Report
Banking Support Portal Requirements:

➢ Project Overview

The Banking Customer Support Portal aims to provide an efficient and user-friendly platform for customers to post their complaints and queries. The system will intelligently route these complaints to the appropriate teams for resolution. It will also incorporate features like real-time auto-suggestions, email communication, and chat bot integration to streamline customer support operations.

➢ Technology Stack

- **Backend Development:**
- Java
- Spring Boot (for microservices)
-
- **Frontend Development:**
- React.js
- JavaScript
- HTML & CSS
-
- **Database Management:**
- MySQL/Oracle
-
- **Containerization and Deployment:**
- Docker
- Kubernetes
- Version Control and CI/CD:
- Git (Gitlab for version control)
- Jenkins for Continuous Integration/Continuous Deployment (CI/CD)

➢ Functionalities:

1. Customer-Side Functionalities:

➢ **Complaint Posting**

- Customers can submit their complaints via the portal.

➢ **Auto Suggestion of FAQs**

- Real-time suggestions will be provided based on the nature of the complaint.

➢ **Dashboard for Tracking**

- Customers can view and track the status of their complaints.

➢ **Email Notifications**

- Automated emails will be sent to customers upon complaint submission and resolution.

➢ Chat Bot Integration:

- The chat bot will handle repetitive queries and provide links to relevant FAQ documents.

➢ **Repetitive Queries:**

- Repetitive queries are common issues or requests that customers often raise, such as unlocking a user account, updating contact information (like mobile number or address), or resetting a PIN.

➢ **Chat Bot Workflow:**

- When a customer posts a query related to one of these repetitive tasks, the Chat Bot will intercept the message.
- It will analyze the query to determine the nature of the request (e.g., unlock, update, reset).
- The Chat Bot will then use the REST API to communicate with the backend of the system to perform the necessary action.

➢ **Providing FAQ Links:**

- In addition to directly resolving queries, the Chat Bot can also suggest relevant FAQ document links.

➢ Backend Functionalities:

➢ Ticket Routing

- Complaints will be automatically assigned to the appropriate teams based on category.

➢ Ticket Handling

- Support teams can view and update the status and comments on query tickets.

➢ Dashboard for Assignments

- Support teams can view and manage their assigned tickets.

➢ MIS/Reporting Module

➢ **Data Retrieval**

- This module will allow retrieval of data from the system for reporting purposes.

➢ **Downloadable Reports**

- Reports will be available for download in CSV, Excel, and PDF formats.

➢ **Architectural Considerations:**

- Microservice Architecture:
- The application will be built as a collection of loosely coupled microservices.
- Each microservice will handle a specific functionality (e.g., Complaints, Routing, Chat Bot, etc.).

Use Case Name	Post Complaint
Actors	1. Registered Customer (Has an existing account with the bank) 2. Non-registered Customer (Does not have an existing account)
Precondition (if any)	
Flow	1. The customer will indicate that he/she wants to post a complaint. 2. A drop down will appear with some options for the customer to choose the type of complaint. 3. The customer will select an option and add the additional information (if any). 4. The complaint will be redirected to the respected team. 5. The customer can view the status of the complaint.

Use Case Name	Manage Complaint Status
Actors	1. Customer Support·
Precondition (if any)	The member should be a part of the respective team.
Flow	1. The support engineer can view all the complaints and change the status. 2. The status of the complaint will be notified to the customer via Email notifications. 3. The engineer can assign a ticket and view it.

Use Case Name	Register Customer
Actors	1. Admin·
Precondition (if any)	New User
Flow	1. A new customer is added to the bank’s database by the admin.

Use Case Name	Chatbox Integration
Actors	1. Admin
Precondition (if any)	
Flow	1. Whenever a user will login, a chatbox will be present with some common queries link. 2. If the query that user has is present there then he/she can click the link and the solution will be shown. 3. If it is not solved, the user can post a complaint or raise a query.

Use Case Name	Generate FAQ
Actors	2. Admin
Precondition (if any)	
Flow	1. The admin can add new FAQ or edit the existing ones. 2. User can view the FAQs on the UI

Use Case Name	View Profile
Actors	1. Customer
Precondition (if any)	The customer should have an account in the bank.
Flow	1. The customer can see the assigned customer ID and all the details. 2. Can logout from the site.