A PROJECT REPORT ON

CUSTOMER CARE REGISTRY

Submitted by

SABARISHINI.M	(952019104020)
MUTHUPRIYA.V	(952019104014)
KAVIYA PRIYA.M	(952019104010)
KOKILA.M	(952019104011)

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

P.S.R.RENGASAMY COLLEGE OF ENGINEERING

SIVAKASI

ANNA UNIVERSITY::CHENNAI 600025

NOV/DEC 2022

ABSTRACT

An online comprehensive CustomerCare Solution is to manage customer interaction and complaints with the ServiceProviders over phone or throughand e-mail. The system should have capability to integrate with any Service Provider from any domain or industry like Banking, Telecom, Insurance, etc

Customer Servicealso known as Client Serviceis the provision of serviceto customers its significance varies by product, industry and domain. In many cases customer services is more important if the purchaserelates to a service as opposed to a product.

Customer Service may be provided by a Person or Sales & Service Representatives Customer Service is normally an integral part of a company's customer value proposition.

CONTENTS

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION TO PROJECT

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 services is more important if the information relates to a services opposed to a Customer. Customer Service may be provided by a Service Representatives Customer Service is normally an integral part of a company's customer value proposition.

ORGANIZATION PROFILE

SOFTWARE SOLUTION

Software Solutions is an IT solution provider for a dynamic environment where business and technology strategies converge. Their approachfocuses on new ways of business combiningIT innovation and adoption while also leveraging an organization's current IT assets. Their work with large global corporations and new products or services and to implement prudent business and technology strategies in today's environment.

RANGE OF EXPERTISE INCLUDES:

- Software Development Services
- Engineering Services
- Systems Integration
- Customer Relationship Management
- Product Development
- Electronic Commerce
- Consulting
- IT Outsourcing

We apply technology with innovation and responsibility to achieve two broad objectives:

- Effectively address the business issuesour customers face today.

THIS APPROACH RESTS ON:

- A strategy wherewe architect, integrate and manage technology services and solutions - we call itAIM for success.
- A robustoffshore development methodology and reduced demand
- on customerresources.
- A focusonthe use ofreusable frameworks to provide cost and times benefits.

They combinethe best people,processes and technology to achieve excellentresults - consistency. Weoffer customers the advantages of:

SPEED:

They understand the importance of timing, of getting there before the competition. A rich portfolioof reusable, modular frameworks helps jump-start projects. Tried and tested methodology ensures that we follow a predictable, low - risk path to achieve results. Our track record is testimony to complex projects delivered within andevens before schedule.

EXPERTISE:

Our teams combine cutting edge technology skills with rich domain expertise. What's equally important - they share a strong customer orientation that means they actually start by listening to the customer. They're focused on coming up with solutions that serve customer requirements today and anticipate future needs.

A FULL SERVICE PORTFOLIO:

They offer customers the advantage of being able to Architect, integrate and manage technology services. This means that they can rely on one, fully accountable source instead of trying to integrate disparate multi vendor solutions.

SERVICES:

Xxx is providing it’s services to companies which are in the field of production, quality control etcWith their rich expertise and experience and information technology they are in best position to provide software solutionsto distinct business requirements.

1.2 PURPOSEOF THE PROJECT

An online comprehensive Customer Care Solution is to manage customer interaction and complaints with the Service Providers over phone or through and e-mail. The system should have capability to integrate with any Service Provider from any domain or industry like Banking, Telecom, Insurance, etc.

Customer Service also known as Client Service is the provisionof service to customers Its significance varies by product, industryand 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 is normally an integral part of a company’s customer.

CHAPTER 2

LITERATURE SURVEY

2.1 PROBLEMS IN EXISTING SYSTEM

The existing system is a semi-automated at where the information is stored in the form of excel sheets in disk drives. The information sharing to the Volunteers, Group members, etc. is through mailing feature only. The information storage and maintenance is more critical in this system.Tracking the member’s activities and progress of the work is a tedious job here. This system cannot provide the information sharing by 24x7 days.

2.2 REFERENCE

TITLE: Customer Care and Relationship Support Office

AUTHOR NAME: Hubert Baumeister & Piotr Kosi

YEAR: September 2000

DESCRIPTION:

Customer Relationship Management (CRM) is an inherent business strategy for companies big and small. The technology has reached a point where it is truly enabling the way enterprises manage their customer relationships. The goal of the EU funded project CARUSO is the design of a software toolkit that facilitates the building and maintaining of high quality business-to-business and business-to-customer relationships. CARUSO is designed to allow a multi-dimensional way of looking at markets, customers, suppliers, products, personnel, internal and external information, communication and action flow. This will be accomplished by the following core features: front-office application builder with customer care and marketing desk, basic technologies comprising a general communication server, intelligent information, document and contact access, unified messaging, and a customizable user interface. Emphasis will be put on exploiting existing tool packages as much as possible. The CARUSO toolkit is targeted at European Small and Medium Sized Enterprises (SME) and allows them to optimize their business operations to the mutual benefit of both the supplier and the consumer.

TITLE: Customer Care Management Model for Service Industry

AUTHOR NAME: Muthusamy Nataraj , Nallasamy Gunasekaran

YEAR: Received January 12th, 2010; revised February 23rd, 2010; accepted April 11th, 2010.

DESCRIPTION:

This describes a model for Customer care management in an automotive service industry. Design/ methodology/approach – Customer care management (CCM) model is developed using TQM techniques, Quality Function Deployment (QFD) and Six Sigma. The matrix structure in QFD is used to transform customer complaints into Critical-to-Quality (CTQ) parameters. By using Six Sigma DMAIC approach, the customer complaint parameters are analyzed for improvement. Findings – The application of CCM model in an automobile service industry has determined that the workload planning is the chronic problem for customer complaint. Further analysis through this model leads to restructuring of existing workload planning practice through a set of algorithms. Research limitations/implications – CCM model lacks to accommodate the effect of relationship between rectification factors. Also competitor technical contemplation is not possible in this model. Originality/value – Customer is the focal point and early response to their complaint is the key to success of every business. This paper has developed a structured complaint management practice which warrants the timely response to customer complaints and speedy resolution for survival in today's customer driven market

TITLE: BUILDING CARE THROUGH CUSTOMER CARE

AUTHOR NAME: Brian R. Wood

DESCRIPTION:

Building maintenance has long been portrayed as a 'Cinderella' activity (Seeley, 1976), unattractive and often poorly considered; and yet it is a very substantial part of the construction economy- in the UK £28 Billion compared with £10 Billion for new-build (Barbour, 1998). Research by the author over recent years has identified a shift from the 'received wisdom' of Planned Preventive Maintenance (PPM) programmes to more responsive practices using technology to get closer to the customer. This paper integrates work published by the author under titles such as Just In Time Maintenance (1995, 1997), Call-Centred Maintenance (1998) and Intelligent Building Care (1999) to demonstrate how the new approach to building maintenance with a focus on care for the customer and a service culture is evolving.

TITLE: Human Resource Development Systems for Customer Care Services Management in Telecommunications Companie

AUTHOR NAME: Mudiyanselage Saman Dassanayake

YEAR: September 2004

DESCRIPTION:

The telecommunications sector in Sri Lanka has experienced deep and remarkable transformations since the mid-1990s. They include, inter alia, Sri Lanka Telecom Limited (SLT), the state-owned telecommunications monopoly, forming a strategic partnership with Nippon Telegraph and Telephone Corporation (NTT) under its privatization program, Suntel Limited (Suntel), a joint venture between intemational and local partners, commencing commercial operations, and many other new firms penetrating the telecom market. Amid structural and regulatory changes, telecommunications companies, more than other utilities, have dramatized a gradual paradigm shift from price to customer service as the major source of building and sustaining a competitive edge for long-term growth and survival. A central issue in the research presented in this thesis is the configuration of human resource management and human resource development (HRD) systems to support the delivery of customer service, maximizing customer retention

TITLE: CUSTOMER SATISFACTION DETERMINATION AND LEVEL OF COMPLAINT

AUTHOR NAME: Yusuf Indra Wibowo

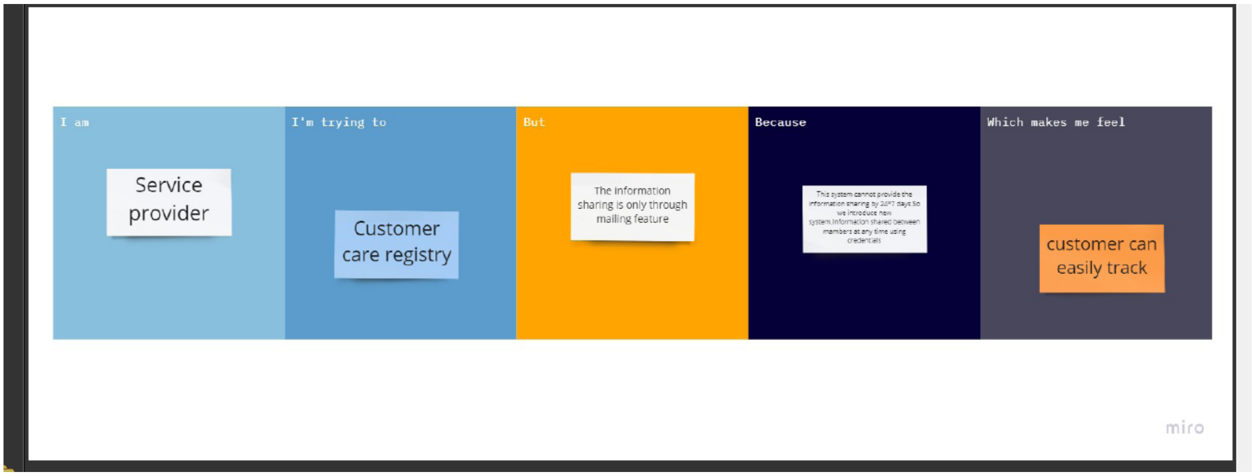
DESCRIPTION:

Previous research or relevant research is very important in a scientific research or article. Previous research or relevant research serves to strengthen the theory and influence of relationships or influences between variables. Article ini review customer satisfaction determination and complaint level: Product Quality and Service Quality Analysis, A Study of Marketing Management Literature. The purpose of writing this article is to build a hypothesis of influence between variables to be used in future research. The result of this risearch library is that: 1) Product Quality affects Customer Satisfaction; 2) Service Quality affects Customer Satisfaction; 3) Product Quality affects complaint level; 4) Service Quality affects complaint level; and 5) Customer Satisfaction affects complaint level.

2.3 PROBLEM STATEMENT DEFINITION

The development of this new system objective is to provide the solution to the problems of existing system. By using this new system, we can fully automate the entire process of the current system. The new system would like to make as web-enabled so that the information can be shared between the members at any time using the respective credentials. To track the status of an individual process, the status update can be centralized using the new system. Being a web-enabled system, the process can be accessed across the world over net.

This system also providing the features like Chatting, Mailing between the members; Images Upload – Download via the web site; updating the process status in centralized location; generated reports can also be exporting to the applications like MS-Excel, PDF format, etc. In this new system, the members like Donors can give their valuable feedback to the Volunteers so that the Volunteers can check their progress of the tasks.The entire process categorized as different modules like Admin module, Volunteermodule, etc. at where we can classify the functionality as an individual process.Using the new systementering into Admin module we can perform…. In this new system using the Volunteermodule we can do….



