CUSTOMER CARE REGISTRY

A PROJECT REPORT

Submitted by

Team ID – PNT2022TMID52252

Team Leader	**Ashina R**
Team Member	**Priskila B**
Team Member	**Pushpa Lincy J**
Team Member	**Sanju P**

in partial fulfilment for the award of

the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

STELLA MARY'S COLLEGE OF ENGINEERING

ARUTHENGANVILAI

ANNA UNIVERSITY : CHENNAI 600 025

ABSTRACT

Customer service is a key component in the success of any organization. However, gaps abound in the execution of this important service such that the organization's progress is frustrated. A review of relevant current research and literature underlines the respectability of this important segment of communication. The purpose of this study was to investigate the skills of customer service agents in corporate communication with particular emphasis on KenCall with the aim of making relevant recommendations that can enhance the role played by customer service in ensuring that the corporate remains profitable and competitive in the market.

The key findings of the study established that majority of the CSA's possess the requisite skills in customer service. However, the findings revealed that skills such as (1) competitors' products (2) stress management and (3) communication skills received less attention and were lacking among the agents. Lack of the above skills by the CSA's could negatively affect the level of customer service delivery thereby negatively affecting corporate communication at KenCall.

From the data interpretation the following recommendations were made: that since the customer service job involves frequent stressful encounters by CSAs, there is need for frequent training on stress management, that there is need to constantly equip the CSAs with skills in competitor's products and services, and that CSAs receive training in communication skills such as message repletion, two-sided presentation of messages, order of message presentation and skills of communicating to different audiences.

ACKNOWLEDGEMENT

1. INTRODUCTION

1.1 Project Overview

Customer service is the support you offer your customers — both before and after they buy and use your products or services — that helps them have an easy and enjoyable experience with you. Offering amazing customer service is important if you want to retain customers and grow your business. Today's customer service goes far beyond the traditional telephone support agent. It's available via email, web, text message, and social media. Many companies also provide self-service support, so customers can find their own answers at any time day or night. Customer support is more than just providing answers; it's an important part of the promise your brand makes to its customers.

1.2 Purpose

Customer service is the support you offer your customers — both before and after they buy and use your products or services — that helps them have an easy, enjoyable experience with your brand. But customer service is more than solving a customer's problems and closing tickets. Today, customer service means delivering proactive and immediate support to customers anytime on the channel of their choice — phone, email, text, chat, and more.

Customer service is also a differentiator that sets your brand apart from competitors that offer similar products or services. Service teams not only answer questions; they make each experience personalized to the customer. In fact, 80% of customers say that the experience a company provides is as important as its products or services.

Meanwhile, frustrating customer experiences contribute to churn. Eighty percent of shoppers will abandon a retailer after three bad experiences, for example. Great customer service is important for your brand reputation, too. After all, customers are quick to share negative experiences with the masses online.

2. LITERATURE SURVEY

2.1 Existing Problem

Customer service is critical to competing effectively. In the past, people chose which companies they did business with based on price, or the product or service offered, but today the overall experience is often the driver. "89% of companies now expect to compete mostly on the basis of customer experience. "Great customer support drives an amazing customer experience, especially when your support team moves beyond just reacting to problems and toward anticipating customers' problems. When support agents are empowered to go above-and-beyond with customers, or have a help desk solution that makes it easy for them to upsell or cross-sell relevant services, they can create winning experiences that help you stand out from the competition.

2.2 Reference

1. *Lucas, Robert (2015). Customer Service Skills For Success. New York: McGraw-Hill. ISBN 978-0-07-354546-2.*

2. **^** *Buchanan, Leigh (1 March 2011). "A Customer Service Makeover". Inc. magazine. Retrieved 29 Oct 2012.*

3. **^** Teresa Swartz, Dawn Iacobucci. *Handbook of Services Marketing and Management*. Thousand Oaks, CA: Sage

4. **^** *Bordoloi, Sanjeev (2019). Service Management Operations, Strategy, Information Technology. New York: McGraw-Hill. ISBN 978-1-260-09242-4.*

5. ^ Jump up to:*ᵃ ᵇ* businessdictionary.com > customer support Archived 2018-07-23 at the Wayback Machine Retrieved March 2011

6. **^** *Crittenden, Victoria (2020-01-01). "Customer support services: more than administrative support – it has to be strategic!". European Journal of Marketing. **54** (7): 1807–1808. doi:10.1108/EJM-07-2020-972. ISSN 0309-0566. S2CID 225558345.*

7. **^** *"10 reasons why AI-powered, automated customer service is the future". ibm.com. 16 October 2017. Retrieved 2020-05-17.*

8. ^ Jump up to:*ᵃ ᵇ ᶜ ᵈ Kongthorn, Alisa & Sangkeettrakarn, Chatchawal & Kongyoung, Sarawoot & Haruechaiyasak, Choochart (2009). "Implementing an online help desk system based on conversational agent". Bibliometrics*

2.3. Problem Statement Definition

Customer Problem Statement

A Customer Problem Statement is a detailed description of an issue that needs to be addressed. This document thoroughly elaborates on the problem that your product or your service solves for your particular customers. It takes into consideration your customer's unique pain points and how your product goals about solving their situation. A customer problem statement helps you and your team understand the detailed experience you are attempting to transform by analyzing and empathizing with your customers. The statement also helps to ensure teams are focusing on the right thing during the innovation or the development process, by analyzing the problem and coming up with ways to solve them before the product is fully built.

The Importance of a Customer Problem Statement

The customer problem statement is a critical component of a project. It benefits everyone involved with the project because it helps people understand why they're working on the project, providing clarity on the reasons behind the product or service. Team members will consider how your customers will be impacted by your project, what their thoughts and needs are, and thus come up with truly effective and valuable ways to improve their experience. A well-rounded customer problem statement will have a big impact on how successful your final product is. This document helps your team define the problem and guide them during the product development process

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

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.

 Reference:

https://app.mural.co/t/customercareregistry6511/home
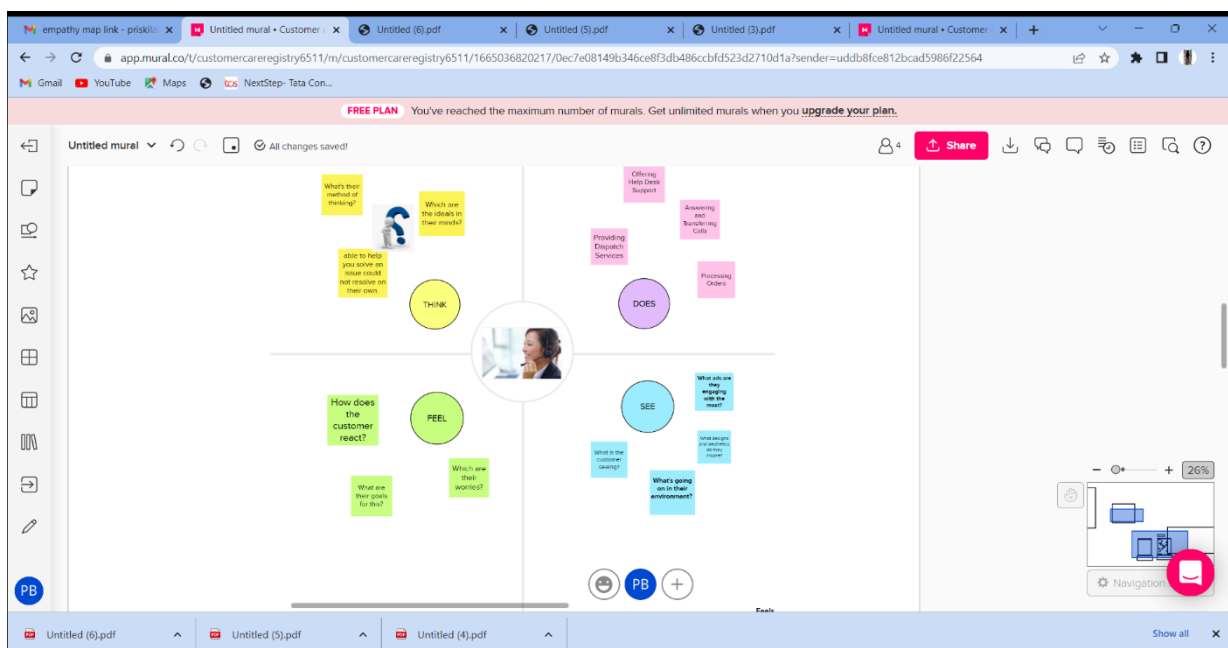
3.2 Ideation & Brainstorming

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 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 are not sitting in the same room.

Reference:

https://app.mural.co/t/customercareregistry6511/m/customercareregistry6511/166546546349

9/3f c8f831dd5cb230def0b6cbeb18670d4c0bac4b?sender=u18848d23baa956eddcfb4578.

