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	Place Order
Actors	1. Registered Shopper (Has an existing account, possibly with billing and shipping information) 2. Non-registered Shopper (Does not have an existing account)
Precondition (if any)	User has selected the items to be purchased.
Flow	1. The user will indicate that she wants to order the items that have already been selected. 2. The system will present the billing and shipping information that the user previously stored. 3. The user will confirm that the existing billing and shipping information should be used for this order. 4. The system will present the amount that the order will cost, including applicable taxes and shipping charges.