CUSTOMER CARE REGISTRY

A PROJECT REPORT

Submitted by

TEAM : PNT2022TMID48083

“R.Tamil Murasu	**(912419104034)**
A.Micheal Raj	**(912419104017)**
J.Kamaleshwaran	**(912419104011)**
A.Sebastin John Paul	**(912419104027)”**

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SHANMUGANATHAN ENGINEERING COLLEGE

ARASAMPATTI

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

BONAFIDE CERTIFICATE

Certified that this project report “**CUSTOMER CARE REGISTRY**” is the bonafide work of **TEAM: PNT2022TMID48083,**

“R.Tamil Murasu	**(912419104034)**
A.Micheal Raj	**(912419104017)**
J.Kamaleshwaran	**(912419104011)**
A.Sebastin John Paul	**(912419104027)”**

who carried out the project work under the supervision.

SIGNATURE

HEAD OF THE DEPARTMENT

Mrs. R. KAVITHA,
M.Tech.,

Assistant Professor, Department of
Computer Science and Engineering,
Shanmuganathan Engineering
College,Arasampatti-622507.

SIGNATURE

SUPERVISOR

Mrs. R. KAVITHA,
M.Tech.,

Assistant Professor, Department of
Computer Science and Engineering,
Shanmuganathan Engineering
College,Arasampatti-622507.

Submitted for the viva voce to be held on ___________________.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

“HARDWORK NEVER FAILS” So we thank god for having gracefully blessed us to come up till now and thereby giving strength and courage to complete the project successfully. We sincerely submit this project to the almighty lotus feet.

We wish to acknowledge with thanks to the significant contribution given by the management of our college chairman **"KALVI VALLAL Mrs. Pichappa Valliammai"**, Correspondent **Dr. P. Manikandan**, and Secretary **Mr. M. Vishwanathan**, Shanmuganathan Engineering College, Arasampatti, for their extensive support.

We convey our indebted humble thanks to our energetic Principal **Dr. KL. Muthuramu M.E., (W.R), M.E.(S.E),M.I.S.T.E., F.I.E., Ph.D.,** for his moral support to complete this project.We are grateful to our Head of the Department **Mrs. R. Kavitha M.Tech.,** for her valuable guidelines during the courses of the project.

Gratitude will never fail towards our dynamic and effective internal guide and also the project coordinator **Mrs. R. Kavitha M.Tech.,** for his valuable, innovative suggestion, constructive instruction, constant encouragement, unending help that have been provided to us throughout the project and efforts that he has taken during project proceedings and during code debugging.

We also convey our humble thanks to all the staff members of the CSE Department who had provided their contribution to steer our project towards the successful completion.

Finally, we thank our family members who have spent their wealth for our studies and have motivated us to complete this project successfully.

TABLE OF CONTENTS

CHAPTER NO **TITLE**

1. INTRODUCTION

The Customer Service Desk is a web based project. Customer Service also known as Client Service is the provision of service to customers' Its significance varies by product, industry and domain. In many cases customer servicesis more important if the information relates to a service as opposed to a Customer. Customer Service may be provided by a Service Representatives Customer Service isnormally an integral part of a company's customer value proposition.

1.1 PROJECT OVERVIEW

This Web Application has been developed to help the customer in processing their complaints. The customers can raise the ticket with a detailed description of the issue. An Agent will be assigned to the Customer to solve the problem. Whenever the agent is assigned to a customer they will be notified with an email alert. Customers can view the status of the ticket till the service is provided.

ADMIN : The main role and responsibility of the admin are to take care of the whole process. Starting from Admin login followed by the agent creation and assigning the customer's complaints. Finally, He will be able to track the work assigned to the agent and a notification will be sent to the customer.

USER : They can register for an account. After the login, they can create the complaint with a description of the problem they are facing. Each user will be assigned with an agent. They can view the status of their complaint.

ADMIN : After the user successful Registered their Admin has Quickly Response to the user.

1.2 PURPOSE

The purpose of the customer care registry project is to develop a centralized database of customer service information that can be accessed by companies in order to improve customer service.

The registry will contain customer service records from a variety of companies, including contact information, customer service ratings, and comments. This information will be used to help companies improve their customer service practices and make informed decisions about customer service policies.

The purpose of this project is to develop a customer care registry for a company. This registry will help the company to track and manage customer care related information. It will also help the company to improve customer care services by providing better and more efficient care to customers.

2. LITERATURE SURVEY

A literature survey was conducted on customer care registry projects. The following is a summary of the findings: It was found that customer care registry projects are typically implemented to improve customer service and/or to reduce costs. In many cases, the registry project is used to streamline customer service processes and/or to reduce the number of customer service calls. In some cases, the registry project is used to improve customer satisfaction scores. It was also found that customer care registry projects can be implemented using a variety of different technologies, including web-based applications, customer relationship management (CRM) systems, and enterprise resource planning (ERP) systems. Finally, it was found that customer care registry projects can be successful if they are properly planned and implemented. However, if the project is not properly planned or implemented, it is likely to fail.

S.NO	Author	Title	Publish Date	Findings
1	Olutayo Boyinbode , Akure	E- Customer Care Service System fo rBenin Electricity Distribution Company	April 2015	Face to face customer contact is time consuming, since theconsumers have to be physically present for both parties to meet before solution can be proffered. Telephone communication with BEDC enables the customers to call and make enquiries and receive responses about information or services required. The BEDC existing customer service system is saddled with limitations such as poor customerservices and relations, high response time. This paper aims to eliminate these weaknesses by automating the process throughan e-Customer Care Service System by which customer makesan enquiries and complaints on the services delivered by the company.
2	D Riananingrum,R R S Hari, F Nursaori and WA Astuti	Integrated Information System fo rCustomer Care	January2021	The application of customer Facing and Ecosystem Facing is implied as computerization of customer data. The goal is to understand customers' needs and expectations to establish good relationships with customers. This study aims to determine Customer Facing and Ecosystem Facing services in the digital transformation business. The research method used a descriptive research method with a qualitative approach. Customer Facing and Ecosystem Facing is a technology that can increase the production process for business.

S.NO	Author	Title	Publish Date	Findings
3	Khalid Rababah, Haslina Mohd, and Huda Ibrahim	Customer Relationship Management (CRM) Processes from Theory to Practice: The Pre-implementation Plan of CRM System	April 2011	This paper provides an extensive review of the literature regarding the CRM processes. This review aims to increase the understanding of the different perspectives and the various types and levels of CRM processes. This paper reveals that there are four major perspectives of CRM processes which are customer facing level processes, customer oriented processes, cross functional CRM processes, and CRM macro- level processes.
4	Mutegyeki, Walter	A mobile based customer communication management tool to improve customer care services in micro finance institutions	May 2016	This report document describes project design and development of a digital customer care communication solution that encompasses all major communication channels that will enable microfinance bankcustomers to conveniently communicate with their banks and also enable banks to manage and evaluate customercare and service delivery. The project followed a software development process that included software requirements specifications and requirements engineering, software design and development, verification & validation which constituted of dynamic testing of each individual core modules.

2.1 Existing problem

The existing system for the customer care registry project is a paper-based system. This system is used to track customer service requests and complaints. The system is cumbersome and time-consuming, and it is difficult to track and report on customer service issues. The customer care registry project is designed to replace the existing system with a new, web-based system. The new system will be easier to use and will provide more accurate and timely information on customer service issues.Out of the previous related work done concerning customer complaint, the most recent research was: Razali et al. (2011) [6] develop a new complaint management system called (e-Aduan) as a platform for UiTM Pahang's customers to complaint and comment regarding the services and facilities provided by the university. The researcher found out that the most appropriate to the research topic handling customer complaint using SOA was: Najar et al. (2010) [5] tried to improve relation between Citizens and Government by presenting a new model based on Service Oriented Architecture (SOA). With utilizing the presented model in Government body on one hand Governments will have the ability to minimize Citizens' dissatisfaction and on the other hand it can encourage Citizens to participate in controlling Government body such as Governments' staffs and organizations.

2.2 References

[1] Cho Y., Hiltz R., & Fjermestad J., "An Analysis of Online Customer Complaints: Implications for Web Complaint Management." in Proceedings of the 35th Hawaii International Conference on System Sciences, Hawaii, (2002).

[2] Najar, A. S., Al-Sukhni, H. A., & Aghakhani, N., "The Application of Service-Oriented Architecture in E-complaint System." Paper presented at (ICCSN '10) the Second International Conference on Communication Software and Networks, (2010, 26-28 Feb. 2010).

[3] Razali R., Abd Halim K. N., & Jusoff K., "Quality Improvement of Services in Unversiti Teknologi Mara Pahang from a Management Perspective." Management Science & Engineering Vol.5, No.1, (2011), pp. 71-80.

2.3 Problem Statement Definition

A problem statement is a concise description of the problem or issues a projectseeks to address. The problem statement identifies the current state, the desired future state and any gaps between the two. A problem statement is an important communication tool that can help ensure everyone working on a project knows what the problem they need to address is and why the project is important.

Defining the problem :

Website isn't Showing orRunning	
Why this happens?	This issue often happens when the WebBrowser is not compatible.
Who does the problem affect?	The user and the users who are usingtheWebsite
What is the issue?	This issue can be related to theincompatibility of the Browser
What is the solution?	To check and download the compatibleversion of the Website

Over Data Utilization on connecting to Desktop	
Why this happens?	This happens because the backgroundwindows update process is on.
Who does the problem affect?	The user and the users who areconnectedto the Desktop.
What is the issue?	This issue is over utilization of mobiledataover connecting to the Desktop

What is the solution?	This issue can be solved by disabling thewindows update option in settings

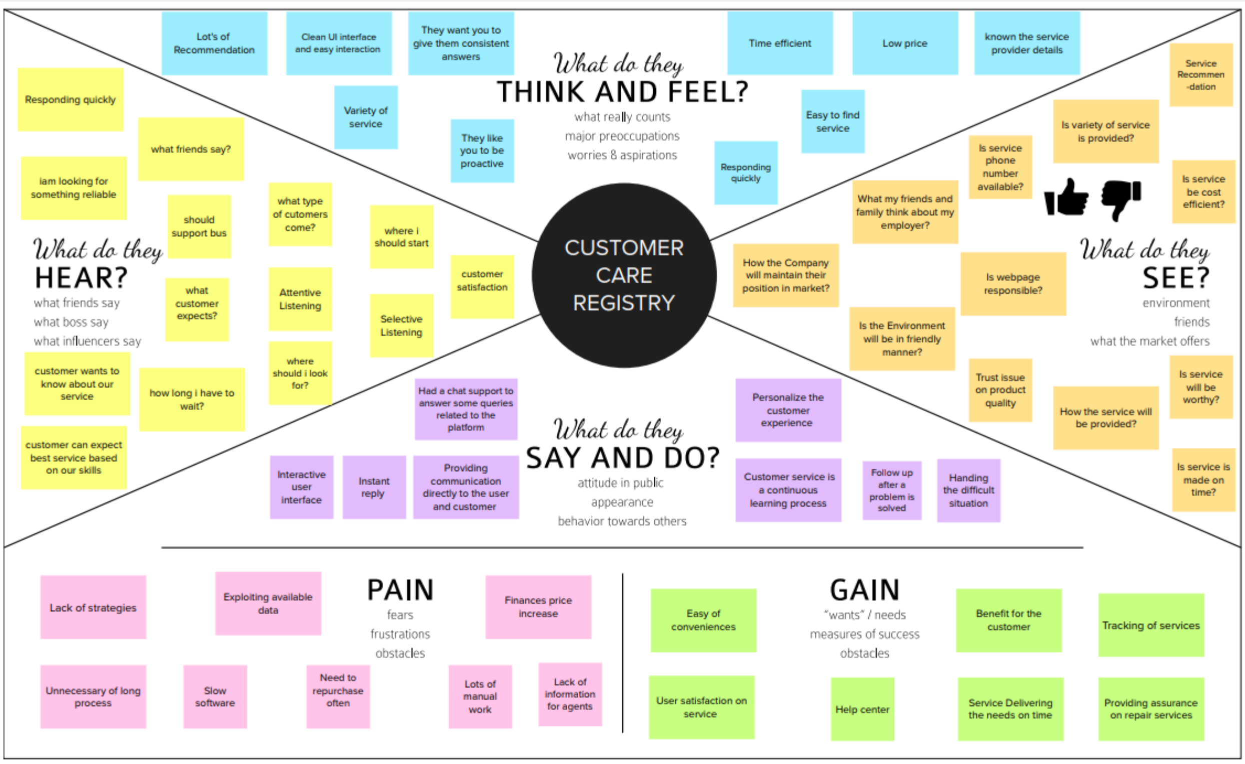
Customer wants to fix a blue screen of death?	
Who does the Problem Affect?	Customer who use the particular thing
What are the boundariesof theproblem?	Customer who use the thing for their personal work, office work etc
What is the issue?	Failure of Hardware or driver sometimesit maybe in software too
When does the issue occur?	It frequently occurs after the customerinstallednew drivers or new piece of software
Where does the issue occur?	It often lies in the Hardware or one ofthe drivers
Why is it important that we fix theproblem?	It is necessary to run the computer or Laptop todo their task or work in order tocomplete it.
What solution to solve this issue?	A quick reboot is sometimes enough tosolve theproblem
What methodology used to solve theissue?	By means of troubleshoot the process orresetting of the software and hardware

Customer wants to fix the Payment issue?	
Who does the Problem Affect?	Customer who use the particular thing
What is the solution to solve thisissuetemporarily?	Check payment method is up todate orTry another payment method
How the issue occurs?	Customer who has entered incorrect card information, payment gateway, or the bankinstitution issue
When does the issue occur?	It occurs when there is insufficientbalance inbank account
Why is it important that we fix theproblem?	For the welfare of the customer needs

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is **a collaborative tool teams can use to gain a deeper insight into their customers**. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agilecommunity.



