

Project Report - TMID – PNT2022TMID14584
Customer Care Registry

1. INTRODUCTION

1.1. Project Overview:

Customer care registry is an application to help customers in processing their complaints. Customers can raise tickets with the description of the problem. An agent will be assigned to the ticket and will be tasked to solve the customer's problem.

1.2. Purpose:

Companies today want to satisfy their customers so that customers would be more inclined to visit their company for services next. For those companies that prioritize customer satisfaction would prefer to use a customer care registry to keep track of complaints and solve it.

2. LITERATURE SURVEY

2.1 Existing problem

Customer support has been a separate department in almost any big company. But for small companies the amount of queries that they receive can overwhelm them. An application that can keep track of customers' requests will be helpful solve most of the requests raised by customers.

2.2 References:

2.2.1. CUSTOMER CARE AND COMPLAINT REGISTRATION MANAGEMENT SYSTEM

2.2.2. NEED FOR SYSTEM- COMPLAINT REGISTRATION MANAGEMENT SYSTEM

2.2.3. CUSTOMER CARE AND COMPLAINT REGISTRATION MANAGEMENT SYSTEM PROJECT REPORT

2.2.4. Online Customer Care and Service Center Project in Java

2.2.5. Pooria Rashvand and Muhd Zaimi Abd Majid (2014) Critical Criteria on Client and Customer Satisfaction for the Issue of Performance Measurement.

2.2.6. Zeljko M. Torbica and Robert C. Stroh(2001) Customer Satisfaction in Home Building.

2.2.7. Natalia Yanovaa(2015) Assessment of Satisfaction with the Quality of Education: Customer Satisfaction Index.

2.2.8. Jam Shahzaib Khan and Salim Khoso (2014) Strategic Management for Customer Satisfaction within Construction Projects.

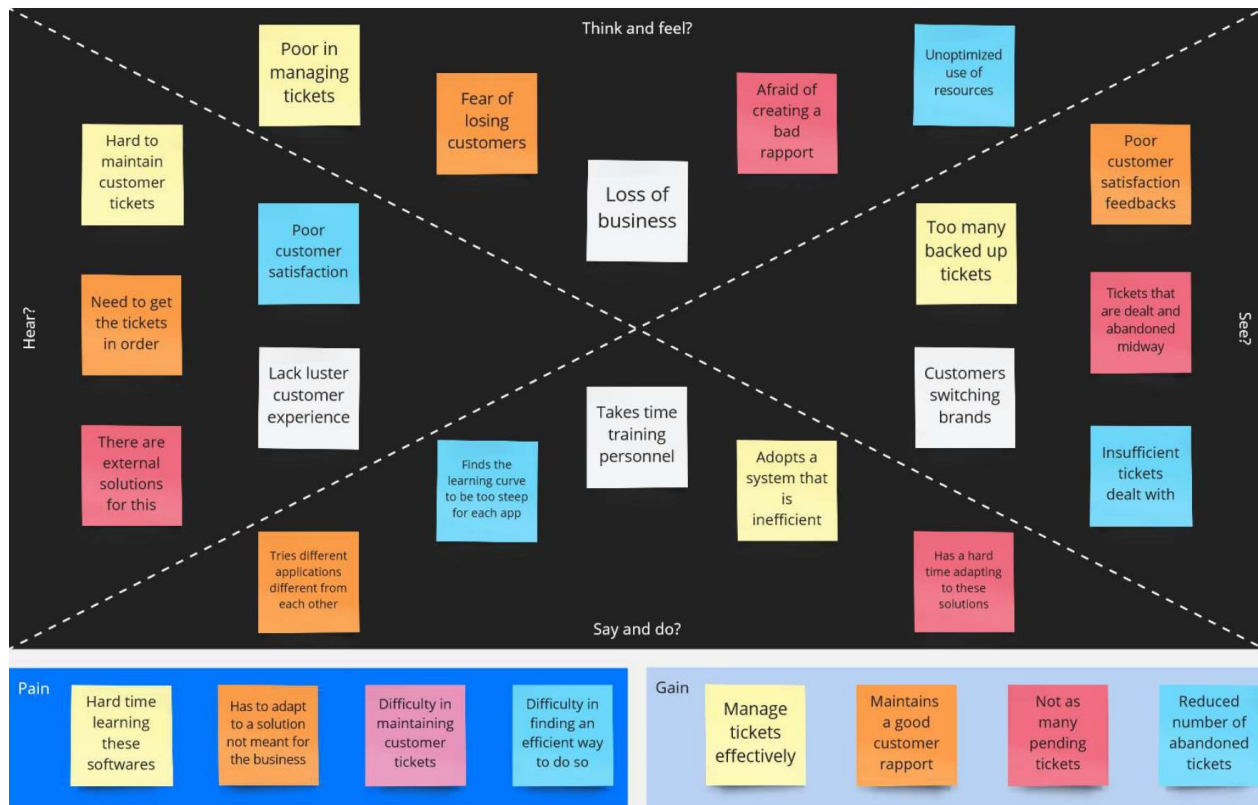
- 2.2.9.** Samira Dezhkam, Farzaneh Ghanbarpoor, Nazanin Shabani, Forooz Pishgar and Mahboobeh Ashoori (2013) The Impact of Product Innovation on Customer Satisfaction and Customer Loyalty.
- 2.2.10.** Tunde Akinola Folorunso and Oluwaseyi Alabi Awodele(2015) Assessment of Clients Needs and Satisfaction at Various Stages of Building Projects Delivery
- 2.2.11.** Ghaleb J. Sweis, Rana M. Imam, Ghaith M. Kassab and Rateb Sweis (2013) Customer Satisfaction in Apartment Buildings: The Case of Jordan.
- 2.2.12.** Abayomi Omonori and Akinloye Lawal (2014) Understanding Customers Satisfaction in Construction Industry in Nigeria.
- 2.2.13.** Abdulhamid Shebob, Amit Mhalas, and Raj Kapur Shah (2013) A Review of Customer Satisfaction Factors in Libyan Housing Projects.