I am	I'm trying to	But	Because	Which makes me feel
Service provider	Customer care registry	The information sharing is only through mailing feature	This system cannot provide the information sharing by 24*7 days.So we introduce new system.Information shared between mambers at any time using credentials	customer can easily track

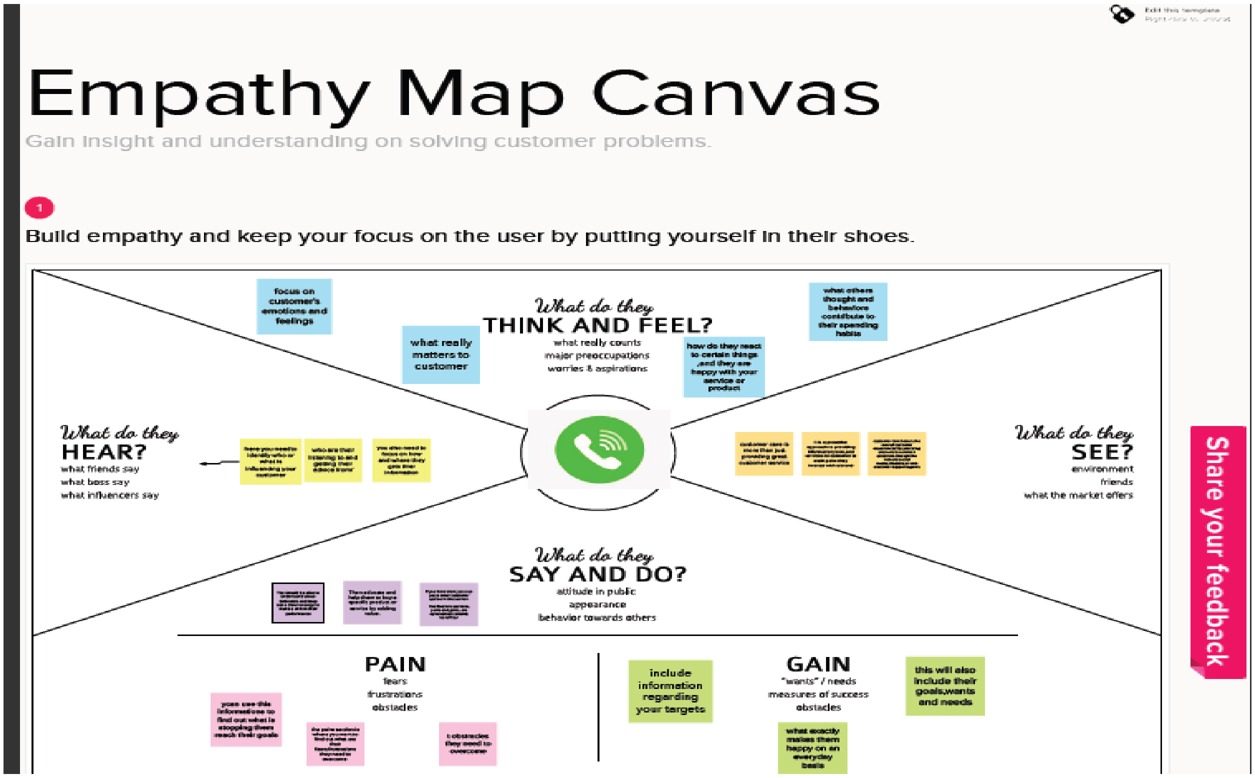
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



Empathy Map Canvas

Gain insight and understanding on solving customer problems.

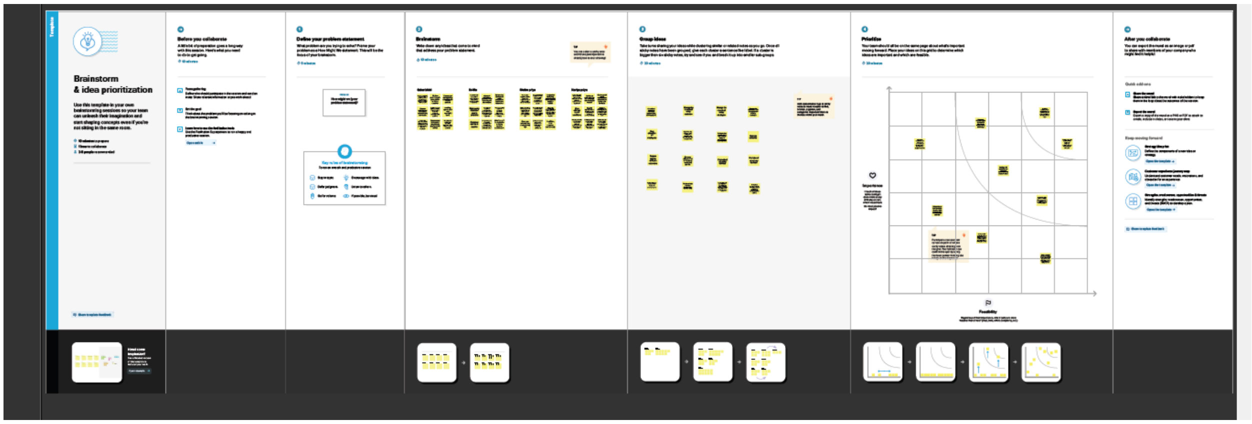
1

Build empathy and keep your focus on the user by putting yourself in their shoes.

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
focus on customer's emotions and feelings
what really matters to customer
how do they react to certain things ,and they are happy with your service or product
what others thought and behaviors contribute to their spending habits
What do they HEAR?
what friends say
what boss say
what influencers say
here you need to identify who or what is influencing your customer
who are their listening to and getting their advice from
you also need to focus on how and where they gets their information
What do they SEE?
environment
friends
what the market offers
customer care is more than just providing great customer service
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
PAIN
fears
frustrations
obstacles
yoan use this informations to find out what is stopping them reach their goals
1 obstacles they need to overcome
GAIN
"wants" / needs
measures of success
obstacles
include information regarding your targets
this will also include their goals,wants and needs
what exactly makes them happy on an everyday basis
Share your feedback

3.2 IDEATION AND 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.



Template

Brainstorm
& idea prioritization

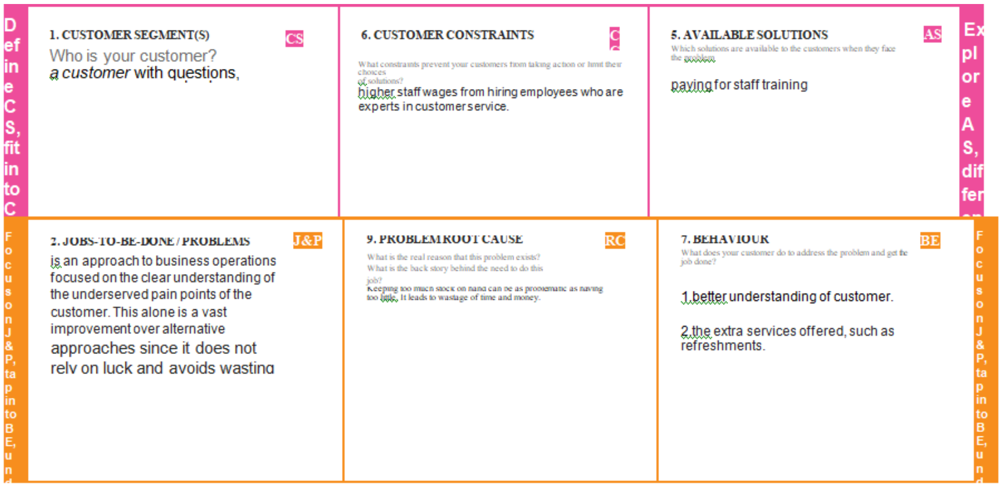
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

3.3 PROPOSED SOLUTIONS

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	A customer problem statement **outlines problems that your customers face**. It helps you figure out how your product or service will solve this problem for them. The statement helps you understand the experience you want to offer your customers
2.	Idea / Solution description	Know Your Customer and How to Solve Their Problem Foster a Customer-Centric Culture Assign One Customer Service Representative to a Single Customer / Account

3.	Novelty / Uniqueness	Improve Customer Service with Chat bots Offer a **Self service** Channel
4.	Social Impact / Customer Satisfaction	Recognize that the behavior and expectations of customers has changed and that this has implications for customer service
5.	Business Model (Revenue Model)	**Fixed package**: Whenever clients hire you to provide customer service for their business, they have to pay a fixed package cost. The amount is billed either annually, semi-annually or quarterly. **Customised packages**: Here, you set the costs for each client based on the services they require and the volume of customer queries/complaints you handle. **Commission-based**: In this model, your revenue is based on how many customer conversions you drive for the client.
6.	Scalability of the Solution	Customer support services should be able to grow and adapt to meet the changing demands of customers. As the climate for your industry changes, your services will need to change accordingly to ensure that consumers are always being served efficiently. **Scalable customer support** means ramping up or scaling back the efforts needed to take care of all customers, and to maintain a loyal base.

3.4 PROBLEM SOLUTION FIT



Define CS, fit into C

1. CUSTOMER SEGMENT(S) CS

Who is your customer?

a customer with questions,

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions?

higher staff wages from hiring employees who are experts in customer service.

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem

paying for staff training

Explore AS, differen

Focus on J&P, tap into BE, und

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

is an approach to business operations focused on the clear understanding of the underserved pain points of the customer. This alone is a vast improvement over alternative approaches since it does not relv on luck and avoids wastina

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?

Keeping too much stock on hand can be as problematic as having too little. It leads to wastage of time and money.

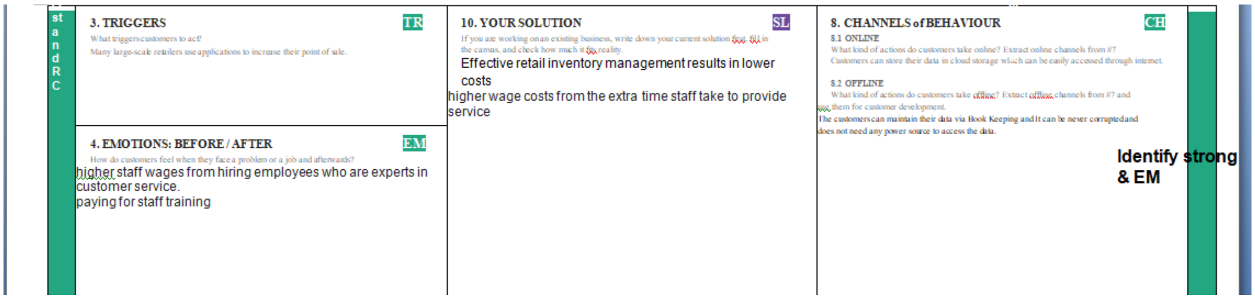
7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?

1.better understanding of customer.

2.the extra services offered, such as refreshments.

Focus on J&P, tap into BE, und



st a n d R C

3. TRIGGERS TR

What triggers customers to act?

Many large-scale retailers use applications to increase their point of sale.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?

higher staff wages from hiring employees who are experts in customer service.
paying for staff training

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.

Effective retail inventory management results in lower costs
higher wage costs from the extra time staff take to provide service

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

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

Customers can store their data in cloud storage which can be easily accessed through internet.

8.2 OFFLINE

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

The customers can maintain their data via Book Keeping and It can be never corrupted and does not need any power source to access the data.

Identify strong & EM

CHAPTER 4
REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMNETS

FR No.	Functional Requirement (Epic)	Sub Requirement (Story/ Sub-Task)
1	User Registration	Registration through Form Registration through Gmail Registration throughLinkedIN
2	User Confirmation	Confirmation via Email Confirmation viaOTP
3	Defining problem	Type what is the problem.
4	Allocating agents	According to the problem agent will be allocated.
5	Analysing problem	Problem and its requirements are analysed by theagents.
6	Tracking problem solution	Track whatis the condition of the problem solutionthrough credentials.
7	Solving problem	Agents solvethe problem and inform to userthrough mail.
8	Customer feedback	User can sendfeedback through credentials.