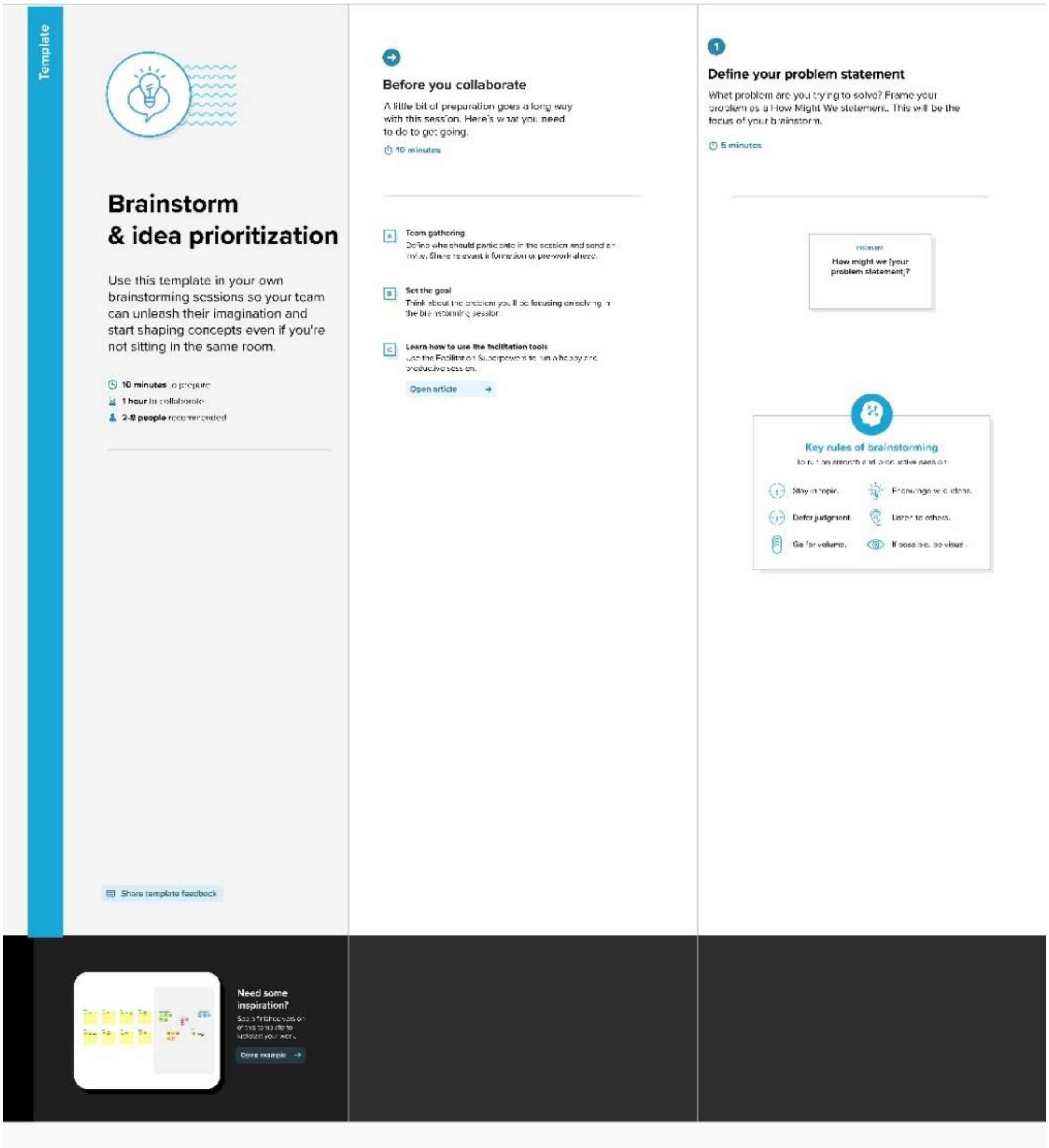
CUSTOMER CARE REGISTRY
What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Lot's of Recommendation
Clean UI interface and easy interaction
They want you to give them consistent answers
Variety of service
They like you to be proactive
Time efficient
Low price
known the service provider details
Easy to find service
Responding quickly
What do they HEAR?
what friends say
what boss say
what influencers say
Responding quickly
what friends say?
iam looking for something reliable
should support bus
what type of cutomers come?
where i should start
what customer expects?
Attentive Listening
Selective Listening
customer satisfaction
customer wants to know about our service
how long i have to wait?
where should i look for?
customer can expect best service based on our skills
What do they SEE?
environment
friends
what the market offers
Service Recommen -dation
Is variety of service is provided?
Is service phone number available?
Is service be cost efficient?
What my friends and family think about my employer?
How the Company will maintain their position in market?
Is webpage responsible?
Is the Environment will be in friendly manner?
Trust issue on product quality
How the service will be provided?
Is service will be worthy?
Is service is made on time?
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
Had a chat support to answer some queries related to the platform
Interactive user interface
Instant reply
Providing communication directly to the user and customer
Personalize the customer experience
Customer service is a continuous learning process
Follow up after a problem is solved
Handing the difficult situation
PAIN
fears
frustrations
obstacles
Lack of strategies
Exploiting available data
Finances price increase
Unnecessary of long process
Slow software
Need to repurchase often
Lots of manual work
Lack of information for agents
GAIN
"wants" / needs
measures of success
obstacles
Easy of conveniences
Benefit for the customer
Tracking of services
User satisfaction on service
Help center
Service Delivering the needs on time
Providing assurance on repair services

3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Brainstorming & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- **10 minutes** to prepare
- **1 hour** to collaborate
- **2-8 people** recommended

Share template feedback

Need some inspiration?
See a finished version of this template to kickstart your work.

Open example →

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

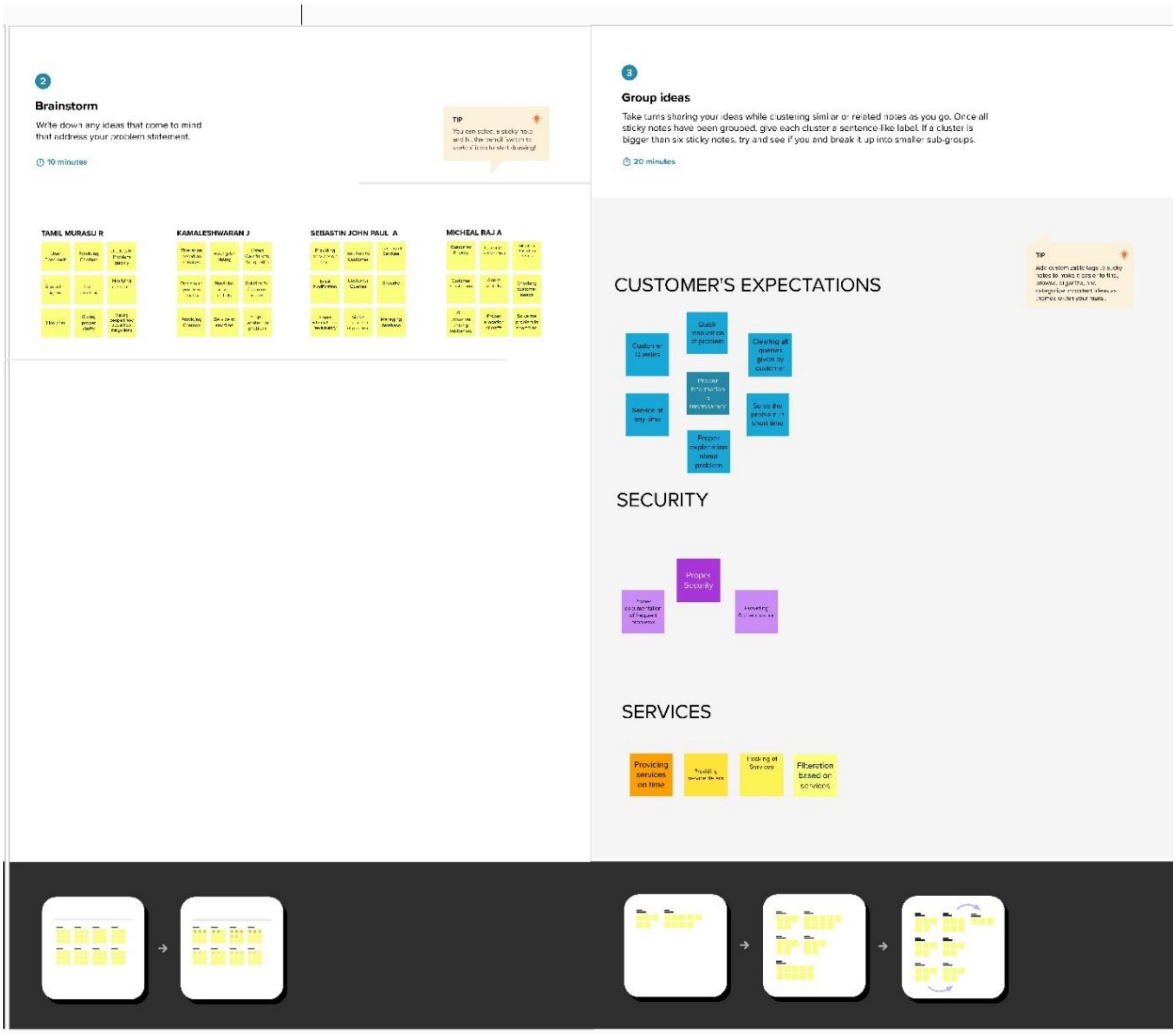
5 minutes

PROBLEM
How might we [your problem statement]?

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.	Encourage wild ideas.
Defer judgment.	Listen to others.
Go for volume.	If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

3

Group ideas

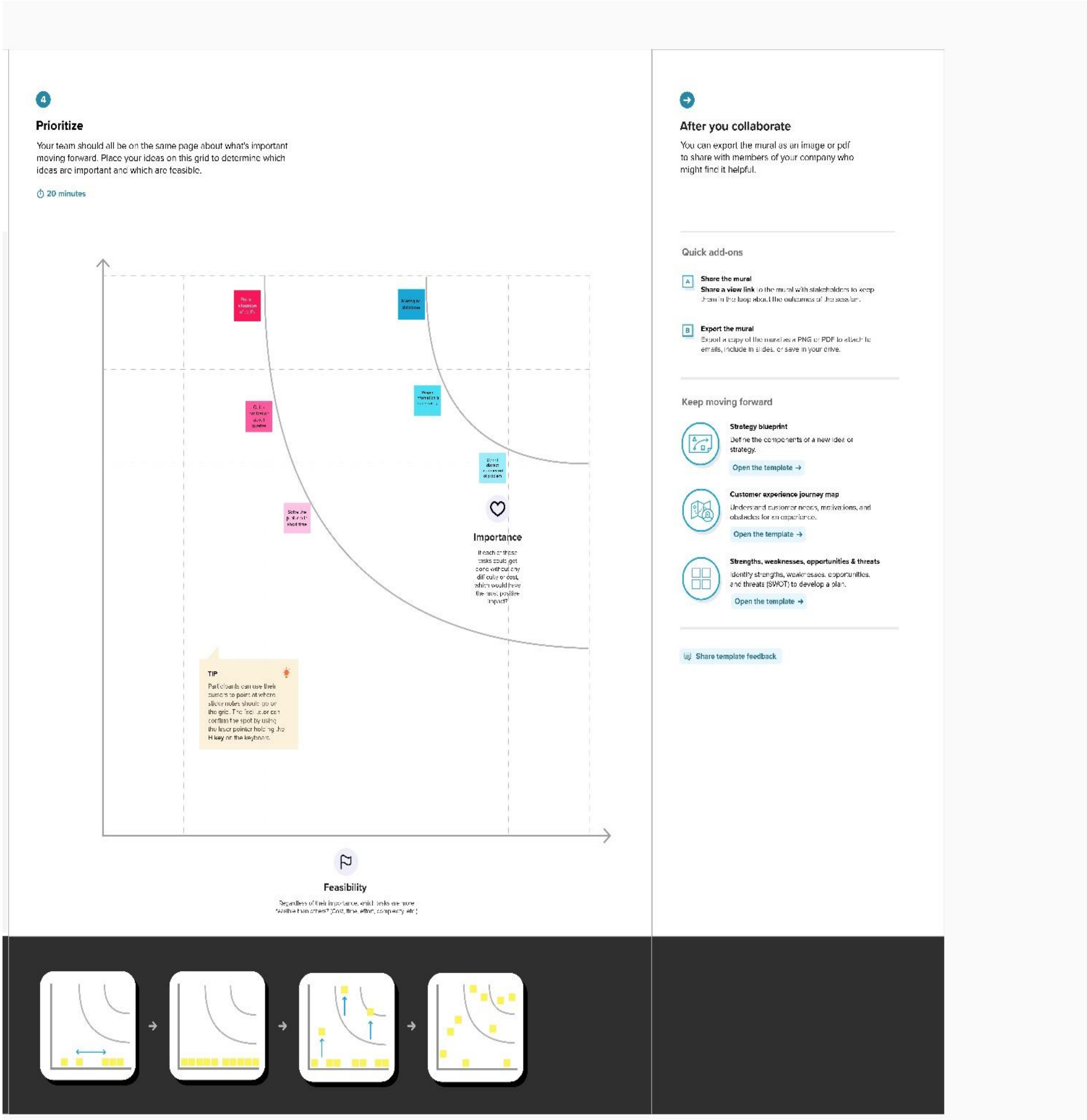
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Step-3: Idea Prioritization



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the H key on the keyboard.

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

A Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B Export the mural
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

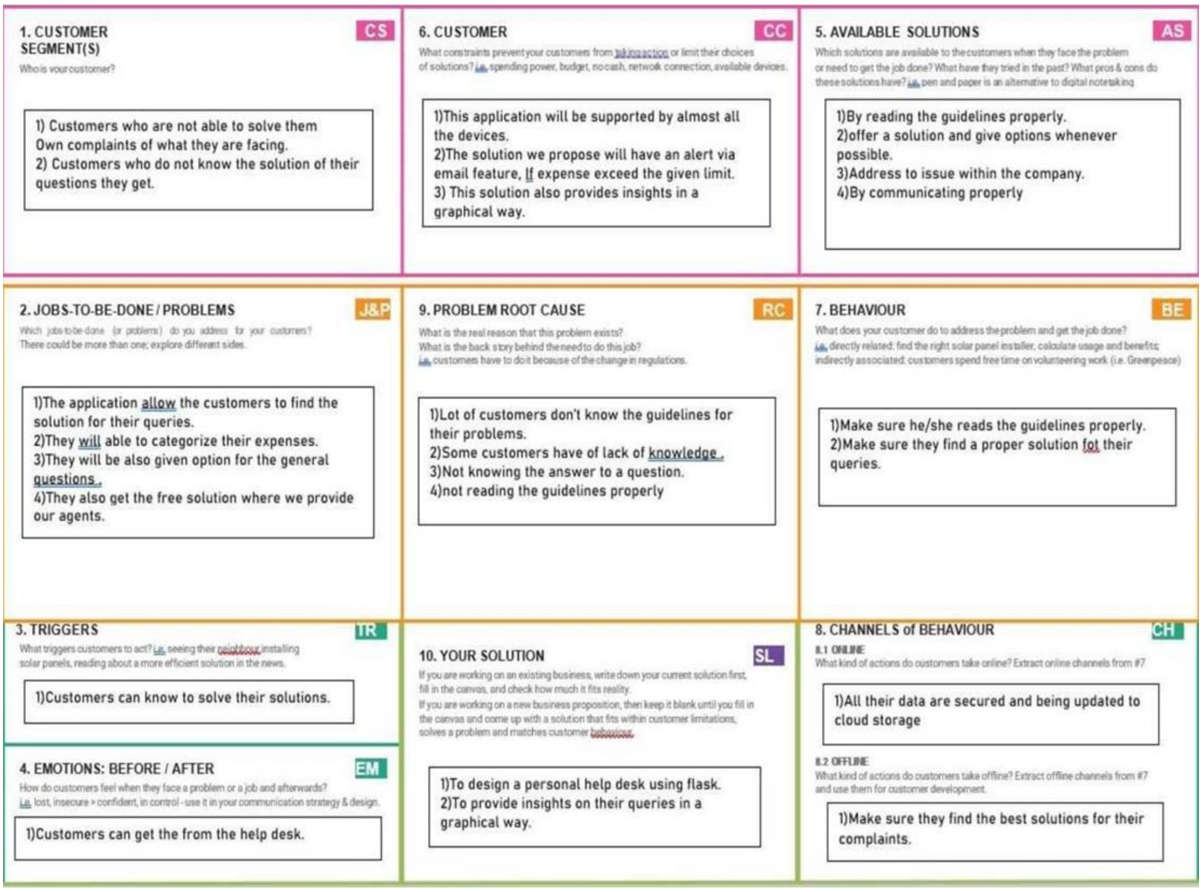
Share template feedback

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To solve customer issues using CloudApplication Development.
2.	Idea / Solution description	Assigned Agent routing can be solved by directly routing to the specific agent about the issue using the specific Email. Automated Ticket closure by using daily sync of the daily database. Status Shown to the Customer can display the status of the ticket to the customer.Regular data retrieval in the form of retrieving lost data.
3.	Novelty / Uniqueness	Assigned Agent Routing, Automated Ticket Closure, Status Shown to the Customer, and Backup data in case of failures.
4.	Social Impact / Customer Satisfaction	Customer Satisfaction, Customer can track their status and Easy agent communication.
5.	Business Model (Revenue Model)	◆ Key Partners are Third-party applications, agents, and customers. ◆ Activities held as Customer Service,System Maintenance. ◆ Key Resources support

		Engineers,Multi-channel. ◆ Customer Relationship have 24/7 Email Support, Knowledge-based channel.
6.	Scalability of the Solution	The real goal of scaling customer service is providing an environment that will allow your customer service specialists to be as efficient as possible. An environment where they will be able to spend less time on gruntwork and more time on actually resolving critical customer issues

3.4 Problem Solution Fit



1. CUSTOMER SEGMENT(S) CS

Who is your customer?

1) Customers who are not able to solve them Own complaints of what they are facing.
2) Customers who do not know the solution of their questions they get.

6. CUSTOMER CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

1)This application will be supported by almost all the devices.
2)The solution we propose will have an alert via email feature, If expense exceed the given limit.
3) This solution also provides insights in a graphical way.