2.3. Problem Statement Definition

Customer Care Registry provides efficient support to the customers in solving their problems or queries. This Application has been developed to help the companies in processing their complaints put forth by the customers. The customers can raise the ticket with a detailed description of the issue. An Agent will be assigned to the Customer to solve the problem. Whenever the agent is assigned to a customer, they will be notified with an email alert. Customers can view the status of the ticket till the service is provided. Companies today wants to satisfy their customers so that customers would be more inclined to visit their company for services next. For those companies that prioritizes customer satisfaction would prefer to use a customer care registry to keep track of complaints and solve it.

3. IDEATION & PROPOSED SOLUTION

3.1. Empathy Map Canvas



3.2. Ideation & Brainstorming

Reference:

<https://app.mural.co/t/kumaraguru8114/m/kumaraguru8114/1667246740963/7df12b9a1a7600a4e328faaa886f118f89e4339f?sender=udd2bbcd686c2cd4fefc55238>

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

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.

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

Before you collaborate

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

10 minutes

A

Team gathering

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

B

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

C

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) →

1

Define your problem statement

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

5 minutes

PROBLEM

How might we help the customers in processing their complaint using a cloud application?

Key rules of brainstorming

To run a smooth and productive session

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

Need some inspiration?

See a finished version of this template to kickstart your work.

[Open example](#) →

Share template feedback

Step-2: Brainstorm, Idea Listing and Grouping

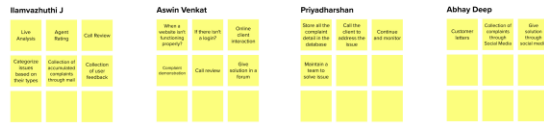
2

Brainstorm

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

10 minutes

TIP
You can select a sticky note and hit the pencil (switch to sketch) icon to start drawing!



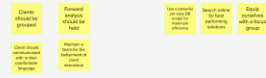
3

Group ideas

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

20 minutes

Customer communication Learning (to learn about)



Preprocessing



UI/UX



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



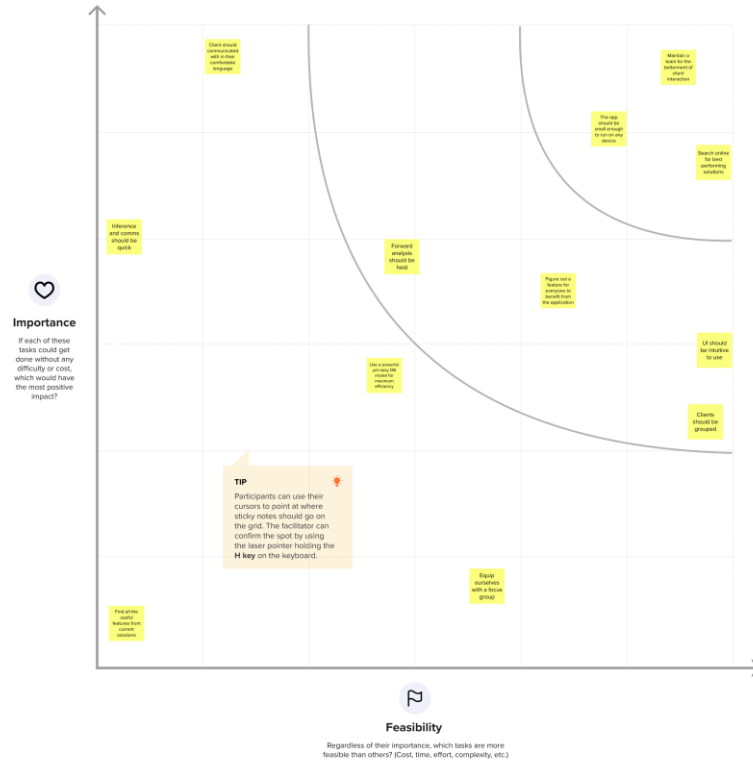
Step-3: Idea Prioritization:

4

Prioritize

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

🕒 20 minutes



➔

After you collaborate

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

Quick add-ons

A

Share the mural

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

B

Export the mural

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

Keep moving forward



Strategy blueprint

Define the components of a new idea or strategy.

[Open the template →](#)



Customer experience journey map

Understand customer needs, motivations, and obstacles for an experience.

[Open the template →](#)



Strengths, weaknesses, opportunities & threats

Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.

[Open the template →](#)

[Share template feedback](#)



3.3. Proposed Solution:

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Customer care service plays a big part in any company's business. They need to satisfy their customers by responding to their needs and queries. But the sheer number of queries generated makes it difficult for the staffs to respond to all the queries produced by several customers.
2.	Idea / Solution description	To create a customer care registry that keeps track of very query until they are responded by a staff.

