Citizen Service Request Platform Documentation

Team: Tecresearch Leader: Mr.Brijesh Nishad (9506563150)

September 26, 2024

Contents

1 Project Introduction

The Citizen Service Request Platform is designed to facilitate communication between citizens and local government authorities for both rural and urban development. It enables users to submit service requests, track their status, and provide feedback on completed services. The platform aims to improve the efficiency and transparency of public services.

2 Aim

The primary objective of this project is to create a robust and user-friendly platform for managing citizen service requests. The goals include:

- Streamlining the process of submitting and managing service requests.
- Enhancing communication between citizens and authorities.
- Promoting transparency and accountability in public services.

3 Use Cases

The platform supports several use cases, including:

- **Submitting a Service Request:** Citizens can submit requests for various public services, such as road repairs, waste management, or street lighting.
- **Tracking Request Status:** Users can track the status of their submitted requests in real time.
- **Feedback and Ratings:** After the service is completed, citizens can provide feedback and rate the quality of service.

4 Project Structure

The project is structured into two main parts: the frontend, developed using React, and the backend, developed using Spring Boot.

4.1 Frontend (React)

The frontend of the application is responsible for providing an interactive user interface. Key files include:

- `index.jsx` - Entry point of the React application.
- `App.jsx` - Main component that defines routes and renders other components.

- `components/ServiceRequestForm.jsx` - Form component for submitting new service requests.
- `components/RequestStatus.jsx` - Component for displaying the status of submitted requests.
- `components/FeedbackForm.jsx` - Form component for providing feedback on completed services.
- `utils/api.js` - Utility file for handling API requests to the backend.

4.2 Backend (Spring Boot)

The backend of the application provides RESTful APIs to support the frontend functionalities. Key files include:

- `CitizenServiceApplication.java` - Main application file.
- `controller/ServiceRequestController.java` - Handles incoming HTTP requests related to service requests.
- `service/ServiceRequestService.java` - Business logic for managing service requests.
- `repository/ServiceRequestRepository.java` - Interface for database interactions related to service requests.
- `model/ServiceRequest.java` - Entity class representing a service request.

5 Code

Below are some essential code snippets and explanations:

5.1 Frontend Example: API Call

Listing 1: Example API call using Axios

```
import axios from 'axios';

export const createServiceRequest = async (data) => {
    try {
        const response = await axios.post('/api/service-requests', data);
        return response.data;
    } catch (error) {
        console.error('Error creating service request', error);
        throw error;
    }
};
```

5.2 Backend Example: REST Controller

Listing 2: Service Request REST Controller

```
@RestController
@RequestMapping("/api/service-requests")
public class ServiceRequestController {

    @Autowired
    private ServiceRequestService serviceRequestService;

    @PostMapping
    public ResponseEntity<ServiceRequest> createServiceRequest(@RequestBody Serv
        ServiceRequest savedRequest = serviceRequestService.save(request);
        return new ResponseEntity<>(savedRequest, HttpStatus.CREATED);
    }

    @GetMapping("/{id}")
    public ResponseEntity<ServiceRequest> getServiceRequest(@PathVariable Long i
        ServiceRequest request = serviceRequestService.findById(id);
        return new ResponseEntity<>(request, HttpStatus.OK);
    }
}
```

6 User Training

To ensure effective use of the Citizen Service Request Platform, the following training materials are provided:

6.1 Getting Started Guide

- Overview of the platform's features and functionalities.
- Step-by-step instructions on how to submit a service request.
- Guide to tracking the status of a request and providing feedback.

6.2 Video Tutorials

- Submitting a Service Request
- Tracking Request Status
- Providing Feedback

6.3 Frequently Asked Questions (FAQ)

- What types of service requests can be submitted?
- How can I track the status of my request?
- Who can I contact for further support?