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

1)By reading the guidelines properly.
2)offer a solution and give options whenever possible.
3)Address to issue within the company.
4)By communicating properly

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

1)The application allow the customers to find the solution for their queries.
2)They will able to categorize their expenses.
3)They will be also given option for the general questions .
4)They also get the free solution where we provide our agents.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

1)Lot of customers don't know the guidelines for their problems.
2)Some customers have of lack of knowledge .
3)Not knowing the answer to a question.
4)not reading the guidelines properly

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

1)Make sure he/she reads the guidelines properly.
2)Make sure they find a proper solution fot their queries.

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

1)Customers can know to solve their solutions.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

1)Customers can get the from the help desk.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

1)To design a personal help desk using flask.
2)To provide insights on their queries in a graphical way.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

1)All their data are secured and being updated to cloud storage

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

1)Make sure they find the best solutions for their complaints.

4. REQUIREMENT ANALYSIS

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.The main types of requirements analysis include business, customer, product, functional, and non-functional requirements. Each one represents a stakeholder or stage of the project and communicates the project needs.

4.1 Functional Requirements

FR No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn Register with valid mobile number
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP Two step verification for new device login.
FR-3	Agent Registration	Registration through Form Registration through Gmail Registration through LinkedIn

		Register with valid mobile number
FR-4	Agent Confirmation	Confirmation via Email Confirmation via OTP Two step verification for new device login.
FR-5	Admin	Admin have both user details and agent detail. Admin maintain agent allotment to the user based onproblem's category.

4.2 Non-functional Requirements:

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	To provide optimal usability for our proposed solution wehave mainly concentrated on easier navigation throughoutour website. For user, they can easily login with their credentials and also they can register by themselves eitherwith unique valid email id or with their mobile number if they don't have any prior account. After good navigation we have concentrated on visual clarity and developed web application which looks pleasant and simple thus making easier accessible to anyaged person. For the first time users, Guide tour will alsobe available in order to provide

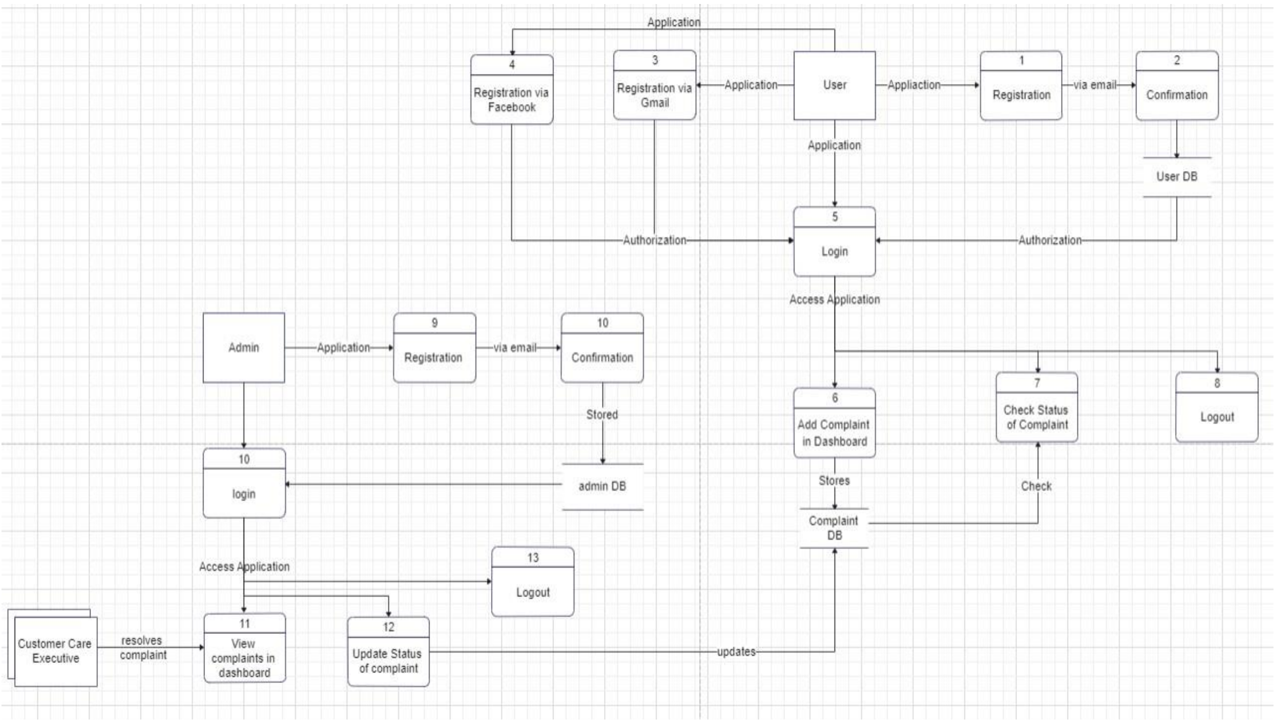
		better user satisfaction. Also, made our web application flexible to all type of devices such as android, mac and desktops.
NFR-2	**Security**	Before any user trying to login their account to any newdevice ,verification code will be sent either to their registered email id or to their registered mobile number.Only after entering their code, they will be allowed to login. That code will also made expire within particular time limit. Also notification will be sent for each and every user activity. Thus everyone will have a secured accountand also their details will be maintained securely in the admin side.
NFR-3	**Reliability**	Since we had split the agents into categories, system's response time for each and every individual will be lesser. Thus making our web application more reliable.
NFR-4	**Performance**	In order to bring best performance, we have concentratedon overload of user requests. To minimize the overloads and to minimize the system's response time we have created more agents service. Agents will be separated and categorized according to the user's needs. For example toresolve product missing category some agents will be assigned and to resolve damaged products category someagents will

		be assigned. so every individual user will be allotted with individual agents.
NFR-5	**Availability**	Customer care registry will be made available even in the weekends and our agents will also be allotted at anytime to any individual user. User can interact with their respective agents 24*7 by following proper user-agent guidelines.
NFR-6	**Scalability**	With respect to increase in user’s requests ,allotment willbe increased. Data storage will increase accordingly. Rescaling is always adaptable.

5. PROJECT DESIGN

5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

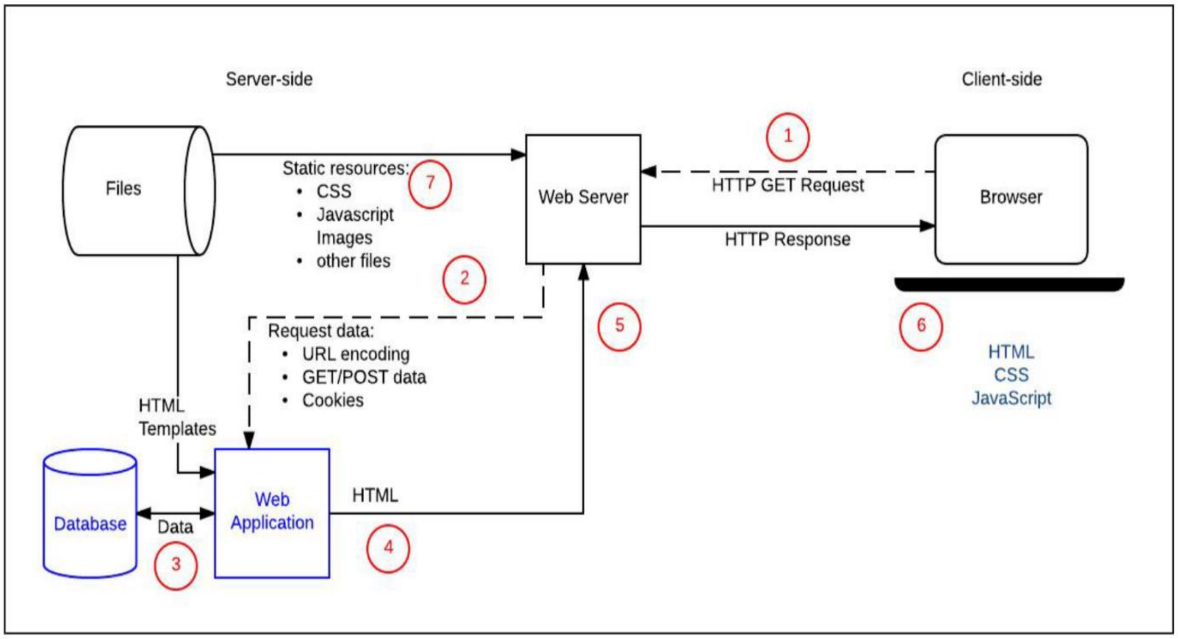


5.2 Solution and Technical Architecture

Assigned Agent routing can be solved by directly routing to the specific agent about theissue using the specific E-mail. Automated Ticket closure by using daily sync of the daily database. Status Shown to the Customer can display the status of the ticket to the customer. Regular data retrieval in the form of retrieving lost data.

Solution Architecture :

The Deliverable shall include the architectural diagram as below and the information.



Technologies needed for Minimum Viable Product deployment

- ➢ Working with HTML & CSS Using Frontend
- ➢ Working with image processing technique
- ➢ Working with Tensorflow capabilities
- ➢ Working with Keras capabilities
- ➢ Working Trained CNN model
- ➢ Build a web application using the Flask framework

Platform

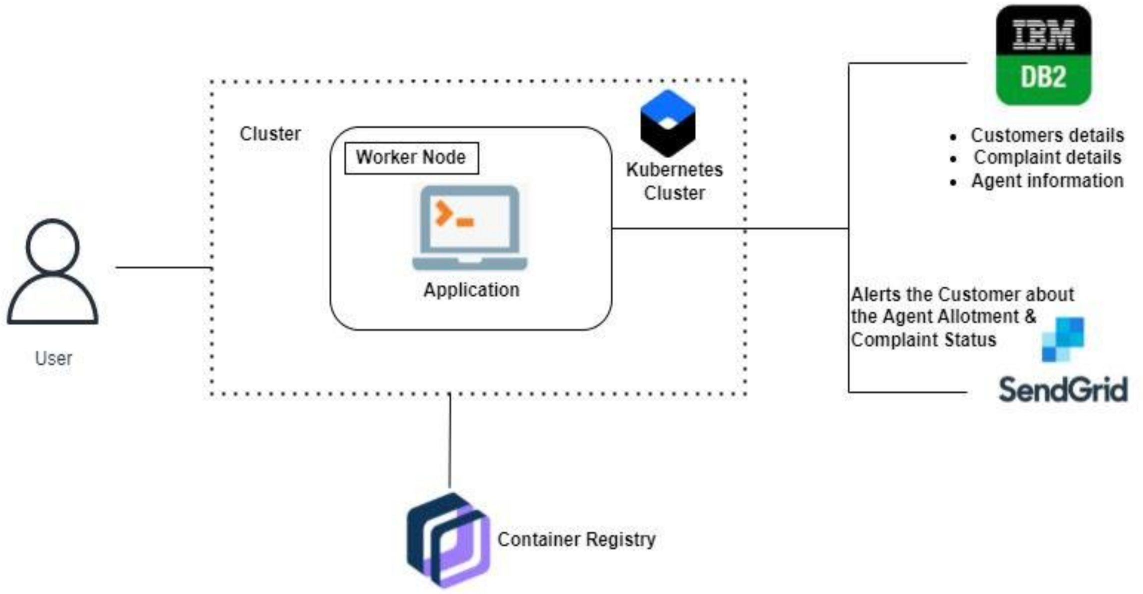
➢ Git & GitHub - Project Management

➢ IBM Cloud - Hosting

➢ IBM Watson – Using Customer Care

Technical Architecture

The Deliverable shall include the architectural diagram as below and the information



5.3 User Stories :

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1

		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-2
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a user , I can register the complaint in the register complaint page	I can register complaint(s)	High	Sprint-1
		USN-7	As a user , I can view the status of the complaint.	I can view status of complaint	Medium	Sprint-1
		USN-8	As a user, I can logout of the application	I can logout from the application	Low	Sprint-2
Customer Care Executive	Dashboard	USN-8	As a customer care Executive, I can resolve a complaint registered by user.	I can provide solution to a problem.	High	Sprint -1
Administrator	Registration	USN-9	As an admin, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-10	As an admin, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	Sprint-1	
	Login	USN-11	As an admin I can log into the application(admin panel) by entering email & password		High	Sprint-1
	Dashboard	USN-12	As an admin, I can update the status of the complaint to the user with the help of customer care executive.	I can satisfy the customer on his/her query.	Medium	Spritn-2
		USN-13	As an admin , I can logout from the application	I can logout from the application	Low	Sprint -2

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Customer (Web User)	Registration	USN-1	As a customer, I can register for the application byentering my email, password, and confirming my password.	2	High	Micheal Raj, Kamalesh
Sprint-1		Login	USN-2	As a customer, I can login to the application by entering correct email and password	1	High	Sebastin John Paul
Sprint-1		Dashboard	USN-3	As a customer, I can see all the tickets raised by meand lot more	3	High	Tamil Murasu
Sprint-2		Ticket creation	USN-4	As a customer, I can create a new ticket with the detailed description of my query	2	High	Micheal Raj
Sprint-3		Address Column	USN-5	As a customer, I can have conversations with the assigned agent and get my queries clarified	3	High	Kamaleshwaran, Sebastin John Paul
Sprint-4		Forgot password	USN-6	As a customer, I can reset my password by this optionin case I forgot my old password	2	Medium	Tamil Murasu, Kamaleshwaran

S20.4Sprint	User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4		Ticket details	USN-7	As a customer, I can see the current status of my tickets	2	Medium	Micheal Raj, Sebastin John Paul
Sprint-3	Agent (Web user)	Login	USN-1	As an agent, I can login to the application by enteringcorrect email and password	2	High	Tamil Murasu
Sprint-3		Dashboard	USN-2	As an agent, I can see all the tickets assigned to me by the admin	3	High	Micheal Raj
Sprint-3		Address Column	USN-3	As an agent, I get to have conversations with thecustomer and clear his/her queries	3	High	Kamaleshwaran, Sebastin John Paul
Sprint-4		Forgot password	USN-4	As an agent, I can reset my password by this option incase I forgot my old password	2	Medium	Tamil Murasu, Micheal Raj
Sprint-1	Admin (Web user)	Login	USN-1	As an admin, I can login to the application by enteringcorrect email and password	1	High	Kamleshwaran, Sebastin John Paul
Sprin		Dashboar	U	As an admin, I can see all the tickets	3	High	Micheal Raj,

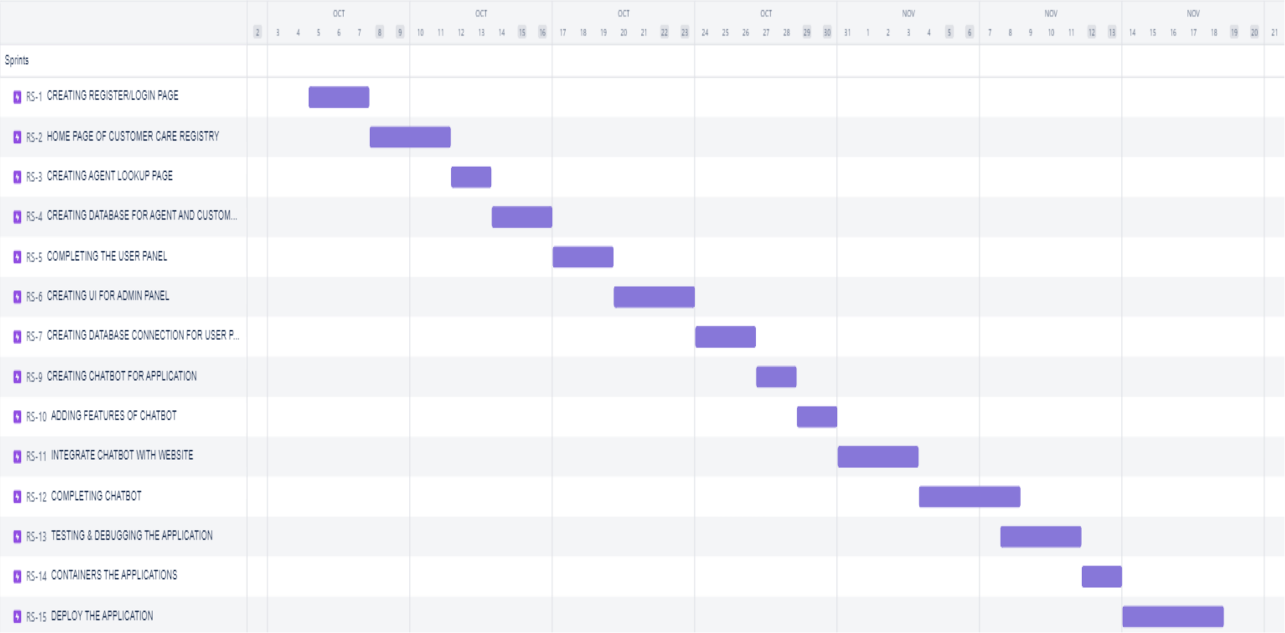
t-1		d	SN-2	raised in the entire system and lot more			Kamaleshwaran
Sprint-2		Agent creation	USN-3	As an admin, I can create an agent for clarifying the customer's queries	2	High	Tamil Murasu, Sebastin John Paul
Sprint-2		Assigning agent	USN-4	As an admin, I can assign an agent for each ticketcreated by the customer	3	High	Micheal Raj, Sebastin John Paul
Sprint-4		Forgot password	USN-4	As an admin, I can reset my password by this option incase I forgot my old password	2	Medium	Tamil Murasu, Kamaleshwaran