- | | | |
|----|---------------------------------------|---|
| 3. | Novelty / Uniqueness | This is not a particularly unique idea as many people tried to develop it. Maybe not the exact same idea. |
| 4. | Social Impact / Customer Satisfaction | The company will be satisfied if they can satisfy most of their customers through their customer care service. |
| 5. | Business Model (Revenue Model) | We could provide this in a service based model and can charge the companies for maintaining their customer care registries. |
| 6. | Scalability of the Solution | The solution can be uniquely tailored to each company and can support the growing customer base of the company. |

3.4. Problem Solution fit:

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) <small>Who is your customer? i.e. working parents of 0-5 y.o. kids</small>	6. CUSTOMER CONSTRAINTS <small>What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.</small>	5. AVAILABLE SOLUTIONS <small>Which solutions are available to the customers when they face the or need to get the job done? What have they tried in the past? What pros &</small>	Explore AS, differentiate
	<ul style="list-style-type: none"> Customers/ companies that need help to keep track of the complaints they face. Companies that wants to provide good customer service to all customers 	<ul style="list-style-type: none"> Customer needs constant internet connection to operate the solution. 	<p>One available solution is automated call service where customers can call and can explain their problem briefly and wait for agent to be assigned to them</p>	
Focus on J&P, tap into BE, understand RC	2. JOBS-TO-BE-DONE / PROBLEMS <small>Which jobs-to-be-done (or problems) do you address for your</small>	9. PROBLEM ROOT CAUSE <small>What is the real reason that this problem exists? What is the back story behind the need</small>	7. BEHAVIOUR <small>What does your customer do to address this problem, and not this i.e. directly related: find the right solar panel installer, calculate usage and</small>	Focus on J&P, tap into BE, understand RC
	<ul style="list-style-type: none"> Allowing customers to track the complaints better. Assign agents to queries that are raised up. 	<ul style="list-style-type: none"> The number of queries that are generated are outrageously high. So it becomes difficult to track and respond to each queries 	<ul style="list-style-type: none"> To keep track of the queries until they are responded If not look into it 	

Identify strong TR & EM	3. TRIGGERS <small>What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.</small>	10. YOUR SOLUTION <small>If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.</small>	8. CHANNELS of BEHAVIOUR <small>8.1 ONLINE What kind of actions do customers take online? Extract online channels from #7</small>
	<p>Amount of queries and the inability to respond every query.</p>	<p>Our solution is to create a registry to track of the queries until an agent is assigned and the query is resolved.</p>	<p>They can upload the data to cloud.</p>
Identify strong TR & EM	4. EMOTIONS: BEFORE / AFTER <small>How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.</small>		8.2 OFFLINE <small>What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.</small>
	<p>Customer can feel worried and unsatisfied.</p>		<p>Make sure to find the best solutions to get the queries resolved.</p>

4. REQUIREMENT ANALYSIS

4.1. Functional Requirements:

Following are the functional requirements of the proposed solution.

Fr NO	Functional Requirements(Epic)	Sub Requirements(Story/Sub-Task)
FR-1.	User Registration	Registration through From Registration through Gmail Registration through Google
FR-2.	User Configuration	Confirmation via Email Confirmation via OTP
FR-3.	User Login	Login via Google Login with Email id and Password
FR-4.	Admin Login	Login via Google Login with Email id and Password
FR-5.	Query Form	Description of the issues Contact information
FR-6.	E-mail	Login alertness
FR-7.	Feedback	Customer feedback

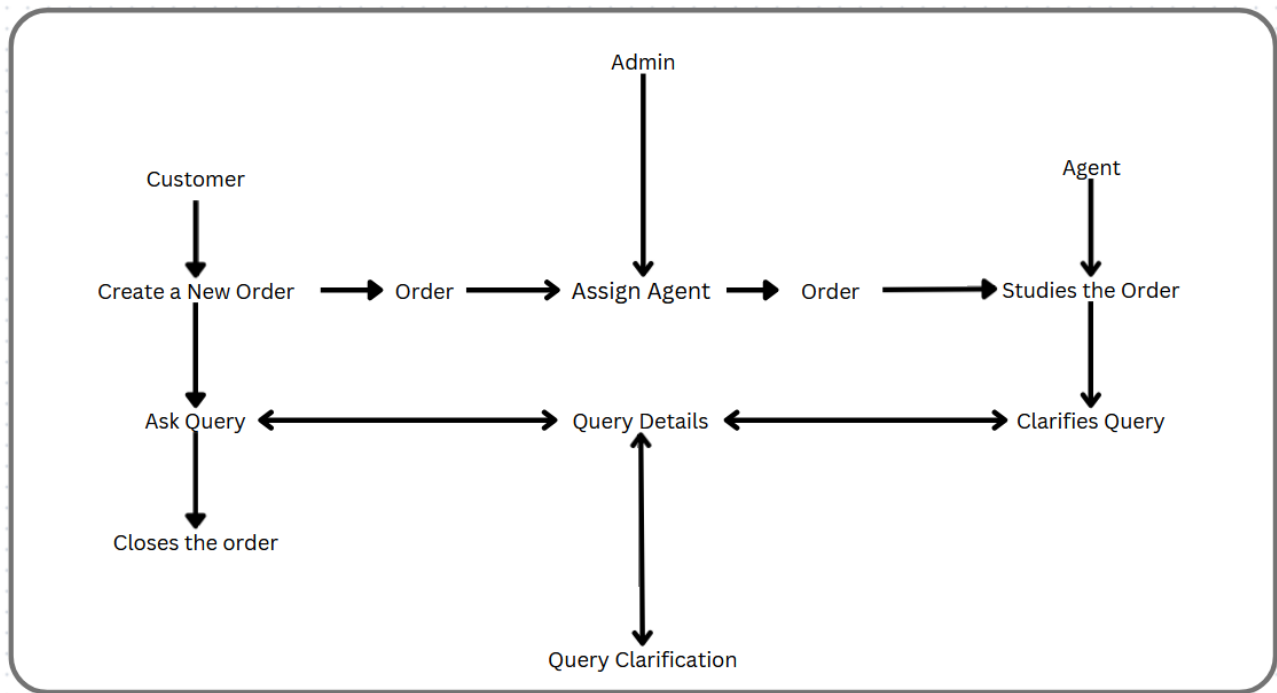
4.2. Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

