability, speed and programming methodology used in developing codes for an application. Code efficiency is directly linked with algorithmic efficiency and the speed of runtime execution for software. It is the key element in ensuring high performance. The goal of code efficiency is to reduce resource consumption and completion time as much as possible with minimum risk to the business or operating environment. The software product quality can be accessed and evaluated with the help of the efficiency of the code used.

Code efficiency plays a significant role in applications in a high-execution-speed environment where performance and scalability are paramount.

One of the recommended best practices in coding is to ensure good code efficiency. Well-developed programming codes should be able to handle complex algorithms. Recommendations for code efficiency include:

- To remove unnecessary code or code that goes to redundant processing.
- To make use of optimal memory and nonvolatile storage.
- To ensure the best speed or run time for completing the algorithm.
- To make use of reusable components wherever possible.
- To make use of error and exception handling at all layers of software, such as the user interface, logic and data flow.
- To create programming code that ensures data integrity and consistency.
- To develop programming code that's compliant with the design logic and flow.
- To make use of coding practices applicable to the related software.
- To optimize the use of data access and data management practices.
- To use the best keywords, data types and variables, and other available programming concepts to implement the related algorithm.

CODE OPTIMIZATION:

- Optimization is a program transformation technique, which tries to improve the code by making it consume less resources (i.e. CPU, Memory) and deliver high speed.
- In optimization, high-level general programming constructs are replaced by very efficient low-level programming codes. A code optimizing process must follow the three rules given below:
- The output code must not, in any way, change the meaning of the program.
- Optimization should increase the speed of the program and if possible, the program should demand less number of resources.
- Optimization should itself be fast and should not delay the overall compiling process.
- Efforts for an optimized code can be made at various levels of compiling the process.
- At the beginning, users can change/rearrange the code or use better algorithms to write the code.
- After generating intermediate code, the compiler can modify the intermediate code by address calculations and improving loops.
- While producing the target machine code, the compiler can make use of memory hierarchy and CPU registers.
- Optimization can be categorized broadly into two types : machine independent and machine dependent.

5.3 TESTING APPROACH:-

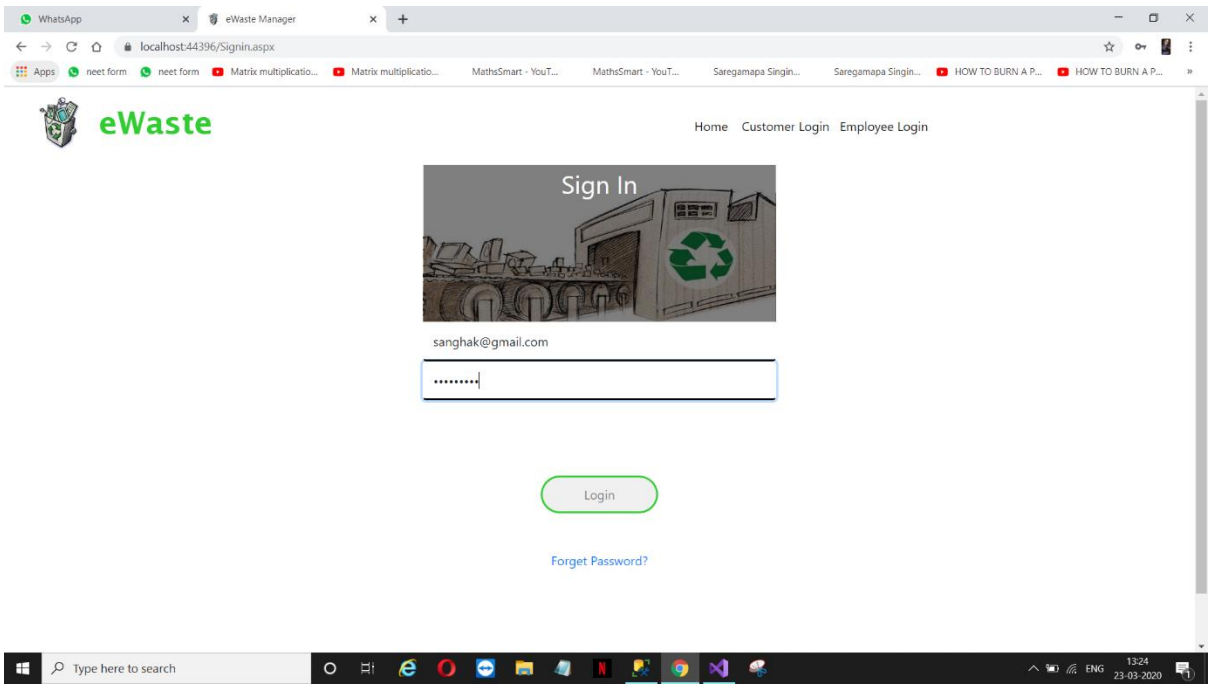
A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

- Proactive - An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
- Reactive - An approach in which the testing is not started until after design and coding are completed.

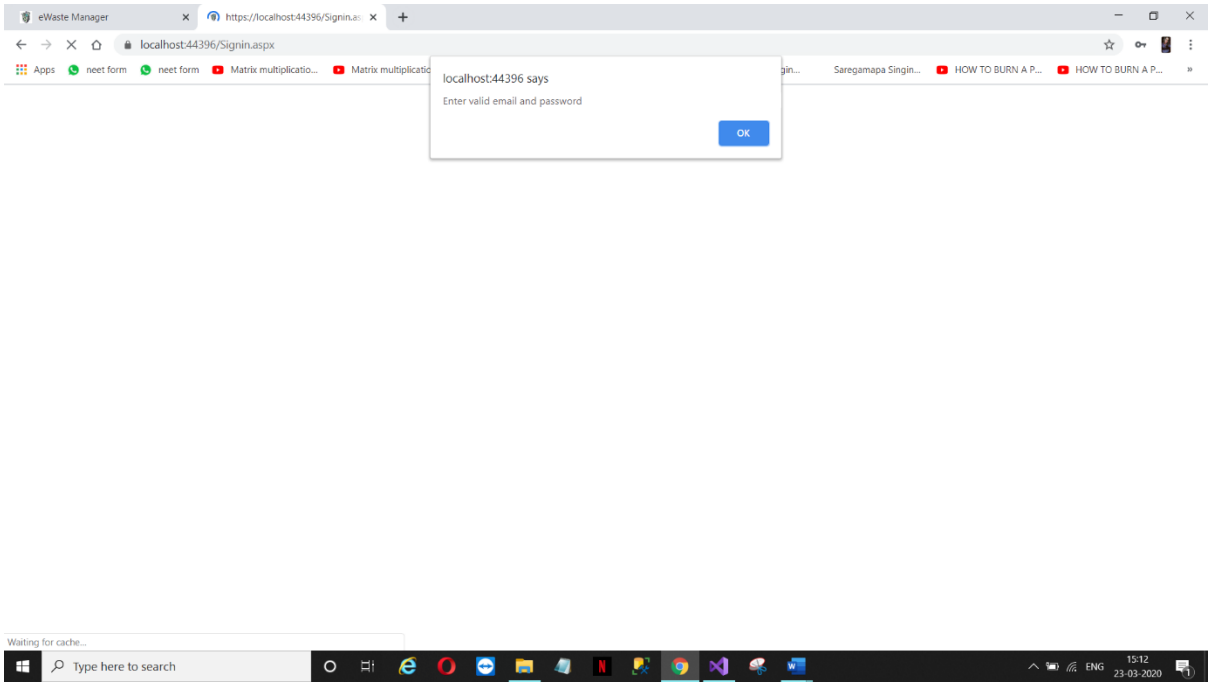
5.3.1 UNIT TESTING:

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input.

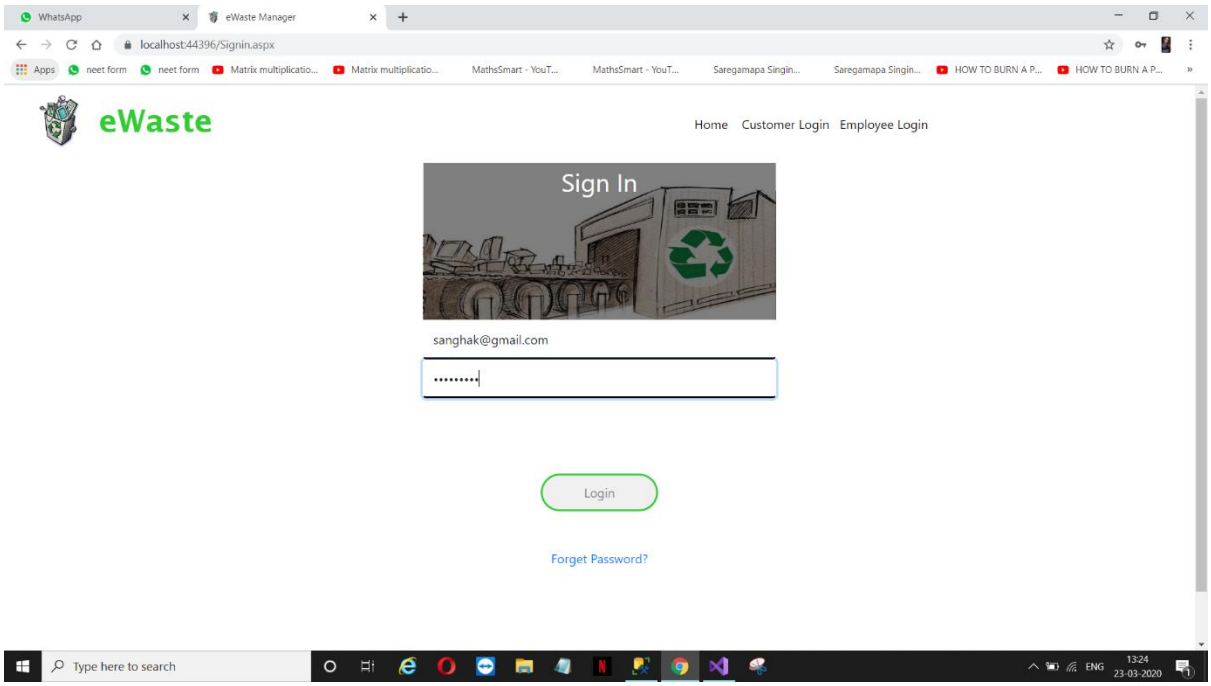
Customer will enter the email and password to login in to the website.



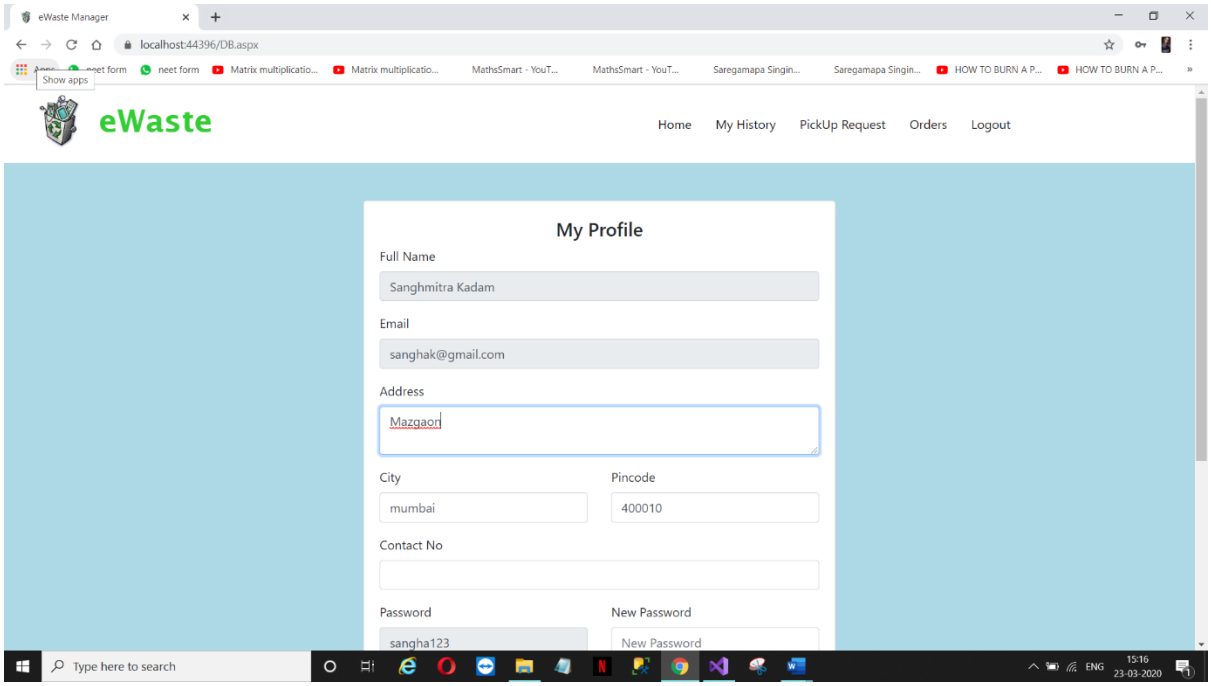
Customer will get a message as “Enter valid email and password” if he/she will enter wrong details at the time of login.



Customer will enter the email and password to login in to the website.