3.3 Proposed Solution

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement(Problem to be solved)	This Application has been developed to help the customer in processing their complaints. The recruitment of the customer care is solving customer problems that customer face in daily life. A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face.
2.	Idea/Solution description	The proposed Solution implements a cloud based web application as a problem. An web application is created using frontend technologies like HTML, CSS, JS, BOOTSTRAP and for the backend python Flask and for the database IBM DB2 is used. Customer details will be embedded with clouds storage. The admin is responsible for accepting issues from the customer and assign an agent to the respective customer and the agent solves problem.
3.	Novelty/Uniqueness	The refund will be converted into the next travelling fee of the customer

4.	Social Impact /Customer Satisfaction	Customer care registry can provide fast , convenient customer support and immediately react to the customer queries.
5.	Business Model (Revenue Model)	By providing this service to the companies for better customer support
6.	Scalability of the Solution	As this is an web application and uses cloud storage .It can be measured by the quality of the services we provide , quick responses for the issues from the agents we have

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement	Sub Requirement
1	User Registration	Registration through Form Registration through Gmail Registration through Google
2	User Confirmation	Confirmation via Email Confirmation via OTP
3	User Login	Login via Google login with Email id and Password
4	Admin Login	Login via Google login with Email id and Password
5	Query Form	Description of the issues contact information
6	Email	Login alertness
7	Feedback	Customer Feedback

NON FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements of proposed system

FR No.	Non Functional Requirements	Description
1	Usability	To provide the solution to the problem
2	Security	Track of login authentication
3	Reliability	Tracking of decade status through email
4	Performance	Effective development of web application
5	Availability	24/7 service
6	Scalability	Agents scalability as per the number of customers

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 amount of the system requirement graphically. It shows how data enters and leaves the system , what changes the information, and where data stored
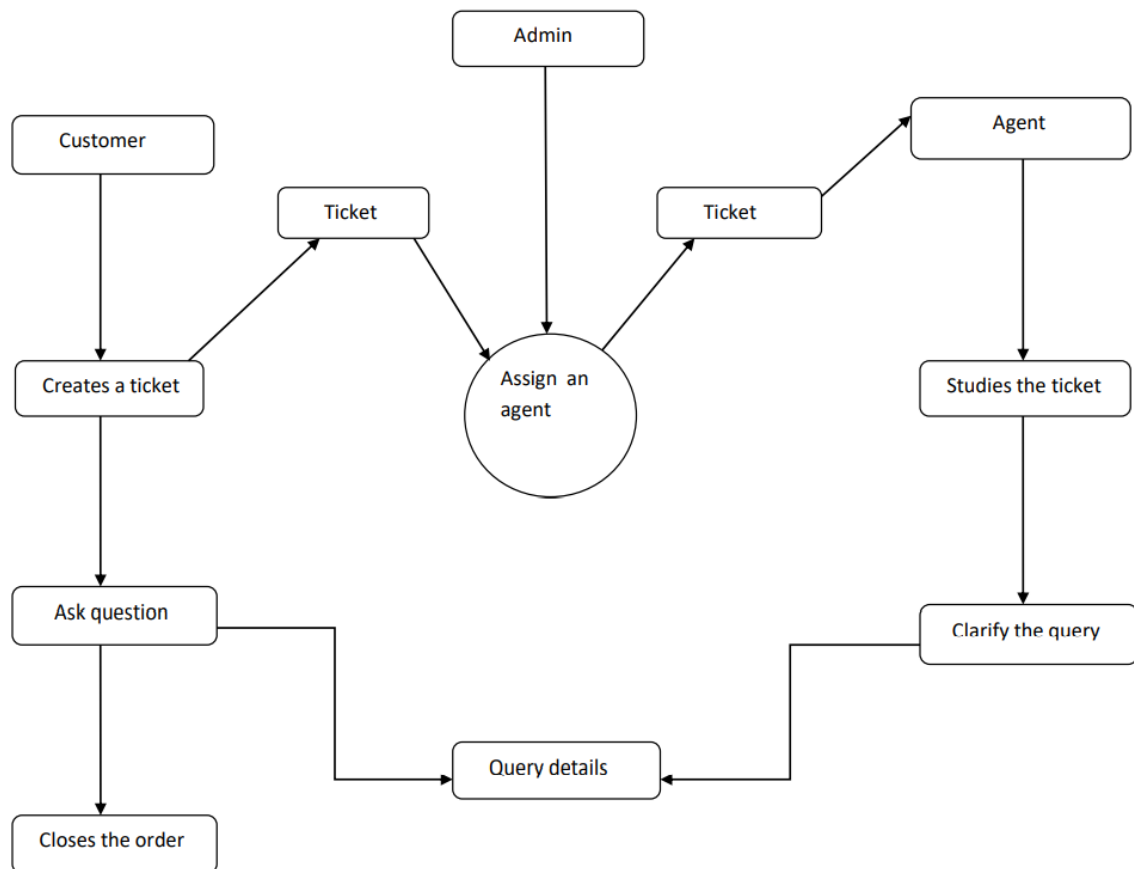
5.2 Solution & Technical Architecture
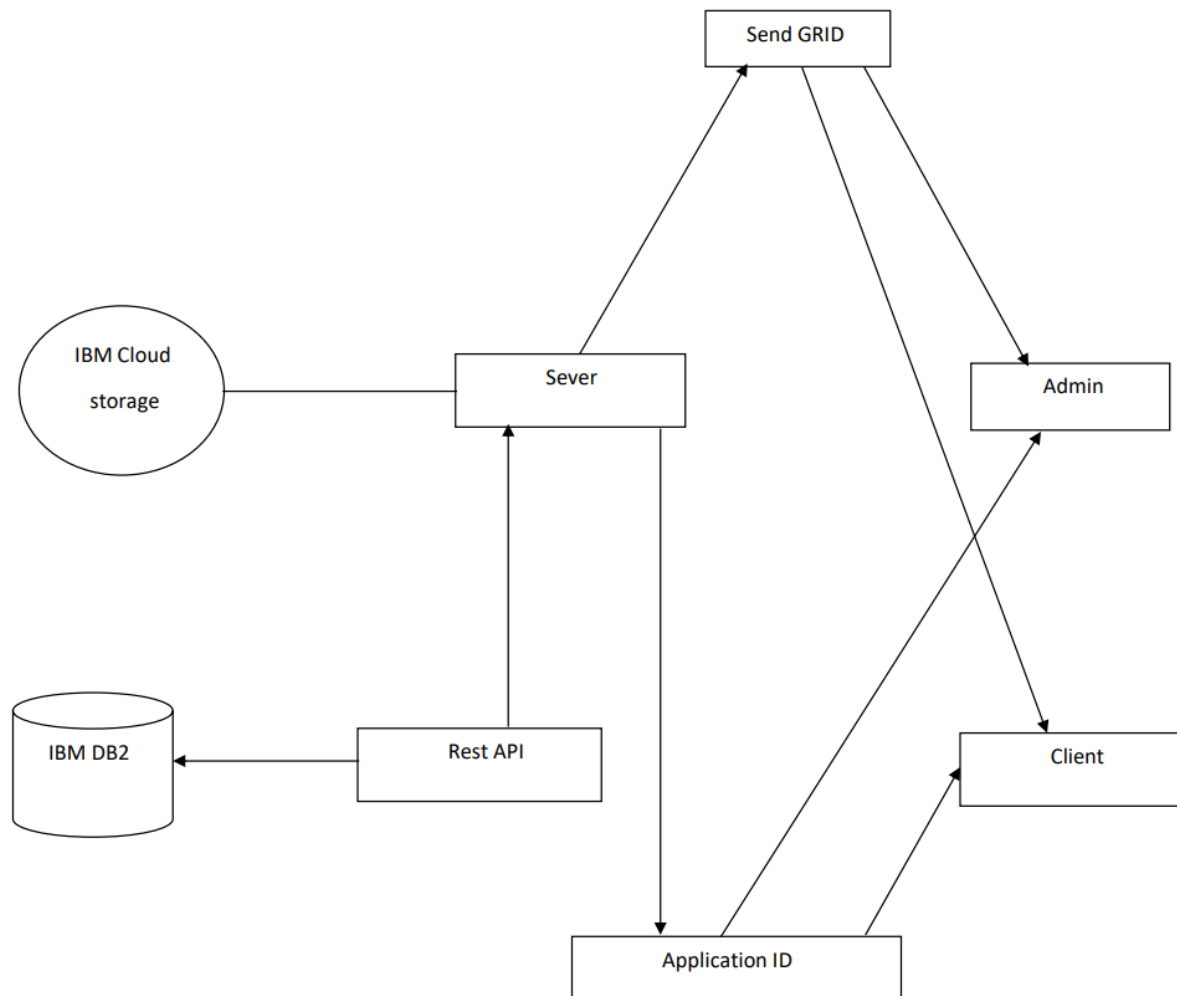
Table-1 Components & Technologies:

S.NO	Components	Description	Technology
1.	User Interface	How user interacts with application e.g, Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configuration etc.	MySQL
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	Infrastructure (Server/Cloud)	Application Deployment on Local System/ Cloud Local Sever Configuration: Cloud Sever Configuration:	IBM Cloud Foundry, Kubernetes

Table-2 Application Characteristics:

S.NO	Characteristics	Description	Technology
1.	Open-Source Framework	List the open-source framework used	Python Flask
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g, Encryption, Intrusion, firewalls
3.	Scalable Architecture	Justify the scalability of architecture (3-tier, Micro-services)	Virtual Machines
4.	Availability	Justify the availability of application (e.g use of load balance, distributed servers etc.)	Microservices
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN'S) etc.	CI/CD

5.3 User Stories

Use the below template to list all the user stories for the product

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a customer, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
	login	USN-2	As a customer, I can login to the application by entering correct email and password.	I can access my account/dashboard.	High	Sprint-1
	Dashboard	USN-3	As a customer, I can see all the orders raised by me.	I get all the info needed in my dashboard.	Low	Sprint-2
	Order creation	USN-4	As a customer, I can place my order with the detailed description of my query	I can ask my query	Medium	Sprint-2
	Address Column	USN-5	As a customer, I can have conversations with the assigned agent and get my queries clarified	My queries are clarified.	High	Sprint-3
	Forgot password	USN-6	As a customer, I can reset my password by this option incase I forgot my old password.	I get access to my account again	Medium	Sprint-4
	Order details	USN-7	As a Customer ,I can see the current stats of order.	I get abetter understanding	Medium	Sprint-4
Agent (web user)	Login	USN-1	As an agent I can login to the application by entering Correct email and password.	I can access my account / dashboard.	High	Sprint-3
	Dashboard	USN-2	As an agent, I can see the order details assigned to me by admin.	I can see the tickets to which I could answer.	High	Sprint-3
	Address column	USN-3	As an agent, I get to have conversations with the customer and clear his/er dobuts	I can clarify the issues.	High	Sprint-3
	Forgot password	USN-4	As an agent I can reset my password by this option in case I forgot my old password.	I get access to my account again.	Medium	Sprint-4

Admin (Mobile user)	Login	USN-1	As a admin, I can login to the appliaction by entering Correct email and password	I can access my account/dashboard	High	Sprint-1
	Dashboard	USN-2	As an admin I can see all the orders raised in the entire system and lot more	I can assign agents by seeing those order.	High	Sprint-1
	Agent creation	USN-3	As an admin I can create an agent for clarifying the customers queries	I can create agents.	High	Sprint-2
	Assignment agent	USN-4	As an admin I can assign an agent for each order created by the customer.	Enable agent to clarify the queries.	High	Sprint-1
	Forgot password	USN-5	As an admin I can reset my password by this option in case I forgot my old password.	I get access to my account.	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Milestone & Activity list

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers,research publications etc.	28 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	24 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	25 SEPTEMBER 2022

Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	23 SEPTEMBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	30 SEPTEMBER 2022
Solution Architecture	Prepare solution architecture document.	28 SEPTEMBER 2022

Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	20 OCTOBER 2022
Functional Requirement	Prepare the functional requirement document.	8 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	9 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram.	10 OCTOBER 2022

Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	22 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS..

6.2 Sprint Delivery Schedule

Sprint	Functional Requirements (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration (Admin and Customer)	USN – 1	As a user, I can register for the application by entering My Registration Credentials like name, username, password, confirm password	3	Medium	Ashina R, Priskila B
Sprint-1	Login and Dashboard (Admin and Customer)	USN – 2	As a user, I will Login into the Dashboard Page by using the Login Credentials once after the Mail Activation link Authentication	5	Medium	Pushpa Lincy J, Sanju p
Sprint-1	Authentication and IBM DB2	USN – 3	As a user, I will be Authenticated from the Administration and store the Credentials back to the Database using IBM DB2	5	High	Ashina R, Pushpa Lincy J
Sprint-2	Email integration and SendGrid API	USN – 4	As a user, I will send the email to the Customer automatically using SendGrid API with the Subject to ticket id, agent name	5	Medium	Priskila B, Sanju P
Sprint-2	DB Schema for queries	USN – 5	As a user, I will create and map the credentials of the Customer from the Application through Tables and Schema	3	High	Priskila B, Pushpa Lincy J
Sprint-3	Watson Assistant	USN – 6	As a user, I will walk through the customer to resolve the queries and also connect the live agent to the Application	3	High	Pushpa Lincy J
Sprint-3	Knowledge Base Assistant	USN – 7	As a user, I will provide predefined Queries like FAQ's so that the customer can be solved by DIY Mechanism	5	Medium	Sanju P
Sprint-4	Deployment with Docker	USN – 8	As a user, I will deploy the entire Application using Docker	1	Medium	Ashina R, Sanju P
Sprint-4	Orchest with Kubernates	USN – 9	As a user, I will allocate the server nodes and balance the workloads in server	2	Medium	Ashina R, Priskila B, Pushpa Lincy J, Sanju P

Project Tracker, Velocity Burndown Chart:

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	13	6 Days	24 Oct 2022	29 Oct 2022		
Sprint – 2	8	6 Days	31 Oct 2022	05 Nov 2022		
Sprint – 3	8	6 Days	07 Nov 2022	12 Nov 2022		
Sprint – 4	3	6 Days	14 Nov 2022	19 Nov 2022		

Velocity:

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

Average velocity of sprint – 1 = 13/6 = 2.17

Average velocity of sprint – 2 = 8/6 = 1.25

Average velocity of sprint – 3 = 8/6 = 1.25

Average velocity of sprint – 4 = 3/6 = 0.5

6.3 Reports from JIRA

7. CODING &SOLUTIONING

7.1Feature 1

Registration Page Code

Registration .html:

```html
<!DOCTYPEhtml>
<htmllang="en">
<head>
    <metacharset="UTF-8">
    <metahttp-equiv="X-UA-Compatible"content="IE=edge">
    <metaname="viewport"content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <linkrel="stylesheet"href="first reg.css">
    <linkrel="stylesheet"href="https://cdn.tutorialjinni.com/intl-tel-
input/17.0.8/css/intlTelInput.css"/>
    <scriptsrc="https://cdn.tutorialjinni.com/intl-tel-
input/17.0.8/js/intlTelInput.min.js"></script>

</head>
<body>
    <divclass="container">
        <formid="contact"action=""method="post">
            <h1>Registration Form</h1>
            <h3>Fill the form below and press the submit button!</h3>
            <divclass="row">
                <!-- first column -->
                <divclass="column">
                    <fieldset>
                        <inputtype="text"placeholder="Full Name
*"name="name"requiredautofocus>
                    </fieldset>
                    <fieldset>
```

```html
                <input type="text" placeholder="Father name
*" name="fname" required autofocus>
            </fieldset>
            <fieldset>
                <input type="email" placeholder="Your email
*" name="email" required autofocus>
            </fieldset>
            <fieldset>
                <input type="text" placeholder="Date of birth
*" name="date" onfocus="(this.type= 'date')" required autofocus>
            </fieldset>
            <!-- adding all country code list -->
            <fieldset>
                <input type="text" placeholder="Phone number
*" name="phone" id="phone" required autofocus>
            </fieldset>
        </div>
        <!-- second column -->
        <div class="column">
            <fieldset>
                <p>What is your gender?</p>
                <div class="radio">
                    <input type="radio" id="male" name="gender" required>
                    <label for="male">Male</label>
                </div>
                <div class="radio">
                    <input type="radio" id="female" name="gender" required>
                    <label for="female">Female</label>
                </div>
            </fieldset>
            <fieldset>
              <div class="idCard">
                  <label for="idCard">How did you hear about us??</label><br>
              </div><div class="idCard">
```

```html
  <select name=" types" id="cars">
  <option value="blank"> Please select</option>
  <option value="">Newspaper</option>
  <option value="">Internet</option>
  <option value="">Magazine</option>
  </select>
                </div>
            </fieldset>


          </div>
        </div>
        <!-- submit button -->
        <fieldset>


<a href ="file:///C:/ibm%20project/sprint%20coding/log.html">
<input type="button" value="Submit Now" id="button" class =" hello"></a>


        </fieldset>
      </form>
    </div>



    <!--Javascript to initialize the code list -->
    <script>
    var input = document.querySelector("#phone");
    window.intlTelInput(input, {
     separateDialCode:true
     });
    </script>
  </body>
</html>
```