Fr NO	Non-Functional Requirements(Epic)	Sub Requirements(Story/Sub-Task)
NFR-1.	Usability	To provide the solution to the problem
NFR-2.	Security	Track of login authentication
NFR-3.	Reliability	Tracking of decade status through Email
NFR-4.	Performance	Effective Development of Web Application
NFR-5.	Availability	24/7 service
NFR-6.	Scalability	Agents scalability as per the number of customers

5. PROJECT DESIGN

5.1. Data Flow Diagrams



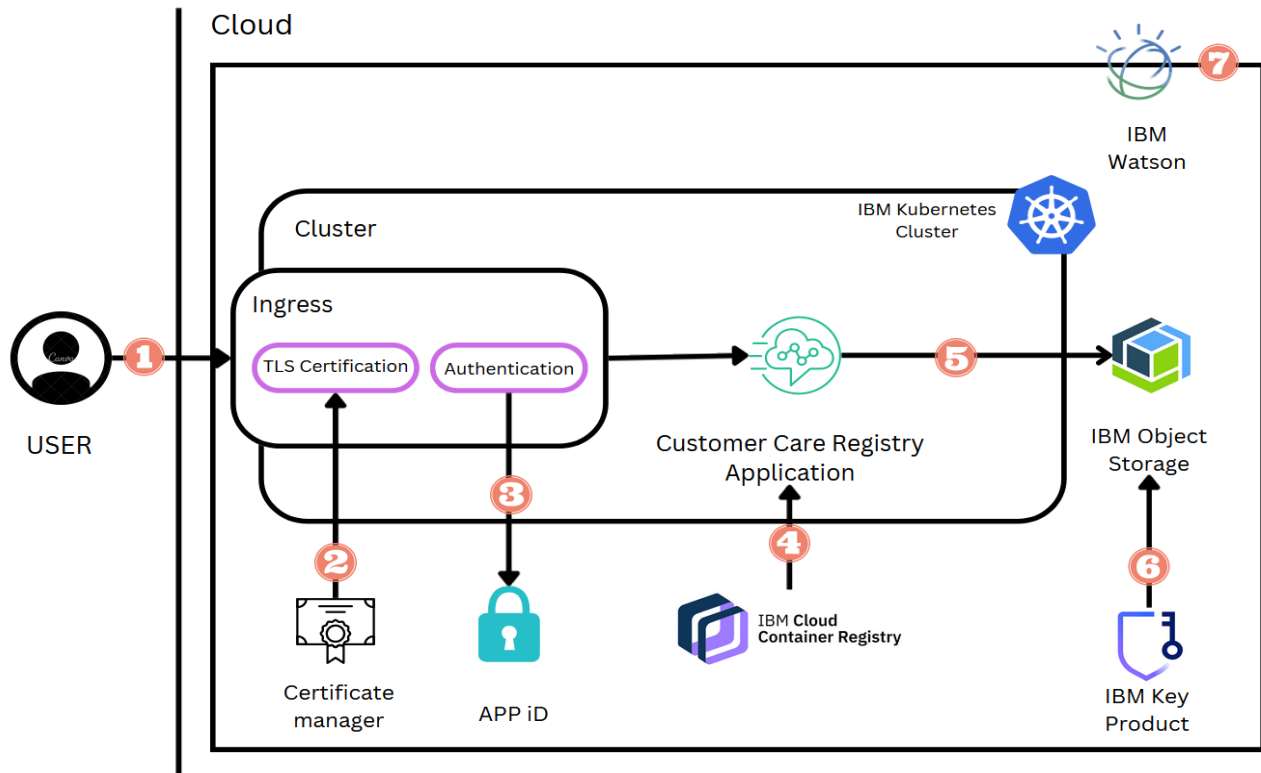
5.2. Solution & Technical Architecture

5.2.1. Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed and delivered.