4.2 NON FUNCTIONAL REQUIREMENTS

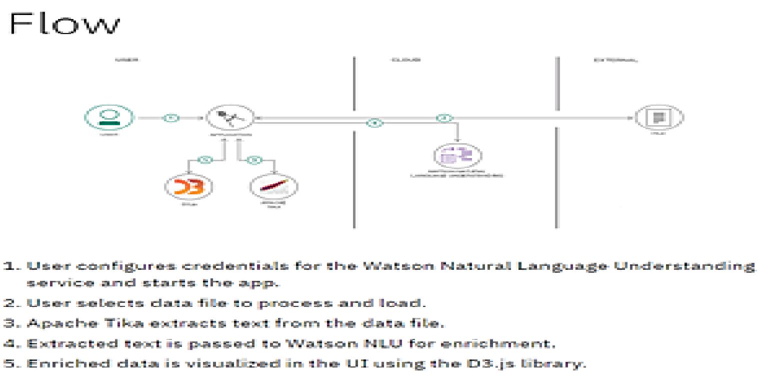
FR No.	Non-Functional Requirement	Description
1	**Usability**	The error rate of users submitting their problemdetails at the ticketmustn’t exceed 10 percent.

2	**Security**	Assures All the data inside the system or in the part will be protected against the malwareattackor unauthorized access.
3	**Reliability**	The system must perform withoutfailure in 95percent of use cases during a month.
4	**Performance**	The landing page supporting 3,000 users per hour must provide 5 second or less response time in a Chrome desktopbrowser, including therendering of text and images.
5	**Availability**	This must be available to US users99.98 percentof the time every month during business hours IST.
6	**Scalability**	The system must be scalable enoughto support1,00,000 visits at the same time while maintaining optimal performance.

CHAPTER 5
PROJECT DESIGN

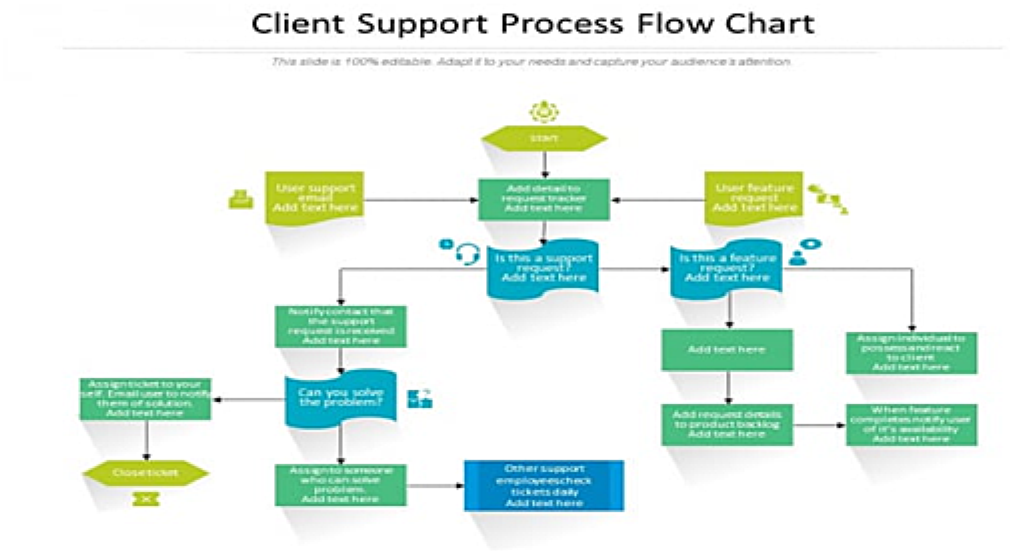
5.1 DATAFLOW DIAGRAMS

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.



Flow

1. User configures credentials for the Watson Natural Language Understanding service and starts the app.
2. User selects data file to process and load.
3. Apache Tika extracts text from the data file.
4. Extracted text is passed to Watson NLU for enrichment.
5. Enriched data is visualized in the UI using the D3.js library.



Client Support Process Flow Chart

This slide is 100% editable. Adapt it to your needs and capture your audience's attention

Start
User support email Add text here
Add detail to request tracker Add text here
User feature request Add text here
Is this a support request? Add text here
Is this a feature request? Add text here
Notify contact that the support request is received Add text here
Add text here
Assign individual to possess and react to client Add text here
Assign ticket to your self. Email user to notify them of solution. Add text here
Can you solve the problem?
Add request details to product backlog Add text here
When feature completes notify user of it's availability Add text here
Close ticket
Assign to someone who can solve problem. Add text here
Other support employees check tickets daily Add text here

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)
6. Hire the Right Employees.
7. Set Goals for Customer Service.
8. Train on Service Skills.
9. Hold People Accountable.
10. Reward and Recognize Good Service

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Strage	File storage requirements	IBM Block Storage or Other Storage Service

8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9.	External API-2	Purpose of External API used in the application	Aadhar API, etc.
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Technology used
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc.	Technology used

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile 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-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard					

CHAPTER 6
PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

ProductBacklog,Sprint Schedule,andEstimation

Sprint	Functional Requirement (Epic)	user story Number	UserStory/Task	Story Points	Priority	Team Members
Sprint-1	UserPanel	USN-1	The user will go in into the website and go through the Services available on the webpage	20	High	SABARISHNI .M KOKILA.M
Sprint-2	Adminpanel	USN-2	The role of the admin is to check out the database about the availability and have a track of all the things that the users are going to service	20	High	MUTHU PRIYA.V KAVIYA PRIYA.M
Sprint-3	ChatBot	USN-3	The user can directly talk to Chatbot regarding the services. Get the recommendations based on information provided by the user.	20	High	SABARISHNI .M KOKILA.M
Sprint-4	finaldelivery	USN-4	Container of applications using docker kubernetes and deployment the application. Create the documentation and final submit the application	20	High	MUTHU PRIYA.V KAVIYA PRIYA.M

6.2 PROJECT PLANNING

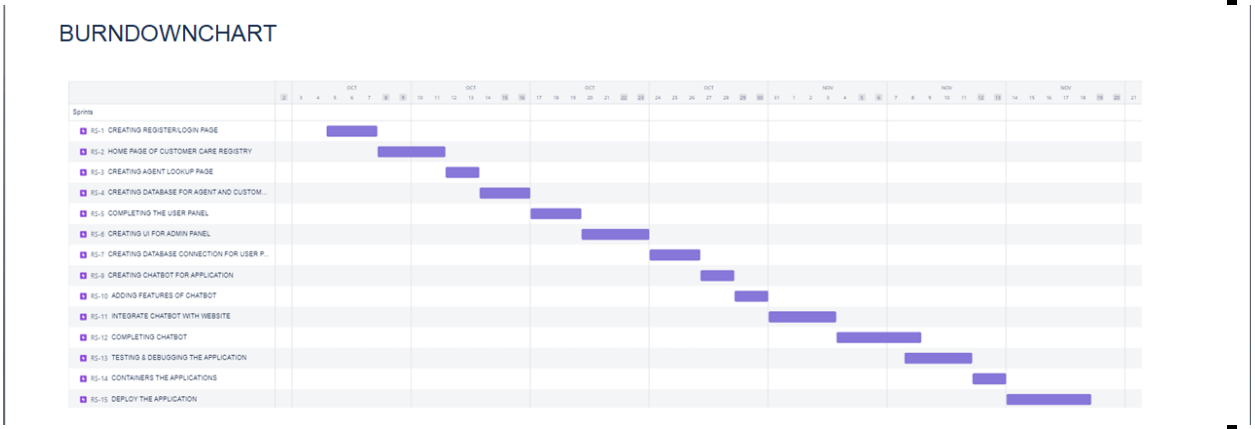
Sprint	Total story points	Duration	Sprint start date	Sprint end date	Story point completion	Sprint release date
Sprint 1	20	6Days	24Oct2022	29Oct2022		29Oct2022
Sprint 2	20	6Days	31Oct2022	05Nov2022		05Nov2022
Sprint 3	20	6Days	07Nov2022	12Nov2022		12Nov2022
Sprint 4	20	6Days	14Nov2022	19Nov2022		19Nov2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20(points per sprint). Let’s calculate the team’s average velocity(AV) periteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

BURNDOWNCHART

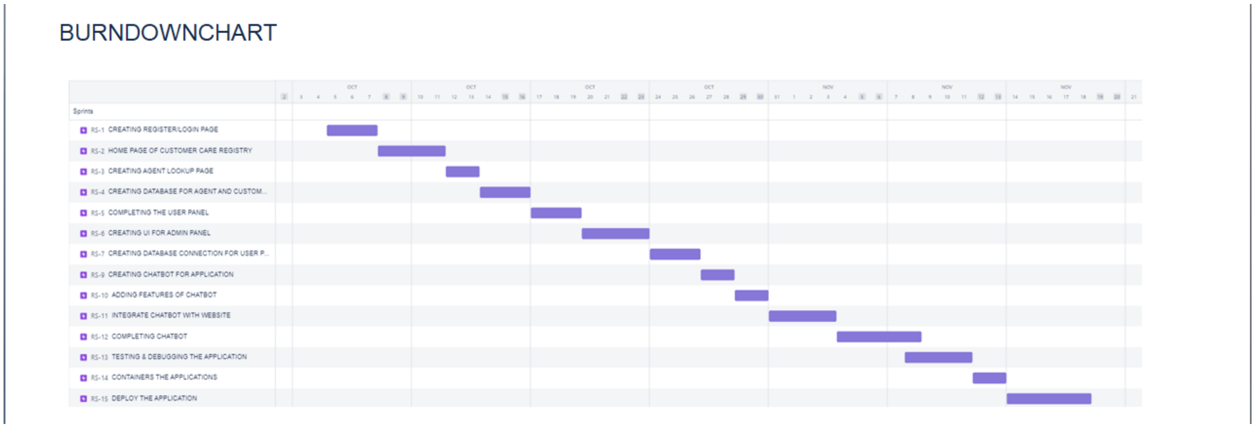


6.3 SPRINT DELIVEYRY SCHEDULE

TITLE	DESCRIPTION	DATE
Literature Survey &InformationGathering	Literature survey on theselected project & gatheringinformation by referring the,technical papers,researchpublicationsetc.	28SEPTEMBER2022
PrepareEmpathyMap	Prepare Empathy Map Canvasto capture the user Pains &Gains, Prepare list of problemstatements	24SEPTEMBER2022
Ideation	List the by organizing thebrainstormingsessionandprioritize the top 3 ideasbased on the feasibility &importance.	25SEPTEMBER2022
ProposedSolution	Preparetheproposedsolutiondocument, which includes thenovelty, feasibility of idea,business model, social impact,scalabilityofsolution,etc.	23SEPTEMBER2022
ProblemSolutionFit	Prepare problem - solution fitdocument.	30SEPTEMBER2022
SolutionArchitecture	Prepare solution architecturedocument.	28SEPTEMBER2022
CustomerJourney	Prepare the customer journeymaps to understand the userinteractions & experienceswith the application (entry toexit).	20OCTOBER2022

FunctionalRequirement	Prepare the functionalrequirementdocument.	8OCTOBER2022
DataFlowDiagrams	Draw the data flowdiagrams and submit forreview.	9OCTOBER2022
TechnologyArchitecture	Prepare thetechnology architecturediagram.	10OCTOBER2022
Prepare Milestone & ActivityList	Preparethemilestones&activitylistoftheproject.	22OCTOBER2022
Project Development - Deliveryof Sprint-1,2,3&4	Develop & submit thedevelopedcodebytestingit.	INPROGRESS..

6.4 REPORTS FROM JIRAA


BURNDOWNCHART
Sprints
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
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

CHAPTER 7
CODING & SOLUTIONING

7.1 FEATURE 1
CUSTOMER LOGIN PAGE

The login page allows a user to gain access to an application by entering their username and password or by authenticating using a social media login.

A user navigates to an application and is presented with a login page as a way to gain access to the application. There are two possible results:

- Authentication is successful and the user is directed to the application landing page.
- Authentication fails and the user remains on the login page. If authentication fails, the screen should show an informational or error message about the failure.

A user is automatically logged out due to inactivity. In this event, they will be returned to the login page, which will display an informational message explaining what happened. Once the user logs in again, they should be taken back to the page they were previously on before being timed out. Thirty minutes is the suggested duration before a session timeout, but this is subject to change based on your product's security requirements.