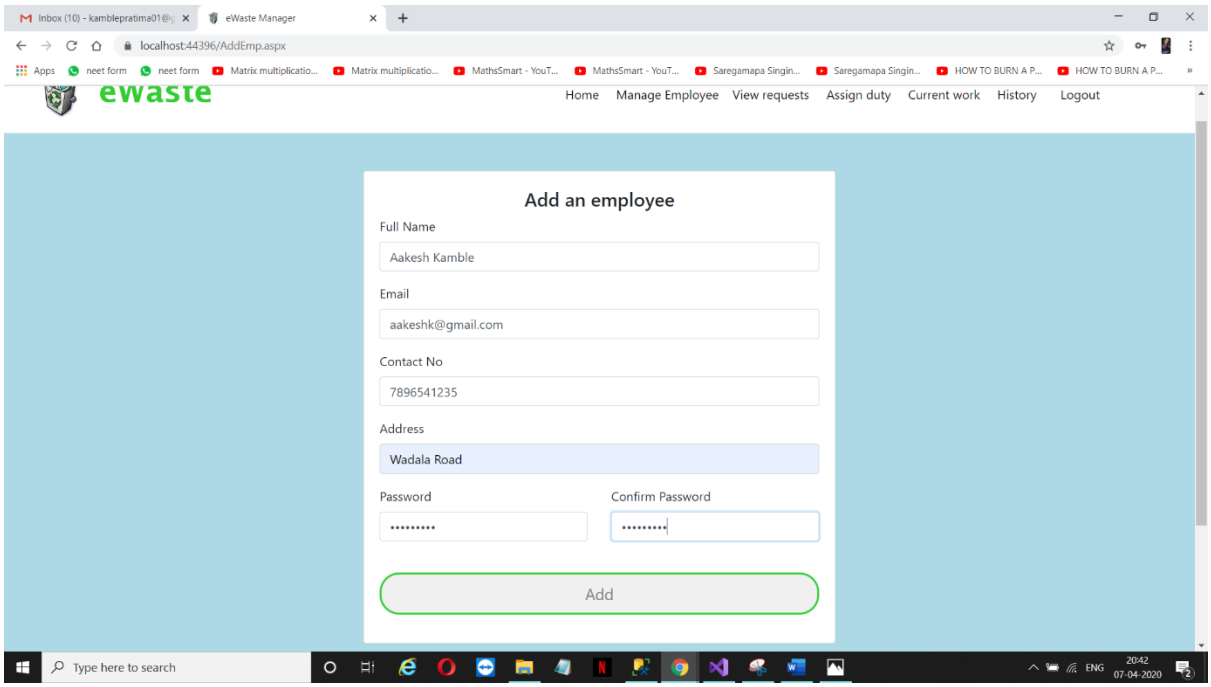
Customer will be successfully logged in.



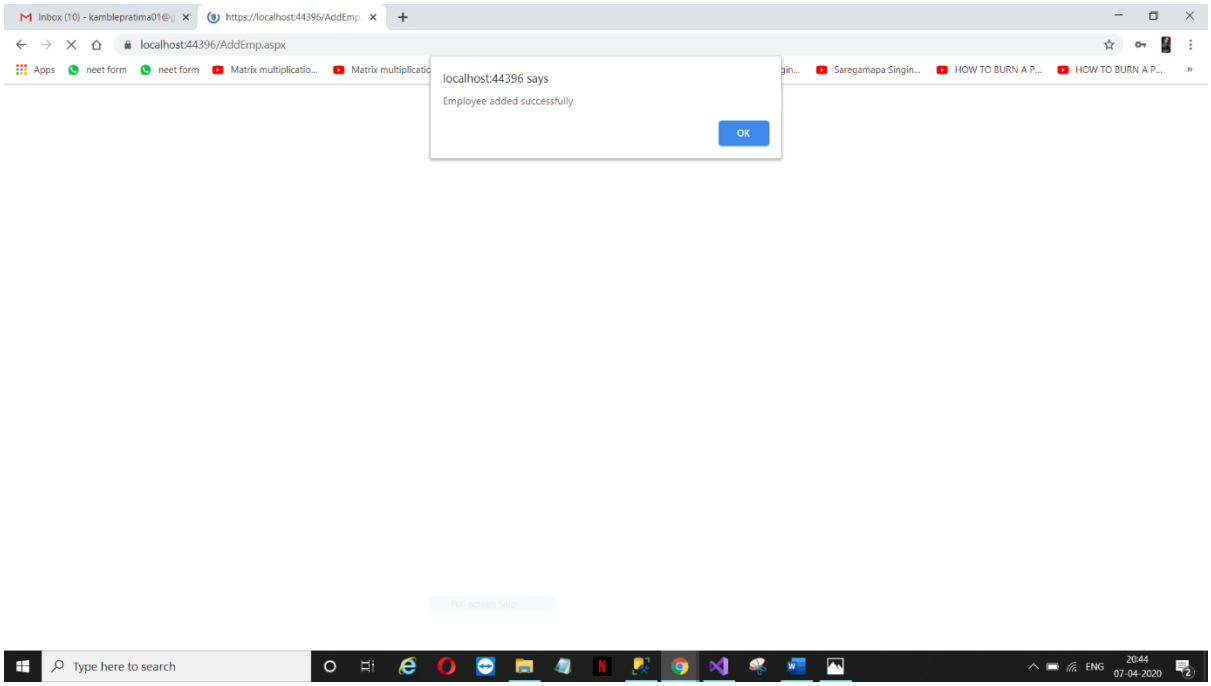
5.3.2 Integrated Testing:-

Integrated testing is a testing in which a group of components are combined to produced output. Also, the interaction between software and hardware is tested in integrated testing if software and hardware components have any relation. It may fall under the both white box testing and black box testing.

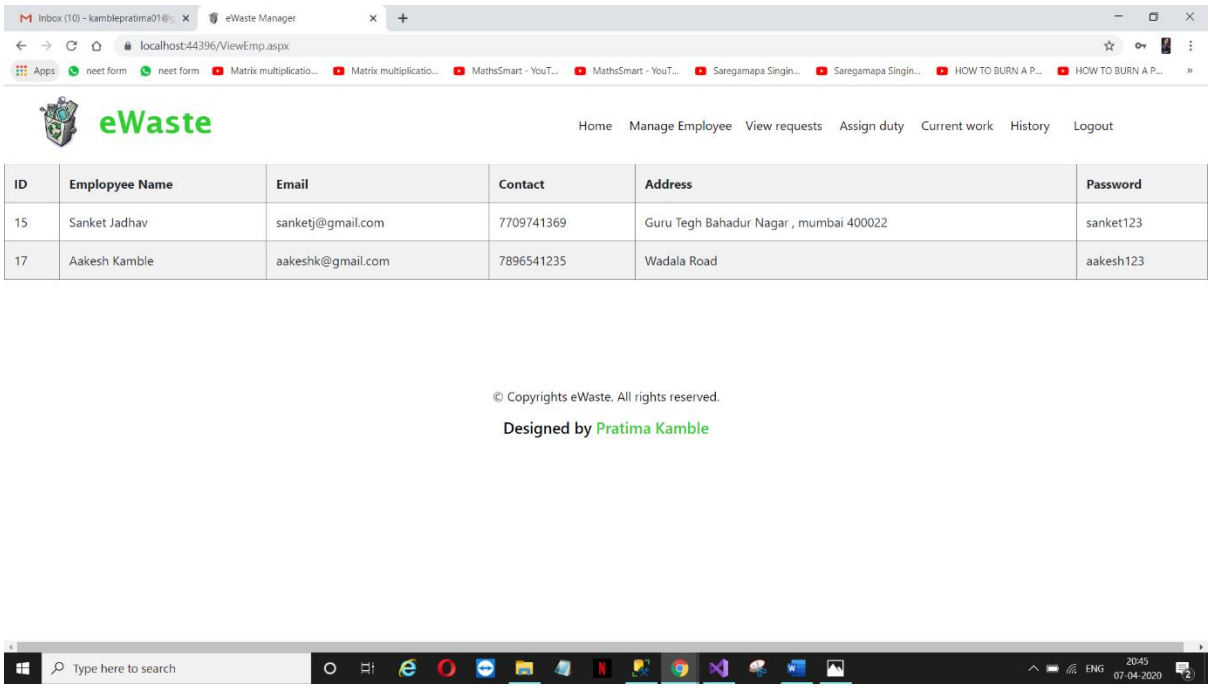
Admin will add an employee.



Employee added successfully.



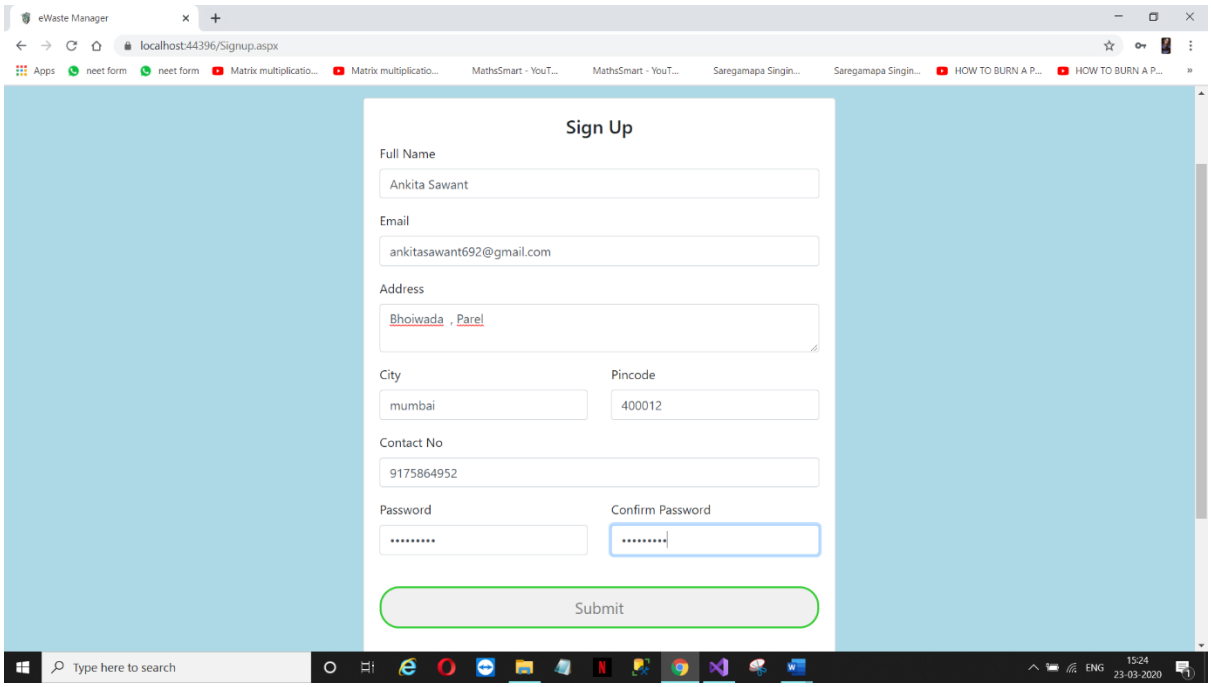
Admin can see list of employees



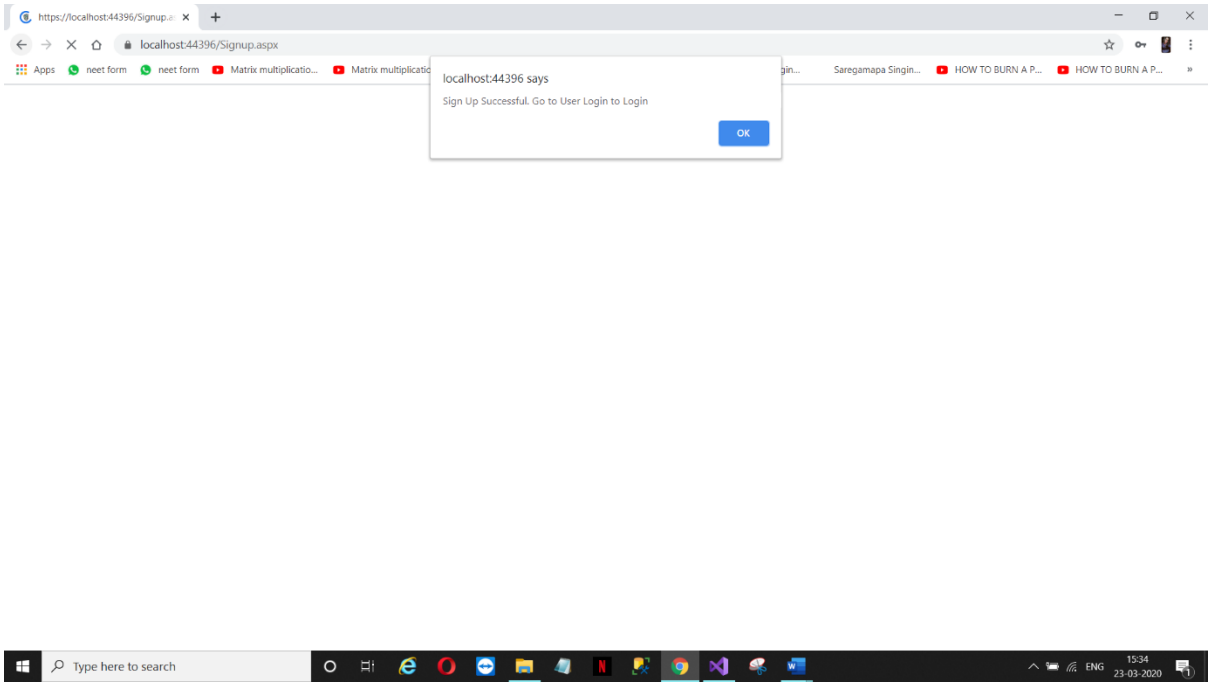
5.3.3 Beta Testing :-

Beta testing is the testing which is done by end users, a team outside the development , or publicly releasing full pre-version of the product which is known as beta version. The aim of beta testing is to cover unexpected errors. It falls under the class of the black box testing.