5.2.2. Technical Architecture:



5.3. User Stories:

User Type	Functional Requirement (Epic)	User Story NO	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 log into 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	USN-4	As a customer, I can place	I can ask my	Medium	Sprint-2

	creation		my order with the detailed description of my query	query		
	Address Column	USN-5	As a customer I can have conversation with the assigned agent and get my query classified.	My queries are clarified.	High	Sprint-3
	Forgot password	USN-6	As a customer, 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
	Order details	USN-7	As a Customer, I can see the current stats of order.	I get a better 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/her doubts	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
	Login	USN-1	As an admin, I can login to the application 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. Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task
Sprint-1	User Panel	USN-1	The user will login into the website and go through the services available on the web page
Sprint-2	Admin panel	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
Sprint-3	Chat Bot	USN-3	The user can directly talk to Chat bot regarding the services. Get the recommendations based on information provided by the user.
Sprint-4	final delivery	USN-4	Container of applications using docker kubernetes and deployment the application. Create the documentation and final submit the application

6.2. Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Total Story Points
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	20
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	20
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	20
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	20

7. CODING & SOLUTION (Explain the features added in the project along with code)

7.1 Feature 1:

Different Types of Users are able to login/register from the same webpage.

Code:

```
<div class="content-section">
  <form method="POST" action="">
    {{ form.hidden_tag() }}
    <fieldset class="form-group">
      <legend class="border-bottom mb-4">Log In</legend>
```

```

<div class="form-group">
  {{ form.email.label(class="form-control-label") }}
  {% if form.email.errors %}
    {{ form.email(class="form-control form-control-lg is-invalid") }}
    <div class="invalid-feedback">
      {% for error in form.email.errors %}
        <span>{{ error }}</span>
      {% endfor %}
    </div>
  {% else %}
    {{ form.email(class="form-control form-control-lg") }}
  {% endif %}
</div>
<div class="form-group">
  {{ form.password.label(class="form-control-label") }}
  {% if form.password.errors %}
    {{ form.password(class="form-control form-control-lg is-invalid") }}
    <div class="invalid-feedback">
      {% for error in form.password.errors %}
        <span>{{ error }}</span>
      {% endfor %}
    </div>
  {% else %}
    {{ form.password(class="form-control form-control-lg") }}
  {% endif %}
</div>
<div class="form-check">
  {{ form.remember(class="form-check-input") }}
  {{ form.remember.label(class="form-check-label") }}
</div>
</fieldset>
<div class="form-group">
  {{ form.submit(class="btn btn-outline-info") }}
  <small class="text-muted ml-2">
    <a href="{{ url_for('reset_request') }}">Forgot Password?</a>
  </small>
</div>
</form>
</div>

```

7.2 Feature 2:

Different Users are directed to their respective dynamic home pages base on their user role.

Code:

```

<div class="collapse navbar-collapse" id="navbarNav">
  <ul class="navbar-nav ml-auto">
    {% if current_user.is_authenticated %}
      <li class="nav-item active">

```

```

        <a class="nav-link" href="{ {url_for('home')}} ">Home</a>
    </li>
    <li>
        <a class="nav-link" href="{ {url_for('account')}} ">Account</a>
    </li>
    <li>
        <a class="nav-link" href="{ {url_for('new_ticket')}} ">New Ticket</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="{ {url_for('logout')}} ">Logout</a>
    </li>
    {% else %}
    <li class="nav-item">
        <a class="nav-link" href="{ {url_for('index')}} ">About us</a>
    </li>
    <li>
        <a class="nav-link" href="{ {url_for('register')}} ">Register</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="{ {url_for('login')}} ">Login</a>
    </li>
    {% endif %}
</ul>
</div>

```

7.3 Feature 3:

The webpages for Ticket Creation, Updating Tickets and Assigning Agents to tickets are only available to the Users with respective User roles.

Code:

```

{% if current_user.userrole == 'Admin'%}
    <div>
        <a class="btn btn-secondary btn-sm mt-1 mb-1" href="{ {url_for('assign_ticket', ticket_id =
ticket.id )} } ">Assign Agent</a>
        <button type="button"
            class="btn btn-danger btn-sm m-1" data-toggle='modal' data-
target='#deleteModal'>Delete</button>
    </div>
    {% endif %}
    {% if current_user.userrole == 'Agent' %}
    <div>
        <a class="btn btn-secondary btn-sm mt-1 mb-1" href="{ {url_for('update_ticket', ticket_id =
ticket.id )} } ">Update</a>
        <button type="button"
            class="btn btn-danger btn-sm m-1" data-toggle='modal' data-
target='#deleteModal'>Delete</button>
    </div>
    {% endif %}

```

7.4 Feature 4:

The Author/Customer receives a ticket whenever a ticket is updated based on its status or based on its completion.

Code:

```
def ticket_status_email(ticket, user):
    message = Mail(
        from_email='ilamvazhuthi.j@gmail.com',
        to_emails= user.email,
        subject = 'Ticket Status Update',
        html_content='<p>The status of your ticket has been updated to<br>{ }<br>This is an automated
message.<br><br><br>For any queries contact our customer support</p>'.format(ticket.status))
    try:
        sg =
SendGridAPIClient("SG.Oyc8TVoqRKKdZEg_hbEI3A._bX8I1Wklmlhw1Vkw62KACRu2kkEYePEJBNvM1FZ
Jy8")
        response = sg.send(message)
        print(response.status_code)
        print(response.body)
        print(response.headers)
    except Exception as e:
        print(e)
```

7.5 Feature 5:

Users are able to reset their password only with their registered Mail_ID and this mail is verified using Serializers.