6.2 Sprint Delivery Shedule:

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	10	6 Days	24 Oct 2022	29 Oct 2022	10	29 Oct 2022
Sprint-2	7	6 Days	31 Oct 2022	05 Nov 2022	7	05 Nov 2022
Sprint-3	11	6 Days	07 Nov 2022	12 Nov 2022	11	12 Nov 2022
Sprint-4	8	6 Days	14 Nov 2022	19 Nov 2022	8	19 Nov 2022

6.3 Reports from JIRA

BURNDOWN CHART


OCT
OCT
OCT
OCT
NOV
NOV
NOV
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
Sprints
RS-1 CREATING REGISTER/LOGIN PAGE
RS-2 HOME PAGE OF CUSTOMER CARE REGISTRY
RS-3 CREATING AGENT LOOKUP PAGE
RS-4 CREATING DATABASE FOR AGENT AND CUSTOM...
RS-5 COMPLETING THE USER PANEL
RS-6 CREATING UI FOR ADMIN PANEL
RS-7 CREATING DATABASE CONNECTION FOR USER P...
RS-9 CREATING CHATBOT FOR APPLICATION
RS-10 ADDING FEATURES OF CHATBOT
RS-11 INTEGRATE CHATBOT WITH WEBSITE
RS-12 COMPLETING CHATBOT
RS-13 TESTING & DEBUGGING THE APPLICATION
RS-14 CONTAINERS THE APPLICATIONS
RS-15 DEPLOY THE APPLICATION

7. CODING & SOLUTIONING

College graduates with prior programming expertise or technical degrees are recruited and transitioned into professional positions with Alabama firms and organisations through the highly competitive Coding Solutions job accelerator andtalent refinement programme at no cost to the graduates. We provide a pool of varied, well-trained, techs-savvy individuals that wants to launch and advance their career in Alabama.

The mission of veteran- and woman-owned CodingSolutions is to mobilise the nextgeneration of IT talent and provide them the tools and resources they require to make your business successful. Innovative talent is necessary for innovative technologies. We wish to provide CodingSolutions prospects to assist you expand your Alabama team. Our applicants are swiftly hired at the top of the list by growing businesses forlucrative, long-term positions.

7.1 Feature 1 :

(Frontend)

Admin Dashboard :

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="UTF-8">
    <title> Responsiive Admin Dashboard | CodingLab </title>
    <!-- Boxicons CDN Link -->
     <meta name="viewport" content="width=device-width, initial-scale=1.0">
   </head>
<body>
```

```
<script>
window.watsonAssistantChatOptions = {
integrationID: "66576f0c-5408-4edc-803b-d9de1f553e8b", // The ID of this integration.region: "eu-gb", // The region your integration is hosted in.
serviceInstanceID: "2607efc7-375b-465c-9e61-399a0f694519", // The ID of your service
instance.
onLoad: function(instance) { instance.render(); }
};
setTimeout(function(){
const t=document.createElement('script');
t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" +
(window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
document.head.appendChild(t);
});
</script>
<style>
/* Googlefont Poppins CDN Link */
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;600;700&display=swap');
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Poppins', sans-serif;
}
.sidebar{
  position: fixed;
  height: 100%;
  width: 240px;
  background: #0A2558;
  transition: all 0.5s ease;
```

```
}
.sidebar.active{
  width: 60px;
}
.sidebar .logo-details{
  height: 80px;
  display: flex;
  align-items: center;
}
.sidebar .logo-details i{
  font-size: 28px;
  font-weight: 500;
  color: #fff;
  min-width: 60px;
  text-align: center
}
.sidebar .logo-details .logo_name{
  color: #fff;
  font-size: 24px;
  font-weight: 500;
}
.sidebar .nav-links{
  margin-top: 10px;
}
.sidebar .nav-links li{
  position: relative;
  list-style: none;
  height: 50px;
}
.sidebar .nav-links li a{
  height: 100%;
  width: 100%;
  display: flex;
  align-items: center;
  text-decoration: none;
  transition: all 0.4s ease;
}
.sidebar .nav-links li a.active{
  background: #081D45;
}
.sidebar .nav-links li a:hover{
  background: #081D45;
}
.sidebar .nav-links li i{
  min-width: 60px;
  text-align: center;
  font-size: 18px;
  color: #fff;
}
.sidebar .nav-links li a .links_name{
  color: #fff;
  font-size: 15px;
  font-weight: 400;
  white-space: nowrap;
}
.sidebar .nav-links .log_out{
  position: absolute;
  bottom: 0;
  width: 100%;
}
```

```
.home-section{
  position: relative;
  background: #f5f5f5;
  min-height: 100vh;
  width: calc(100% - 240px);
  left: 240px;
  transition: all 0.5s ease;
}
.sidebar.active ~ .home-section{
  width: calc(100% - 60px);
  left: 60px;
}
.home-section nav{
  display: flex;
  justify-content: space-between;
  height: 80px;
  background: #fff;
  display: flex;
  align-items: center;
  position: fixed;
  width: calc(100% - 240px);
  left: 240px;
  z-index: 100;
  padding: 0 20px;
  box-shadow: 0 1px 1px rgba(0, 0, 0, 0.1);
  transition: all 0.5s ease;
}
.sidebar.active ~ .home-section nav{
  left: 60px;
  width: calc(100% - 60px);
```

```
}
.home-section nav .sidebar-button{
  display: flex;
  align-items: center;
  font-size: 24px;
  font-weight: 500;
}
nav .sidebar-button i{
  font-size: 35px;
  margin-right: 10px;
}
.home-section nav .search-box{
  position: relative;
  height: 50px;
  max-width: 550px;
  width: 100%;
  margin: 0 20px;
}
nav .search-box input{
  height: 100%;
  width: 100%;
  outline: none;
  background: #F5F6FA;
  border: 2px solid #EFEEF1;
  border-radius: 6px;
  font-size: 18px;
  padding: 0 15px;
}
nav .search-box .bx-search{
  position: absolute;
  height: 40px;
  width: 40px;
  background: #2697FF;
  right: 5px;
  top: 50%;
  transform: translateY(-50%);
  border-radius: 4px;
  line-height: 40px;
  text-align: center;
  color: #fff;
  font-size: 22px;
  transition: all 0.4 ease;
}
.home-section nav .profile-details{
  display: flex;
  align-items: center;
  background: #F5F6FA;
  border: 2px solid #EFEEF1;
  border-radius: 6px;
  height: 50px;
  min-width: 190px;
  padding: 0 15px 0 2px;
}
nav .profile-details img{
  height: 40px;
  width: 40px;
  border-radius: 6px;
  object-fit: cover;
}
nav .profile-details .admin_name{
  font-size: 15px;
  font-weight: 500;
  color: #333;
  margin: 0 10px;
  white-space: nowrap;
}
nav .profile-details i{
  font-size: 25px;
  color: #333;
}
.home-section .home-content{
  position: relative;
  padding-top: 104px;
}
.home-content .overview-boxes{
  display: flex;
  align-items: center;
  justify-content: space-between;
  flex-wrap: wrap;
  padding: 0 20px;
  margin-bottom: 26px;
}
.overview-boxes .box{
  display: flex;
  align-items: center;
  justify-content: center;
  width: calc(416% / 4 - 15px);
  background: #fff;
  padding: 30px 14px;
  border-radius: 12px;
  box-shadow: 0 5px 10px rgba(0,0,0,0.1);
}
.overview-boxes .box-topic{
  font-size: 20px;
  font-weight: 500;
}
.home-content .box .number{
  display: inline-block;
  font-size: 35px;
  margin-top: -6px;
  font-weight: 500;
}
.home-content .box .indicator{
  display: flex;
  align-items: center;
}
.home-content .box .indicator i{
  height: 20px;
  width: 20px;
  background: #8FDACB;
  line-height: 20px;
  text-align: center;
  border-radius: 50%;
  color: #fff;
  font-size: 20px;
  margin-right: 5px;
}
.box .indicator i.down{
  background: #e87d88;
}
```

```
.home-content .box .indicator .text{
  font-size: 12px;
}
.home-content .box .cart{
  display: inline-block;
  font-size: 32px;
  height: 50px;
  width: 50px;
  background: #cce5ff;
  line-height: 50px;
  text-align: center;
  color: #66b0ff;
  border-radius: 12px;
  margin: -15px 0 0 6px;
}
.home-content .box .cart.two{
   color: #2BD47D;
   background: #C0F2D8;
 }
.home-content .box .cart.three{
   color: #ffc233;
   background: #ffe8b3;
 }
.home-content .box .cart.four{
   color: #e05260;
   background: #f7d4d7;
 }
.home-content .total-order{
  font-size: 20px;
  font-weight: 500;
```

```css
}
.home-content .sales-boxes{
  display: flex;
  justify-content: space-between;
  /* padding: 0 20px; */
}

/* left box */
.home-content .sales-boxes .recent-sales{
  width: 28%;
  background: #fff;
  padding: 20px 30px;
  margin: 0 20px;
  border-radius: 12px;
  box-shadow: 0 5px 10px rgba(0, 0, 0, 0.1);
}
.home-content .sales-boxes .sales-details{
  display: flex;
  align-items: center;
  justify-content: space-between;
}
.sales-boxes .box .title{
  font-size: 24px;
  font-weight: 500;
  /* margin-bottom: 10px; */
}
.sales-boxes .sales-details li.topic{
  font-size: 20px;
  font-weight: 500;
}
```

```
.sales-boxes .sales-details li{
  list-style: none;
  margin: 8px 0;
}
.sales-boxes .sales-details li a{
  font-size: 18px;
  color: #333;
  font-size: 400;
  text-decoration: none;
}
.sales-boxes .box .button{
  width: 100%;
  display: flex;
  justify-content: flex-end;
}
.sales-boxes .box .button a{
  color: #fff;
  background: #0A2558;
  padding: 4px 12px;
  font-size: 15px;
  font-weight: 400;
  border-radius: 4px;
  text-decoration: none;
  transition: all 0.3s ease;
}
.sales-boxes .box .button a:hover{
  background:  #0d3073;
}

/* Right box */
```

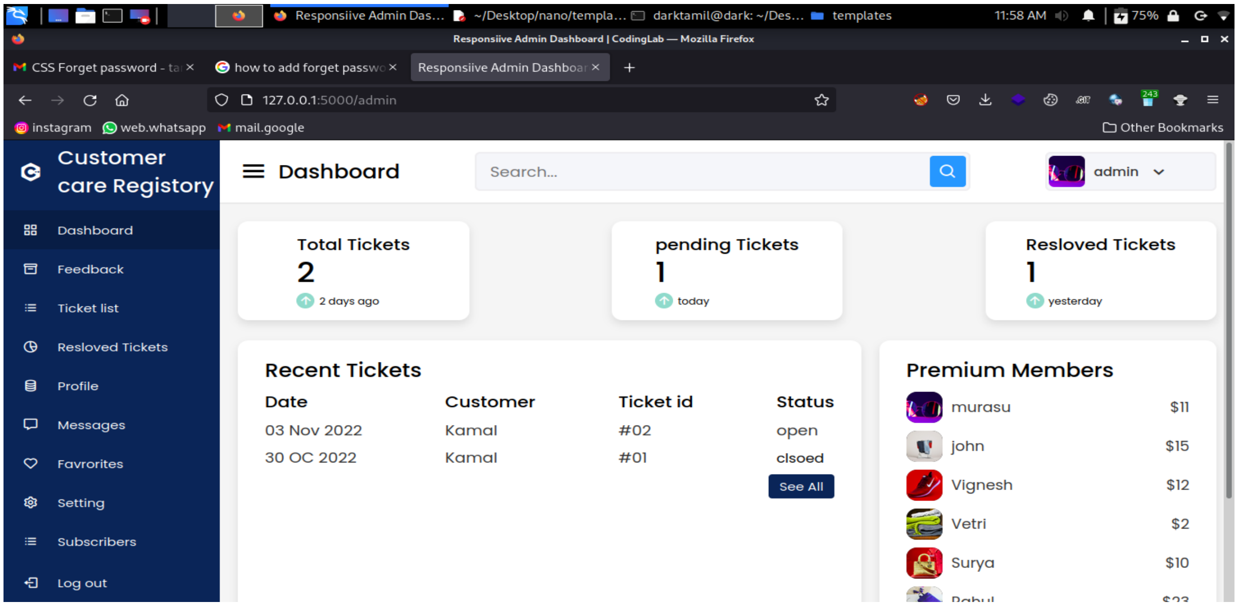
```
.home-content .sales-boxes .top-sales{
  width: 62%;
  background: #fff;
  padding: 20px 30px;
  margin: 0 20px 0 0;
  border-radius: 12px;
  box-shadow: 0 5px 10px rgba(0, 0, 0, 0.1);
}
.sales-boxes .top-sales li{
  display: flex;
  align-items: center;
  justify-content: space-between;
  margin: 10px 0;
}
.sales-boxes .top-sales li a img{
  height: 40px;
  width: 40px;
  object-fit: cover;
  border-radius: 12px;
  margin-right: 10px;
  background: #333;
}
.sales-boxes .top-sales li a{
  display: flex;
  align-items: center;
  text-decoration: none;
}
.sales-boxes .top-sales li .product,
.price{
  font-size: 17px;
  font-weight: 400;
  color: #333;
}
/* Responsive Media Query */
@media (max-width: 1240px) {
  .sidebar{
    width: 60px;
  }
  .sidebar.active{
    width: 220px;
  }
  .home-section{
    width: calc(100% - 60px);
    left: 60px;
  }
  .sidebar.active ~ .home-section{
    left: 220px;
    width: calc(100% - 220px);
    overflow: hidden;
  }
  .home-section nav{
    width: calc(100% - 60px);
    left: 60px;
  }
  .sidebar.active ~ .home-section nav{
    width: calc(100% - 220px);
    left: 220px;
  }
}
@media (max-width: 1150px) {
  .home-content .sales-boxes{
    flex-direction: column;
  }
  .home-content .sales-boxes .box{
    width: 100%;
    overflow-x: scroll;
    margin-bottom: 30px;
  }
  .home-content .sales-boxes .top-sales{
    margin: 0;
  }
}
@media (max-width: 1000px) {
  .overview-boxes .box{
    width: calc(175% / 1 - -137px);
    margin-bottom: 15px;
  }
}
@media (max-width: 700px) {
  nav .sidebar-button .dashboard,
  nav .profile-details .admin_name,
  nav .profile-details i{
    display: none;
  }
  .home-section nav .profile-details{
    height: 50px;
    min-width: 40px;
  }
  .home-content .sales-boxes .sales-details{
    width: 560px;
  }
}
@media (max-width: 550px) {
  .overview-boxes .box{
    width: 100%;
    margin-bottom: 15px;
  }
  .sidebar.active ~ .home-section nav .profile-details{
    display: none;
  }
}
@media (max-width: 400px) {
  .sidebar{
    width: 0;
  }
  .sidebar.active{
    width: 60px;
  }
  .home-section{
    width: 100%;
    left: 0;
  }
  .sidebar.active ~ .home-section{
    left: 60px;
    width: calc(100% - 60px);
  }
  .home-section nav{
    width: 100%;
    left: 0;
  }
  .sidebar.active ~ .home-section nav{
    left: 60px;
    width: calc(100% - 60px);
  }
}
*{
text-decoration: none;
}
h1{
    text-align: center;
}
</style>