CUSTOMER REGISTER PAGE

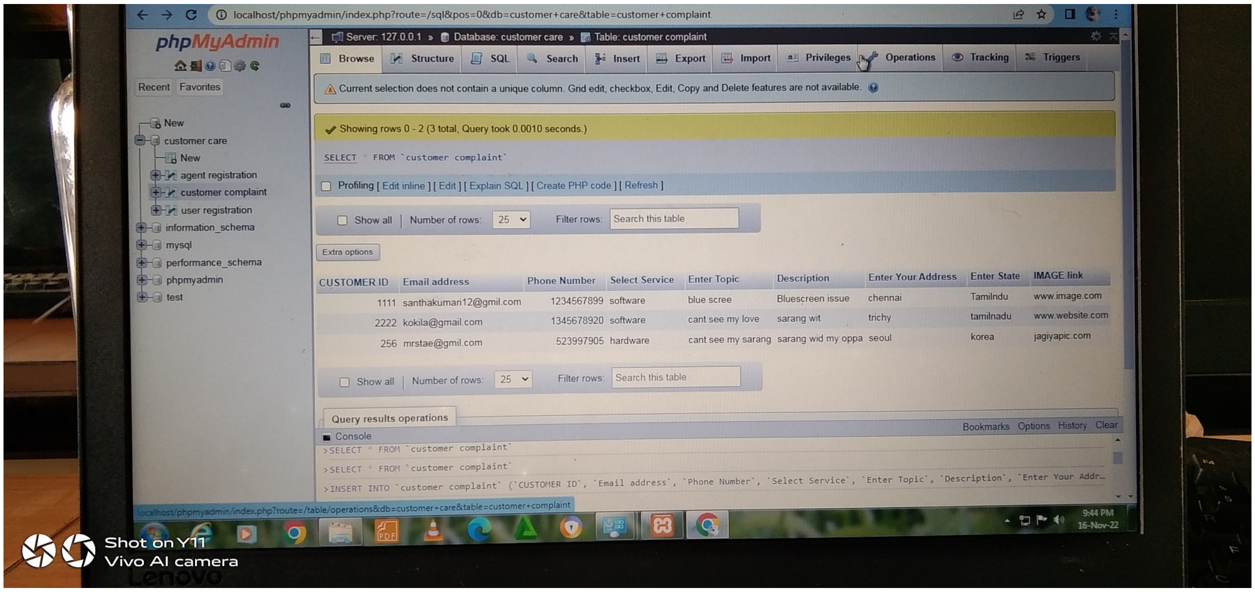
All customers that have created online account need to provide customer registration information, which is used to capture customer profile as well as generate and issue commercial registration certificate. After logging-in to the system for the first time, customers are provided with a wizard-like interface that allows them to provide information required for capturing customer profile and generating commercial registration certificate.

7.2 FEATURE 2

DASHBOARD

A customer dashboard allows the business to evaluate any number of metrics about their customers. They can also look at the metrics collected over time to see how the business is doing. Additionally, a customer dashboard allows a business to test different hypotheses about the structure of their sales, marketing and customer service operations. The data they collect can help them identify what works well for their business and identify areas of improvement. It can also help them make decisions about how various elements are operating within their company.

7.3 DATABASE SCHEMA


localhost/phpmyadmin/index.php?route=/sql&pos=0&db=customer+care&table=customer+complaint
phpMyAdmin
Recent Favorites
New
customer care
New
agent registration
customer complaint
user registration
information_schema
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: customer care » Table: customer complaint
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
Extra options
Query results operations
Console
Bookmarks Options History Clear
9:44 PM
16-Nov-22
Shot on Y11
Vivo AI camera
Lenovo

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

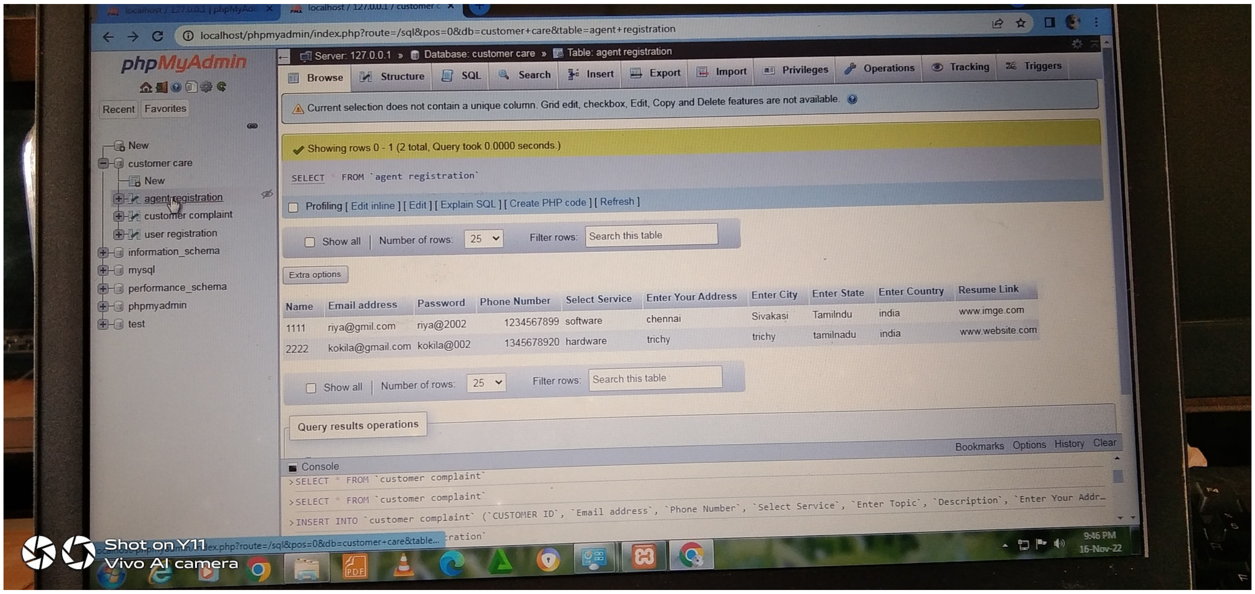
✓ Showing rows 0 - 2 (3 total, Query took 0.0010 seconds.)

```
SELECT * FROM `customer complaint`
```

CUSTOMER ID	Email address	Phone Number	Select Service	Enter Topic	Description	Enter Your Address	Enter State	IMAGE link
1111	santhakumari12@gmil.com	1234567899	software	blue scree	Bluescreen issue	chennai	Tamilndu	www.image.com
2222	kokila@gmail.com	1345678920	software	cant see my love	sarang wit	trichy	tamilnadu	www.website.com
256	mrstae@gmil.com	523997905	hardware	cant see my sarang	sarang wid my oppa	seoul	korea	jagiyapic.com

```
>SELECT * FROM `customer complaint`
>SELECT * FROM `customer complaint`
>INSERT INTO `customer complaint` (`CUSTOMER ID`, `Email address`, `Phone Number`, `Select Service`, `Enter Topic`, `Description`, `Enter Your Addr…
```

localhost/phpmyadmin/index.php?route=/table/operations&db=customer+care&table=customer+complaint



localhost/phpmyadmin/index.php?route=/sql&pos=0&db=customer+care&table=agent+registration

Server: 127.0.0.1 » Database: customer care » Table: agent registration

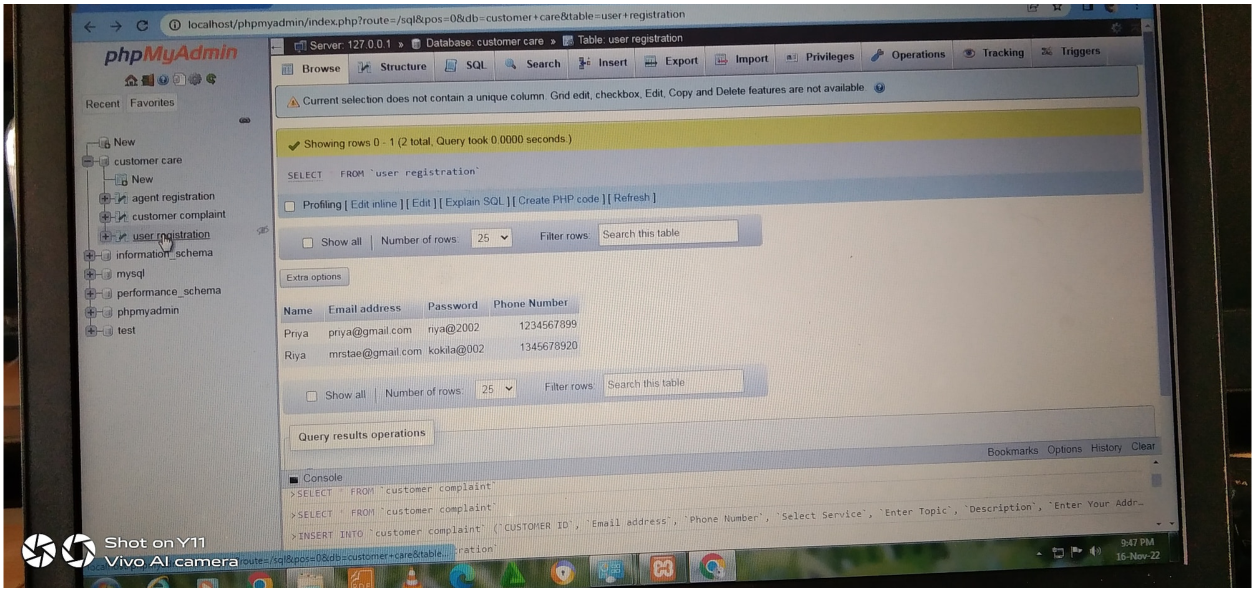
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 1 (2 total, Query took 0.0000 seconds.)

```
SELECT * FROM `agent registration`
```

Name	Email address	Password	Phone Number	Select Service	Enter Your Address	Enter City	Enter State	Enter Country	Resume Link
1111	riya@gmil.com	riya@2002	1234567899	software	chennai	Sivakasi	Tamilndu	india	www.imge.com
2222	kokila@gmail.com	kokila@002	1345678920	hardware	trichy	trichy	tamilnadu	india	www.website.com

```
>SELECT * FROM `customer complaint`
>SELECT * FROM `customer complaint`
>INSERT INTO `customer complaint` (`CUSTOMER ID`, `Email address`, `Phone Number`, `Select Service`, `Enter Topic`, `Description`, `Enter Your Addr…
```



localhost/phpmyadmin/index.php?route=/sql&pos=0&db=customer+care&table=user+registration

Server: 127.0.0.1 » Database: customer care » Table: user registration

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 1 (2 total, Query took 0.0000 seconds.)