Registration.css:

```css
/* import google font family */
@import url('https://fonts.googleapis.com/css2?family=Open+Sans:ital,wght@0,600;1,600&display=swap');
/* selecting everything */
* {
 padding: 0;
 margin: 0;
 box-sizing: border-box;
}
body {
 font-family: 'Open Sans', sans-serif, helvetica, Arial;
 font-weight: 400;
 font-size: 14px;
 color: black;
 /* body background image */
 background-image: linear-gradient(to bottom, rgba(128, 128, 128, 0.541), rgba(204, 0, 255, 0.637)), url(image.jpg);
 background-attachment: fixed;
 background-size: cover;

}
.container {
 max-width: 800px;
 width: 100%;
 margin: 0auto;
}
#contact {
 background-color: rgba(220, 255, 66, 0.7);
 padding: 20px;
 margin: 50px0;
```

```css
}
#contactinput, button {
 font: 40015px'Open Sans', sans-serif, helvetica, Arial;
}
#contacth1 {
 font-size: 35px;
 font-weight: bold;
 text-align: center;
 color: blue;
}
#contacth3 {
 margin: 5px0px15px;
 text-align: center;
}
.row {
 display: flex;
 width: 100% !important;
 padding: 20px0px;
}
.row.column {
 margin: 0px20px;
 width: 50%;
}
fieldset {
 border: mediumnone!important;
 margin:  0010px;
 min-width: 100%;
 width: 100%;
}
#contactinput {
 width: 100%;
 border: 1pxsolidrgb(150, 150, 150);
 background-color: white;
 padding: 10px;
```

```css
  margin: 5px0;
}
.idcard{
width: 200%;
 border: 1pxsolidrgb(150, 150, 150);
 background-color: white;
 padding: 10px;
 margin: 8px0;




 }
input[type = "radio"] {
 width: 10%!important;
}
#contact.row.radio {
 border: 1pxsolidrgb(150, 150, 150);
 background-color: white;
 margin: 7px010px;
 padding: 5px;
}
#contact.row.idCard{
 display: flex;
 height: 45px!important;
 margin: 5px0;
}


}
#contact.row#phone {
 max-width: 100%!important;
 padding-right: 80px;
}
#contactinput:hover {
```

```css
  transition: border-color0.3sease-in-out;
  border: 1pxsolidrgb(68, 68, 68);
}
#contactbutton {
  outline: none;
  border: none;
  background-color: blue;
  color: white;
  margin: 005px40%;
  padding: 10px;
  font-size: 17px;
  width: 150px;
}

#contactbutton:hover {
  background-color: rgba(0, 0, 255, 0.8);
}

#contactinput:focus {
  outline: 0;
  border: 1pxsolidred;
}

/* mobile responsive mode */
@mediascreen and (max-width: 580px) {
.row {
  flex-direction: column;
}
.row.column {
  width: 90% !important;
}
#contact.row#phone {
padding-right: 180px;
}
```

```
a{
text-decoration: none;
color:white;
}
.hello{
background-color:blue;
color :white;



}
```


Login code

Login.html:

```
<!DOCTYPEhtml>
<htmllang="en">
<head>
    <metacharset="UTF-8">
    <title>Validated Login Form</title>
    <linkrel="stylesheet"href="log.css">
```

```html
</head>
<body>
    <div class="container">
        <h1 class="label">User Login</h1>


<form class="login_form" action="home.html" method="post" name="form" onsubmit="returnv
alidated()">
        <div class="font">Email or Phone</div>
        <input autocomplete="off" type="text" name="email">
        <div id="email_error">Please fill up your Email or Phone</div>
        <div class="font font2">Password</div>
        <input type="password" name="password">
        <div id="pass_error">Please fill up your Password</div>
        <button type="submit">Login</button>
    </form>
    </div>
    <script src="log.js"></script>
</body>
</html>
```

login.css

```css
*{
    padding: 0;
    margin: 0;
}
body{
    background-size: cover;
    align-items: center;
    justify-content: center;
    display: flex;
    font-family: sans-serif;
}
.container{
```

```css
    position: relative;

    margin-top: 100px;

    width: 450px;

    height: auto;

    background: #dedede;

    border-radius: 5px;

}

.label{

    padding: 20px130px;

    font-size: 35px;

    font-weight: bold;

    color: #130f40;

}

.login_form{

    padding: 20px40px;

}

.login_form.font{

    font-size: 18px;

    color: #130f40;

    margin: 5px0;

}

.login_forminput{

    height: 40px;

    width: 350px;

    padding: 05px;

    font-size: 18px;

    outline: none;

    border: 1pxsolidsilver;

}

.login_form.font2{

    margin-top: 30px;

}

.login_formbutton{

    margin: 45px030px0;
```

```css
    height: 45px;

    width: 365px;

    font-size: 20px;

    color: white;

    outline: none;

    cursor: pointer;

    font-weight: bold;

    background: #1A237E;

    border-radius: 3px;

    border: 1pxsolid#3949AB;

    transition: .5s;

}
.login_formbutton:hover{

    background: #151c6a;

}
.login_form#email_error,
.login_form#pass_error{

    margin-top: 5px;

    width: 345px;

    font-size: 18px;

    color: #C62828;

    background: rgba(255,0,0,0.1);

    text-align: center;

    padding: 5px8px;

    border-radius: 3px;

    border: 1pxsolid#EF9A9A;

    display: none;

}
```

Login.js:

```javascript
//Validtion Code For Inputs


var email = document.forms['form']['email'];
var password = document.forms['form']['password'];


var email_error = document.getElementById('email_error');
var pass_error = document.getElementById('pass_error');


email.addEventListener('textInput', email_Verify);
password.addEventListener('textInput', pass_Verify);


function validated(){
if (email.value.length<9) {
email.style.border = "1px solid red";
email_error.style.display = "block";
email.focus();
return false;
}
if (password.value.length<6) {
password.style.border = "1px solid red";
pass_error.style.display = "block";
password.focus();
return false;
}


}
function email_Verify(){
if (email.value.length>= 8) {
email.style.border = "1px solid silver";
email_error.style.display = "none";
return true;
}
```

```
}
functionpass_Verify(){
if (password.value.length>= 5) {
password.style.border = "1px solid silver";
pass_error.style.display = "none";
returntrue;
}
}
```
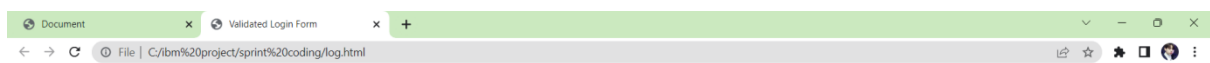

Home code

Home.html:

```
<!DOCTYPEhtml>
<html>
<head>
 <title>Customer service</title>
 <linkrel="stylesheet"type="text/css"href="home.css">
</head>
<!---body section--->
<body>
```

```html
    <div class="container">
      <div class="hello">


      <ul>
          <li><a href="#">HOME</a></li>
          <li><a href="#">NEWS</a></li>
          <li><a href="#">PRODUCT</a></li>
          <li><a href="#">PARTNERS</a></li>
          <li><a href="#">VISIT </a></li>
      </ul>
          </div>
        <hr></hr>


<div class="image">
    <img src="https://www.sender.net/wp-content/uploads/2021/09/SMS-Templates-for-Customer-Service-1-1-1024x658.png">


</div>
<div>
<p class="title">CUSTOMER CARE REGISTRY</p>
<br><br>
<p class="ti">If you need instant support please use <br>live chat we can respond quickly</p>
</div><br><br><br><br><br><br>
<div class="button"><button type="button" value=" Live Chat"><a href="file:///C:/ibm%20project/sprint%20coding/query.html">Live Chat</button></a>
<button type="button" value=" go back"><a href="file:///C:/ibm%20project/sprint%20coding/log.html">Go Back</button></a></div>


</html>
```

