Telecom Customer Onboarding System

Description:

Develop a customer onboarding system for a telecom company to streamline the registration, document verification, and service activation processes.

User Stories and Backlog:

Here are the detailed user stories for each epic:

Epic 1: Customer Registration

User Story 1.1: Create Registration Page

- As a front-end developer I want to build a registration page with form fields for customer information So that customers can input their details for onboarding.

User Story 1.2: Create Login Page and Validate User Credentials

- As a front-end developer, I want to create a login page and validate the username and password combination, So that only registered customers can access their accounts.

User Story 1.3: Design UI

- As a UI/UX designer, I want to design an intuitive and user-friendly interface for registration and login pages, So that customers have a seamless onboarding experience.

User Story 1.4: Connect to MySQL Database to Store Customer Details

- As a backend developer, I want to connect the registration page to the MySQL database, So that customer details are securely stored for future use.

User Story 1.5: Edit and Update Details

* As a backend developer, I want to functionality to edit and update details field.

Epic 2: Document Verification

User Story 2.1: Integrate with Third-Party API for Document Verification

- As a backend developer, I want to integrate with a third-party API for document verification, So that customer documents can be automatically validated during registration.

User Story 2.2: Store Verification Status in the Database

- As a backend developer, I want to store the document verification status in the database, So that the system can track which customers have passed the verification process.

User Story 2.3: Implement Error Handling for Failed Verification

- As a backend developer, I want to implement error handling for document verification failures, So that appropriate actions can be taken when verification fails.

User Story 2.4: Provide Feedback to the Customer Service Representative on Verification Status

- As a customer, I want to receive real-time feedback on the document verification status, So that I can inform customers promptly about their registration progress.

Epic 3: Service Activation

User Story 3.1: Design the Service Activation Workflow

- As a solution architect, I want to design a workflow for activating telecom services,

So that the process is streamlined and automated based on customer information.

User Story 3.2: Implement Logic to Activate Services Based on Customer Plans

- As a backend developer, I want to implement logic to activate services based on the selected customer plans, So that customers receive the services they signed up for.

User Story 3.3: Develop Email Notification Service Using Spring Boot

- As a backend developer, I want to develop an email notification service using Spring Boot, So that customers are informed about the status of their service activation.

User Story 3.4: Update Customer Status in the Database Post-Activation

- As a backend developer, I want to update the customer’s status in the database once their services are activated, So that the system accurately reflects the customer’s service status.

User Story 3.5: Generate and Send Welcome Emails to Customers

- As a backend developer, I want to generate and send welcome emails to customers upon successful service activation, So that they are informed and engaged from the start.

Epic 4: Deploy on AWS

User Story 4.1: Set Up AWS Environment

- As a cloud engineer, I want to set up the necessary AWS environment ,So that the application can be securely deployed in the cloud.

User Story 4.2: Configure CI/CD Pipeline

- As a DevOps engineer, I want to configure a CI/CD pipeline using AWS services (e.g., CodePipeline, CodeBuild, CodeDeploy), So that application updates can be automatically built, tested, and deployed.

User Story 4.3: Deploy Backend Services

- As a backend developer, I want to deploy the Spring Boot application on an AWS EC2 instance, So that the backend services are accessible in the cloud.

User Story 4.4: Deploy Frontend on AWS S3 and CloudFront

- As a front-end developer, I want to deploy the React frontend on AWS S3 with CloudFront as the CDN, So that the application has low-latency access for users.

User Story 4.5: Set Up AWS RDS for MySQL Database

- As a database administrator, I want to set up an AWS RDS instance for the MySQL database, So that customer data is securely stored and managed in the cloud.

User Story 4.6: Monitor Application with AWS CloudWatch

- As a DevOps engineer, I want to configure AWS CloudWatch to monitor application performance and logs, So that potential issues can be detected and addressed in real-time.

User Story 4.7: Implement Security Best Practices

- As a security engineer, I want to implement AWS security best practices (IAM roles, security groups, encryption), So that the application is protected against unauthorized access and data breaches.