Code:

```
def send_reset_email(user):
    token = user.get_reset_token()
    message = Mail(
        from_email='ilamvazhuthi.j@gmail.com',
        to_emails= user.email,
        subject='Password Reset Request',
        html_content=f'"To reset your password, visit the following link:
{url_for('reset_token', token = token, _external = True)}"
If you did not make this request then simply ignore ths email and no changes will be made
"'
    try:
        sg =
SendGridAPIClient("SG.Oyc8TVoqRKKdZEg_hbEI3A._bX8I1Wklmlhw1Vkw62KACRu2kkEYePEJBNvM1FZ
Jy8")
        response = sg.send(message)
        print(response.status_code)
        print(response.body)
        print(response.headers)
```

```
except Exception as e:
    print(e)
```

7.6 Feature 6:

The entire application is base out of a single layout.html such that new pages can be created with ease.

Code:

```
{% extends "layout.html" %}
{% block body %}
{Content for the Current Page}
{%endblock body%}
```

7.7 Feature 7:

The Application is featured in such a way that even if the browser is able to cache any page of the application no data of significance is actually revealed.

Code:

```
@app.route("/index")
def index():
    if current_user.is_authenticated:
        return redirect(url_for('home'))
    return render_template('index.html')
@app.route("/logout")
def logout():
    logout_user()
    return redirect(url_for('index'))
```

7.8 Database Schema:

```
class User(db.Model, UserMixin):
    id = db.Column(db.Integer, primary_key = True)
    username = db.Column(db.String(20), unique = True, nullable = False)
    userrole = db.Column(db.String(20), unique = False, nullable = False)
    email = db.Column(db.String(120), unique = True, nullable = False)
    image_file = db.Column(db.String(120), nullable = True, default = 'default.jpg')
    password = db.Column(db.String(60), nullable = False)
    ticket = db.relationship("Ticket", backref='author', lazy = True)

class Ticket(db.Model):
    id = db.Column(db.Integer, primary_key = True)
    title = db.Column(db.String(100), nullable = False)
```



```

status = db.Column(db.String(20), unique = False, nullable = True)

date_posted = db.Column(db.DateTime, nullable = False, default = datetime.utcnow)

content = db.Column(db.Text, nullable = False)

user_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable = False)

agent_id = db.Column(db.Integer, nullable = True)

```

8. TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result
001	Function al	User	User is able to create a user account	An existing account with the same username and password does not exist	Index- > Register	Username= Testuser Password = 123	Account Created Successfully popup
002	Function al	Agent	Agent is able to create a agent account	An existing account with the same username and password does not exist	Index- > Register	Username= Testagent Password = 123	Account Created Successfully popup
003	Function al	Admin	Admin is able to create a admin account	An existing account with the same username and password does not exist	Index- > Register	Username= Testadmin Password = 123	Account Created Successfully popup
004	Function al	User	User is able to login to the user designated page	Has already created an account	Index->Login	Username= Testuser Password = 123	Redirects to Home
005	Function al	Agent	Agent is able to login to the agent designated page	Has already created an account	Index->Login	Username= Testagent Password = 123	Redirects to Home
006	Function al	Admin	Admin is able to login to the admin designated page	Has already created an account	Index- > Login	Username= Testadmin Password = 123	Redirects to Home
007	Function al	User	Able to create a new ticket	Logged in to home	Home-> New Ticket	Ticket title=trial Ticket Content=trial	Ticket Created Successfully popup
008	Function al	Agent	Able to create a new ticket	Logged in to home	Home-> New Ticket	Ticket title=trial Ticket Conteny=trial	Ticket Created Successfully popup
009	Function al	Admin	Able to create a new ticket	Logged in to home	Home-> New Ticket	Ticket title=trial Ticket Content=trial	Ticket Created Successfully popup
010	Function al	User	Able to view the created ticket	Logged in to home	Home	Ticket title=trial Ticket Content=trial	Ticket is in home

011	Function Admin al	Able to view the created ticket	Logged in to home	Home	Ticket title=trial Ticket Content=trial	Ticket is in home
012	Function Admin al	Able to assign a ticket to a agent	Able to access the created ticket	Home-> ticket	Agent id, Ticket status	Ticket assigned successfully popup
013	Function Agent al	Able to view the assigned ticket	Logged in to home	Home	Ticket title=trial Ticket Content=trial	Ticket is in home
014	Function Agent al	Able to update the assigned ticket	Logged in to home	Home-> ticket	Ticket title=trial Ticket Content=trial	Ticket Updated Successfully pop-up
015	Function User al	Receives email about the status of the ticket	Logged into the registered mail id	Inbox	Ticket title=trial Ticket Content=trial	Receives mail regarding the ticket
016	Function Admin al	Deleting a completed Ticket	Able to access the ticketHome-> ticket	Home-> ticket	Ticket title=trial Ticket Content=trial	Ticket deleted successfully popup
017	Function Agent al	Deleting a completed Ticket	Able to access the ticket	Home-> ticket	Ticket title=trial Ticket Content=trial	Ticket deleted successfully popup
018	Function User al	Logging out of their account	Logged in to home	Home-> Logout	Username= Testuser Password = 123	Logged out successfully popup
019	Function Agent al	Logging out of their account	Logged in to home	Home-> Logout	Username= Testagent Password = 123	successfully popup
020	Function Admin al	Logging out of their account	Logged in to home	Home-> Logout	Username= Testadmin Password = 123	successfully popup
021	Function User al	Forgot password	Logged out of their account	Index-> Login-> Forget Password	Username= Testuser Password = 123	Password reset email received successfully
022	Function Agent al	Forgot password	Logged out of their account	Index-> Login-> Forget Password	Username= Testagent Password = 123	Password reset email received successfully
023	Function Admin al	Forgot password	Logged out of their account	Index-> Login-> Forget Password	Username= Testadmin Password = 123	Password reset email received successfully
024	Function User al	Forgot password	Sent reset email 30 mins ago	Index-> Login-> Forget Password	Username= Testuser Password = 123	Token invalid due to time limit
025	Function Agent al	Forgot password	Sent reset email 30 mins ago	Index-> Login-> Forget Password	Username= Testagent Password = 123	Token invalid due to time limit
026	Function Admin al	Forgot password	Sent reset email 30 mins ago	Index-> Login-> Forget Password	Username= Testadmin Password = 123	Token invalid due to time limit