Home.css:

```css
*{
margin:0;
padding:0;
}

body{



}
.container{


 background-color: white;
}
.hello{
 width: 400px;
 float: left;
 height: 70px;
 margin-left: 800px;
 margin-top: 40px;
}
ul{

 float:left;
 display:flex;
 justify-content:center;
 align-items: center;
}
ulli{
list-style: none;
margin-left: 62px;
margin-top:27px;
font-size:14px;
```

```css
}
ullia{
 text-decoration: none;
 color: black;
 font-family: arial;
 font-weight: bold;


}
ullia:hover{
color:#ff7200;
}
.image{
  float: right;
}
.diffh3{
  float: "left";
}
.title{
  float: left;
  margin-top: 200px;
  margin-left: 50px;
  font-size: 25px;
  font-family: Times New roman;
}
 .hi{
float: left;
text-decoration: bold;
 margin-left: 200px;
 font-family: Times New roman;
}
.ti{
  float: left;
  margin-left: 100px;
  font-size: 15px;
```

```css
    margin-top: 9px;
    font-family: arial;
   font-weight: bold
}
element.style {
}
style.css:245
.button-two {
    background-color: #f9004d;
    color: white;
    text-decoration: none;
    border: 2pxsolidtransparent;
    font-weight: bold;
    padding: 13px30px;
    border-radius: 30px;
    transition: .4s;
    float: left;
    margin-top: 200px;
    margin-left: 150px;
    font-size: 25px;
}

style.css:1
* {
    padding: 0;
    margin: 0;
    font-family: 'Josefin Sans', sans-serif;
    box-sizing: border-box;
}

user agent stylesheet
a:-webkit-any-link {
    color: -webkit-link;
    cursor: pointer;
```

```css
    text-decoration: underline;
}button{
width: 200px;
padding: 15px;
margin-left: 500px;
text-align: center;
margin: 20px10px;
border-radius: 25px;
font-weight: bold;
border: 2pxsolid#ff7200;
background: #ff7200;
color: white;
cursor: pointer;
overflow: hidden;
}
span{
background: #009688;
height: 100%
width:100%;
border-radius: 25px;
position: absolute;
left: 0;
bottom: 0;
z-index: -1;
}
button:hoverspan{
border: 100%;
}
button:hover{
border: none;
}
a{
text-decoration: none;
```
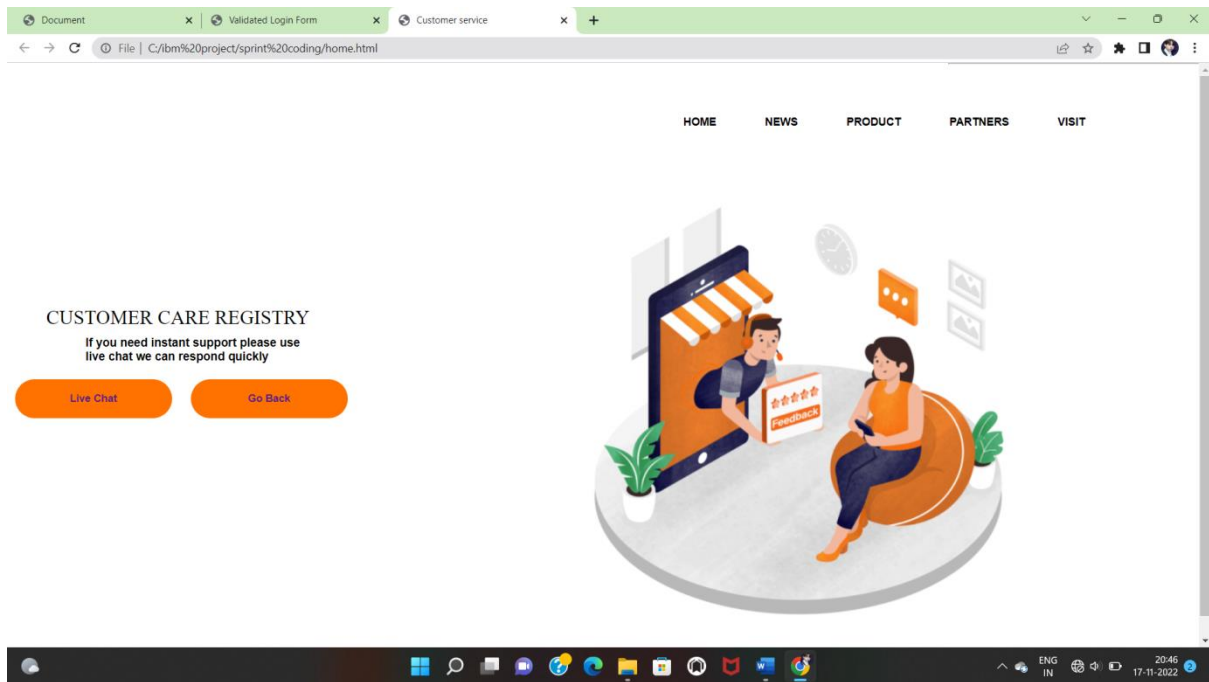
7.2 Feature 2

Query page code

Query.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">


    <!-- custom css file link  -->
    <link rel="stylesheet" href="query.css">


</head>
<body>
<br><br>
<center>
```

```html
<h3class="hello">Please get in touch and our expert support team  will answer all your
questions</h3></center>
<divclass="container">


  <formaction="">


    <divclass="row">


       <divclass="col">



         <divclass="inputBox">
            <span>Full name :</span>
            <inputtype="text"placeholder="full name">
         </div>
         <divclass="inputBox">
            <span>email :</span>
            <inputtype="email"placeholder="example@example.com">
         </div>
         <div>
 <labelfor="company">Company :</label>
<selectid="company">
      <optionvalue="student">Student</option>
<optionvalue="employee">Employee</option>
<optionvalue="others">Others</option>
         </div>
<divclass="inputbox">
 <p>Are you an existing customer?</p>
</div>
          <divclass="inputBox">
             <span>city :</span>
             <inputtype="text">
```

```html
                </div>

                <div class="flex">
                    <div class="inputBox">
                        <span>state :</span>
                        <input type="text" placeholder="india">
                    </div>
                    <div class="inputBox">
                        <span>zip code :</span>
                        <input type="text" placeholder="123 456">
                    </div>
                </div>

            </div>

            <div class="col">


        </form>

</div>
 <a href="file:///C:/ibm%20project/sprint%20coding/success.html"><input type="button" value
="proceed to checkout" class="submit-btn"></a>


</body>
</html>
```

Query.css:

```css
{
  font-family: 'Poppins', sans-serif;
  margin:0; padding:0;
  box-sizing: border-box;
  outline: none; border:none;
```

```css
  text-transform: capitalize;
  transition: all.2slinear;
}
.hello{
  margin-top:100px;
}
.container{
  display: flex;
  justify-content: center;
  align-items: center;
  padding:25px;
  min-height: 10vh;
}

.containerform{
  padding:20px;
  width:700px;
  background: #fff;
  box-shadow: 05px10pxrgba(0,0,0,.1);
}

.containerform.row{
  display: flex;
  flex-wrap: wrap;
  gap:15px;
}

.containerform.row.col{
  flex:11250px;
}

.containerform.row.col.title{
  font-size: 20px;
  color:#333;
```

```css
  padding-bottom: 5px;
  text-transform: uppercase;
}


.containerform.row.col.inputBox{
  margin:15px0;
}


.containerform.row.col.inputBoxspan{
  margin-bottom: 10px;
  display: block;
}


.containerform.row.col.inputBoxinput{
  width: 100%;
  border:1pxsolid#ccc;
  padding:10px15px;
  font-size: 15px;
  text-transform: none;
}

.containerform.row.col.inputBoxinput:focus{
  border:1pxsolid#000;
}

.containerform.row.col.flex{
  display: flex;
  gap:15px;
}


.containerform.row.col.flex.inputBox{
  margin-top: 5px;
}
```

```css
.containerform.row.col.inputBoximg{
  height: 34px;
  margin-top: 5px;
  filter: drop-shadow(001px#000);
}


.containerform.submit-btn{
  width: 100%;
  padding:12px;
  font-size: 17px;
  background: #27ae60;
  color:#fff;
  margin-top: 5px;
  cursor: pointer;
}


.containerform.submit-btn:hover{
  background: #2ecc71;
}
```

Success Page Code

Success.html:

```html
<!DOCTYPE html>
<html>
<head>
<title>
Success Fully..
</title>
</head>
<body>




<style>
    h1{


    margin-top: 60px;
}
  body{


    background-image:
url("https://blog.vantagecircle.com/content/images/size/w1000/2022/09/Employee-
Development.png");



    }


</style>
    <div>


    <center><h1>Your Request done Successfully...</h1></center>
```

```
    </div>
    <divclass="hello">
     <imgsrc="https://blog.vantagecircle.com/content/images/size/w1000/2022/09/Employee-
Development.png"height=100%width=100%>