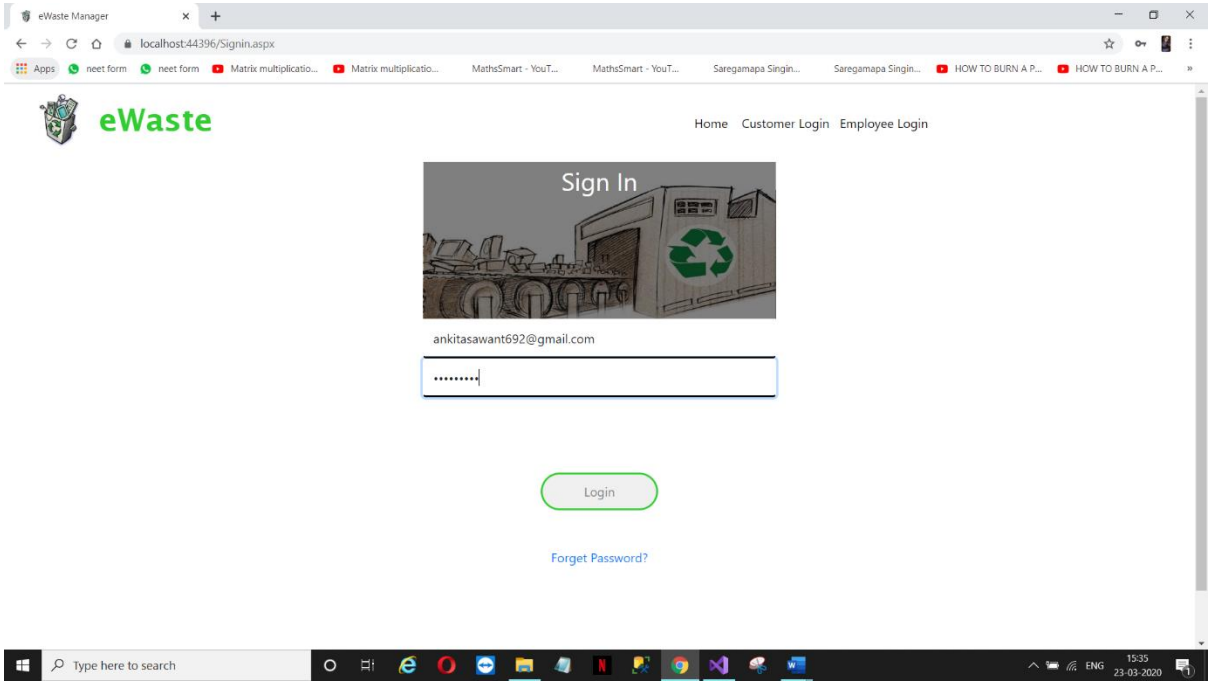
New customer registration for creating an account.



Code has send by admin to customer.

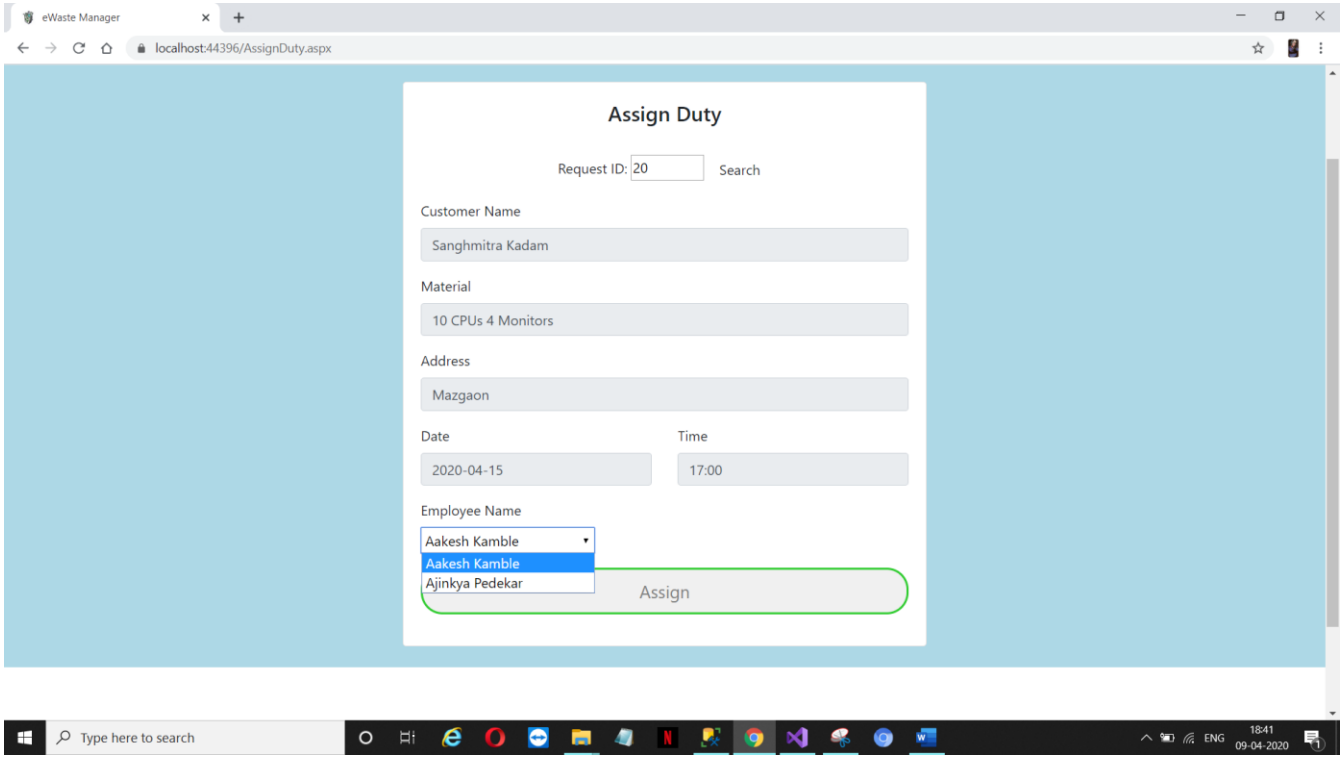


Customer login through email and password successfully.

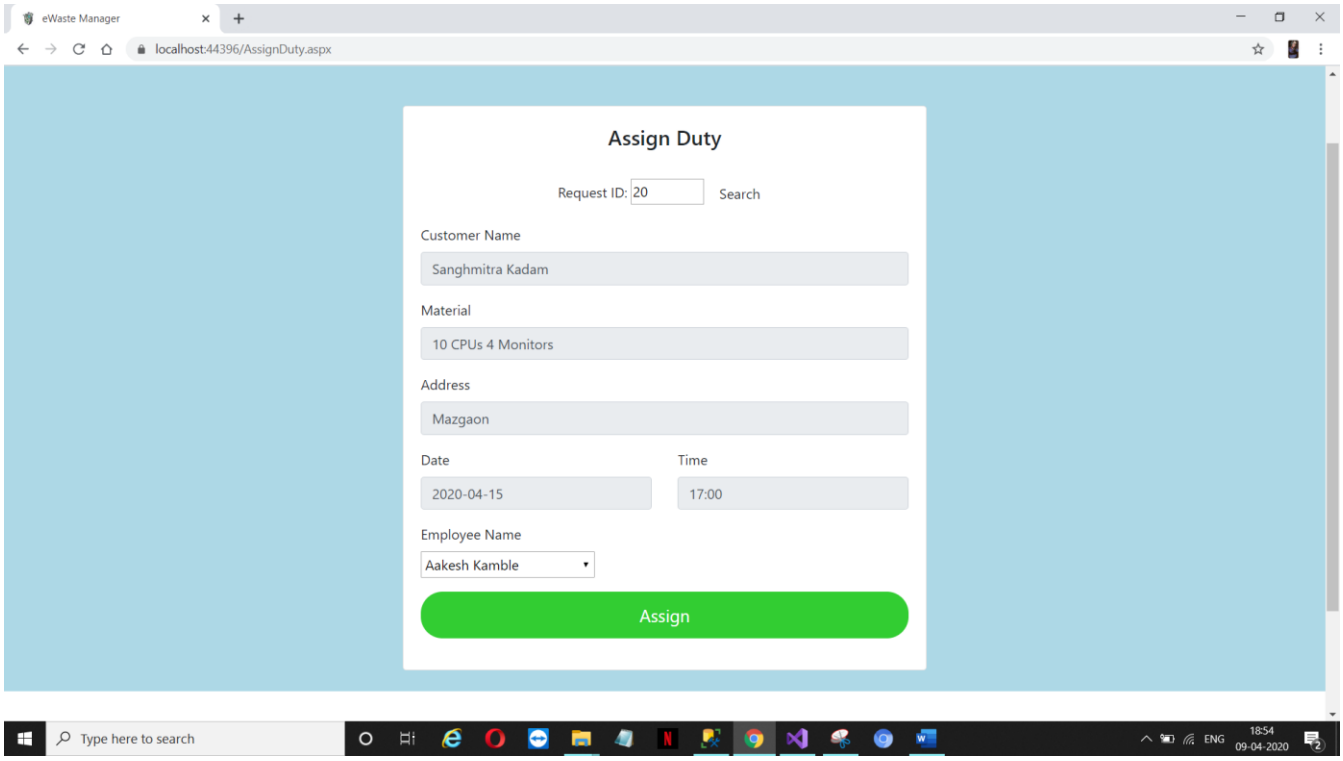


5.4 MODIFICATIONS & IMPROVEMENTS:-

In “Assign Duty” page, it was showing all the employees even if they are doing other duty.



Now it is only showing the employee who is available now.



5.5 TEST CASES:-

LOGIN PAGE TEST CASES:

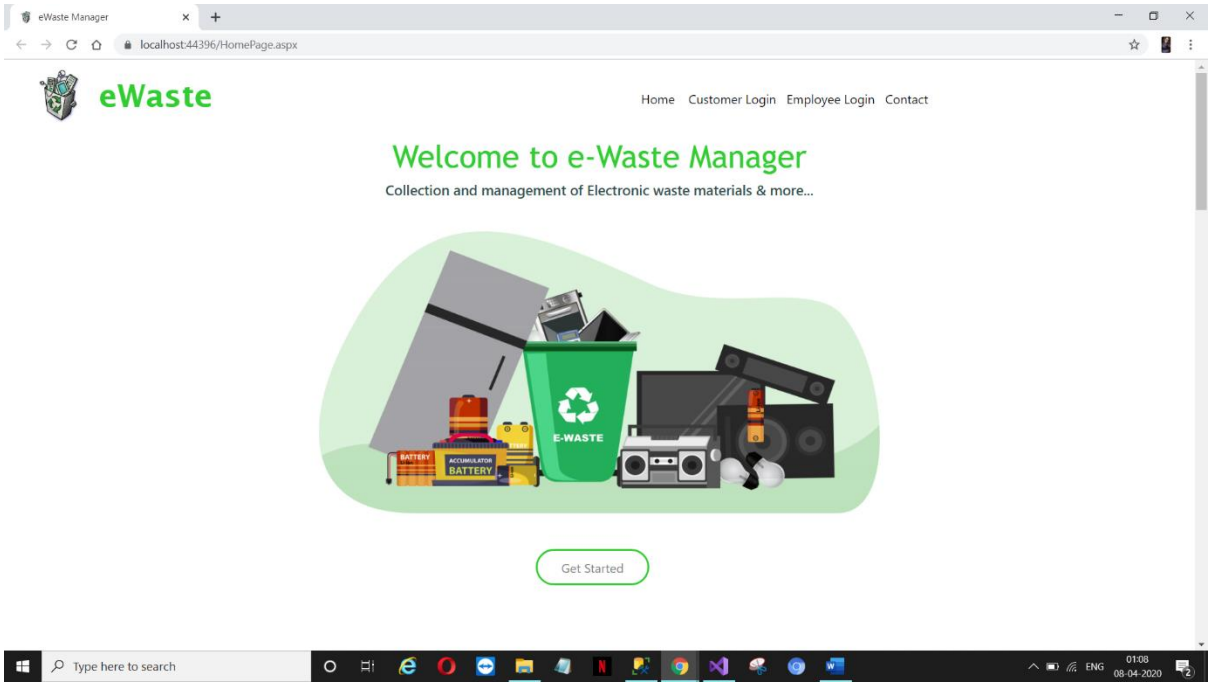
TEST CASE ID	TEST CASE	EXPECTED RESULT	ACTUAL RESULT	REMARK
TC1	Test if registered user is able to login successfully.	User must be logged in to the web page.	User is successfully logged in to the web page	Pass
TC2	Test if unregistered users is not able to login to the site	Proper error must be displayed and prompt to enter login again	Proper error is displayed and prompts to enter login details again	Pass
TC3	Test with empty username and empty password and check if login fails	Proper error message must be displayed and prompt to enter login again	Proper error is displayed and prompts to enter login again	Pass
TC4	Check of the password is masked on the screen i.e., password must be in bullets or asterisks	The password field should display the characters in asterisks or bullets such that the password is not visible on the screen.	The password field display the characters in asterisks.	Pass