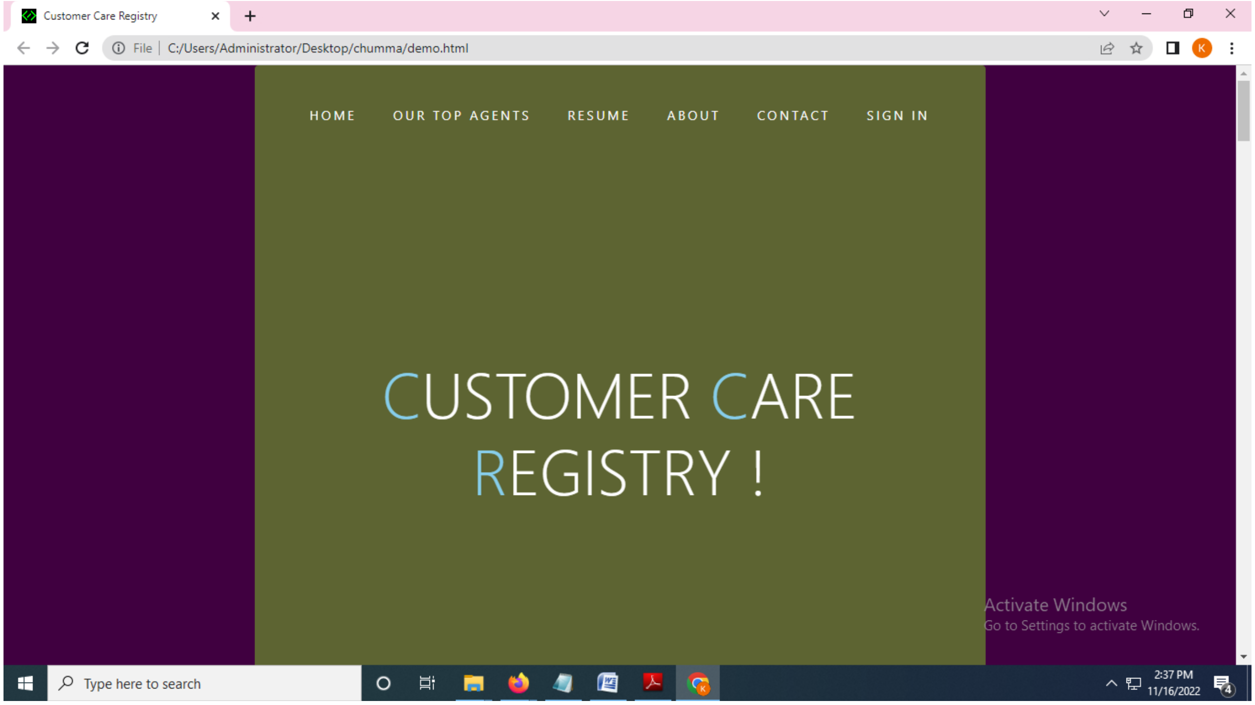
```
SELECT * FROM `user registration`
```

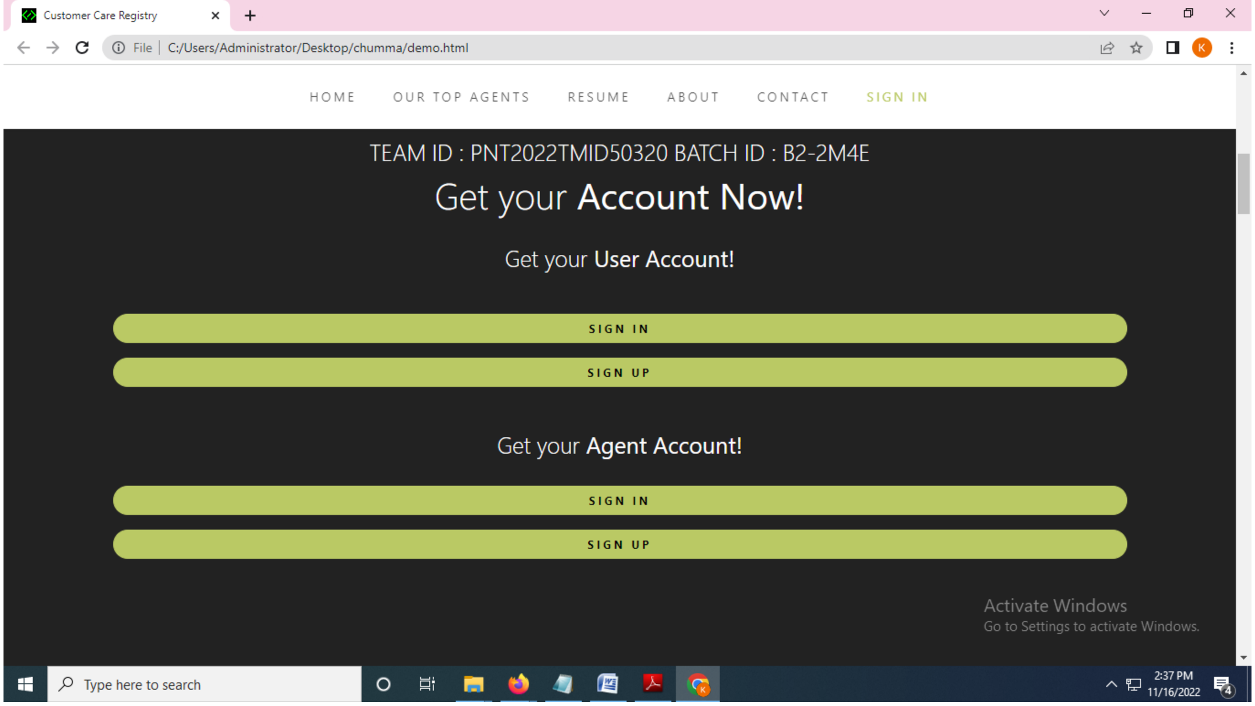
Name	Email address	Password	Phone Number
Priya	priya@gmail.com	riya@2002	1234567899
Riya	mrstae@gmail.com	kokila@002	1345678920

```
>SELECT * FROM `customer complaint`
>SELECT * FROM `customer complaint`
>INSERT INTO `customer complaint` (`CUSTOMER ID`, `Email address`, `Phone Number`, `Select Service`, `Enter Topic`, `Description`, `Enter Your Addr…
```

CHAPTER 8
TESTING

8.1 TEST CASES



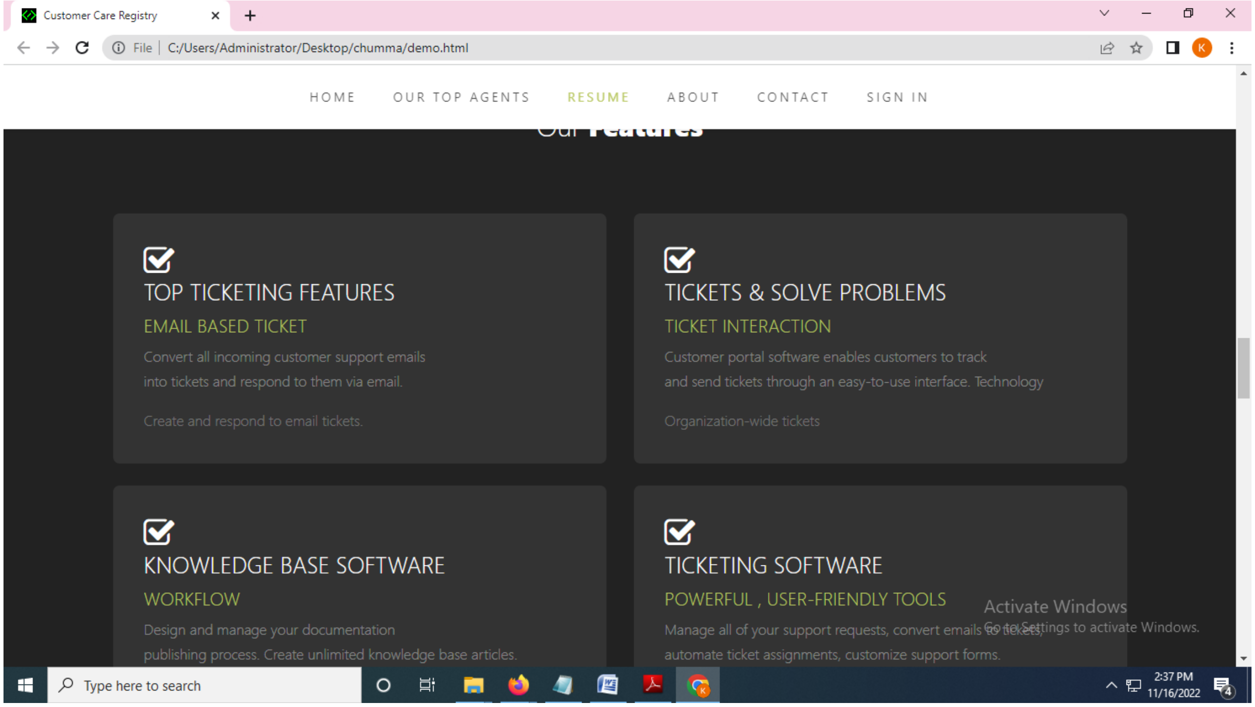


TEAM ID : PNT2022TMID50320 BATCH ID : B2-2M4E

Get your Account Now!

Get your User Account!

Get your Agent Account!



TOP TICKETING FEATURES

EMAIL BASED TICKET

Convert all incoming customer support emails into tickets and respond to them via email.

Create and respond to email tickets.

TICKETS & SOLVE PROBLEMS

TICKET INTERACTION

Customer portal software enables customers to track and send tickets through an easy-to-use interface. Technology

Organization-wide tickets

KNOWLEDGE BASE SOFTWARE

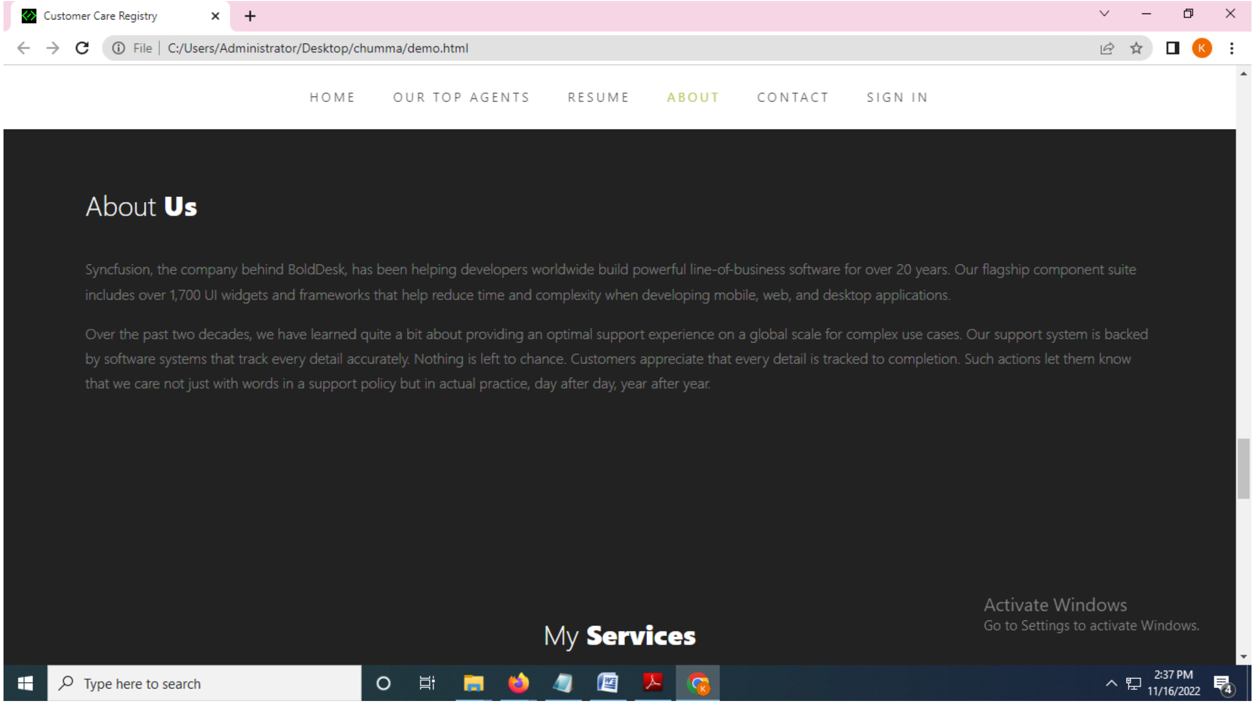
WORKFLOW

Design and manage your documentation publishing process. Create unlimited knowledge base articles.

TICKETING SOFTWARE

POWERFUL , USER-FRIENDLY TOOLS

Manage all of your support requests, convert emails to tickets, automate ticket assignments, customize support forms.