8.2 User Acceptance Testing

Tes	Feature	Compo	Test	Pre-	Steps To	Test	Expecte	Actual	Status	Comment	TC for	BUG	Executed
-----	---------	-------	------	------	----------	------	---------	--------	--------	---------	--------	-----	----------

Case ID	Test Type	Test Scenario	Requisite	Execute	Data	Expected Result	Actual Result	Status	Automation (Y/N)	ID	By
001	Functional	User is able to create a user account	An existing account with the same username and password does not exist	Index -> Register	Username= Testuser Password = 123	Account Created Account Created	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
002	Functional	Agent is able to create a agent account	An existing account with the same username and password does not exist	Index -> Register	Username= Testagent Password = 123	Account Created Account Created	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
003	Functional	Admin is able to create a admin account	An existing account with the same username and password does not exist	Index -> Register	Username= Testadmin Password = 123	Account Created Account Created	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
004	Functional	User is able to login to the user designated page	Has already created an account	Index -> Login	Username= Testuser Password = 123	Redirect s to Home Redirect s to Home	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
005	Functional	Agent is able to login to the agent designated page	Has already created an account	Index -> Login	Username= Testagent Password = 123	Redirect s to Home Redirect s to Home	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
006	Functional	Admin is able to login to the admin designated page	Has already created an account	Index -> Login	Username= Testadmin Password = 123	Redirect s to Home Redirect s to Home	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
007	Functional	User is able to create a new ticket	Logged in to home	Home -> New Ticket	Ticket title=trial Ticket Content=trial	Ticket Created Ticket Created	PASS Successf ully popup	Successful	N		Ilamvazhuthi J
008	Functional	Agent is able to create a new ticket	Logged in to home	Home -> New Ticket	Ticket title=trial Ticket Content=trial	Ticket Created Ticket Created	PASS Successf ully popup	Successful	N		Ilamvazhuthi J

						Content	popup	popup				
009	Functional	Admin	Able to create a new ticket	Logged in to home	Home-> New Ticket	Ticket title=trial	Ticket Created Successfully	Ticket Created Successfully	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content						
						=trial						
010	Functional	User	Able to view the created ticket	Logged in to home	Home	Ticket title=trial	Ticket is in home	Ticket is in home	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content						
						=trial						
011	Functional	Admin	Able to view the created ticket	Logged in to home	Home	Ticket title=trial	Ticket is in home	Ticket is in home	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content						
						=trial						
012	Functional	Admin	Able to assign a ticket to an agent	Able to access the created ticket	Home-> ticket	Agent id, Ticket status	Ticket assigned successfully	Ticket assigned successfully	PASS	Successful	N	Ilamvazhuthi J
							popup	popup				
013	Functional	Agent	Able to view the assigned ticket	Logged in to home	Home	Ticket title=trial	Ticket is in home	Ticket is in home	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content						
						=trial						
014	Functional	Agent	Able to update the assigned ticket	Logged in to home	Home-> ticket	Ticket title=trial	Ticket Updated Successfully	Ticket Updated Successfully	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content	popup	popup				
						=trial						
015	Functional	User	Receives email about the status of the ticket	Logged into the registered mail id	Inbox	Ticket title=trial	Receives mail regarding the ticket	Receives mail regarding the ticket	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content						
						=trial						
016	Functional	Admin	Deleting a completed Ticket	Able to access the ticketHome-> ticket	Home-> ticket	Ticket title=trial	Ticket deleted successfully	Ticket deleted successfully	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content	popup	popup				
						=trial						
017	Functional	Agent	Deleting a completed Ticket	Able to access the ticket	Home-> ticket	Ticket title=trial	Ticket deleted successfully	Ticket deleted successfully	PASS	Successful	N	Ilamvazhuthi J
						Ticket Content	popup	popup				
						=trial						
018	Functional	User	Logging out of their	Logged in to home	Home-> Logout	Username=Testuser	Logged out successfully	Logged out successfully	PASS	Successful	N	Ilamvazhuthi J
							popup					