CHAPTER 6

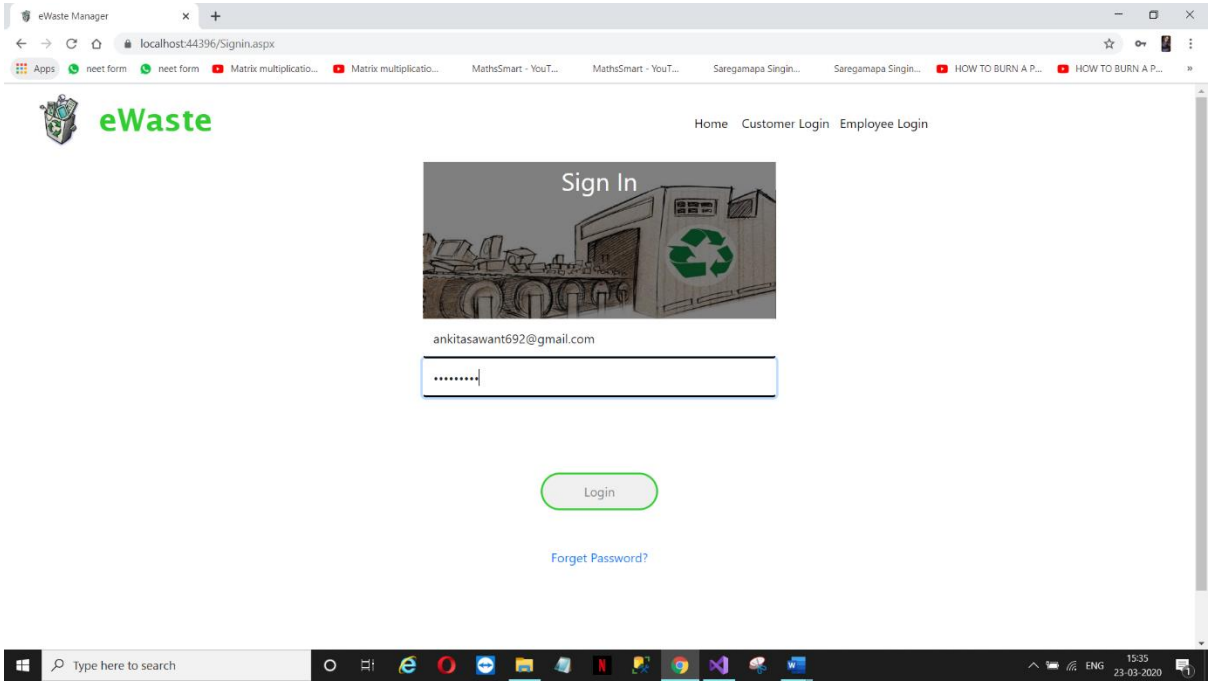
RESULTS AND DISCUSSION

6.1 Test Reports

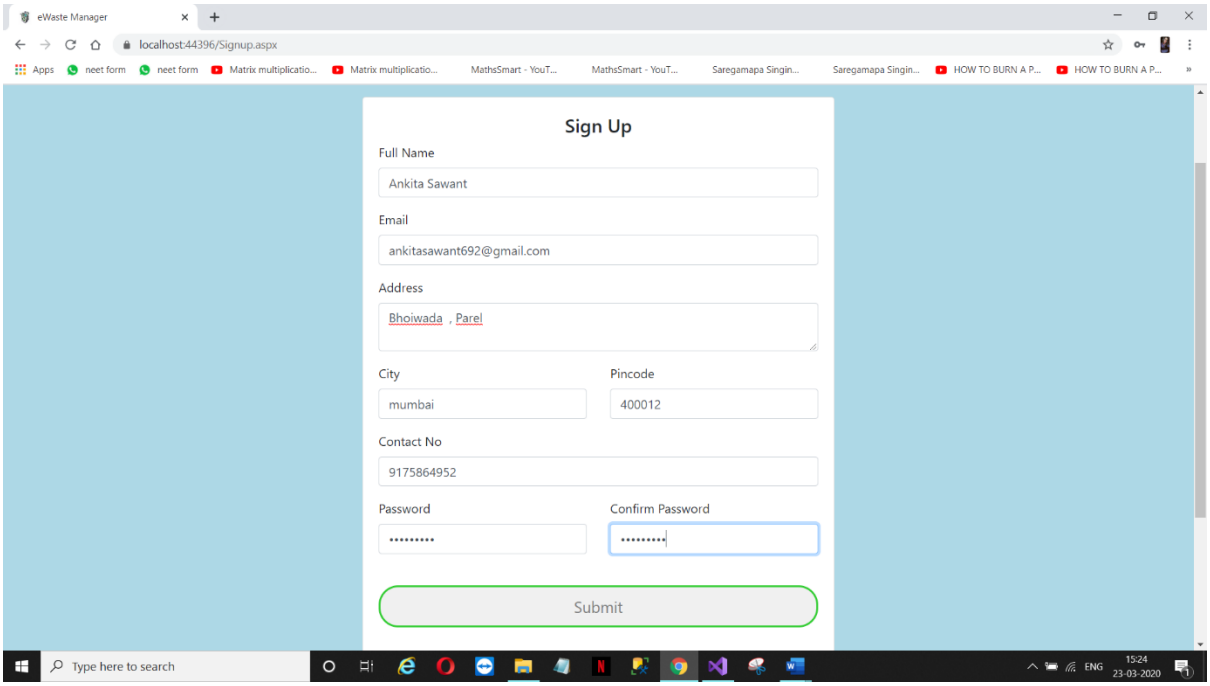
1. Home Page



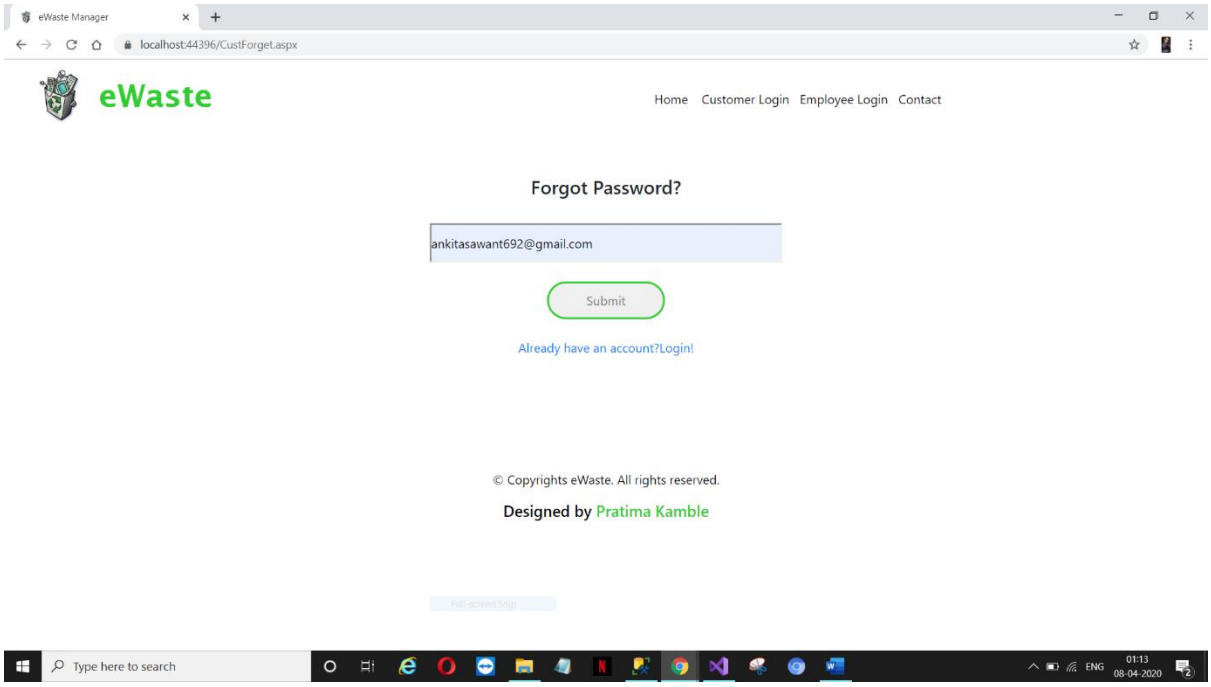
2. Customer Login



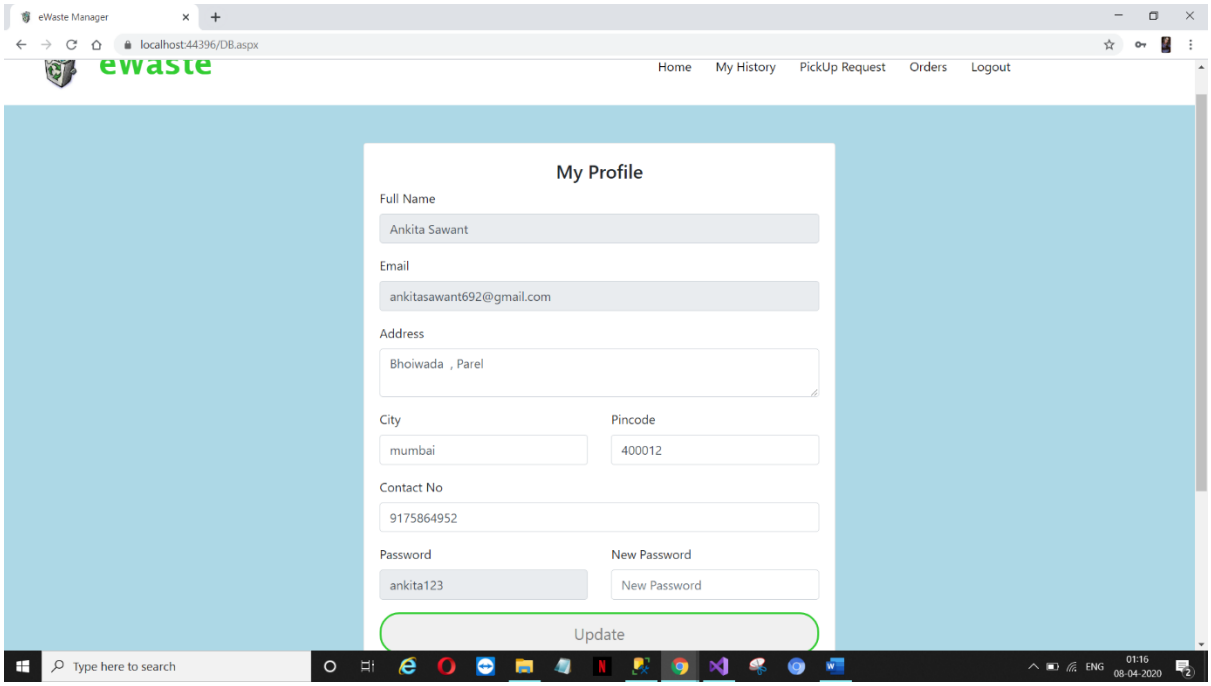
3. Customer Registration



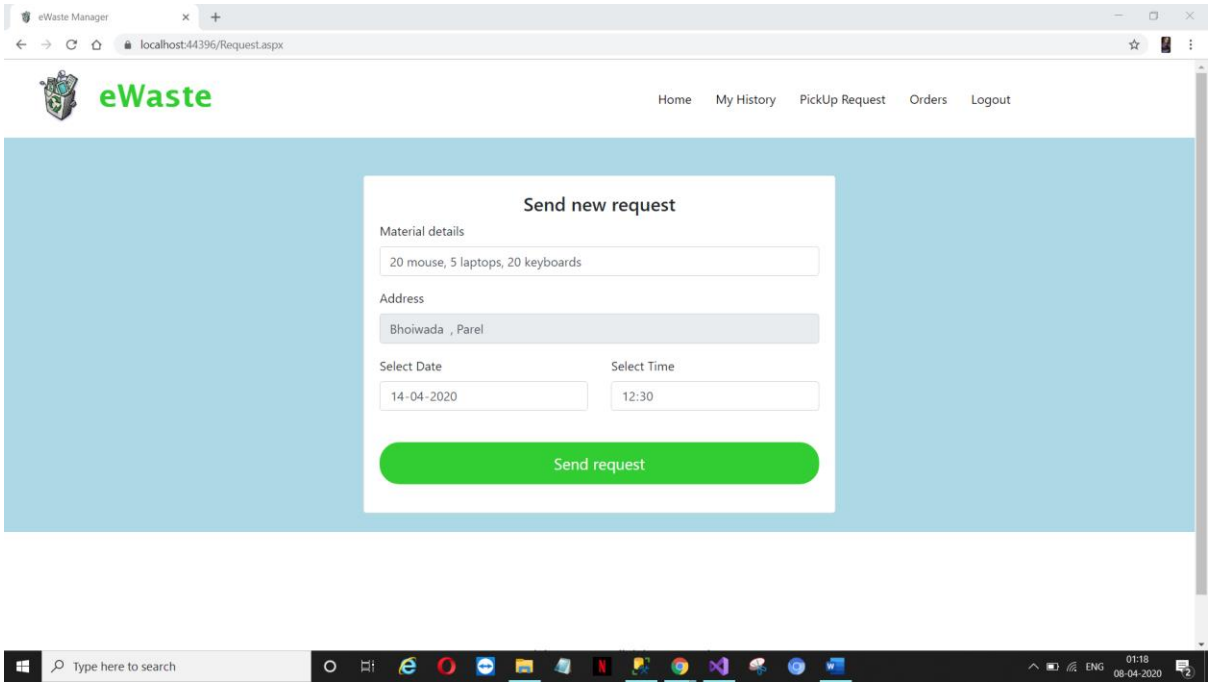
4. Customer Forgot Password



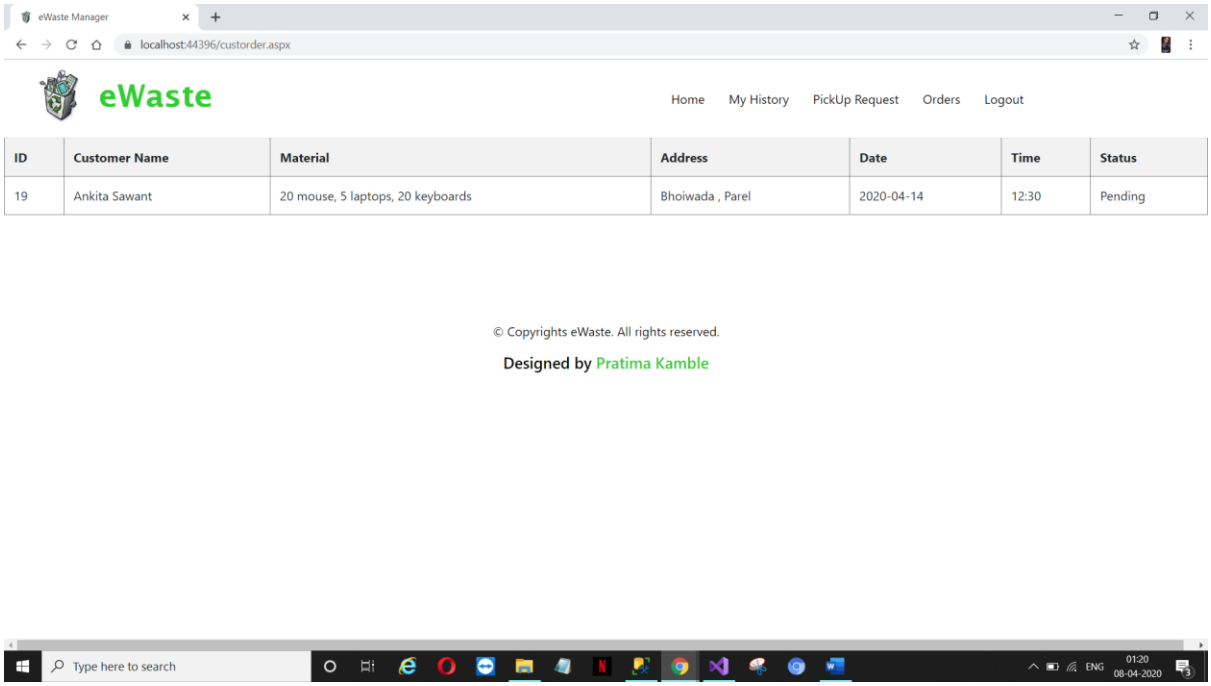
5. Customer Update Profile



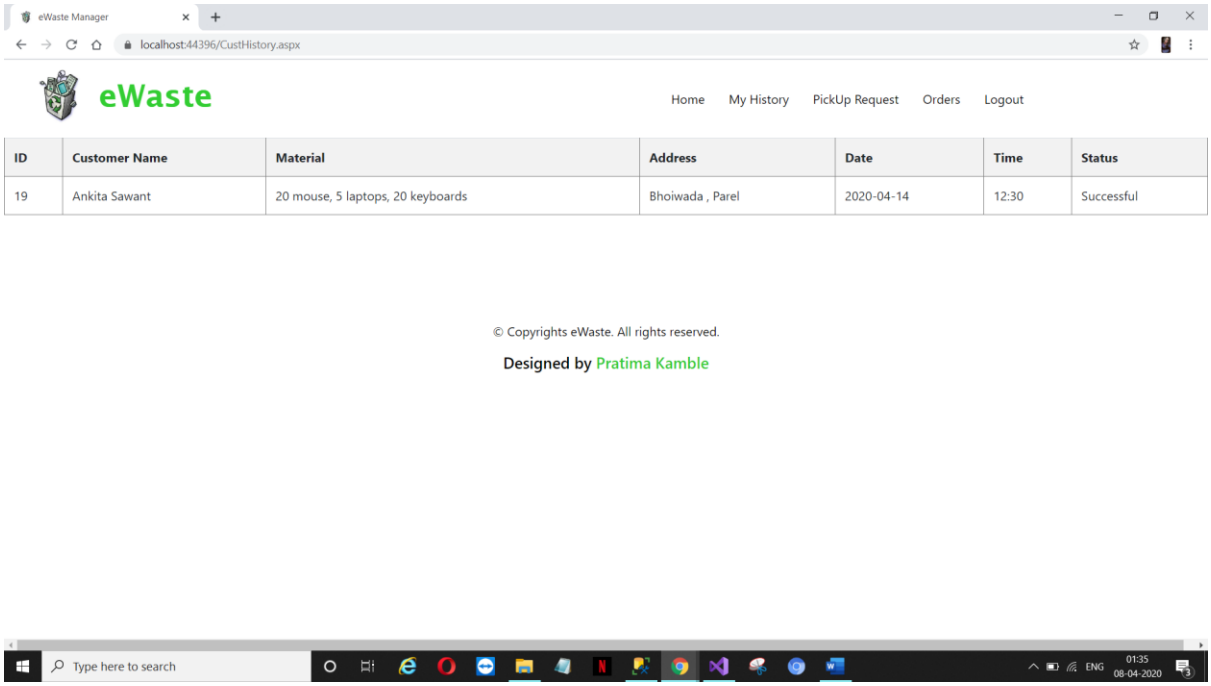
6. Send Order Request



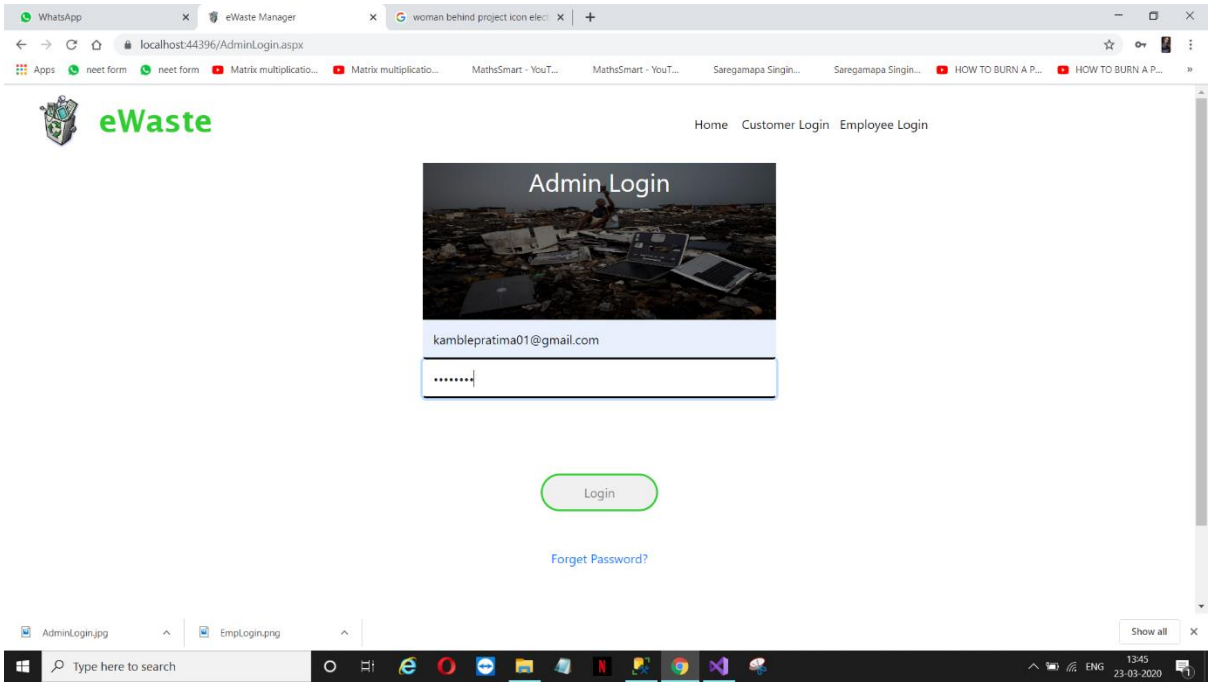
7. View Pending Orders



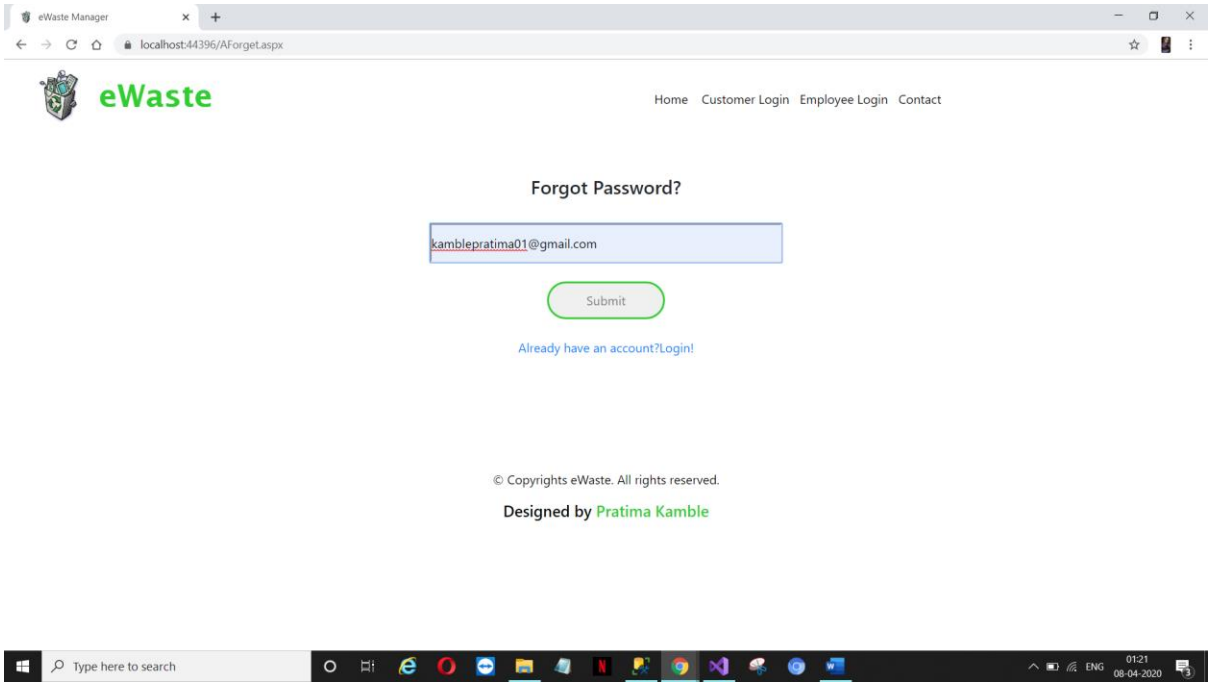
8. View History



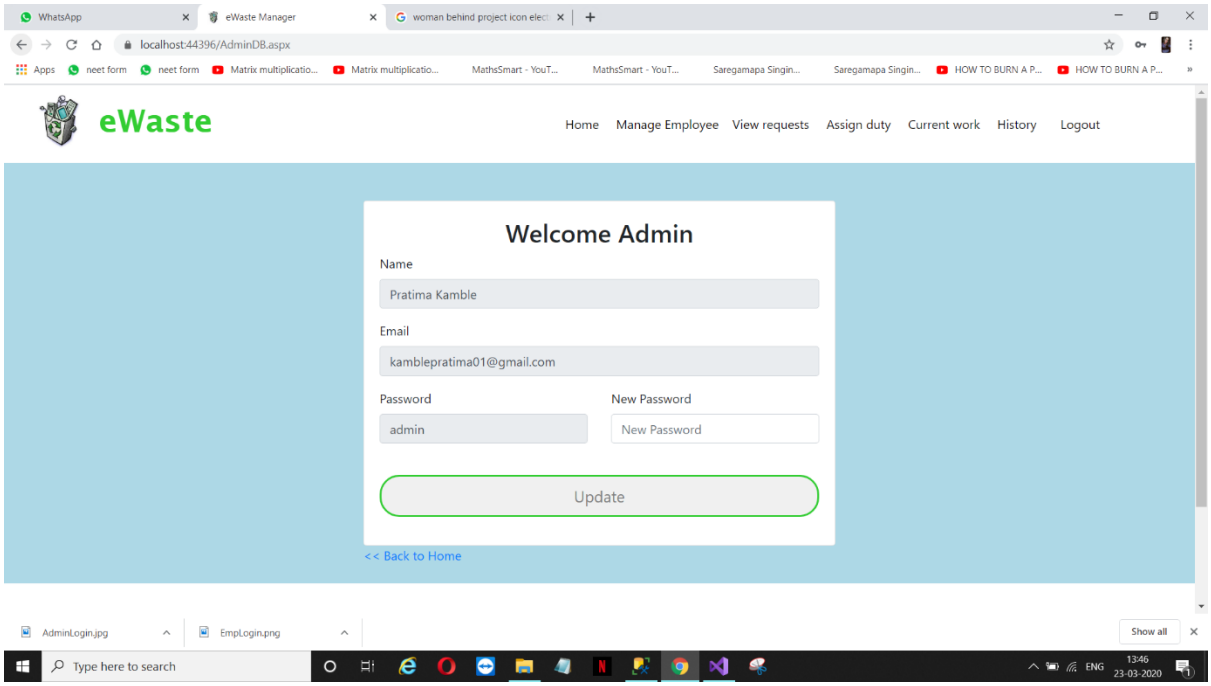
9. Admin Login



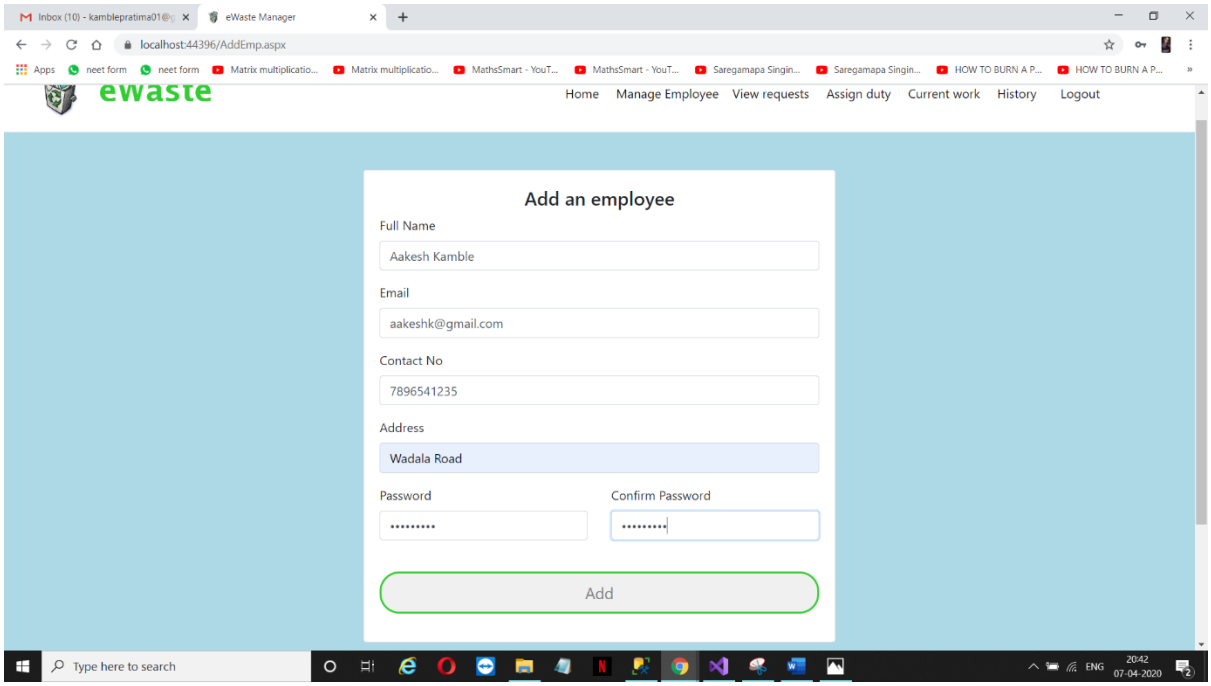
10.Admin Forgot Password



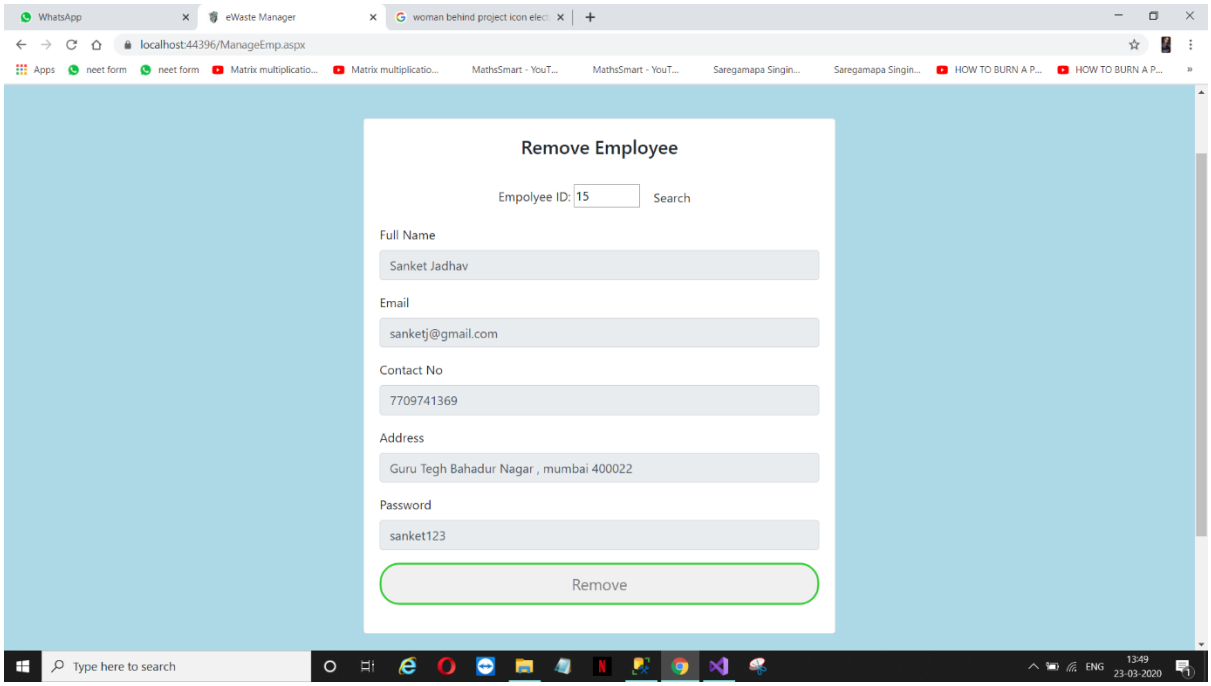
11.Admin Update Profile



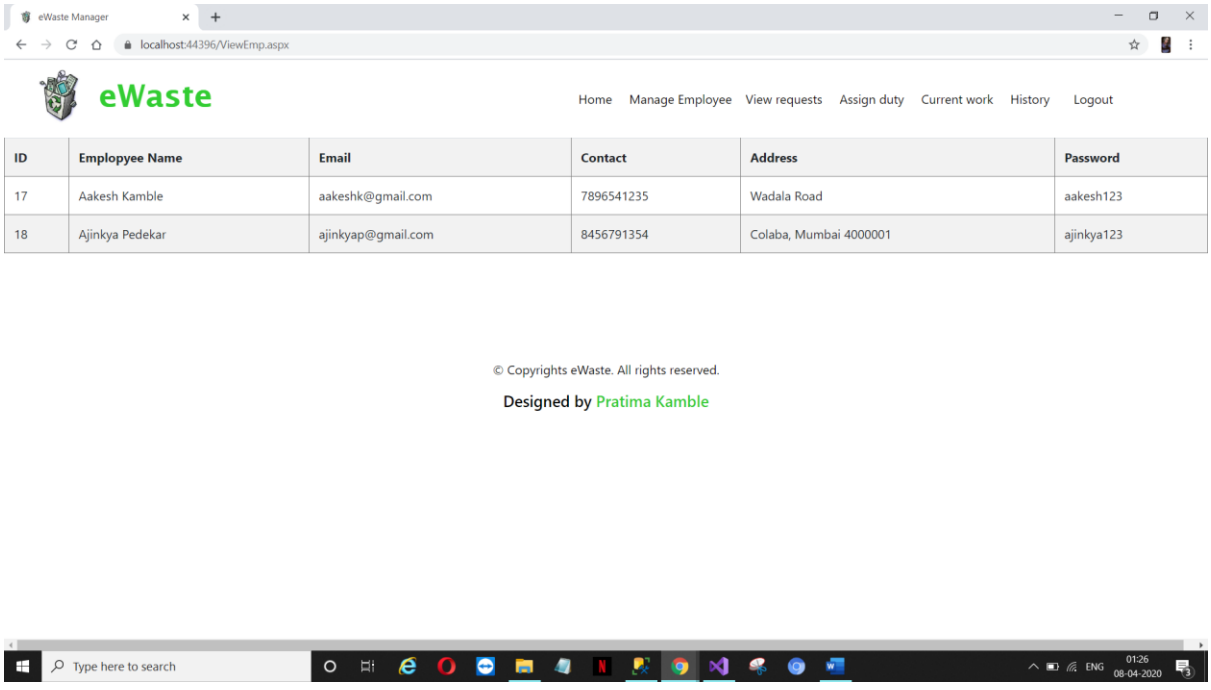
12.Add Employee



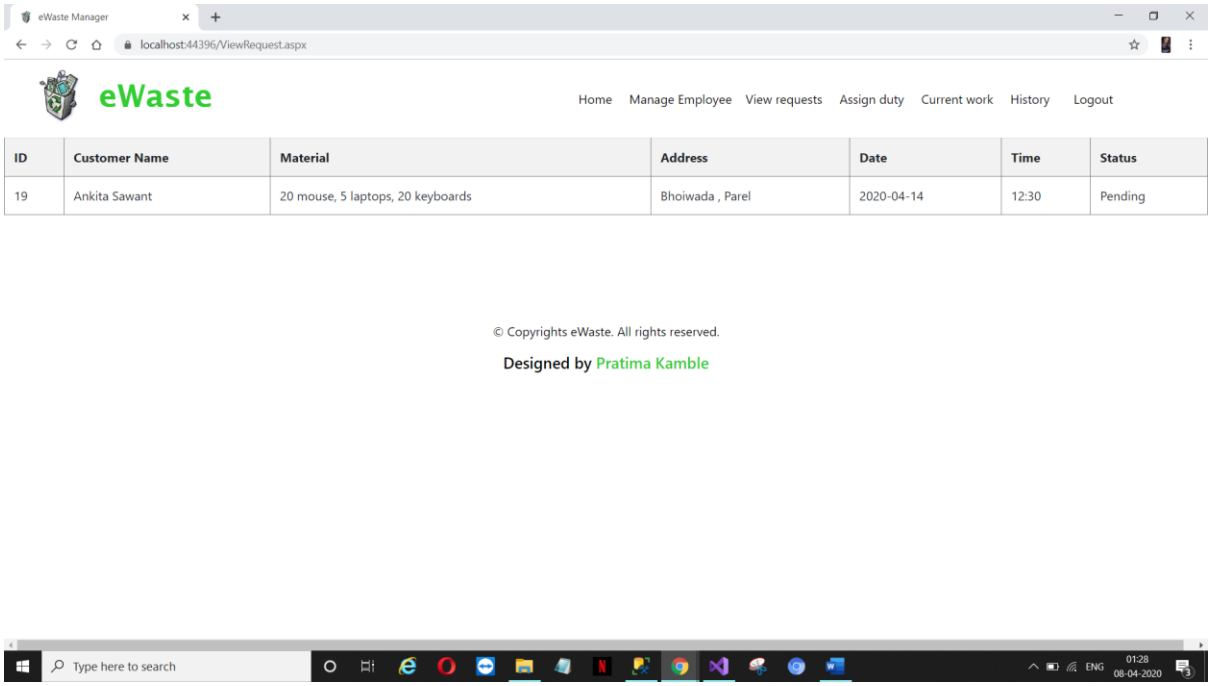
13.Remove Employee



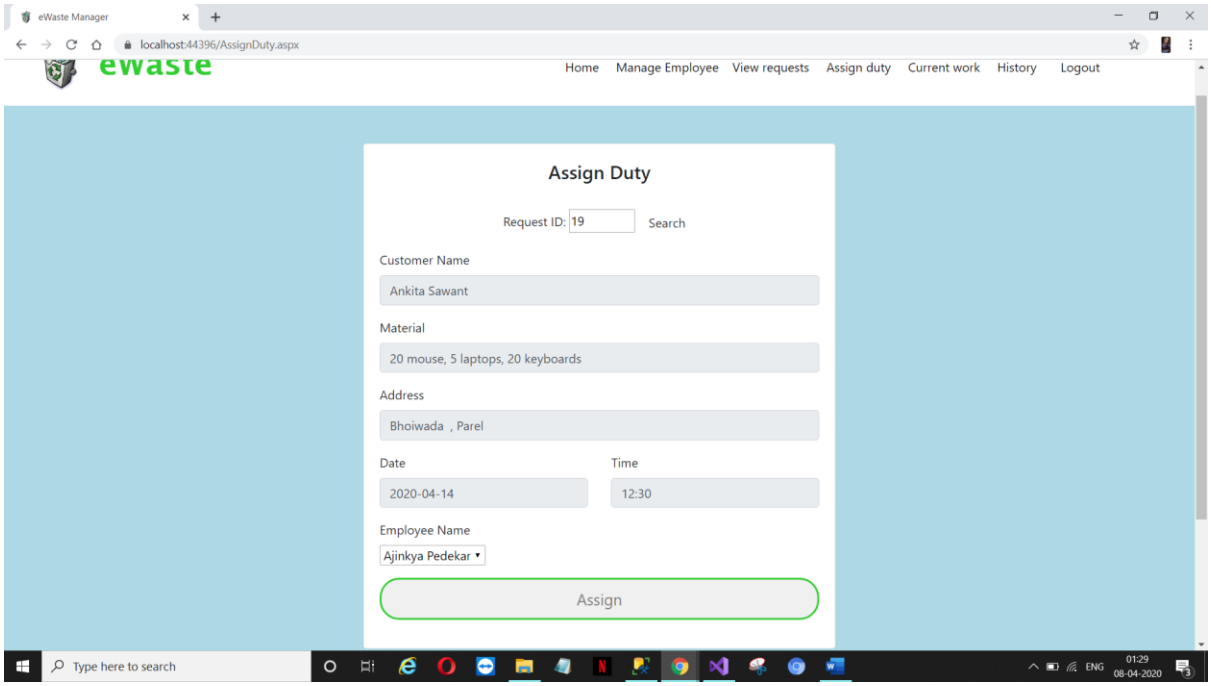
14.View Employee



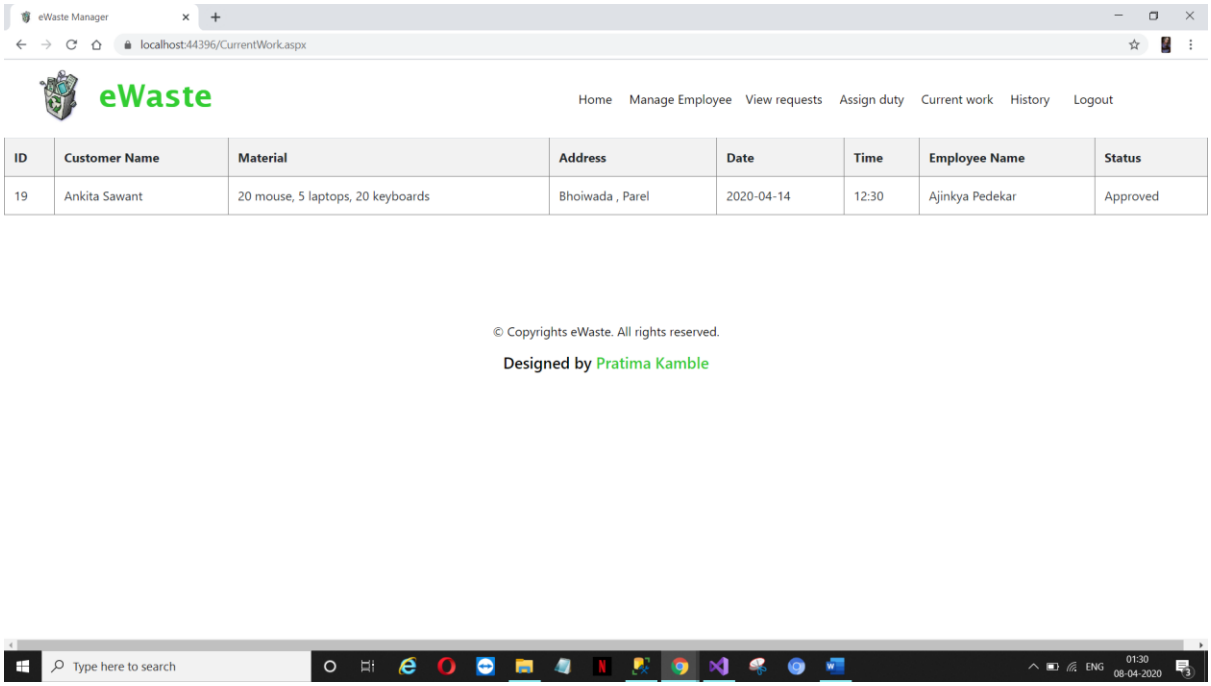
15. View Order Request



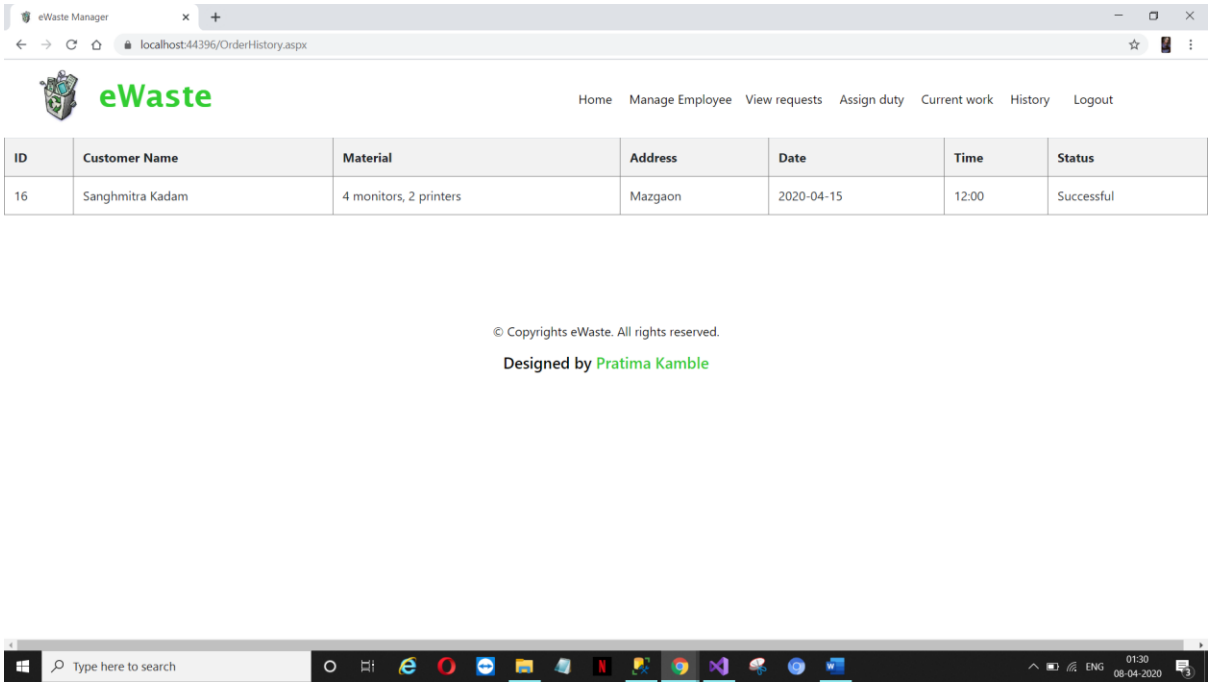
16. Assign Duty



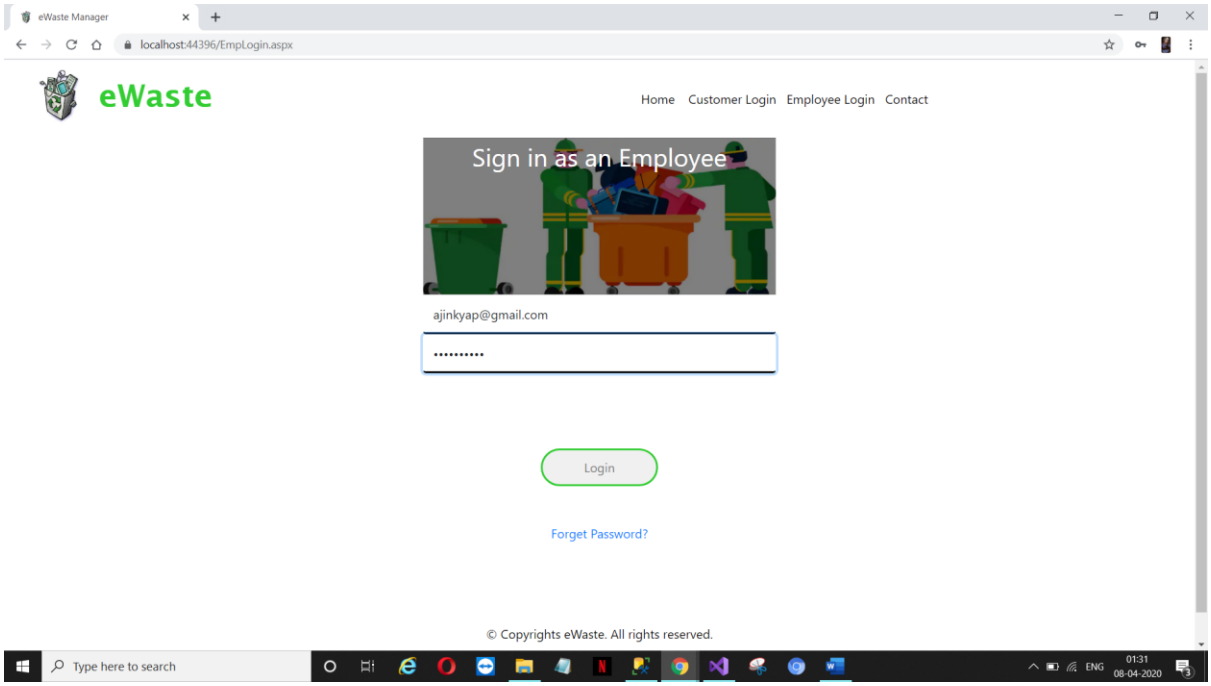
17.Current Approved Work



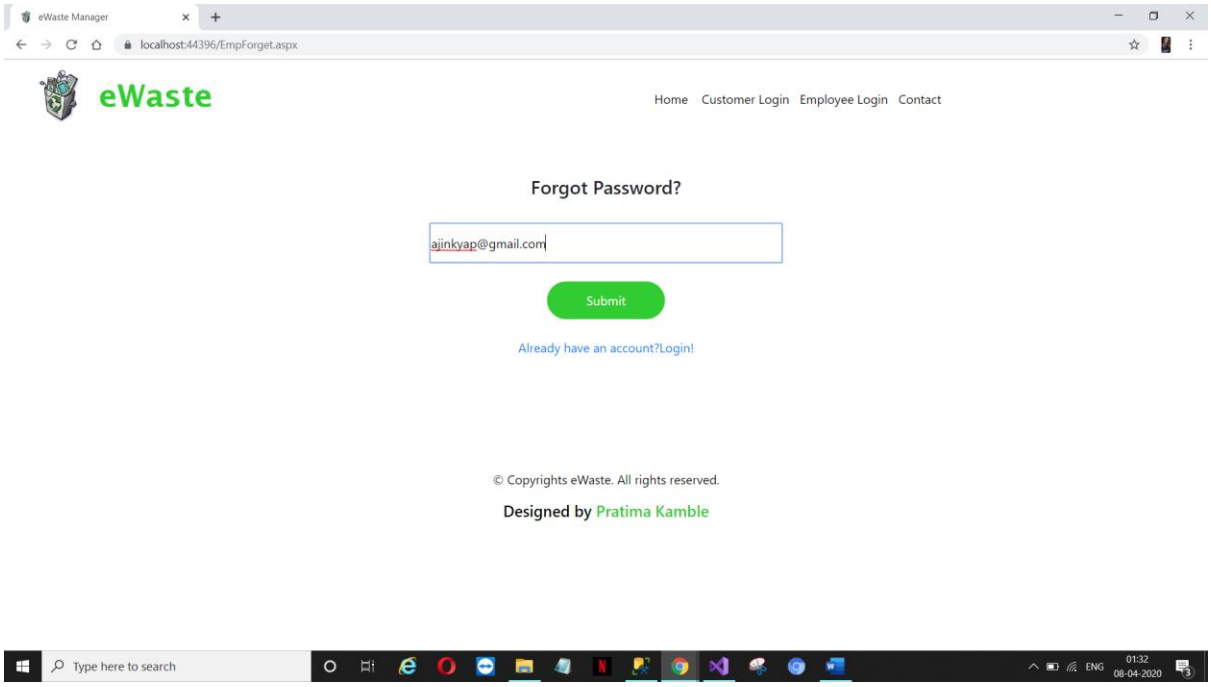
18.Admin History



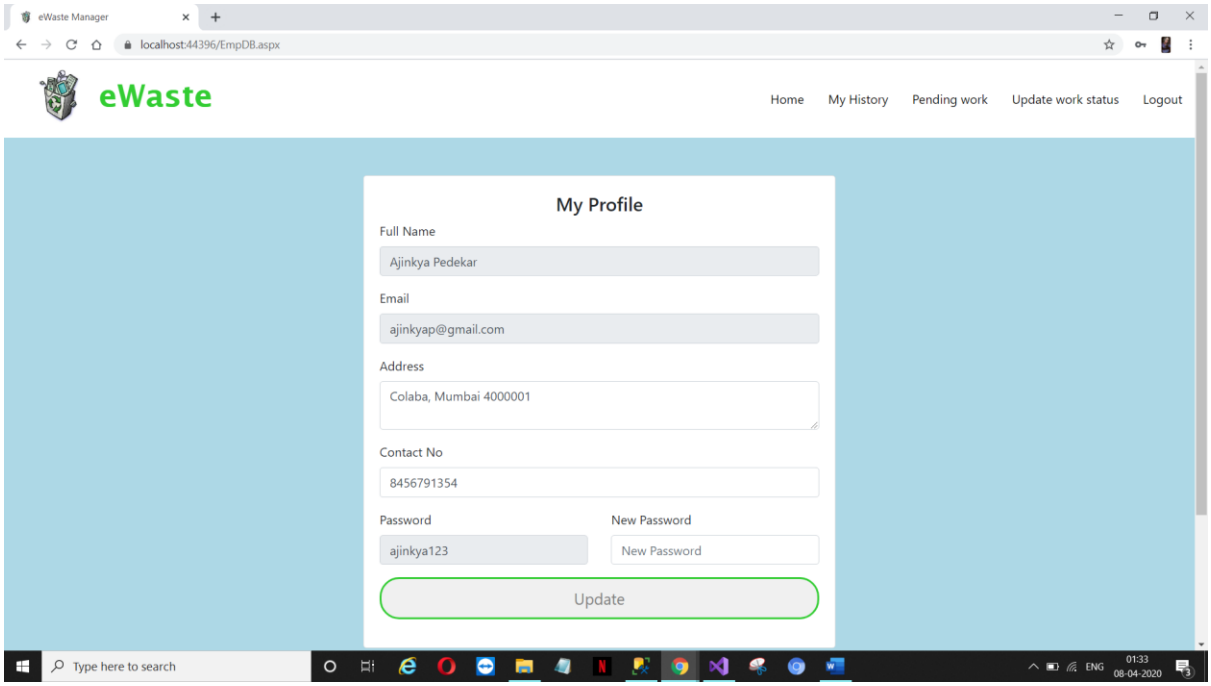
19.Employee Login



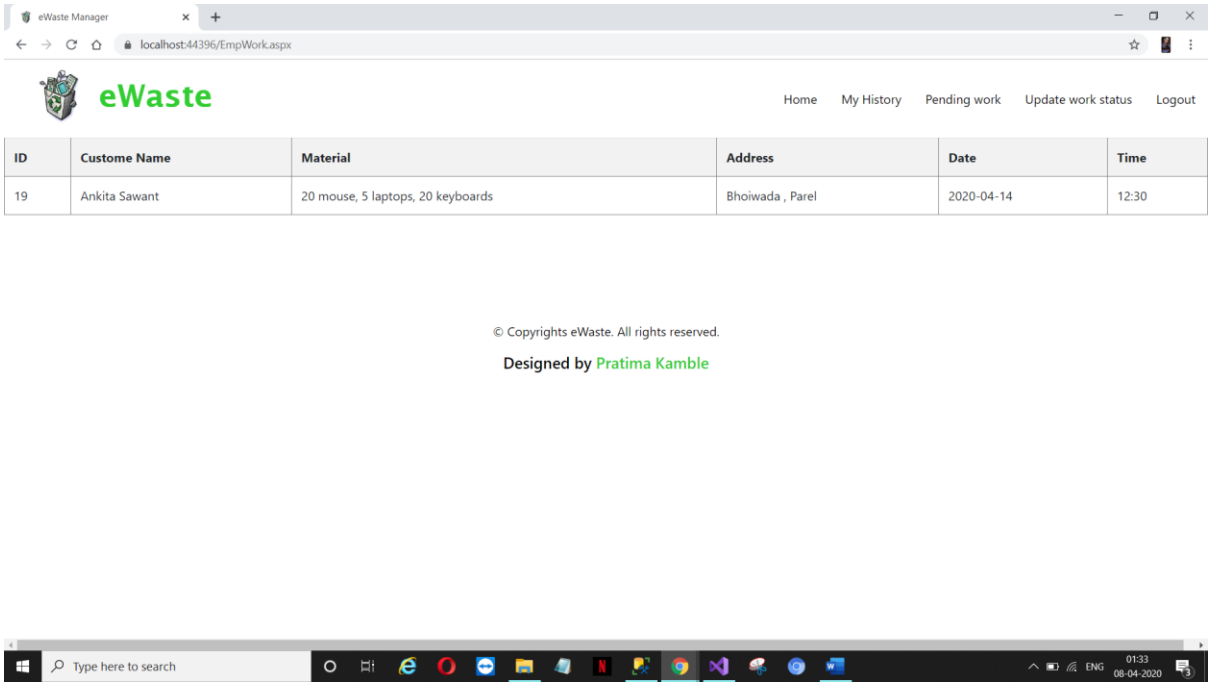
20.Employee Forgot Password



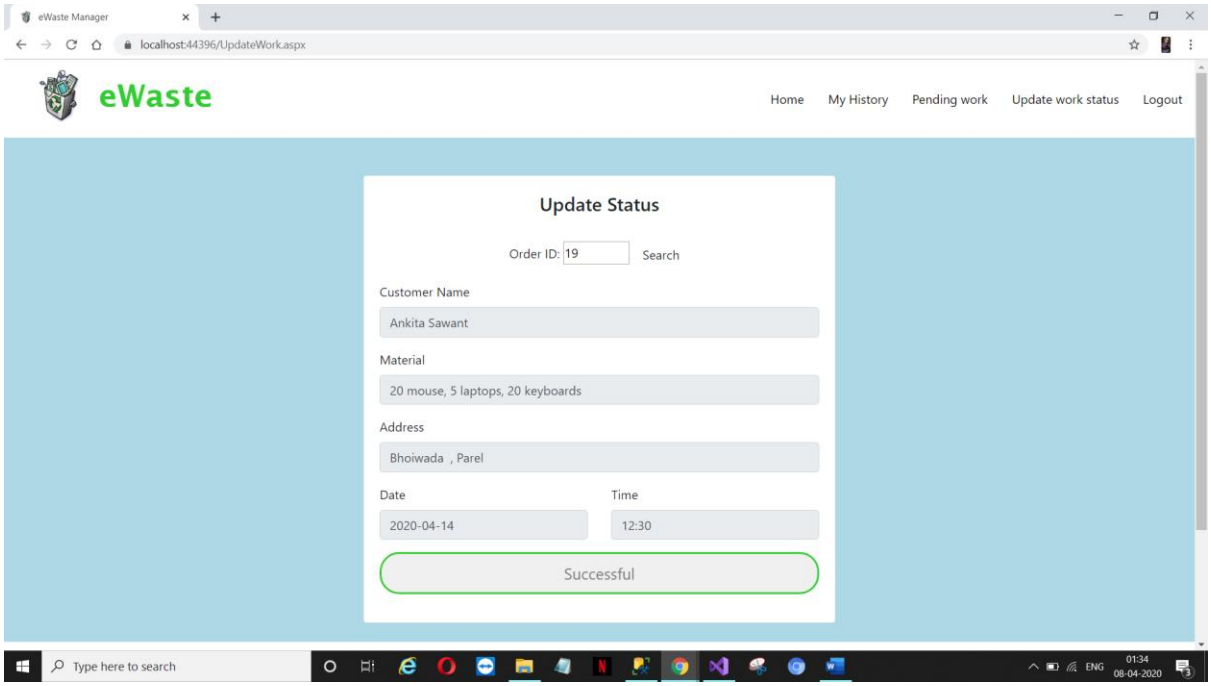
21.Employee Update Profile