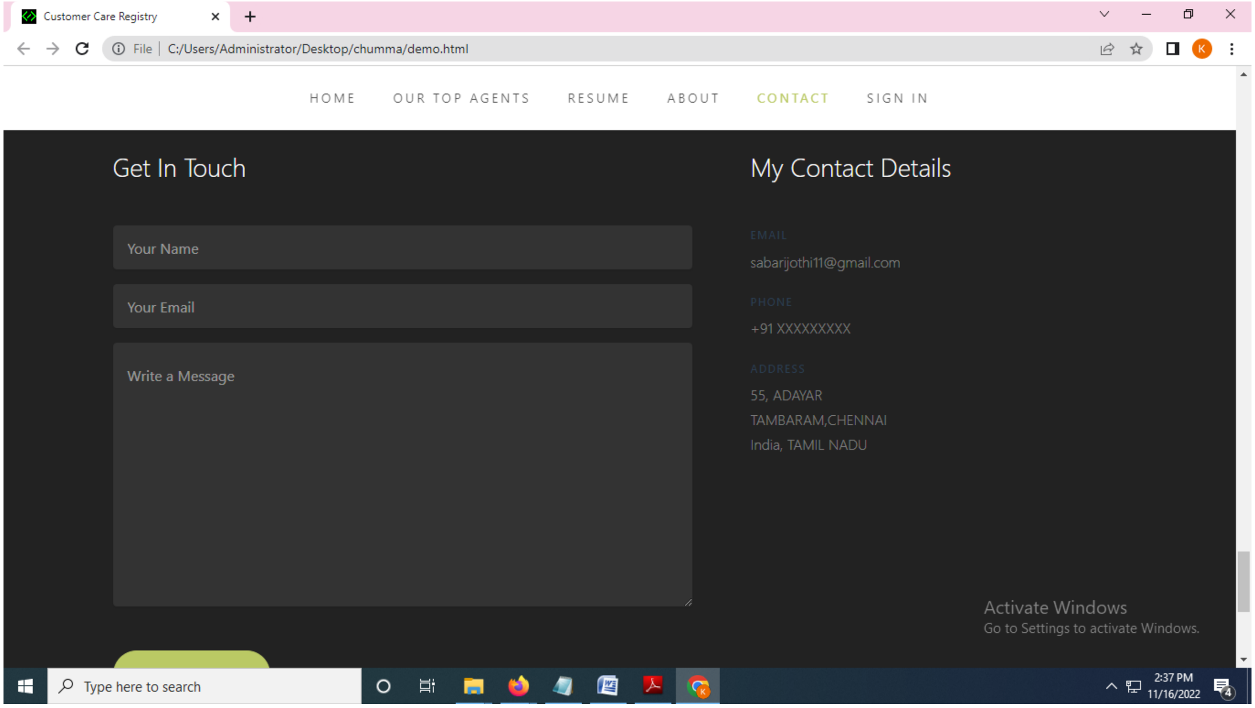


About **Us**

Syncfusion, the company behind BoldDesk, has been helping developers worldwide build powerful line-of-business software for over 20 years. Our flagship component suite includes over 1,700 UI widgets and frameworks that help reduce time and complexity when developing mobile, web, and desktop applications.

Over the past two decades, we have learned quite a bit about providing an optimal support experience on a global scale for complex use cases. Our support system is backed by software systems that track every detail accurately. Nothing is left to chance. Customers appreciate that every detail is tracked to completion. Such actions let them know that we care not just with words in a support policy but in actual practice, day after day, year after year.

My **Services**



Get In Touch

Your Name

Your Email

Write a Message

My Contact Details

EMAIL

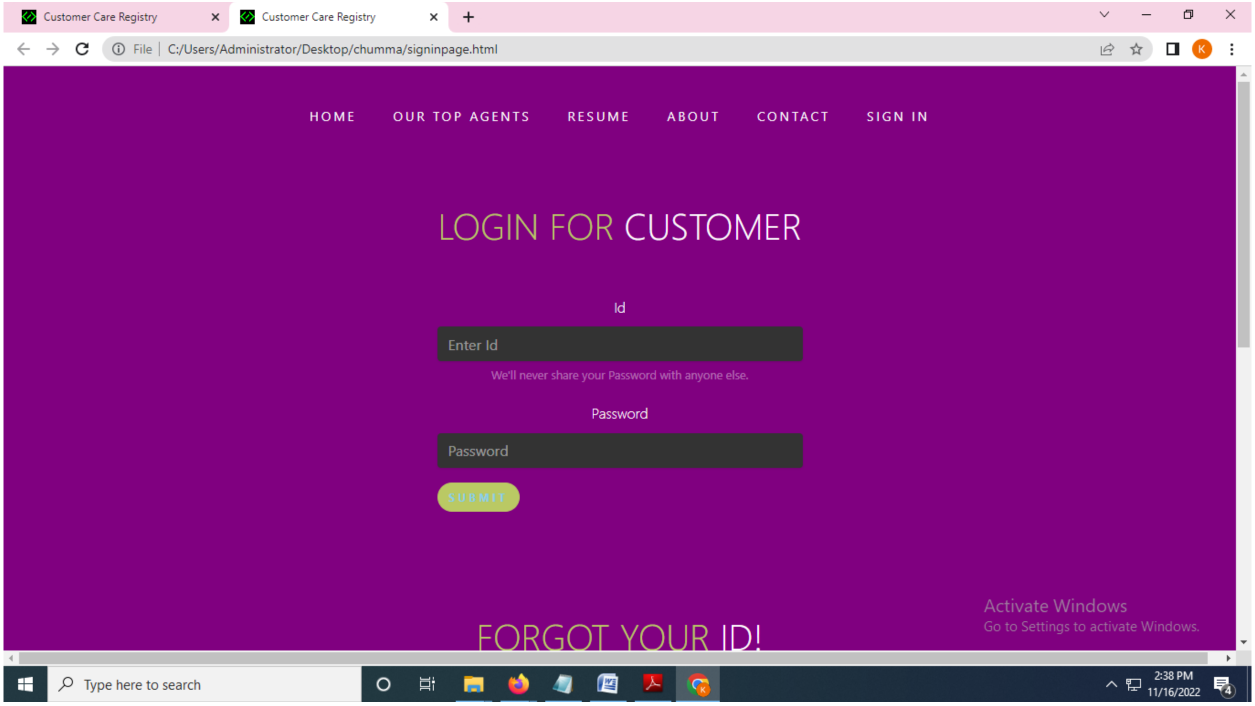
sabarijothi11@gmail.com

PHONE

+91 XXXXXXXXX

ADDRESS

55, ADAYAR
TAMBARAM,CHENNAI
India, TAMIL NADU



LOGIN FOR CUSTOMER

Id

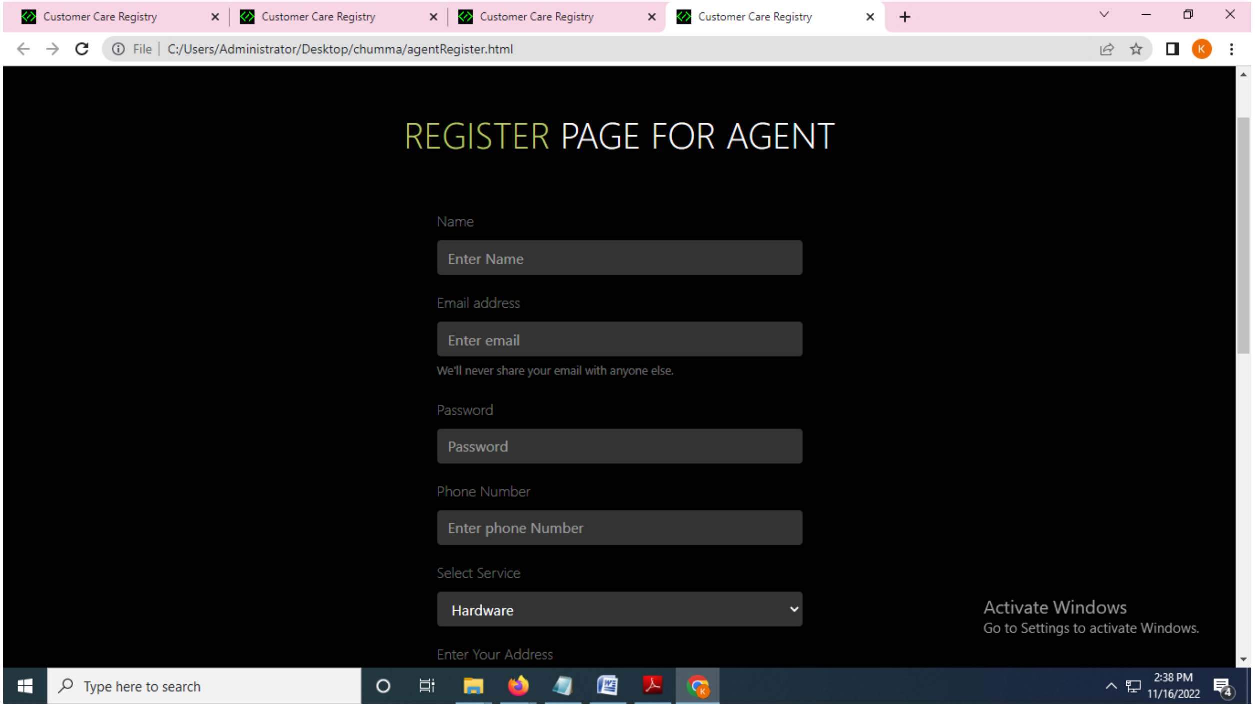
Enter Id

We'll never share your Password with anyone else.

Password

Password

FORGOT YOUR ID!



C:/Users/Administrator/Desktop/chumma/agentRegister.html

REGISTER PAGE FOR AGENT

Name

Enter Name

Email address

Enter email

We'll never share your email with anyone else.

Password

Password

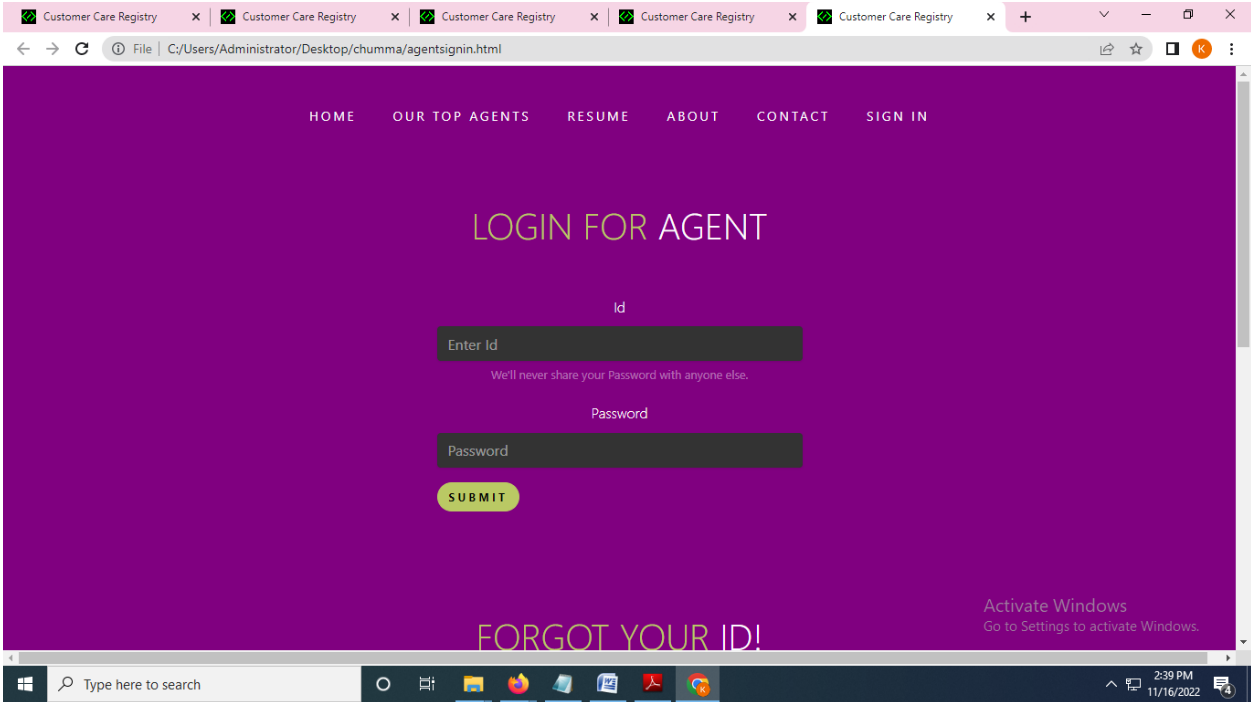
Phone Number

Enter phone Number

Select Service

Hardware

Enter Your Address



C:/Users/Administrator/Desktop/chumma/agentsignin.html

LOGIN FOR AGENT

Id

Enter Id

We'll never share your Password with anyone else.