 <link href='https://unpkg.com/boxicons@2.0.7/css/boxicons.min.css' rel='stylesheet'>
     <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <div class="sidebar">
    <div class="logo-details">
      <i class='bx bxl-c-plus-plus'></i>
      <span class="logo_name">Customer care Registory</span>
    </div>
      <ul class="nav-links">
        <li>
          <a href="#" class="active">
            <i class='bx bx-grid-alt' ></i>
            <span class="links_name">Dashboard</span>
          </a>
        </li>
        <li>

      <a href="/form">
     <i class='bx bx-box' ></i>
     <span class="links_name" >Create New Ticket</span>
    </a>
   </li>

   <li>
    <a href="#">
     <i class='bx bx-list-ul' ></i>
     <span class="links_name">Services</span>
    </a>
   </li>
   <li class="log_out">
    <a href="/register">
     <i class='bx bx-log-out'></i>
     <span class="links_name">Log out</span>
    </a>
   </li>
  </ul>
</div>
<section class="home-section">
 <nav>
  <div class="sidebar-button">
   <i class='bx bx-menu sidebarBtn'></i>
   <span class="dashboard">Dashboard</span>
  </div>
  <div class="profile-details">
   <img src="{{url_for('static', filename='sunglasses.jpg')}}" alt="">
   <span class="admin_name">{{session["username"]}}</span>
   <i class='bx bx-chevron-down' ></i>
  </div>
```

```
</nav>

<div class="home-content">
  <div class="overview-boxes">

    <div class="box">
      <div class="right-side">
        <div class="box-topic"><a href="/admininfo">Welcome Admin</a></div>

      </div>

    </div>
  </div>

  <div class="sales-boxes">
    <div class="recent-sales box">
       <ul class="top-sales-details">

        <a href="#">

          <h1 style="color: rgba(39, 8, 239, 0.656);">Agents</h1>
          <ul class="pricing-plan__list">
            <li class="pricing-plan__feature" style="padding-left: 30px">Tamil Murasu</li>
            <li class="pricing-plan__feature" style="padding-left: 30px">Micheal Raj</li>
            <li class="pricing-plan__feature" style="padding-left: 30px">Kamaleshwaran</li>
            <li class="pricing-plan__feature" style="padding-left: 30px">Sebastin John Paul</li>
        </ul>
        </a>
```

```
</div>
<div class="top-sales box">
  <div class="title" style="color: rgba(39, 8, 239, 0.656)">Membership holders</div>
  <ul class="top-sales-details">
    <li>
    <a href="#">
      <img src="{{url_for('static', filename='sunglasses.jpg')}}" alt="tamil">
      <span class="product">murasu</span>
    </a>
    <span class="price">Premimum user</span>
  </li>
  <li>
    <a href="#">
      <img src="{{url_for('static', filename='jeans.jpg')}}" alt="">
      <span class="product">john</span>
    </a>
    <span class="price">Standard user</span>
  </li>
  <li>
    <a href="#">
      <img src="{{url_for('static', filename='nike.jpg')}}" alt="">
      <span class="product">Vignesh</span>
    </a>
    <span class="price">Premimum user</span>
  </li>
  <li>
    <a href="#">
      <img src="{{url_for('static', filename='scarves.jpg')}}" alt="">
      <span class="product">Vetri</span>
    </a>
```

```
          <span class="price">Premimum user</span>
        </li>
        <li>
          <a href="#">
            <img src="{{url_for('static', filename='bag.jpg')}}" alt="">
            <span class="product">Surya</span>
          </a>
          <span class="price">Standard user</span>
        </li>
        <li>
          <a href="#">
            <img src="{{url_for('static', filename='blueBag.jpg')}}" alt="">
            <span class="product">Rahul</span>
          </a>
          <span class="price">Premimum user</span>
        <li>
          <a href="#">
            <img src="{{url_for('static', filename='addidas.jpg')}}" alt="">
            <span class="product">Asmira</span>
          </a>
          <span class="price">Premimum user</span>
        </li>
<li>
          <a href="#">
            <img src="{{url_for('static', filename='shirt.jpg')}}" alt="">
            <span class="product">Bala</span>
          </a>
          <span class="price">Standard user</span>
        </li>
        </ul>
        </div>
      </div>
    </div>
  </section>
  <script>
   let sidebar = document.querySelector(".sidebar");
let sidebarBtn = document.querySelector(".sidebarBtn");
sidebarBtn.onclick = function() {
  sidebar.classList.toggle("active");
  if(sidebar.classList.contains("active")){
  sidebarBtn.classList.replace("bx-menu" ,"bx-menu-alt-right");
}else
  sidebarBtn.classList.replace("bx-menu-alt-right", "bx-menu");
}
 </script>

</body>
</html>
```

Output :



Customer care Registory

- Dashboard
- Feedback
- Ticket list
- Resloved Tickets
- Profile
- Messages
- Favrorites
- Setting
- Subscribers
- Log out

Dashboard

Search...

admin

Total Tickets
2
2 days ago

pending Tickets
1
today

Resloved Tickets
1
yesterday

Recent Tickets

Date	Customer	Ticket id	Status
03 Nov 2022	Kamal	#02	open
30 OC 2022	Kamal	#01	clsoed

See All

Premium Members

murasu	$11
john	$15
Vignesh	$12
Vetri	$2
Surya	$10
Rahul	$23

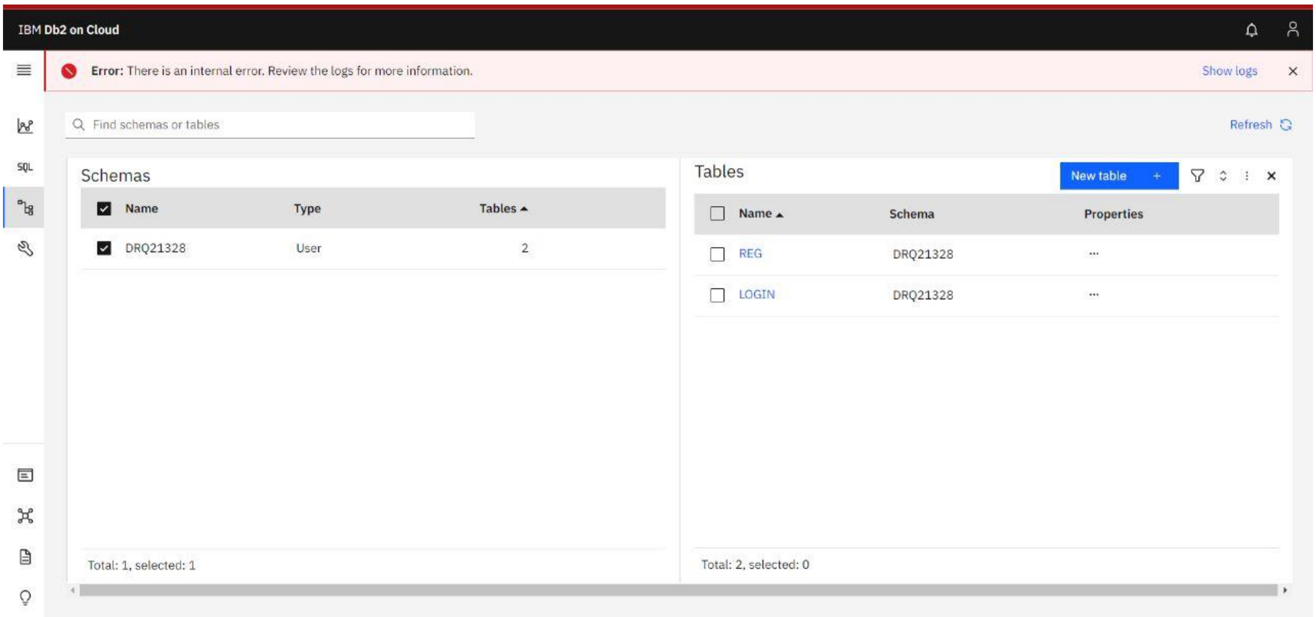
7.2 Feature 2 :

(Backend)

```
from flask import Flask, render_template, request, redirect, url_for, session

from flask_mail import Mail,Message

import ibm_db

import re


app = Flask(__name__)

app.config['MAIL_SERVER'] = 'smtp.gmail.com'

app.config['MAIL_PORT'] = 465

app.config['MAIL_USERNAME'] = 'tmkssolution1234@gmail.com'

app.config['MAIL_PASSWORD'] = 'xyfxjnripeytrqhk'

app.config['MAIL_USE_TLS'] = False

app.config['MAIL_USE_SSL'] = True


mail=Mail(app)


app.secret_key = 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-a1f5-
23dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32716;SECURIT
Y=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=jzc43091;PWD=PI8VtGRvZ
lSVT65A",'','')



@app.route('/')


def homer():

    return render_template('home.html')


@app.route('/agent')

def agent():
    return render_template("agent.html")


@app.route('/admin')
def agnt():
    return render_template("admin.html")
@app.route('/free')
def free():
    return render_template("free.html")



@app.route('/ForgotPassword', methods=['GET','POST'])
def ForgotPassword():
     if request.method=="POST":
        name=request.form['name']
        email=request.form['email']
        message=Message(subject, sender='tmkssolution1234@gmail.com', recipients=['johncaesar07@gmail.com', 'murasutamil2002@gmail.com','kamaleshwaran1123@gmail.com','hellsprince26@gmail.com'])


        message.body="""



        Hey Guys we have work.


        The Client sends a Query,


        Name - {}


        Email - {}


        Thank you


        """.format(name,email)
        mail.send(message)

        return render_template("success.html")
     return render_template("ForgotPassword.html")


@app.route('/login',methods =['GET', 'POST'])
def login():
    global userid
    msg = ''


    if request.method == 'POST' :
        username = request.form['username']
        password = request.form['password']
        sql = "SELECT * FROM users WHERE username =? AND password=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print (account)
        if account:
            session['loggedin'] = True
            session['id'] = account['USERNAME']
            userid=  account['USERNAME']
            session['username'] = account['USERNAME']
            msg = 'Logged in successfully !'

            msg = 'Logged in successfully !'
            return render_template('customer.html', msg = msg)
        else:
            msg = 'Incorrect username / password !'
    return render_template('login.html', msg = msg)




@app.route('/register', methods =['GET', 'POST'])
def registet():
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        sql = "SELECT * FROM users WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO  users VALUES (?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, username)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg = msg)

@app.route('/customer')
def dash():




    return render_template('customer.html')

@app.route('/plan')
def plan():
    return render_template('plan.html')





@app.route('/logout')

def logout():
   session.pop('loggedin', None)
   session.pop('id', None)
   session.pop('username', None)
   return render_template('home.html')





@app.route("/form", methods=['GET', 'POST'])
def form():
    if request.method=="POST":
        name=request.form['name']
        email=request.form['email']

        suppoter=request.form['suppoter']

        subject=request.form['subject']

        query=request.form['query']

        message=Message(subject, sender='tmkssolution1234@gmail.com',
recipients=['johncaesar07@gmail.com',
'murasutamil2002@gmail.com','kamaleshwaran1123@gmail.com','hellsprince26@gmail
.com'])


        message.body="""



        Hey Guys we have work.


        The Client sends a Query,


        Name - {}


        Email - {}


        Support - {}


        Query :  {}



        Thank you


        """.format(name,email,suppoter,query)


        mail.send(message)


        return render_template("success.html")
    return render_template("form.html")
```