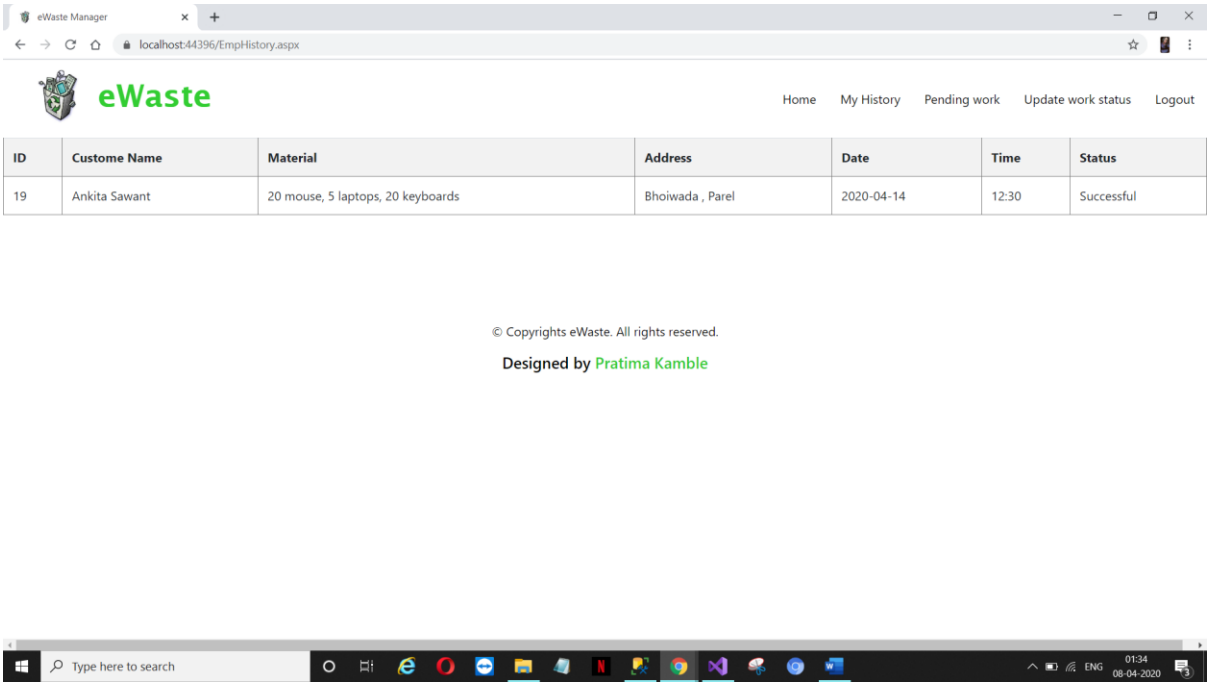
22.Pending Work



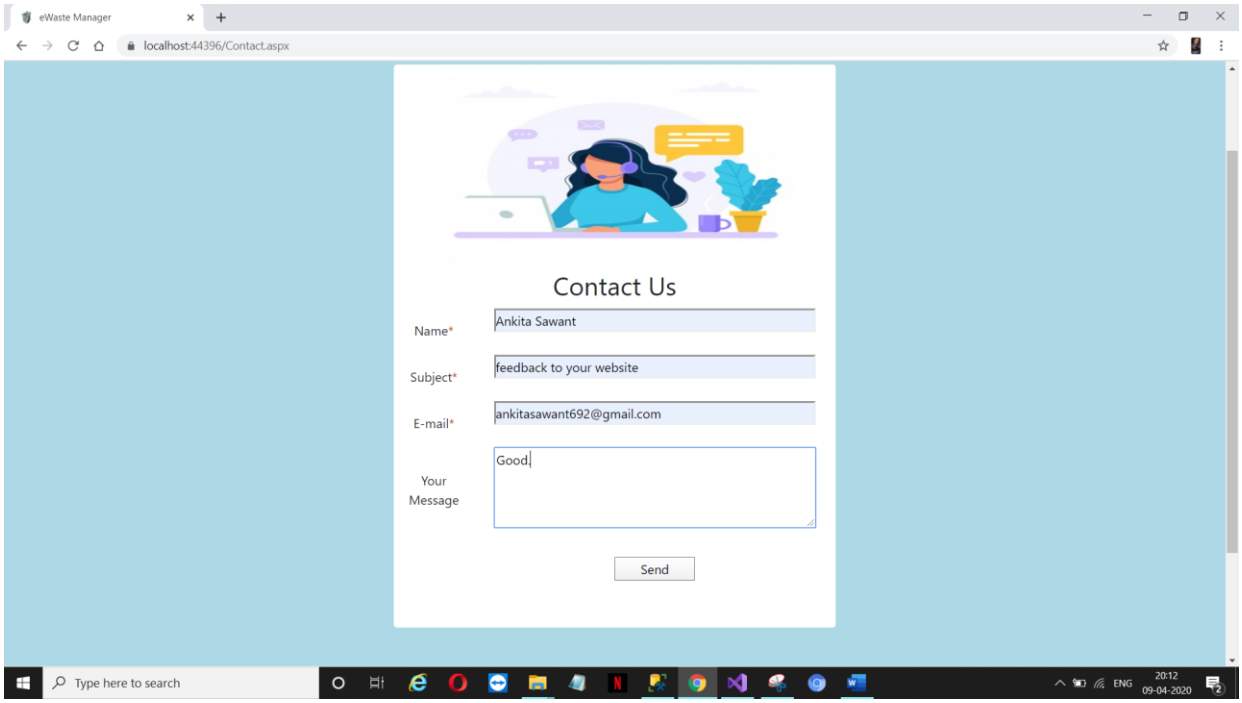
23.Update Work Status



24.Work History



25. Contact


eWaste Manager
localhost:44396/Contact.aspx
Contact Us
Name*
Ankita Sawant
Subject*
feedback to your website
E-mail*
ankitasawant692@gmail.com
Your Message
Good.
Send
Type here to search
ENG
20:12
09-04-2020

6.2 User Documentation

1. Home Page

   This page is the default page which will appear when customer or employee will open the website.

2. Customer Login

   On this page a customer will login to website by using valid email and password if he/she is already exist.

3. Customer Registration

   If a person is new to website then, he/she can register on this page by filling the details on registration form.

4. Customer Forgot Password

   If a customer forget their email or password, they can recover it.

5. Customer Update Profile

   If a customer wants to update their details, they can update it on this page.

6. Send Order Request

   On this page a customer can send a request to an admin by filling the form and successfully completing the phase of sending request.

7. View Pending Orders

   On this page Customer can see their orders which are approved from admin.

8. View History

   When an employee updates work done on their profile, then this page will show the successful pickup status of an order.

9. Admin Login