Password

Password

FORGOT YOUR ID!

8.2 USER ACCEPTANCE TESTING

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).

Defect Analysis

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

Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	10	0	0	10
ClientApplication	40	0	0	40
Security	5	0	0	2

CHAPTER 9
RESULTS

9.1 PERFORMANCE METRICS

The Customer Feedback metric

The most important metric for your contact centre and the broader business is customer feedback. From the business perspective, you want to know if your customers are going to stay or leave and if they will recommend your business to their friends and colleagues. If you are the customer you want the right or incorrect processes improved so that you get the best experience. From the perspective of the employee, you want processes that will help deliver good service to customers.

The Service Efficiency Metric

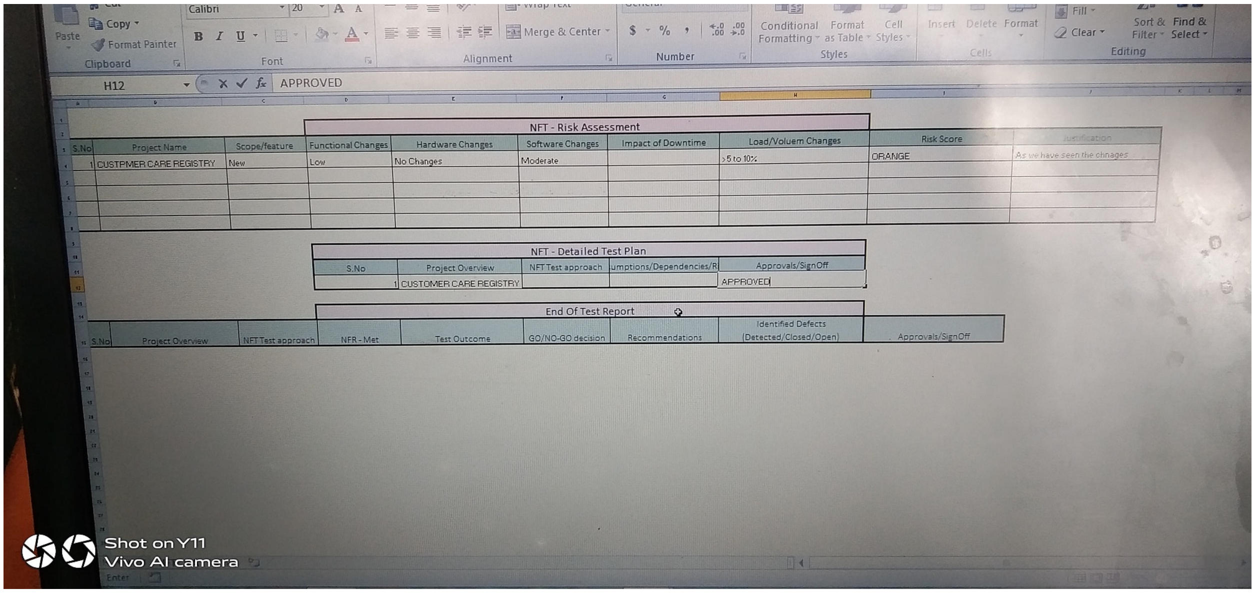
The next thing is service efficiency. The main thing is to measure service efficiency at a macro level rather than at an individual level to avoid bad behaviours. The most common story is the service agent that passes customers to other agents to improve their average handling time. As a service organisation, you want to give your employees the tools that they need to provide the best experience for customers.

Quality, Consistency and Compliance

The next of the 4 key metrics for customer service focuses on the effective running of a contact centre or customer service organisation is quality, consistency and compliance. If you’re a customer you want to ensure that you get the same experience each time they come into contact with the service team. As a customer in addition to great service, you want to ensure that processes are in place to protect and safeguard sensitive information.

Employee Engagement

The final metric that we believe to be important is employee engagement. How engaged is the team that is interacting with customers? We have found that engaged employees provide better service and help deliver increased customer satisfaction. As a business, you want to ensure that employees are being managed properly and that employees.



APPROVED

NFT - Risk Assessment

S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Voluem Changes	Risk Score	Justification
1	CUSTPMER CARE REGISTRY	New	Low	No Changes	Moderate		>5 to 10%	ORANGE	As we have seen the chnages

NFT - Detailed Test Plan

S.No	Project Overview	NFT Test approach	umptions/Dependencies/R	Approvals/SignOff
1	CUSTOMER CARE REGISTRY			APPROVED

End Of Test Report

S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

1. Customer loyalty

Loyal customers have many benefits for businesses. 91% of customers say a positive customer service experience makes them more likely to make a further purchase. Also, investing in new customers is five times more expensive than retaining existing ones (source: Invesp). Creating loyal customers through good customer service can therefore provide businesses with lucrative long-term relationships.

2. Increase profits

These long-term customer relationships established through customer service can help businesses become more profitable. Businesses can grow revenues between 4% and 8% above their market when they prioritise better customer service experiences (source: Bain & Company). Creating a better customer service experience than those offered by competitors can help businesses to standout in their market place, and in turn make more sales.

3. Customer recommendations

Providing good customer service can create satisfied customers, who are then more likely to recommend the business to others. 94% of customers will recommend a company whose service they rate as “very good” (source: Qualtrics XM Institute). This is useful, as 90% of customers are influenced by positive reviews when buying a product (source: Zendesk). Customers recommending a company through word of mouth or online reviews can improve the credibility of the business.

4. Increase conversion

Good customer service can help businesses turn leads into sales. 78% of customers say they have backed out of a purchase due to a poor customer experience (source: Glance). It is therefore safe to assume that providing good customer service will help to increase customer confidence and in turn increase conversion.

5. Improve public image

Customer service can help businesses to improve the public perception of the brand, which can then provide protection if there is a slip up. 78% of customers will forgive a company for a mistake after receiving excellent service (source: Salesforce Research). Meanwhile, almost 90% of customers report trusting a company whose service they rate as “very good.” On the other hand, only 16% of those who give a “very poor” rating trust companies to the same degree(source: Qualtrics XM Institute). Creating positive customer experiences is vital in gaining customer trust and creating a strong public image.

CHAPTER 11
CONCLUSION

- ✓ It is a web-enabled project.
- ✓ With this project the details about the productwill be given to the customers in detail with in a short span of time.
- ✓ Queries regarding the product or the serviceswill also be clarified.
- ✓ It provides more knowledgeabout the varioustechnologies.
- ✓ Seek and promote customer feedback

CHAPTER 12

FUTURE SCOPE

The future of customer service increasingly will be driven by technology innovations. Ideally, these new technologies will improve customer and agent experiences, along with business metrics like revenue, operational costs and customer ratings. But businesses often miss the mark when they try to move too quickly with too much technology, ultimately resulting in consumer dissatisfaction rather than elation.

Customer expectations for what defines a good experience stay fairly consistent over time, but the approach to providing that experience changes. Meanwhile, advanced technologies, largely driven by artificial intelligence, analytics and automation, arm companies with new techniques for driving customer satisfaction and loyalty.

How are customer expectations changing?

Over the years, customer expectations generally haven't changed. Customers want to be served quickly and completely on the first try. If they're speaking to a human agent, they want a friendly, knowledgeable interaction -- the goal being to resolve the customer's problem or answer their question quickly and easily.
Drilling down, however, customer expectations are influenced by the changes in technology. Just five years ago, for example, few customers would have expected to communicate with businesses over SMS or messaging services from their mobile phone. Now, it's common because consumers use those applications in other areas of their lives.

Perhaps the biggest area of change is the interaction channels used to communicate with businesses. Today, 58% of customers interact with digital channels, and 50% of all transactions start digitally, according to Metrigy's research. Consumers now expect to have several options for communication, including messaging apps like Facebook Messenger, WeChat, WhatsApp and Apple Business Chat, along with web chat, SMS, screen-sharing, video, self-service knowledge bases and FAQs, and chatbots.

Consumers also are more open to proactive outreach -- whether the customer service team is inviting them to a customer loyalty program or reminding them of an appointment -- so long as those reminders, confirmations and invitations arrive at their preferred application.

How is technology influencing the future of customer service?

Businesses can provide quick, contextual customer service with tools like analytics, agent assist and workforce optimization (WFO) for agents in the contact center, as well as customer-facing tools such as self-service, chatbots and personalization.

At the core of most new technologies are the three As -- artificial intelligence, automation and analytics. Working together, these technologies can provide organizations with advice, context, results and metrics for improvement, but it's imperative to roll out deployments cautiously instead of trying to boil the ocean. Businesses will then be able to identify how well these customer service tools address specific problems or opportunities and evaluate their performance through analytics.

Technology should improve customer service, customer experiences and agent satisfaction and continue to raise the bar for meeting customer expectations.