```
if __name__ == '__main__':
   app.run(host='0.0.0.0')
```

7.3 Database Schema :



IBM Db2 on Cloud

Error: There is an internal error. Review the logs for more information.

Schemas

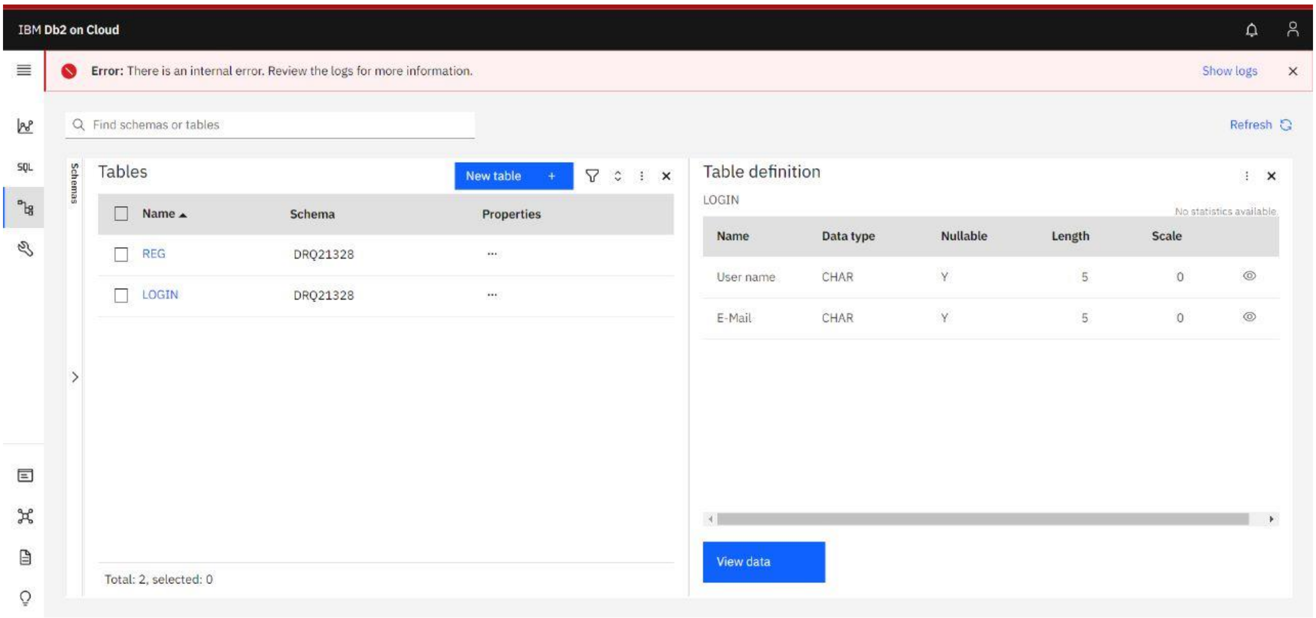
☑	Name	Type	Tables ▲
☑	DRQ21328	User	2

Total: 1, selected: 1

Tables

□	Name ▲	Schema	Properties
□	REG	DRQ21328	···
□	LOGIN	DRQ21328	···

Total: 2, selected: 0



IBM Db2 on Cloud

Error: There is an internal error. Review the logs for more information.

Tables

□	Name ▲	Schema	Properties
□	REG	DRQ21328	···
□	LOGIN	DRQ21328	···

Total: 2, selected: 0

Table definition

LOGIN

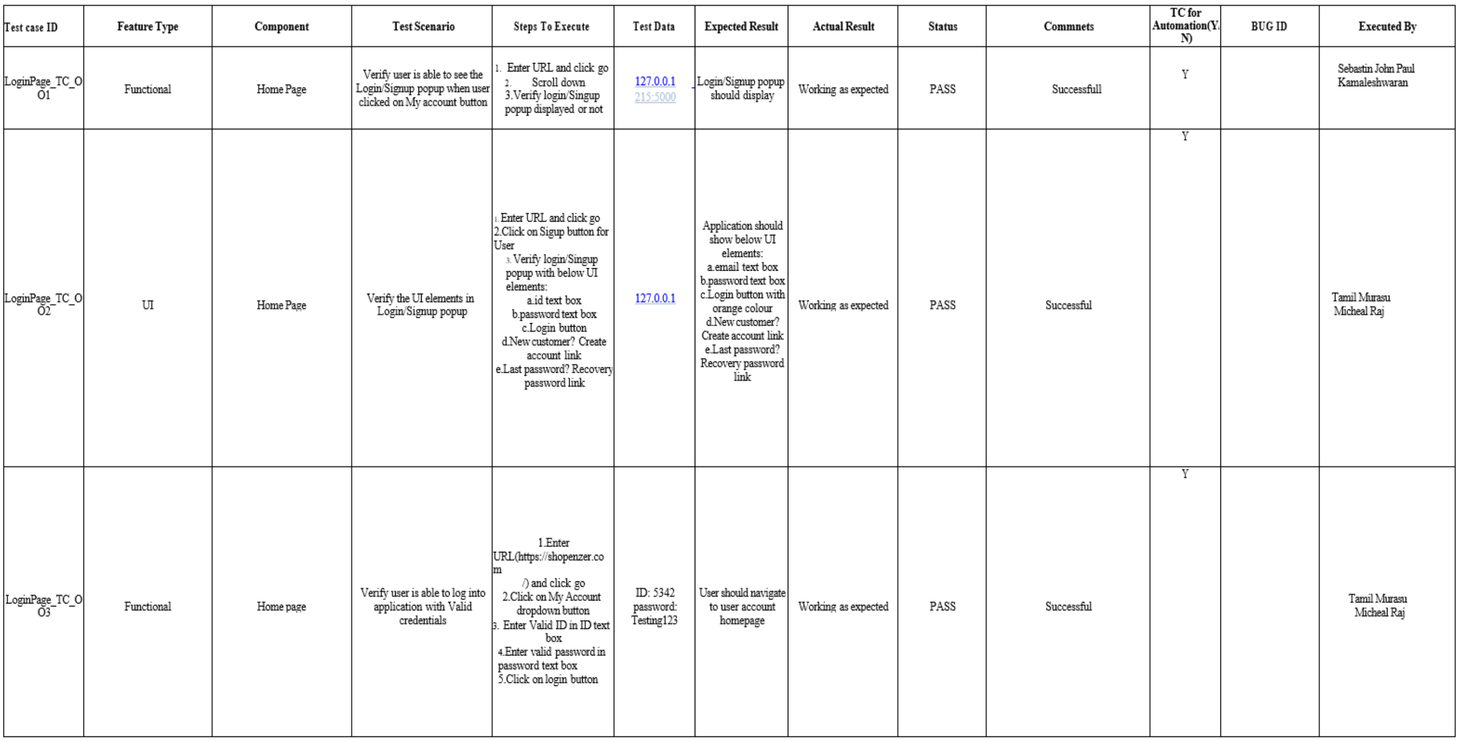
No statistics available

Name	Data type	Nullable	Length	Scale
User name	CHAR	Y	5	0
E-Mail	CHAR	Y	5	0

8. TESTING

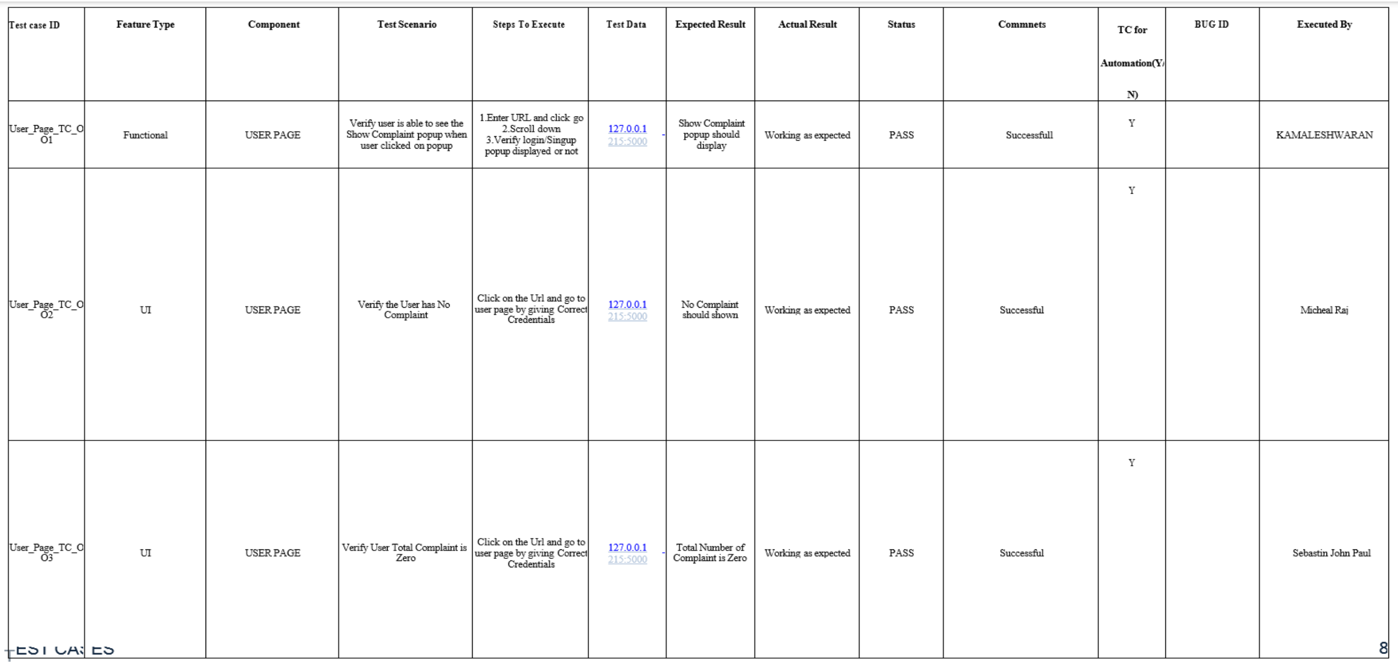
8.1 TEST CASES

A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a particular test scenario in order to verify compliance against a specific requirement. Test Case acts as the starting point for the test execution, and after applying a set of input values



Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_O O1	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button	1. Enter URL and click go 2. Scroll down 3.Verify login/Singup popup displayed or not	127.0.0.1 215:5000	Login/Signup popup should display	Working as expected	PASS	Successfull	Y		Sebastin John Paul Kamaleshwaran
LoginPage_TC_O O2	UI	Home Page	Verify the UI elements in Login/Signup popup	1. Enter URL and click go 2.Click on Sigup button for User 3. Verify login/Singup popup with below UI elements: a.id text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery password link	127.0.0.1	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.Last password? Recovery password link	Working as expected	PASS	Successful	Y		Tamil Murasu Micheal Raj
LoginPage_TC_O O3	Functional	Home page	Verify user is able to log into application with Valid credentials	1.Enter URL(https://shopenzer.co m /) and click go 2.Click on My Account dropdown button 3. Enter Valid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 5342 password: Testing123	User should navigate to user account homepage	Working as expected	PASS	Successful	Y		Tamil Murasu Micheal Raj

LoginPage_TC_OO4	Functional	Login page	Verify user is able to log into application with InValid credentials	1.Enter URL(127.0.0.1215:5000) and click go 2.Click on My Account dropdown button 3.Enter InValid ID in ID text box 4.Enter valid password in password text box 5.Click on login button	ID: 5342 password: Testing123	Application should show 'Incorrect email or password ' validation message.	Working as expected	PASS	Successful	Y		Tamil Murasu
LoginPage_TC_OO5	Functional	Login page	Verify user is able to log into application with InValid credentials	1.Enter URL(127.0.0.1215:5000) and click go 2.Click on My Account dropdown button 3.Enter Valid ID in ID text box 4.Enter Invalid password in password text box 5.Click on login button	ID: 5342 password: Testing12367868 6786876876	Application should show 'Incorrect email or password ' validation message.	Working as expected	PASS	Successful	Y		KAMALESHWARAN
LoginPage_TC OO6	Functional	Login page	Verify user is able to log into application with InValid credentials	1.Enter URL(127.0.0.1215:5000) and click go 2.Click on My Account dropdown button 3.Enter InValid ID in ID text box 4.Enter Invalid password in password text box 5.Click on login button	ID: 5342 password: Testing123	Application should show 'Incorrect email or password ' validation message.	Working as expected	PASS	Successful	Y		Micheal Raj



Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
User_Page_TC_OO1	Functional	USER PAGE	Verify user is able to see the Show Complaint popup when user clicked on popup	1.Enter URL and click go 2.Scroll down 3.Verify login/Singup popup displayed or not	127.0.0.1 215:5000	Show Complaint popup should display	Working as expected	PASS	Successfull	Y		KAMALESHWARAN
User_Page_TC_OO2	UI	USER PAGE	Verify the User has No Complaint	Click on the Url and go to user page by giving Correct Credentials	127.0.0.1 215:5000	No Complaint should shown	Working as expected	PASS	Successful	Y		Micheal Raj
User_Page_TC_OO3	UI	USER PAGE	Verify User Total Complaint is Zero	Click on the Url and go to user page by giving Correct Credentials	127.0.0.1 215:5000	Total Number of Complaint is Zero	Working as expected	PASS	Successful	Y		Sebastin John Paul

Admin_Page_TC_OO4	Functional	Admin Page	Admin can see the Agent DataBase	1.Enter URL(127.0.0.1215:5000) and click go 2.Enter the Credentials for the admin page and submit	127.0.0.1215:5000	Agent Database should display on show agent database.	Working as expected	PASS	Successful	Y		KAMALESHWARAN
Admin_Page_TC_OO5	Functional	Admin Page	Admin can delect the Agent Database	1.Enter URL(127.0.0.1215:5000) and click go 2.Click on submit by giving correct credentials to the admin Page	127.0.0.1215:5000	Delect the agent Database	Working as expected	PASS	Successful	Y		Micheal Raj
Admin_Page_TC_OO6	Functional	Admin Page	Verify the overall Delect the database for User	1.Enter URL(127.0.0.1215:5000) and click go 2.Click on submit by giving correct credentials to the admin Page 3.After type the "A" in the Text box for the agent database delect	127.0.0.1215:30106	Delect the overall Agent database delect.	Working as expected	PASS	Successful	Y		Tamil Murasu

Agent_Register_TC_O11	Functional	AGENT REGISTER	Verify Id sent to customer email address	1.Enter URL(127.0.0.1215:5000) and click go 1.Register the account by giving credentials 2. Click on button Submit	127.0.0.1215:30106/	Email sent successfully	Working as expected	PASS	Successful	Y		TAMIL MURASU
Web_Chat_TC_O11	Functional	WEB CHAT	Click on the Web chat button	1.Enter URL(127.0.0.1215:5000) and click go 1.Click on the Web Chat Button	127.0.0.1215:30106/	Web chat popup	Working as expected	PASS	Successful	N		MICHEAL RAJ
Web_chat_TC_O12	UI	WEB CHAT	Web chat button visible	1.Enter URL(127.0.0.1215:5000) and click go 1.shown on the Web Chat Button	127.0.0.1215:30106/	Web chat visible	Working as expected	PASS	Successful	N		TAMIL MURASU
Admin_Login_TC_O13	Functional	AGENT LOGIN	Verify user is able to get login id on emails	1. Enter URL(127.0.0.1215:5000) and click go 2.To the Agent Login page getting of emails	127.0.0.1215:30106/	Get Notified by Emails	Working as expected	PASS	Successful	Y		KAMALESHWARAN

7.2 USER ACCEPTANCE TESTING

1. PURPOSE OF DOCUMENT

The purpose of this document is to briefly explain the test coverage and open issues of the [Customer Care Registry] project at the time of the release to User Acceptance Testing (UAT).

2. DEFECT ANALYSIS

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	5	5	24
Duplicate	2	0	2	0	4
External	5	3	2	1	11
Fixed	15	5	5	10	35
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	32	17	17	18	84

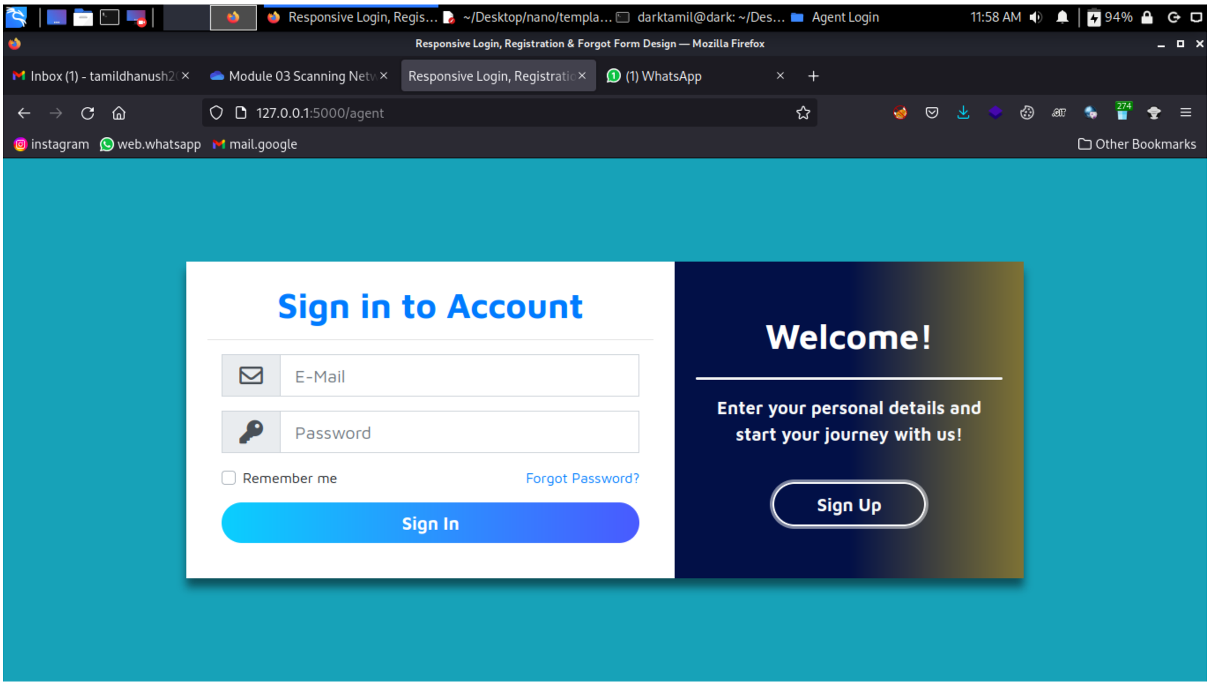
3. TEST CASE ANALYSIS

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	10	0	0	10
Client Application	40	0	0	40