   On this page admin will login into website by using valid email and password.

10. Admin Forgot Password

    If an admin forget their email or password, they can recover it.

11. Admin Update Profile

    If an admin wants to update their details, they can update it on this page.

12. Add Employee

    On this page admin can add a new employee to the website.

13. Remove Employee

    On this page admin can remove any existing employee from the website.

14. View Employee

    On this page admin can view a list of employees which are added to website.

15. View Order Request

    An admin can see the request sent by the customers.

16.Assign Duty

On this page admin can assign a duty to an employee by selecting employee's name.

17.Current Approved Work

On this page admin can see the orders which are assigned to employee which are counted as an approved orders.

18.Admin History

This page will show the history of work done by an employee to the admin.

19.Employee Login

On this page an employee will login by using their valid email and password.

20.Employee Forgot Password

If an employee forget their email or password, they can recover it.

21.Employee Update Profile

If an employee wants to update their details, they can update it on this page.

22.Pending Work

This page will show the work assigned by the admin to an employee.

23.Update Work Status

On this page, an employee update the status of work as a successful when it is done.

24.Work History

This page will show the history of work done by an employee.

25.Contact

On this page any user can send a feedback or a message to an email.

CHAPTER 7
CONCLUSION

7.1 CONCLUSIONS:

- From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product is a highly efficient GUI based component.
- This component can be easily plugged in many other systems.
- Also, the component is user friendly.
- This System helps to store all the data about the eWaste Management, users in the computer and there is no need to do paper work.
- Data is going to be preserved carefully for longer period hence proper backup is required otherwise there is chance of losing entries or data.
- Generally, admin of the waste center faces the problem to data storage etc. so this can be enhanced in future for auto backup of electronic waste material data and user data.
- This project helps the homeless, abandoned dogs to get a new home and foster parents by using modern technology.
- While making this project, I learnt how to apply all the practical concepts we studied during our BSC.IT course.
- The proper smooth and successful of a project depends largely upon proper project planning and clear understanding as well as documentation of the requirements.
- The making of this project was certainly a learning experience and will certainlyhelp our future professional endeavors.

7.1.1 SIGNIFICANCE OF THE SYSTEM:-

- Electronic and electrical equipment's cannot be avoided in today's world. So also in the case of waste electronic and electrical materials.
- As long as this is the necessary evil, it has to be best managed to minimize its adverse impacts on environment. Electronic waste piles are growing as is their pollution potentials. Most of these problems have their source in the development and design of the product concerned.
- Using this type of system we can conclude that using the methodology of reduce decreases the piles of electronic and electrical equipment and make environment to be cleaned and healthy.

7.2 LIMITATIONS OF THE SYSTEM:

- This not only cause widespread pollution but is harmful for dedicated people who recycle such products. Long-term effects on environment is still unknown. So increasing the education about the e-waste is needed.

7.3 FUTURE SCOPE OF THE PROJECT:

- E-waste Collection System is future to reduce the electric waste and reusability of the electronic material in future.
- The E-waste Collection System that develop the information and awareness about the recycling and problem affect by the collection. Decrease the Electronic and Electric Waste from the World and less development of electronic and more recycling of the material.

REFERENCES

- https://www.tutorialspoint.com
- https://www.creately.com
- https://www.draw.io.com
- https://www.teamgantt.com
- https://www.youtube.com
- https://www.google.com
- https://www.lucidchart.com