CHAPTER 13

APPENDIX

SOURCE CODE

```
from __future__ import print_function
from audioop import add
import datetime
from unicodedata import name
from pprint import pprint
from flask import Flask, render_template, request, redirect, url_for, session, flash
from markupsafe import escape
from flask import *
import ibm_db
import datetime
conn =
ibm_db.connect("DATABASE=;HOSTNAME=;PORT=;SECURITY;=SSL;SSLServerCertificat
e=;UID=;PWD=", '', '')
print(conn)
print("connection successful...")
app = Flask(__name__)
app.secret_key = 'your secret key'
@app.route('/')
def home():
    message = "TEAM ID : PNT2022TMID37544" +" "+ "BATCH ID : B1-1M3E "
    return render_template('index.html',mes=message)
@app.route('/home', methods=['POST', 'GET'])
def index():
    return render_template('index.html')
@app.route('/signinpage', methods=['POST', 'GET'])
def signinpage():
    return render_template('signinpage.html')
@app.route('/agentsignin', methods=['POST', 'GET'])
def agentsignin():
  return render_template('signinpageagent.html')
@app.route('/signuppage', methods=['POST', 'GET'])
def signuppage():
    return render_template('signuppage.html')
@app.route('/agentRegister', methods=['POST', 'GET'])
def agentRegister():
    return render_template('agentregister.html')
@app.route('/forgotpass', methods=['POST', 'GET'])
def forgotpass():
    return render_template('forgot.html')
@app.route('/newissue/<name>', methods=['POST', 'GET'])
def newissue(name):
    name = name
    return render_template('complaint.html',msg=name)
@app.route('/forgot', methods=['POST', 'GET'])
def forgot():
try:
        global randomnumber
        ida = request.form['custid']
        print(ida)
        global id
        id = ida
        sql = "SELECT EMAIL,NAME FROM Customer WHERE id=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, ida)
        ibm_db.execute(stmt)
        emailf = ibm_db.fetch_both(stmt)
        while emailf != False:
            e = emailf[0]
```

```
         n = emailf[1]
         break
configuration = sib_api_v3_sdk.Configuration()
      configuration.api_key['api-key']
api_instance = sib_api_v3_sdk.TransactionalEmailsApi(
         sib_api_v3_sdk.ApiClient(configuration))
      subject = "Verification for Password"
      html_content = "<html><body><h1>Your verification Code is : <h2>" + \
         str(randomnumber)+"</h2> </h1> </body></html>"
      sender = {"name": "IBM CUSTOMER CARE REGISTRY",
             "email": "ibmdemo6@yahoo.com"}
      to = [{"email": e, "name": n}]
      reply_to = {"email": "ibmdemo6@yahoo.com", "name": "IBM"}
      headers = {"Some-Custom-Name": "unique-id-1234"}
      params = {"parameter": "My param value",
             "subject": "Email Verification"}
      send_smtp_email = sib_api_v3_sdk.SendSmtpEmail(
         to=to, reply_to=reply_to, headers=headers, html_content=html_content,
params=params, sender=sender, subject=subject)
 api_response = api_instance.send_transac_email(send_smtp_email)
pprint(api_response)
      message = "Email send to:"+e+" for password"
      flash(message, "success")
 except ApiException as e:
      print("Exception when calling SMTPApi->send_transac_email: %s\n" % e)
      flash("Error in sending mail")
   except:
      flash("Your didn't Signin with this account")
   finally:
      return render_template('forgot.html')
@app.route('/agentforgot', methods=['POST', 'GET'])
def agentforgot():
try:
      global randomnumber
      ida = request.form['custid']
      print(ida)
      global id
      id = ida
      sql = "SELECT EMAIL,NAME FROM AGENT WHERE id=?"
      stmt = ibm_db.prepare(conn, sql)
      ibm_db.bind_param(stmt, 1, ida)
      ibm_db.execute(stmt)
      emailf = ibm_db.fetch_both(stmt)
      while emailf != False:
         e = emailf[0]
         n = emailf[1]
         break
 configuration = sib_api_v3_sdk.Configuration()
      configuration.api_key['api-key']
api_instance = sib_api_v3_sdk.TransactionalEmailsApi(
         sib_api_v3_sdk.ApiClient(configuration))
      subject = "Verification for Password"
      html_content = "<html><body><h1>Your verification Code is : <h2>" + \
         str(randomnumber)+"</h2> </h1> </body></html>"
      sender = {"name": "IBM CUSTOMER CARE REGISTRY",
             "email": "ibmdemo6@yahoo.com"}
      to = [{"email": e, "name": n}]
      reply_to = {"email": "ibmdemo6@yahoo.com", "name": "IBM"}
      headers = {"Some-Custom-Name": "unique-id-1234"}
      params = {"parameter": "My param value",
             "subject": "Email Verification"}
      send_smtp_email = sib_api_v3_sdk.SendSmtpEmail(
         to=to, reply_to=reply_to, headers=headers, html_content=html_content,
params=params, sender=sender, subject=subject)
```

```
api_response = api_instance.send_transac_email(send_smtp_email)
pprint(api_response)
        message = "Email send to:"+e+" for OTP"
        flash(message, "success")
 except ApiException as e:
        print("Exception when calling SMTPApi->send_transac_email: %s\n" % e)
        flash("Error in sending mail")
    except:
        flash("Your didn't Signin with this account")
    finally:
        return render_template('forgot.html')
@app.route('/admin', methods=['POST', 'GET'])
def admin():
    userdatabase = []
    sql = "SELECT * FROM customer"
    stmt = ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_both(stmt)
    while dictionary != False:
        userdatabase.append(dictionary)
        dictionary = ibm_db.fetch_both(stmt)
    if userdatabase:
        sql = "SELECT COUNT(*) FROM customer;"
        stmt = ibm_db.exec_immediate(conn, sql)
        user = ibm_db.fetch_both(stmt)

    users = []
    sql = "select * from ISSUE"
    stmt = ibm_db.exec_immediate(conn, sql)
    dict = ibm_db.fetch_both(stmt)
    while dict != False:
        users.append(dict)
        dict = ibm_db.fetch_both(stmt)
    if users:
        sql = "SELECT COUNT(*) FROM ISSUE;"
        stmt = ibm_db.exec_immediate(conn, sql)
        count = ibm_db.fetch_both(stmt)
agent = []
    sql = "SELECT * FROM AGENT"
    stmt = ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_both(stmt)
    while dictionary != False:
        agent.append(dictionary)
        dictionary = ibm_db.fetch_both(stmt)
   if agent:
        sql = "SELECT COUNT(*) FROM AGENT;"
        stmt = ibm_db.exec_immediate(conn, sql)
        cot = ibm_db.fetch_both(stmt)
return
render_template("admin.html",complaint=users,users=userdatabase,agents=agent,mes
sage=user[0],issue=count[0],msgagent = cot[0])
@app.route('/remove', methods=['POST', 'GET'])
def remove():
otp = request.form['otpv']
    if otp == 'C':
        try:
            insert_sql = f"delete from customer"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.execute(prep_stmt)
            flash("delected successfully the Customer", "success")
        except:
            flash("No data found in Customer", "danger")
        finally:
            return redirect(url_for('signuppage'))
    if otp == 'A':
        try:
            insert_sql = f"delete from AGENT"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.execute(prep_stmt)
            flash("delected successfully the Agents", "success")
        except:
            flash("No data found in Agents", "danger")
        finally:
           return redirect(url_for('signuppage'))
 if otp == 'C':
        try:
            insert_sql = f"delete from AGENT"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.execute(prep_stmt)
            flash("delected successfully the Complaints", "success")
        except:
            flash("No data found in Complaints", "danger")
        finally:
            return redirect(url_for('signuppage'))
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        try:
id = request.form['idn']
            global hello
            hello = id
            password = request.form['password']
            print(id, password)
            if id == '1111' and password == '1111':
                return redirect(url_for('admin'))
 sql = f"select * from customer where id='{escape(id)}' and
password='{escape(password)}'"
```

```
            stmt = ibm_db.exec_immediate(conn, sql)
            data = ibm_db.fetch_both(stmt)

            if data:
                session["name"] = escape(id)
                session["password"] = escape(password)
                return redirect(url_for("welcome"))
else:
                flash("Mismatch in credetials", "danger")
        except:
            flash("Error in Insertion operation", "danger")
return render_template('signinpage.html')
@app.route('/welcome', methods=['POST', 'GET'])
def welcome():
    try:
        id = hello
        sql = "SELECT
ID,DATE,TOPIC,SERVICE_TYPE,SERVICE_AGENT,DESCRIPTION,STATUS FROM ISSUE
WHERE CUSTOMER_ID =?"
        agent = []
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, id)
        ibm_db.execute(stmt)
        otpf = ibm_db.fetch_both(stmt)
        while otpf != False:
            agent.append(otpf)
            otpf = ibm_db.fetch_both(stmt)
sql = "SELECT COUNT(*) FROM ISSUE WHERE CUSTOMER_ID = ?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, id)
        ibm_db.execute(stmt)
        t = ibm_db.fetch_both(stmt)
        return render_template("welcome.html",agent=agent,message=t[0])
    except:
        return render_template("welcome.html")
@app.route('/loginagent', methods=['GET', 'POST'])
def loginagent():
    if request.method == 'POST':
        try:
            global loginagent
            id = request.form['idn']
            loginagent = id
            password = request.form['password']
    sql = f"select * from AGENT where id='{escape(id)}' and
password='{escape(password)}'"
            stmt = ibm_db.exec_immediate(conn, sql)
            data = ibm_db.fetch_both(stmt)

            if data:
                session["name"] = escape(id)
                session["password"] = escape(password)
                return redirect(url_for("agentwelcome"))
else:
                flash("Mismatch in credetials", "danger")
        except:
            flash("Error in Insertion operation", "danger")
return render_template("signinpageagent.html")
@app.route('/delete/<ID>')
def delete(ID):
    sql = f"select * from customer where Id='{escape(ID)}'"
    print(sql)
    stmt = ibm_db.exec_immediate(conn, sql)
    student = ibm_db.fetch_row(stmt)
    if student:
        sql = f"delete from customer where id='{escape(ID)}'"
        stmt = ibm_db.exec_immediate(conn, sql)

        flash("Delected Successfully", "success")
        return redirect(url_for("admin"))
@app.route('/agentform', methods=['GET', 'POST'])
def agentform():
    if request.method == 'POST':
 try:
            x = datetime.datetime.now()
            y = x.strftime("%Y-%m-%d %H:%M:%S")
            name1 = request.form['name']
            email = request.form['email']
            password = request.form['password']
            phonenumber = request.form['phonenumber']
            service = request.form['service']
            address = request.form['address']
            city = request.form['city']
            state = request.form['state']
            country = request.form['country']
            link = request.form['link']
 sql = "SELECT * FROM AGENT WHERE EMAIL = ?"
            stmt = ibm_db.prepare(conn, sql)
            ibm_db.bind_param(stmt, 1, email)
            ibm_db.execute(stmt)
            account = ibm_db.fetch_assoc(stmt)
if account:
                flash("Record Aldready found", "success")
            else:
                print("exec")
                insert_sql = "INSERT INTO AGENT
```

```
(NAME,EMAIL,PASSWORD,PHONENUMBER,SERVICE_AGENT,ADDRESS,CITY,STATE,COU
NTRY,RESUME_LINK,DATE) VALUES(?,?,?,?,?,?,?,?,?,?,?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, name1)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, password)
            ibm_db.bind_param(prep_stmt, 4, phonenumber)
            ibm_db.bind_param(prep_stmt, 5, service)
            ibm_db.bind_param(prep_stmt, 6, address)
            ibm_db.bind_param(prep_stmt, 7, city)
            ibm_db.bind_param(prep_stmt, 8, state)
            ibm_db.bind_param(prep_stmt, 9, country)
            ibm_db.bind_param(prep_stmt, 10, link)
            ibm_db.bind_param(prep_stmt, 11, y)
            ibm_db.execute(prep_stmt)
            flash("Record stored Successfully", "success")
            sql = "SELECT ID FROM AGENT WHERE email=?"
            stmt = ibm_db.prepare(conn, sql)
            ibm_db.bind_param(stmt, 1, email)
            ibm_db.execute(stmt)
            hi = ibm_db.fetch_tuple(stmt)
configuration = sib_api_v3_sdk.Configuration()
            configuration.api_key['api-key']
api_instance = sib_api_v3_sdk.TransactionalEmailsApi(
            sib_api_v3_sdk.ApiClient(configuration))
            subject = "Registering Account in Customer Care Registry"
            html_content = " <html><body><h1>Thanks for Registering into Customer Care
Registry</h1> <h2>Your Account Id is :"+str(hi[0])+"</h2><h2>With
Regards:</h2><h3>Customer Care Registry</h3> </body></html>"
            sender = {"name": "IBM CUSTOMER CARE REGISTRY",
              "email": "ibmdemo6@yahoo.com"}
            to = [{"email": email, "name": name1}]
```

```
        reply_to = {"email": "ibmdemo6@yahoo.com", "name": "IBM"}
        headers = {"Some-Custom-Name": "unique-id-1234"}
        params = {"parameter": "My param value",
         "subject": "Email Verification"}
        send_smtp_email = sib_api_v3_sdk.SendSmtpEmail(
        to=to, reply_to=reply_to, headers=headers, html_content=html_content,
params=params, sender=sender, subject=subject)
api_response = api_instance.send_transac_email(send_smtp_email)
 pprint(api_response)
   except:
      flash("Error in Insertion Operation", "danger")
    finally:
      return redirect(url_for("agentRegister"))
      con.close()
return render_template('agentregister.html')
@app.route('/completed/<DESCRIPTION>', methods=['GET', 'POST'])
def completed(DESCRIPTION):
  status ="Completed"
  try:
 sql = "UPDATE ISSUE SET STATUS = ? WHERE DESCRIPTION =?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,status)
    ibm_db.bind_param(stmt,2,DESCRIPTION)
    ibm_db.execute(stmt)
flash("Successful","success")
    return redirect(url_for('agentwelcome'))
  except:
    flash("No record found","danger")
    return redirect(url_for('agentwelcome'))
@app.route('/deletecomplaint/<ID>')
def deletecomplaint(ID):
  sql = f"select * from ISSUE where ID='{escape(ID)}'"
```

```
    print(sql)
    stmt = ibm_db.exec_immediate(conn, sql)
    student = ibm_db.fetch_row(stmt)
    if student:
        sql = f"delete from ISSUE where ID='{escape(ID)}'"
        stmt = ibm_db.exec_immediate(conn, sql)
        users = []
        flash("Delected Successfully", "success")

        return redirect(url_for("admin"))
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
```

GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-48159-1660804994

DEMO LINK

https://youtu.be/5nl-kWsoN3o