    </div>
</body>
</html>
```
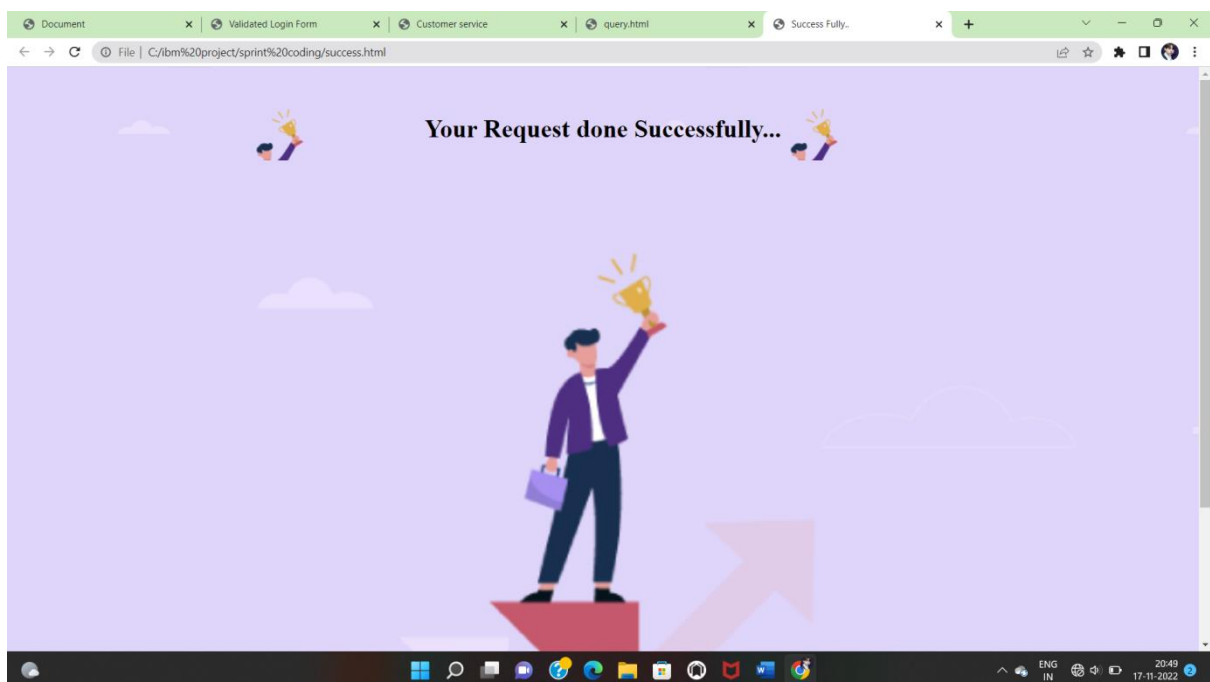
8. TESTING

8.1 Test Cases

A test case is a set of actions performed on a system to determine if it satisfies software requirements and functions correctly. The purpose of a test case is to determine if different features within a system are performing as expected and to confirm that the system satisfies all related standards, guidelines and customer requirements. The process of writing a test case can also help reveal errors or defects within the system. Test cases are typically written by members of the quality assurance (QA) team or the testing team and can be used as step-by-step instructions for each system test. Testing begins once the development team has finished a system feature or set of features. A sequence or collection of test cases is called a test suite. A test case document includes test steps, test data, preconditions and the post conditions that verify requirements.

Why test cases are important:

Test cases define what must be done to test a system, including the steps executed in the system, the input data values that are entered into the system and the results that are expected throughout test case execution. Using test cases allows developers and testers to discover errors that may have occurred during development or defects that were missed during ad hoc tests.

The benefits of an effective test case include:

1.Guaranteed good test coverage.

2.Reduced maintenance and software support costs.

3.Reusable test cases.

4. Confirmation that the software satisfies end-user requirements.

5. Improved quality of software and user experience.

6.Higher quality products lead to more satisfied customers.

7. More satisfied customers will increase company profits.

Overall, writing and using test cases will lead to business optimization. Clients are more satisfied, customer retention increases, the costs of customer service and fixing products

decreases, and more reliable products are produced, which improves the company' s reputation and brand image.

8.2 User Acceptance Testing

User acceptance testing (UAT), also called application testing or end- user testing, is a phase of software development in which the software is tested in the real world by its intended audience. User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done. UAT testing meaning can also be defined as the user methodology where the developed software is tested by the business user to validate if the software is working as per the specifications defined. This type of testing is also known as beta testing, application testing or more commonly end-user testing. The main Purpose of UAT is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is kind of black box testing where two or more end-users will be involved.

UAT is performed by :

☐ Client

☐ End users

Need of User Acceptance Testing:

Need of User Acceptance Testing arises once software has undergone Unit, Integration and System testing because developers might have built software based on requirements document by their own understanding and further required changes during development may not be effectively communicated to them, so for testing whether the final product is accepted by client/end-user, user acceptance testing is needed.

1. Developers code software based on requirements document which is their "own" understanding of the requirements and may not actually be what the client needs from the software.

2. Requirements changes during the course of the project may not be communicated effectively to the developers.

Prerequisites of User Acceptance Testing:

Following are the entry criteria for User Acceptance Testing:

a. Business Requirements must be available.

b. Application Code should be fully developed

c. Unit Testing, Integration Testing & System Testing should be completed

d. No Showstoppers, High, Medium defects in System Integration Test Phase –

e. Only Cosmetic error is acceptable before UAT

f. Regression Testing should be completed with no major defects All the reported defects should be fixed and tested before UAT

g. Traceability matrix for all testing should be completed

h. UAT Environment must be ready

i. Sign off mail or communication from System Testing Team that the system is ready for UAT execution.

9. RESULTS
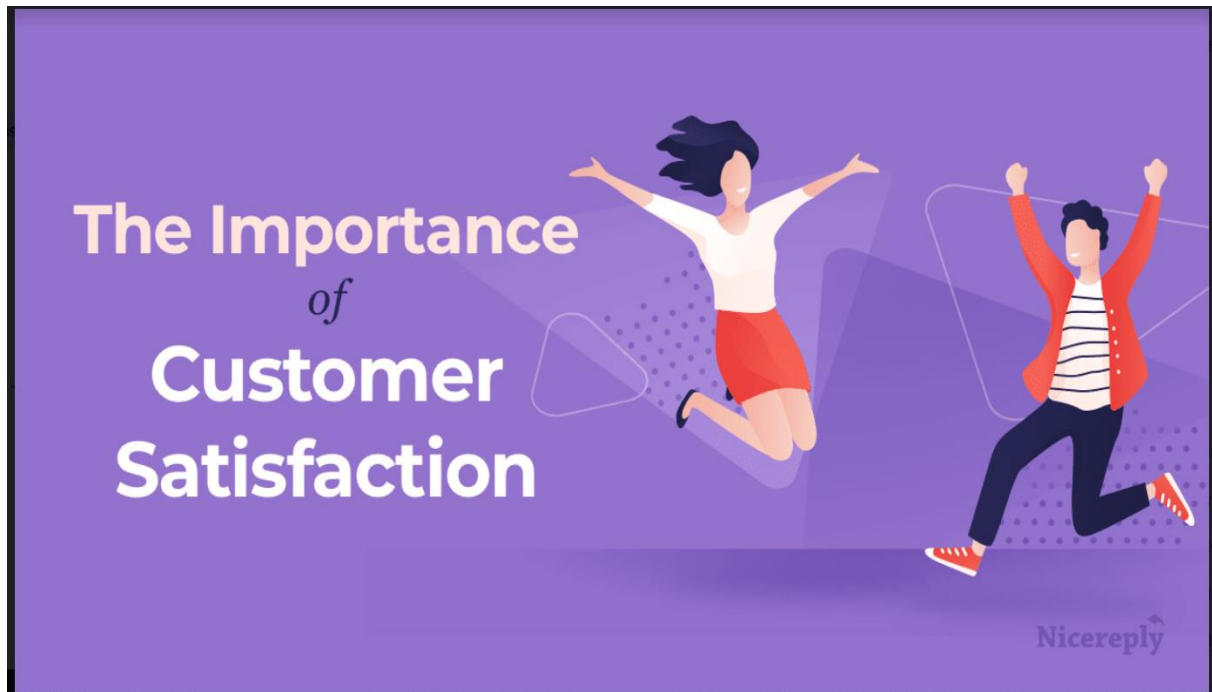
9.1 Performance Metrics

Tracking customer service metrics in 2021 is like monitoring your business' vital signs—its importance cannot be understated. Insights from customer service metrics are invaluable when it comes to improving customer loyalty and your bottom line. But with so many customer service metrics out there, it's hard to know which ones to prioritize to ensure the best customer experience. Our free downloadable guide will help you gain an in-depth

understanding of the most critical metrics and data points. But first, read on for a preview of why these common KPI metrics for customer service matter and what tools you can use to measure them.

Customer relationship metrics

Customer relationship metrics provide insights into buyers' interactions with your company. They help your customer support team gauge customer churn and customer retention. They also show whether buyers are likely to recommend your product or service to others.

Customer satisfaction (CSAT) score

Collect this metric by sending out customer satisfaction surveys that ask customers to rank their recent support experience on a scale of 1 to 5 or to rate it as "good" or "bad." To gain more context about **CSAT scores**, also include open-ended survey questions that prompt customers to give more details about their interaction. For example, you can ask, "In ways did this experience meet your expectations?" to find out what your support team is doing right. You can also state, "Tell us what we can do better" to identify areas of improvement.

Customer Effort Score (CES)

Customer Effort Score evaluates how easy it is for customers to resolve their issue, complete a task, or speak to an agent. The score is collected via surveys that ask customers to rate the ease of their interaction on a scale of "very easy" to "very difficult." Send CES surveys shortly after a customer makes a purchase or engages with a support agent so the experience is still fresh in their mind.

Net Promoter Score℠(NPS)

Net Promoter Score℠(NPS) measures customer loyalty and satisfaction. Track this metric by asking buyers how likely they are to recommend your business to someone else on a scale of 1 to 10.

- Promoters (respondents who answer 9–10) are very satisfied, happy customers and will most likely recommend you to others.

- Passives (respondents who answer 7–8) are satisfied but not likely to recommend your products or services.
- Detractors (respondents who answer 6 and below) are dissatisfied and will not encourage others to patronize your business.

Agent and team performance metrics

Performance metrics track the efficiency and productivity of individual support agents and the team as a whole. Managers can use these insights to identify problems that reduce agents' quality of service and then come up with solutions.

First reply time (FRT)

First reply time, or first response time, captures how long it takes an agent to initially respond to a support request or ticket. Long wait times might indicate that agents are struggling to keep up with a high volume of tickets or that there are slow, inefficient support processes in place. If you provide a customer service-level agreement (SLA), check whether the agreement has a specific first-response time frame. If so, consistently track FRT to ensure you're meeting your SLA requirements.

Ticket reopens

This customer service metric shows how many attempts it takes to solve a customer's problem. A large number of reopens might mean customers have complex support requests, which could point to an issue with your product or service. A high reopen rate could also suggest that agents close tickets before customers' issues are fully resolved. In this situation, customers will likely be unsatisfied with the support they receive and have more questions about their problems.

Resolution time

Resolution time refers to the amount of time it takes for a support agent to solve a problem. Your resolution time really matters—73 percent of customers say fast resolutions are the most important aspect of a good customer service experience. When monitoring this important metric, look for trends across different customer issue types and individual agents. You may find that certain problems are more complicated and require additional time to

resolve. Or, you might see that certain support agents are slower than others and need further training.

Agent touches

This metric measures the number of updates an agent makes to a ticket. A high number of touches indicates that support agents are faced with complex issues that take a lot of time and effort to resolve. It might also point to a problem in your product or gaps in your knowledge base.

Tools for tracking customer service metrics

Manually keeping tabs on every essential customer service metric at once is near impossible. Luckily, there are tools that automate the tracking process for you, so you can easily capture, process, and iterate on customer service performance metrics.

Customer relationship management (CRM) tool

A <u>CRM</u> houses all customer information in one unified place, allowing support agents to quickly access important details and personalize customer interactions. The tool automatically collects and analyzes customer data, too, which saves support teams time and provides them with key insights to deliver better support. Managers can also set goals for agents and track team progress within CRMs like <u>Zendesk</u>. Our software enables you to gather, analyze, and leverage customer support metrics. You can even create custom views using advanced reporting and dashboards to spot trends and improve customer support.

Survey tools

Use survey tools to create, distribute, and analyze customer surveys. These tools reduce errors that often occur during manual survey creation and allow respondents to access surveys from a number of devices—making for a better, easier experience. With an online tool like <u>SurveyMonkey</u>, support teams can gather customer feedback through various channels such as email, social media, and text. SurveyMonkey also provides templates, integrates data with apps in your tech stack, and automates the analysis of survey results.

Social media monitoring tools

These tools track brand mentions, mentions of competitors, and keywords related to your brand on social media. Manually monitoring mentions across numerous social channels is time-consuming, so use social media tools to automate the process. These platforms can also create reports to highlight trends and capture social sentiment, helping you gauge the mood of online conversations. Hootsuite helps brands stay on top of social media conversations by tracking their mentions on all social channels. The software integrates with a review tracking tool, too, so you can monitor opinions from review sites and find positive reviews to share.