		account			Passwor	ully					
01	Function	Agent	Logging out of their account	Logged in to home	Home-> Logout	Userna me= Testage nt	successf uly popup	successf uly popup	PASS	Successful	N
02	Function	Admin	Logging out of their account	Logged in to home	Home-> Logout	Userna me= Testadmp in	successf uly popup	successf uly popup	PASS	Successful	N
02	Function	User	Forgot password	Logged out of their account	Index-> Login-> Forget	Userna me= Testuser Password	Passwor d reset email received	Passwor d reset email receive	PASS	Successful	N
02	Function	Agent	Forgot password	Logged out of their account	Index-> Login-> Forget	Userna me= Testage nt	Passwor d reset email received	Passwor d reset email receive	PASS	Successful	N
02	Function	Admin	Forgot password	Logged out of their account	Index-> Login-> Forget	Userna me= Testadm in	Passwor d reset email received	Passwor d reset email receive	PASS	Successful	N
02	Function	User	Forgot password	Sent reset email 30 mins ago	Index-> Login-> Forget	Userna me= Testuser Password	Token invalid due to time limit	Token invalid due to time limit	PASS	Successful	N
02	Function	Agent	Forgot password	Sent reset email 30 mins ago	Index-> Login-> Forget	Userna me= Testage nt	Token invalid due to time limit	Token invalid due to time limit	PASS	Successful	N
02	Function	Admin	Forgot password	Sent reset email 30 mins ago	Index-> Login-> Forget	Userna me= Testadm in	Token invalid due to time limit	Token invalid due to time limit	PASS	Successful	N

9. RESULTS

9.1 Performance Metrics

9.1.1 Pressure test

NFT - Risk Assessment

S. No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Volumen Changes	Risk Score	Justification
1	Custo mer Care Regist ry	New	Low	No Chang es	Mode rate	Moder ate	>5 to 10%	ORA NGE	only moderat e changes
2	Custo mer Care Regist ry	existing	Low	No Chang es	Low	Low	>5 to 10%	GREE N	little changes required
3	Custo mer Care Regist ry	existing	Low	No Chang es	Low	Low	>5 to 10%	GREE N	small bug fixes
4	Custo mer Care Regist ry	New	moder ate	No Chang es	Mode rate	Low	>5 to 10%	ORA NGE	Moderat e changes

NFT - Detailed Test Plan

S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff
1	Login Page	LOAD	Page will slow down or wont be accessible	Ilamvazhuthi J

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
-------	------------------	-------------------	-----------	--------------	-------------------	-----------------	---	--------------------

1	LOGI N PAGE	LOA D	Not met beca use of login the user	PASS	NO- GO	CLOSED	Ilamvazhu thi J
---	-------------------	----------	---	------	-----------	--------	--------------------

9.1.1. Defect Analysis

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

Resolution	Severit y 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	1	0	1
Duplicate	0	1	2	1	5
External	2	3	0	1	6
Fixed	1	3	3	3	10
Not Reproduced	0	0	0	1	1
Skipped	1	0	0	1	2
Won't Fix	0	1	2	1	4
Totals	3	8	8	8	29

9.1.2. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	1	0	0	1
Client Application	20	0	0	20
Security	2	0	0	2
Outsource Shipping	1	0	0	1
Exception Reporting	3	0	0	3

Final Report Output	4	0	0	4
Version Control	2	0	0	2

10. ADVANTAGES & DISADVANTAGES

10.1.1. Advantages:

1. The UI has been developed in such a way that layers of updates and features can be added on without having to mess around too much with previous iterations of the code.
2. The Back-end of the application is pretty much optimized to have the best performance regardless of the limitations of the device being used to access it.
3. The application has been secured using the liberalizers and several authentication factors such that no unauthorized personnel could actually tamper with the data.
4. The application also has this minimalist theme with more or less an industrial base to make it easier to use.
5. The application also has been created with the intent of not missing any of the customer requests nor the agent request.

10.1.2. Disadvantages:

1. To the untrained eye the UI could feel a bit lacklustre.
2. Some people might find the application boring to interact with while others would prefer that the application adheres to the simple goal of Customer Care.
3. Due to multiple authentication on every request, the application might feel sluggish occasionally
4. The application might be overwhelmed with a wave of tickets and users which can cause the application to be bottlenecked sooner than later.

11. CONCLUSION

We have studied the problem statement carefully and looked at existing solution and decided to proceed with the application navigation in such a way that a User Generated Ticket cannot be flushed out of the system unless it is marked as completed. Our application has met all the criteria for the app we intended to create and it has passed all the Test and exceeded expectations for meeting their match with the existing application. The application has an integrated Chat Bot

12. FUTURE SCOPE

Some Variation of Applicable can be tried to be actually implemented if the aforementioned disadvantages are fine-tuned. We can also add more features and extend the application from a simple Customer Care Registry and made into an enterprise level application. The user interface of the application can also be improved by adding more design elements based on user feedback which we were not able to do due to the fact of it being such a short sprint.

13. APPENDIX

Due to the sheer size of the code and the amount of files in it we have attached the GitHub repository link down below along with the link for the Video demo/presentation.

GitHub: <https://github.com/IBM-EPBL/IBM-Project-10924-1659245354.git>

Project Demo Link: <https://drive.google.com/file/d/19PY0cRNzvtcuBMdu5Gq-Sl0QS0kM7hws/view?usp=sharing>