Security	5	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	10	0	0	10
Final Report Output	4	0	0	4
Version Control	4	0	0	4

9. RESULT

Login page



Sign in to Account

E-Mail

Password

□ Remember me Forgot Password?

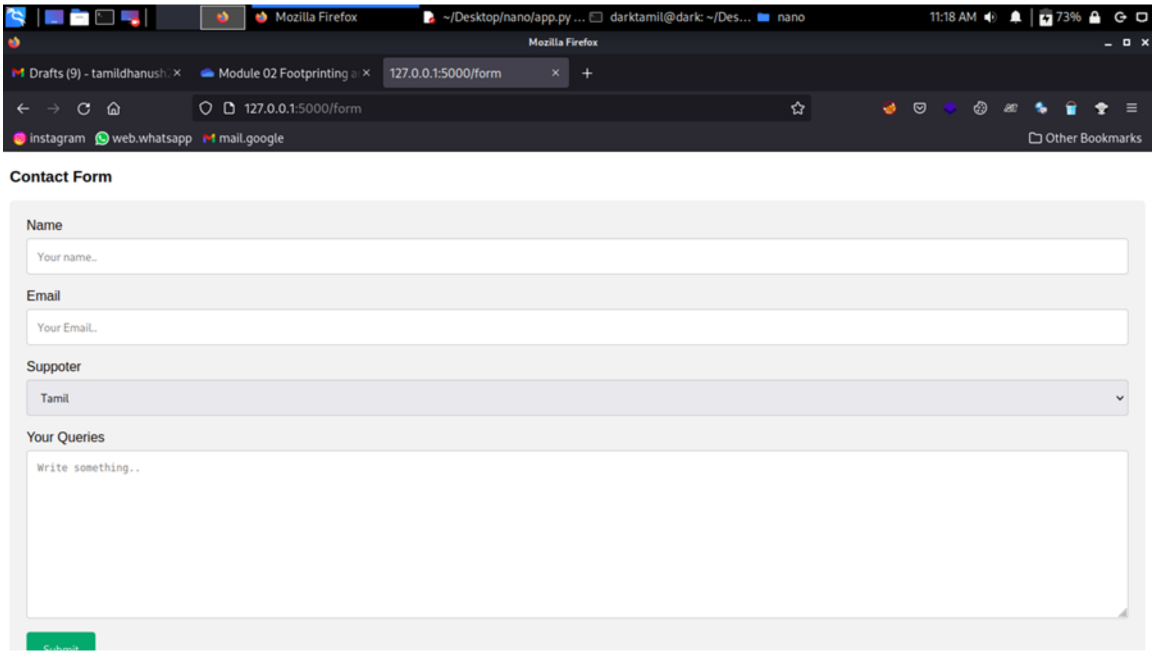
Sign In

Welcome!

Enter your personal details and start your journey with us!

Sign Up

Login form



Contact Form

Name

Your name..

Email

Your Email..

Suppoter

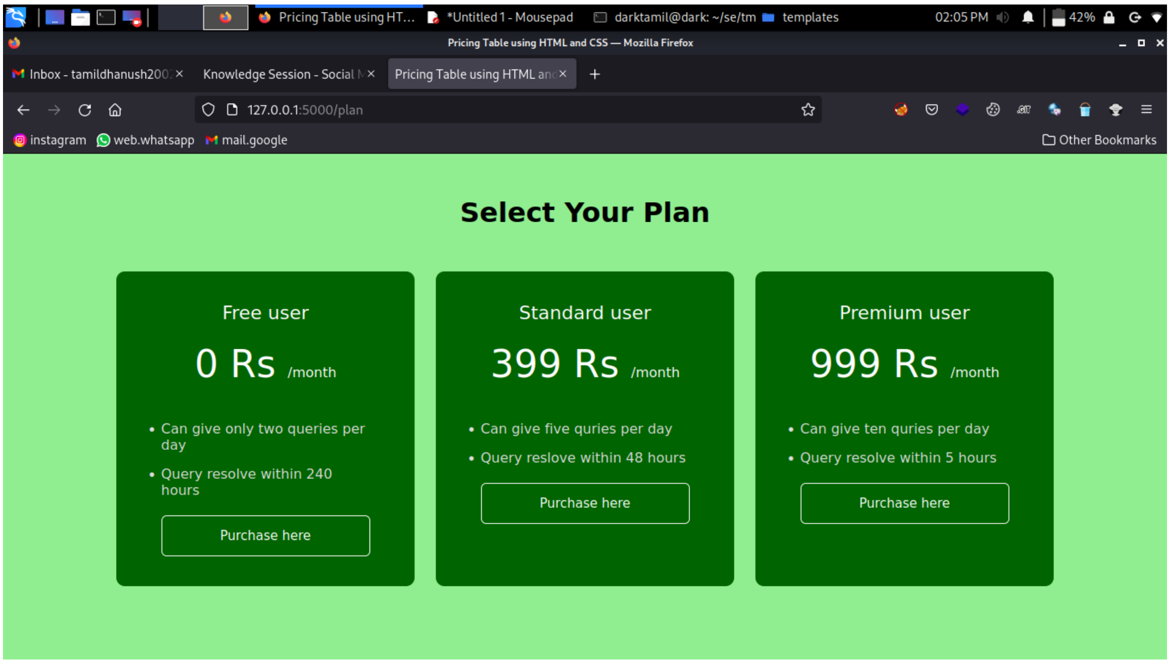
Tamil

Your Queries

Write something..

Submit

Premium package:



Select Your Plan

Free user

0 Rs /month

- Can give only two queries per day
- Query resolve within 240 hours

Purchase here

Standard user

399 Rs /month

- Can give five quries per day
- Query reslove within 48 hours

Purchase here

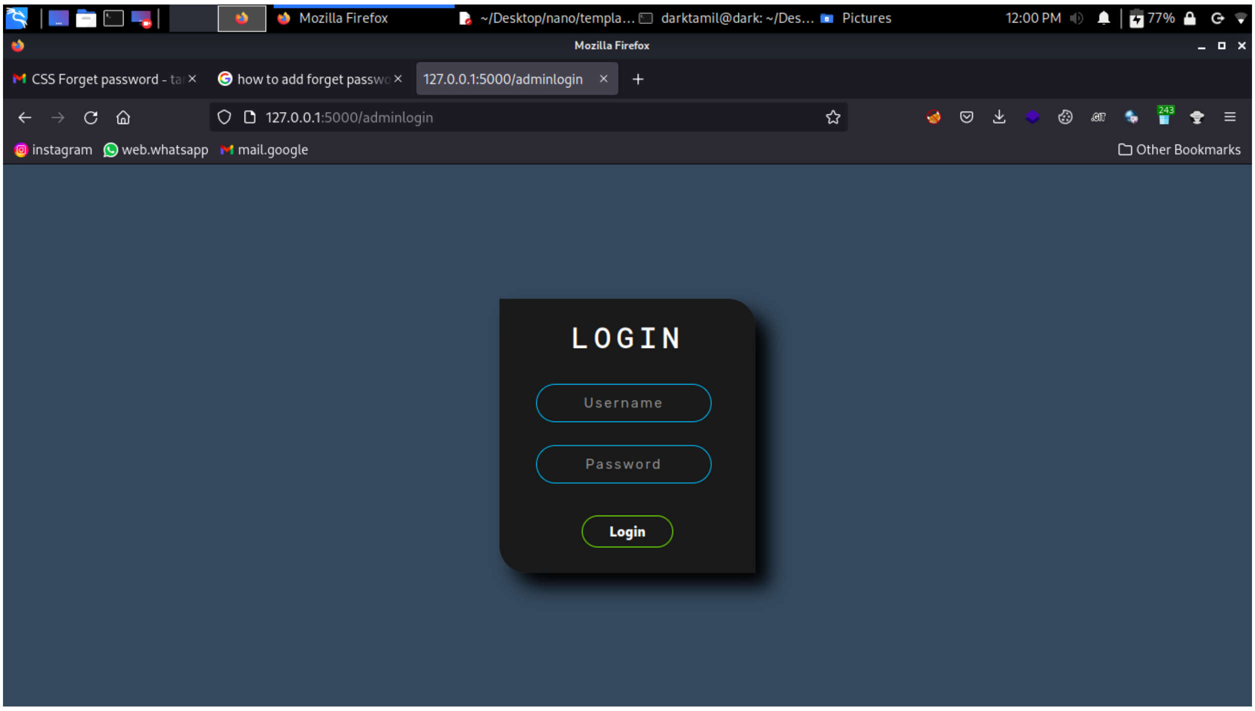
Premium user

999 Rs /month

- Can give ten quries per day
- Query resolve within 5 hours

Purchase here

Admin login page:



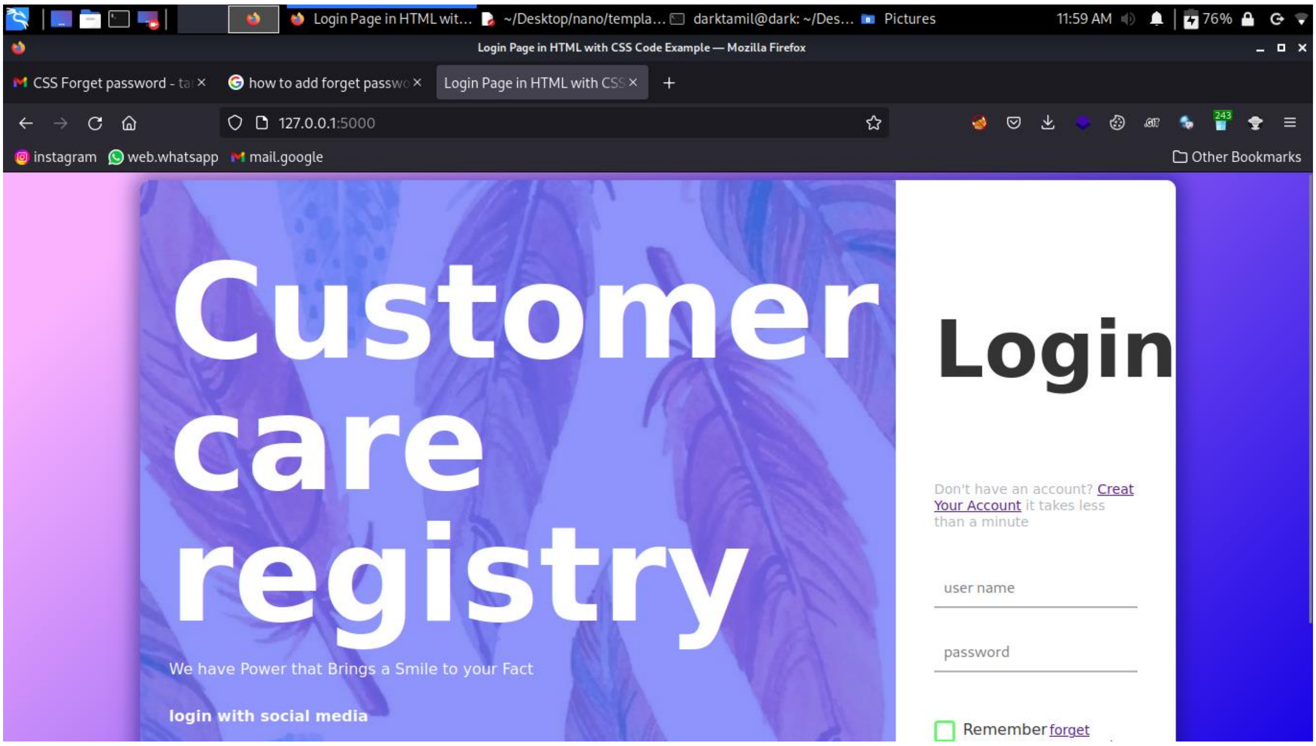
LOGIN

Username

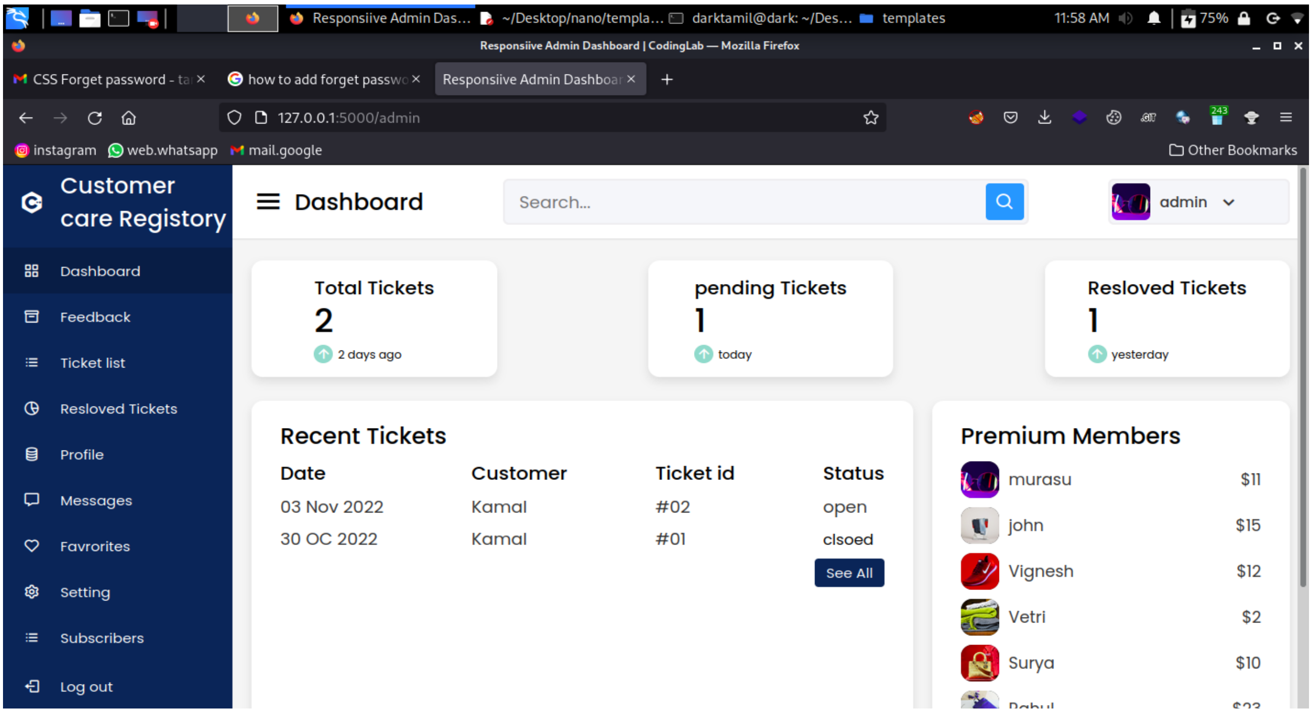
Password

Login

Home page:


Login Page in HTML with CSS Code Example — Mozilla Firefox
127.0.0.1:5000
Customer care registry
We have Power that Brings a Smile to your Fact
login with social media
Login
Don't have an account? Creat Your Account it takes less than a minute
user name
password
□ Remember forget

Admin dashboard:



Total Tickets 2 2 days ago

pending Tickets 1 today

Resloved Tickets 1 yesterday

Recent Tickets

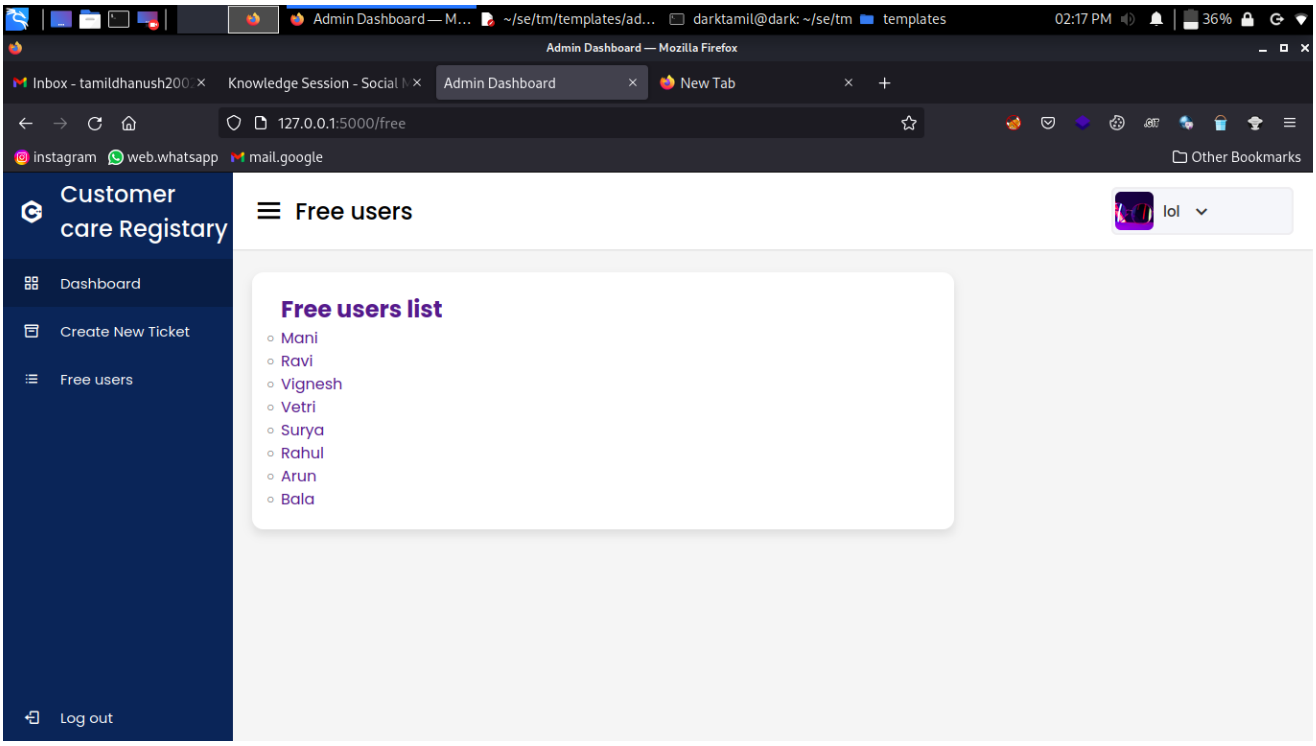
Date	Customer	Ticket id	Status
03 Nov 2022	Kamal	#02	open
30 OC 2022	Kamal	#01	clsoed

See All

Premium Members

murasu	$11
john	$15
Vignesh	$12
Vetri	$2
Surya	$10
Rahul	$23

Admin dashboard for free user:



Customer care Registary

Dashboard

Create New Ticket

Free users

Log out

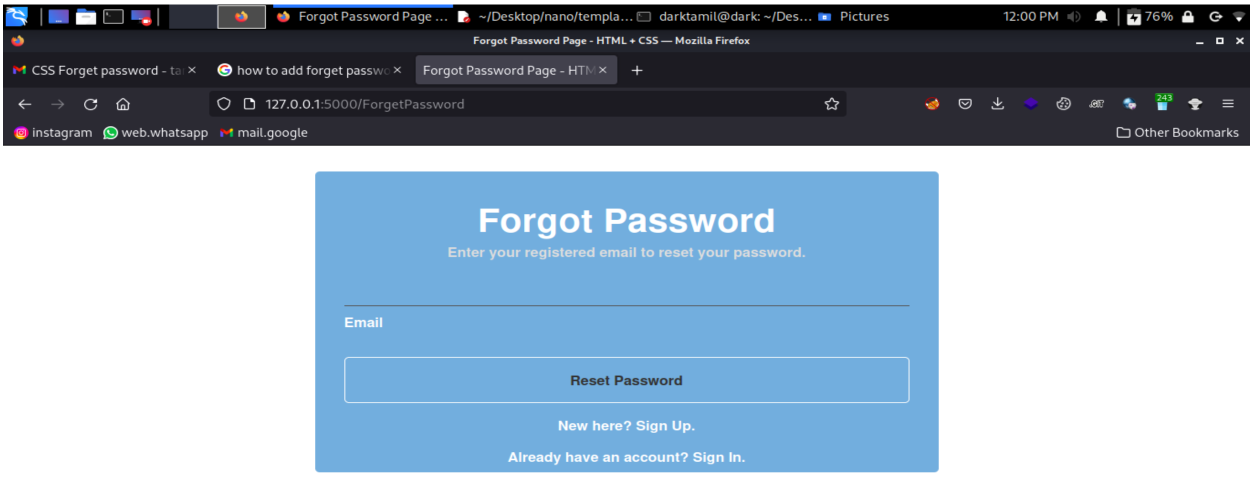
Free users

lol

Free users list

- Mani
- Ravi
- Vignesh
- Vetri
- Surya
- Rahul
- Arun
- Bala

Forgot password:



127.0.0.1:5000/ForgetPassword

Forgot Password

Enter your registered email to reset your password.

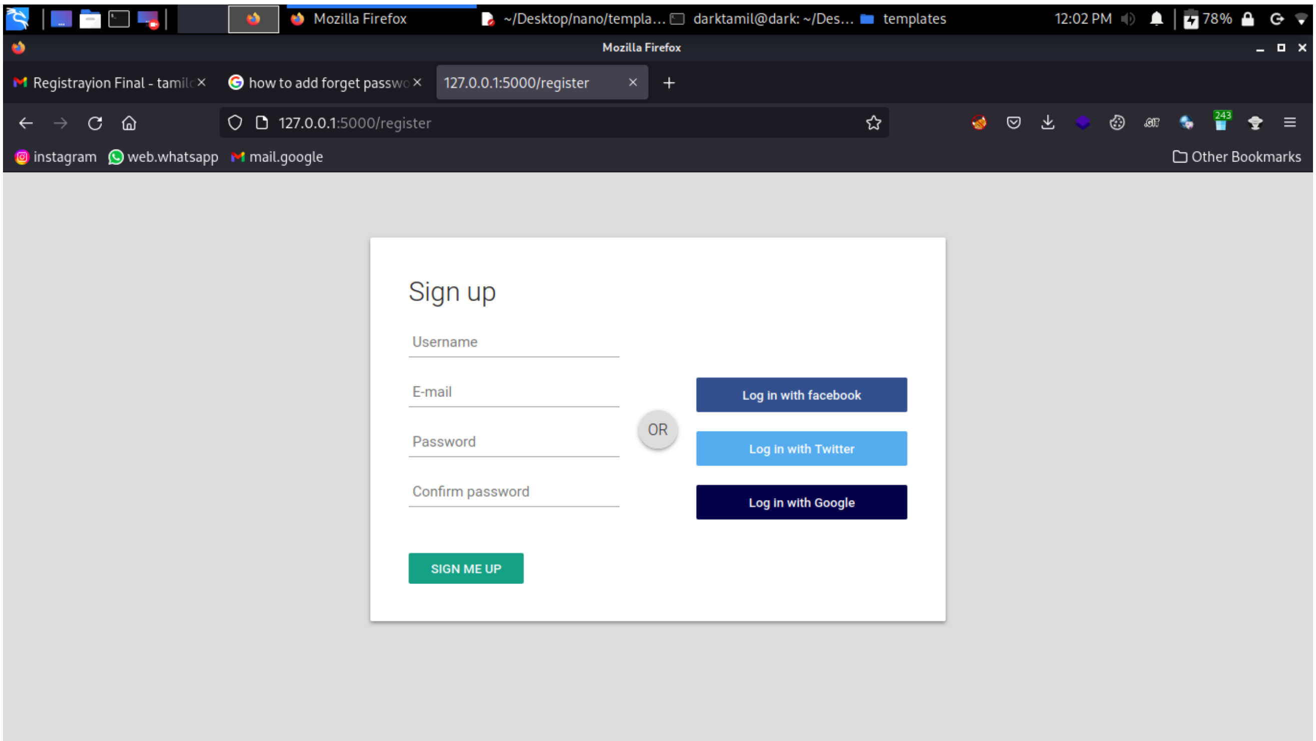
Email

Reset Password

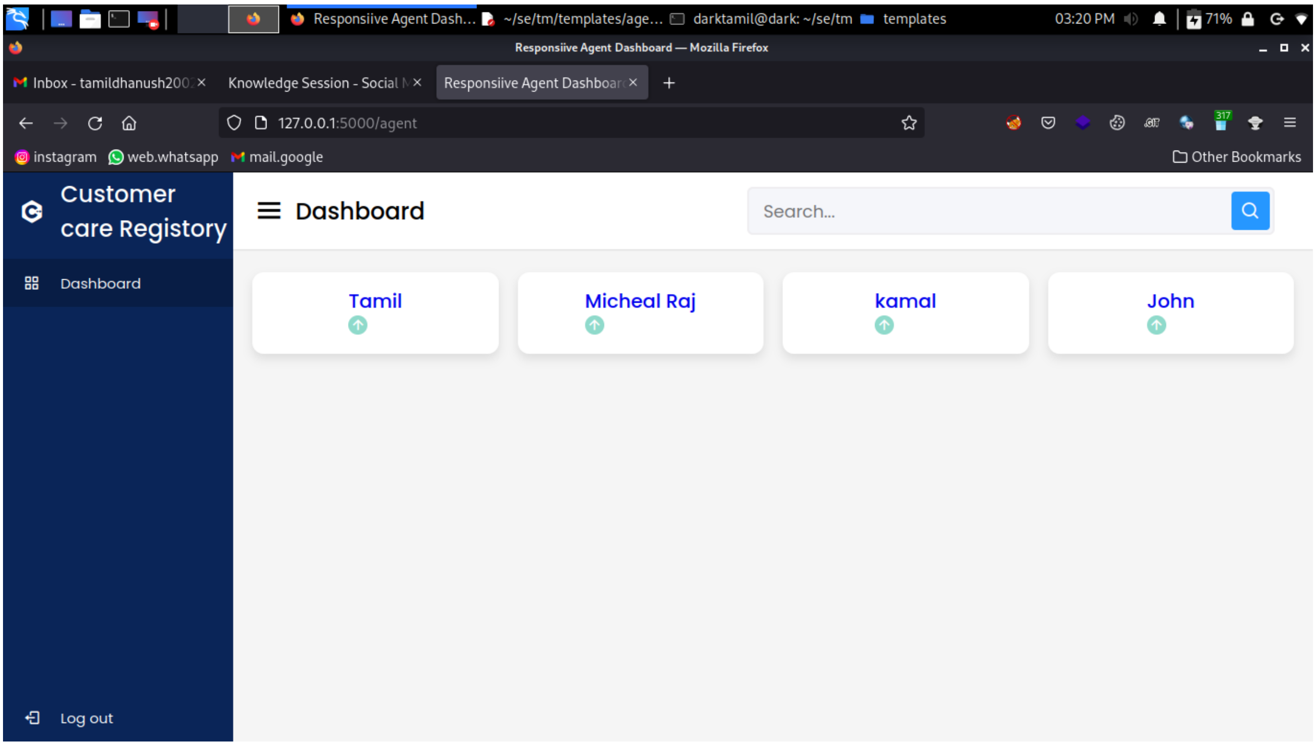
New here? Sign Up.

Already have an account? Sign In.

Registration page:



Backend :



10. ADVANTAGES & DISADVANTAGES

Advantages:

1. Enhances Better Customer Service
2. Customer Care Registry systems provide businesses with numerous strategic advantages. One of such is the capability to add a personal touch to existing relationships between the business and the customers. It is possible to treat each client individually rather than as a group, by maintaining a repository on each customer's profiles. This system allows each employee to understand the specific needs of their customers as well as their transaction file.
3. The organization can occasionally adjust the level of service offered to reflect the importance or status of the customer. Improved responsiveness and understanding among the business employees results in better customer service. This decreases customer agitation and builds on their loyalty to the business. Moreover, the company would benefit more by getting feedback over their products from esteemed customers.
4. The level of customer service offered is the key difference between businesses that lead the charts and those that are surprised with their faulty steps. Customer service efficiency is measured by comparing turnaround time for service issues raised by customers as well as the number of service errors recorded due to misinformation.
5. A good business should always follow – up with customers on the items they buy. This strategy enables a business to rectify possible problems even before they are logged as complaints.

Disadvantages:

1.Only services for which a particular payment has been made are covered under the consumer protection act. However, it does not protect medical professionals, or hospitals, and covers cases when this act does not apply to free medical care.

2. This act does not apply to mandatory services, such as water supply, that are provided by state agencies.

3. Only two clauses related to the supply of hazardous materials are covered by this act. Consumer redress is not given any power by the consumer protection act.

4. The consumer protection act focuses on the supply of ineffective products, but there are no strict regulations for those who produce it.

11. CONCLUSION

The researcher has highlighted how the system works, who are the main users, services and how they can deal with the proposed system. This paper presents an overview of the development and implementation of the Complaint Management System as a web-service based on cloud. The results obtained from the implementation are encouraging and promising for the development or more complex systems in the future as the Complaints Management is a complex and critical problem. Complaints and compliments are valuable source of information that organizations can use to improve program delivery and service. As regulatory and market pressures continue to mount upon companies, industry leaders will need to develop effective solutions or face the high costs inherent in failed technology implementations and weak customer relationships. The preferred alternative is a customer-focused complaints management solution that works. Finally the researcher believes that the presented model can be helpful in other fields of e-complaining in terms of Citizen Adaption and Citizen Loyalty.

12.FUTURE SCOPE

It is quite certain that with great precision the new-gen technology of Customer Care Registry solutions will help in the sales and marketing to a great deal. This will be done while calculating the better results attained by the marketing team.

The progression of CUSTOMER CARE REGISTRY in the future would mostly depend on how faster API's redefines cloud platforms. And it is simply the beginning of API's era of integrating CUSTOMER CARE REGISTRY solution with the business application.

Even though the benefits of CUSTOMER CARE REGISTRY sounds quite interesting while its implementation challenges are quite difficult to overcome. And with the advancement of CUSTOMER CARE REGISTRY technology every year, this trend of implementation barriers will keep rising.

In the near future, CUSTOMER CARE REGISTRY will be mostly analytical & net-based. More trending technologies of CUSTOMER CARE REGISTRY such as data analytics & other matrices will be used to analyze the business performance.

Moreover, more user will be benefited due to the linkage of CUSTOMER CARE REGISTRY along with social media as it will see more popularity in the coming days. Apart from this, the future CUSTOMER CARE REGISTRY will enable its user to exchange data over electronic devices more easily than ever. Not to mention CUSTOMER CARE REGISTRY's immense scope of being integrated with multiple other platforms in the future that will boost its functionalities resulting in great progress and development of company & organizational activities.

13. APPENDIX

Coding :