10. ADVANTAGES & 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 (source: Salesforce Research). 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.

Disadvantages:

1. Less Quality Control

If you want to establish a trustworthy reputation for your business, you must do everything possible to make sure it operates like a well-oiled machine. This implies that each component must adhere to particular requirements. However, if a customer service department runs remotely, there is no way to guarantee compliance with your policy.

Client service is critical in establishing your company's reputation. It can either enhance or detract from the user experience. That is why a single substandard contact can be so costly.

2. Lack of Brand Loyalty and Possibility of Security Breach

In-house personnel is often considerably more concerned about the success of the business. Customer care organizations that support many businesses concurrently cannot possibly cheer hard for the success of each brand. Lack of loyalty can result in indifference and a decline in the quality of work.

What makes this the most significant point in all of the advantages and disadvantages of outsourcing customer support is that workers often have access to sensitive information. These employees can become a security risk with a lack of brand loyalty.

3. Inadequate Communication Among Departments

One of the most beneficial aspects of receiving customer feedback is bettering your business. Your client care team must know which questions to ask to get accurate customer responses.

Additionally, your team must understand where survey information should be routed to use the feedback correctly. Setting up such a system is tricky when working with a remote team.

4. Offshore Collaboration Challenges

Offshore outsourcing offers the most significant advantages, introducing unexpected complications. While the language barrier and time difference are minor issues, they may significantly influence productivity.

One of the most severe difficulties is legislative inconsistency. To protect yourself, you should consult an international law professional before entering into any contract with an offshore company.

5. The Risk of Disseminating Confidential Corporate Information

Perhaps the most significant downside of outsourcing corporate operations is the danger of losing sensitive data and confidentiality. When outsourcing areas like HR or customer service, the third party is often exposed to personal data that may be readily stolen or transmitted to the wrong hands.

Corporations value intellectual property. Therefore, watertight procedures and checks for data loss and confidentiality protection are essential when negotiating outsourced negotiations.

6. Instability of Outsourcing Firms

If an outsourcing firm fails, so does your organization's whole business department, which might have disastrous consequences. Thus, thorough research is necessary before signing any outsourcing agreement. Reviewing past performance or asking for references and feedback from previous clients is a good practice to avoid an outsourcing firm with a troubled history.

7. Absence of Customer Focus

Outsourced enterprises often operate with a diverse range of customers and cater to the expertise requirements of many organizations at the same time. Outsourced suppliers may quickly lose track and attention, resulting in a lackluster performance on organizational responsibilities.

11.CONCLUSION

Customer service is important to every business! Without customers, no business can survive. Customers will go where they are treated fairly and with respect, and even spend more money at such a business.

All of us have bought something at a fast food restaurant or a retail store where the clerk barely said a word, took our money, shoved our purchase at us and never said anything resembling a thank-you. Such experiences don't inspire anyone to go there again.

All that being said, in a laundry/linen operation, a customer has to be happy with the cleanliness of the goods, prompt delivery, clear billing, polite employees, simple procedures, etc. There are lots of ways that customers interact with a company. All of them should be examined for improvements in customer service.

For example, I once heard an end-user complain bitterly about a linen service, not for quality or pricing, but for the complexity of its invoices! This is something that one wouldn't normally consider a serious issue. Yet, this customer was ready to switch to someone else for this reason. When I pressed them, they admitted everything else was great, but they couldn't get past the complicated invoices and the multiple phone calls it took to clear up problems. In a competitive market, little things can mean the difference between keeping a customer and watching them take their business to your competitor.

A manager should be mindful of every interaction with a customer, including sales calls, product delivery, customer complaints, billing and any problem resolution. Each of these interactions needs to be polite, efficient and convenient to the customer. And for the customer to notice, they need to be consistently pleasing.

When creating ordering or delivery procedures, consider your customer's point of view: Are you only thinking of your own time and effort, or are you trying to make everything simple, easy and pleasant for your customer? One goal could be that a real person answers phone calls; automated switchboards are efficient but serve to annoy the caller most of the time! Do your customers have to fill out forms to make changes, or can they speak to someone and have their problem solved—and solved the first time it's mentioned?

I think the hardest thing is to get all employees to share in this goal. Customer service employees probably understand that being polite and efficient is an important part of their job. But, getting all the employees to consider their actions as directly impacting the success of the business is more than sticking up posters and having nice slogans. This takes effort and diligence all throughout the business.

Spending time with individual employees, showing them by example how to deal with customers, creating an environment where everyone is treated like you want them to treat your customers, and repeating all this over and over will be required.

12. FUTURE SCOPE
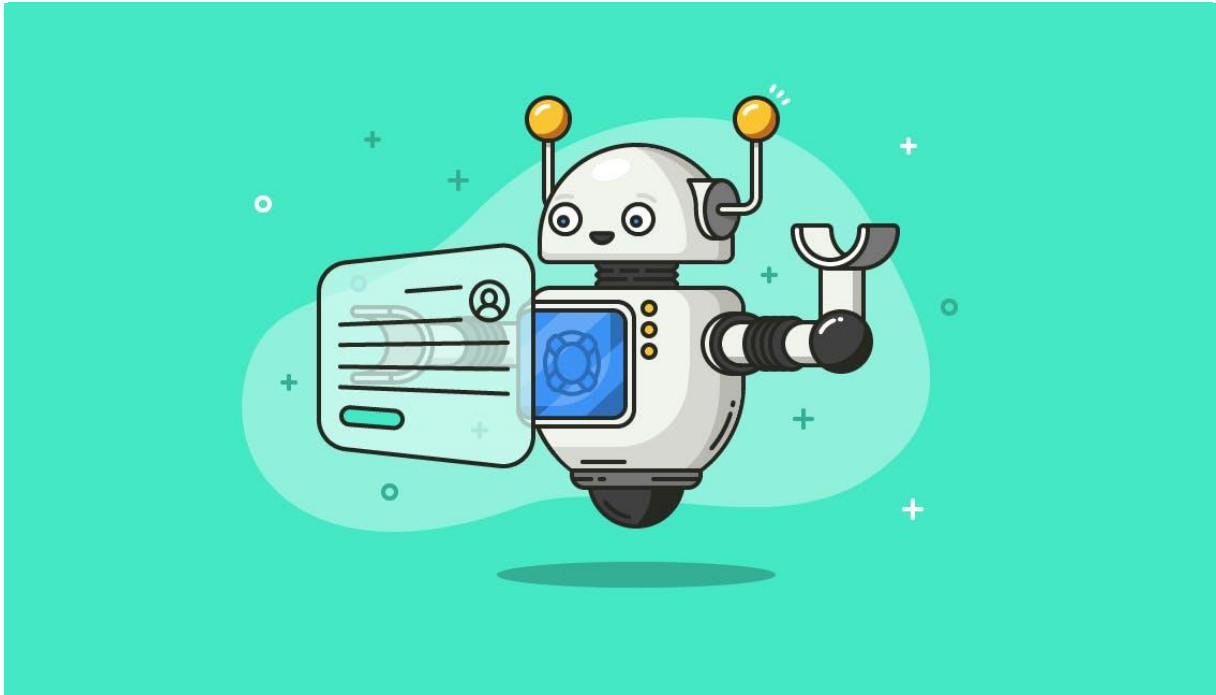
If you ask me about the future of customer service, I'll say it belongs to the companies who view customer support as a competitive advantage rather than a cost center.

In the Support Driven community, we recognize the emergence of <u>customer support as a career</u>, and we're always thinking about ways we can do it better.

We asked a number of industry leaders about what they think is in store for the future of customer service; here's what they told us.

1. Self-service, community-based service, and predictive support

Jay Baer, owner of <u>Convince & Convert</u>, says that there are "three things driving the future of customer support."

- Self-service: Customers want to figure it out themselves, if they can do so without hassle or consequence.

- Community-based service: Customers helping one another figure it out is going to continue as a trend.

- Predictive support: This requires the innovative use of data, but providing "help" to customers before they know they need it is the holy grail of customer support, and it's coming!

2. Personalization

Nick Francis, Co-founder and CEO at Help Scout, says that "As customer support tools get smarter, self-service is going to become a lot more relevant, helpful, and powerful. In most companies, this means many support professionals will spend more of their time doing proactive, success-oriented activities."

"When customers do end up talking with someone, the support team will have everything they need to be helpful in a personalized way: Account information, recent in-app activity, and probably a video of the most recent session where the customer ran into trouble."

"In addition, teams will get assistance from AI-powered suggestions, making it quicker than ever to provide a great experience."

3. Human-to-human connection

Camille E. Acey, VP of Customer Success at Clubhouse, says "I recently had the pleasure of meeting Dr. Taniya Mishra from Affectiva, a company that creates AI for understanding people's emotions through their facial expressions, gestures, and tone of voice."

"While their efforts seem primarily focused on market research at the moment, I see a lot of potential for this sort of technology in customer support."

"Some support desks are indeed asking people to self-report their moods by selecting from a list or clicking on one a few rudimentary cartoon-faced images, but I think most would agree that tapping on a smiley face is a bit primitive and self-reporting can be wildly unreliable."

"If we are truly here to serve the customers, why are we asking them to jump through such annoying additional hoops? It is our job to gauge their mood and — provided they aren't being abusive to us — respond accordingly."

"Support is first and foremost about communication, and human communication has worked in much the same way for far longer than this 'new' digital age. We see a physical presence, we hear a voice, we see a face, we see gestures, and we respond in turn."

"While we as support professionals can benefit immensely from seeing words or images on a screen — especially when trying to tackle a particularly sticky technical challenge — nothing is better than seeing people, hearing the tone of their voice, and making a human connection."

"When we can't be in the same physical space, we are going to have to find ways that our technology can truly bring us the next best thing: a visual experience, enhanced with biometric and behavioral data. The more we know and can glean, the better the user experience."

4. Rising customer expectations

Customer service consultant and speaker Micah Solomon believes that "the only customer service trend you need to be worried about is this: Consciously or unconsciously, customers continue to expect better and better customer service — in every industry, every niche, at every price point."

"These expectations don't come out of the blue. Customers expect you to provide better customer service because they're already getting better customer service elsewhere."

"Whether customer service has been improving in your particular competitive niche or not, it has improved over time at so many companies with such broad consumer reach, including Starbucks, Amazon, Apple, USAA Insurance, Trader Joe's, and Publix, not to mention the great hotels and restaurants that serve so many of your customers every day."

After one of these companies comes into contact with a customer of yours — when USAA expertly assists in filing an insurance claim, or Amazon enables an effortless product return, or a genius at the Apple store debugs an iPhone issue with aplomb — it's inevitable that your customer is going to expect friendlier, speedier, more intuitive service from your company as well."

13.APPENDIX

Source Code

Forget Password .html:

```html
<html>
<head>
<title> Password Matching Validation </title>
<link rel="stylesheet" type="text/css" href="reset.css">
</head>
<script>
function matchPassword() {
  var pw1 = document.getElementById("pswd1");
  var pw2 = document.getElementById("pswd2");
  if(pw1 != pw2)
  {
    alert("Passwords did not match");
  } else {
    alert("Password created successfully");
  }
}
</script>

<body>

<center>
<form>
<div class="container">
<div class="c1">
 <h1> Confirm Password Validation </h1><hr><br><br>
</div>
<!-- Enter Password -->
<div class="c2">
<td><b>Enter Password</b></td>
<input class="hello" type = "password" name = "pswd1"><br><br>
```

```html
</div>
<!-- Enter Confirm password -->
<divclass="c3">
<td><b>Confirm Password</b></td>
<inputclass="hello"type = "password"name = "pswd2"><br><br>
</div><br><br>
<!---Click to validate confirm password -->
<divclass="c4">
<button  type = "submit"onclick="matchPassword()">Submit</button>


<!-- Click to reset fields -->


<buttontype = "reset"value = "Reset">Reset</button>
</div></div>
</div>
</form>
</center>
</body>
</html>
```

Forget Password.css:

```css
.container{

        border :2pxsolid#d9b3ff;
        border-radius:20px;
         margin-top:100px;
         padding:50px;
         margin-left:400px;
         margin-right:400px;
         background-color:#d9b3ff;
         box-shadow:60px1px50pxgray; ;


}
```

```css
h1{
 color:white;
 }
.hello{
 padding:10px;
 border-radius:20px;
  border:3pxsolidwhite;


 }
.c4button{


 padding-top:10px;
 padding-bottom:10px;
 padding-left:40px;
  padding-right:40px;
  border-radius:20px;
  border:3pxsolid#f7e6ff;
  background-color:#f7e6ff;
 }
 .c3{

font-size:20px;

 }
.c2{

font-size:20px;
 }
 button:hover{
  background-color:#b31aff;
  border:3pxsolid#b31aff;


 }
```

User Queries.html:

```html
<html><center>
<head></head>
<body>
<form>
<b><h1>User Queries</b></h1><hr>


<textarearows="5"cols="60"name="comp"placeholder="Type your queries
here..."required></textarea>
<ahref="file:///C:/ibm%20project/implementing%20web%20application/success.html">
<buttontype="submit">request</a></button>
<center><ahref="file:///C:/ibm%20project/implementing%20web%20application/solve.html"
>View Solution</a></center>
</form>
</body>
</center>
</html>
```

Solution Posted Successfully.html:

```html
<!DOCTYPEhtml>
<html>
<head>
<title>
Success Fully..
</title>
</head>
<body>




<style>
```

```
    h1{

    margin-top: 60px;
}
  body{

    background-image:
url("https://blog.vantagecircle.com/content/images/size/w1000/2022/09/Employee-
Development.png");



    }

</style>
    <div>


    <center><h1>Solution Posted  Successfully...</h1></center>

    </div>
    <divclass="hello">
    <imgsrc="https://blog.vantagecircle.com/content/images/size/w1000/2022/09/Employee-
Development.png"height=100%width=100%>


    </div>
</body>
</html>
```

Agent Solution.html:

```html
<html><center>
<head></head>
<body>
<form>
<b><h1>solutions</b></h1><hr>


<textarearows="5"cols="60"name="comp"placeholder="Type your solution
here..."required></textarea>
<ahref="file:///C:/ibm%20project/post.html">
<buttontype="submit">Post</a></button>
</form>
</body>
</center>
</html>
```

Solution Viewing Page.html:

```html
<html>
<head>
<title>queries</title>
</head>
<body>
<b><h1>Queries and Solution</h1><hr>
<h2> Here is your problem solution<h2>
  <h3>using forget password page you can change and reset your password....Then you can
try again to login<br>...If again you have any problem ..u can make query</h3>
<center><h3>Thank You for Using Our Service!!!<h3></center>
</body>
</html>
```

Demo Link --

https://drive.google.com/drive/folders/1xTghl7w6ysk1WMLFTqBTU1Yf5C-pGRV-

Gitup--

https://github.com/IBM-EPBL/IBM-Project-49305-1660817625