```
from flask import Flask, render_template, request, redirect, url_for, session

from flask_mail import Mail,Message

import ibm_db

import re

app = Flask(__name__)

app.config['MAIL_SERVER'] = 'smtp.gmail.com'

app.config['MAIL_PORT'] = 465

app.config['MAIL_USERNAME'] = 'tmkssolution1234@gmail.com'

app.config['MAIL_PASSWORD'] = 'xyfxjnripeytrqhk'

app.config['MAIL_USE_TLS'] = False

app.config['MAIL_USE_SSL'] = True

mail=Mail(app)

  app.secret_key = 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-a1f5-
23dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32716;SECURITY=S
SL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=jzc43091;PWD=PI8VtGRvZlSVT65
A",'','')


@app.route('/')

def homer():

    return render_template('home.html')

@app.route('/agent')

def agent():

    return render_template("agent.html")

@app.route('/admin')

def agnt():

    return render_template("admin.html")

@app.route('/free')

def free():

    return render_template("free.html")
```

```
@app.route('/ForgotPassword', methods=['GET','POST'])

def ForgotPassword():

     if request.method=="POST":

        name=request.form['name']

        email=request.form['email']

        message=Message(subject, sender='tmkssolution1234@gmail.com',
recipients=['johncaesar07@gmail.com',
'murasutamil2002@gmail.com','kamaleshwaran1123@gmail.com','hellsprince26@gmail.co
m'])

message.body="""

        Hey Guys we have work.The Client sends a Query,

      Name - {}

      Email - {}

      Thank you

      """.format(name,email)

      mail.send(message)

      return render_template("success.html")

     return render_template("ForgotPassword.html")

@app.route('/login',methods =['GET', 'POST'])

def login():

    global userid

    msg = ''

    if request.method == 'POST' :

        username = request.form['username']

        password = request.form['password']

        sql = "SELECT * FROM users WHERE username =? AND password=?"

        stmt = ibm_db.prepare(conn, sql)

        ibm_db.bind_param(stmt,1,username)

        ibm_db.bind_param(stmt,2,password)

        ibm_db.execute(stmt)

        account = ibm_db.fetch_assoc(stmt)

        print (account)

        if account:
            session['loggedin'] = True
            session['id'] = account['USERNAME']
            userid=  account['USERNAME']
            session['username'] = account['USERNAME']
            msg = 'Logged in successfully !'
            msg = 'Logged in successfully !'
            return render_template('customer.html', msg = msg)
        else:
            msg = 'Incorrect username / password !'
    return render_template('login.html', msg = msg)
@app.route('/register', methods =['GET', 'POST'])
def registet():
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        sql = "SELECT * FROM users WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO  users VALUES (?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, username)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg = msg)
@app.route('/customer')
def dash():
   return render_template('customer.html')
@app.route('/plan')
def plan():
    return render_template('plan.html')

@app.route('/logout')
def logout():
   session.pop('loggedin', None)
   session.pop('id', None)
   session.pop('username', None)
   return render_template('home.html')

@app.route("/form", methods=['GET', 'POST'])
def form():
    if request.method=="POST":
        name=request.form['name']
        email=request.form['email']
        suppoter=request.form['suppoter']
        subject=request.form['subject']
        query=request.form['query']
        message=Message(subject, sender='tmkssolution1234@gmail.com',
recipients=['johncaesar07@gmail.com',
'murasutamil2002@gmail.com','kamaleshwaran1123@gmail.com','hellsprince26@gmail.co
m'])
```

```
message.body="""
        Hey Guys we have work.
The Client sends a Query,
      Name - {}
      Email - {}
      Support - {}
      Query :  {}
      Thank you
      """.format(name,email,suppoter,query)
   mail.send(message)
return render_template("success.html")
    return render_template("form.html")


if __name__ == '__main__':
   app.run(host='0.0.0.0')
```

Git hub link:

https://github.com/IBM-EPBL/IBM-Project-1439-1663911671

Demo link:

https://github.com/IBM-EPBL/IBM-Project-1439-1663911671/tree/main/Final_Deliverables/Project%20Demostration