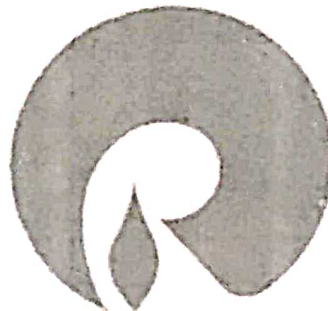


Project Report Approval Certificate

This project report entitled **Blocking & Un-Blocking of 4G & 5G Customers**
by:

Apurv Jadhav

Is approved for the degree of Bachelor of Technology in Computer Science Engineering



Date: 19th Jan, 2024

Place: Reliance Corporate Park, Ghansoli

Vice President of Jio

Manish Vishwakarma
Sr. Manager

Prof. Dr. Aparna Bannore

ACKNOWLEDGEMENT

I would like to take this chance to acknowledge **Mr. Nilesh S Jain**, Vice President of Jio – Platform, for accepting me as a student trainee to accomplish my project training at Jio Headquarters, Navi Mumbai and providing me all the facilities of Jio, Navi Mumbai.

I am fortunate to be placed under the able guidance of **Mr. Rajesh Sinha**, Vice President of Jio – Platform, Navi Mumbai. His encouraging remarks from time to time greatly helped me in improving my skills.

I am fortunate to be placed under the able guidance of **Mr. Manish Vishwakarma**, Senior Manager of Jio – Platform, for the invaluable guidance, co-operation, constant encouragement, and support. All these people have helped me in tough situations of the project.

I would like to pay my sincere regards to **Prof. Dr. Aparna Bannore** Head of the Department of Computer Science and Engineering, Sies GST And all the other staff members of the Department who provided us homely environment and constant encouragement during my B. Tech course.

At last, but above all I would like to pay my sincere regards to my parents for their invaluable support.

Apurv Jadhav

Contents

Abstract	iii
List of Abbreviations	iii
List of Figures	iv
1 Introduction	1-2
1.1 Introduction	1
1.2 Motivation	1
1.3 Problem Statement	1
1.4 Organization of the Report	2
2 Proposed System	3-4
2.1 Proposed system (explain how drawbacks will be overcome)	3
2.2 Architecture/ Framework	3
2.3 Data Collection	4
3 Design and Methodology	5-6
3.1 Design details	5
3.2 Methodology	5
4 Results and Discussions	7-9
4.1 Implementation	7
Chapter 5 Conclusion	10

ABSTRACT

CEIR is a database that helps people track and block their lost or stolen mobile phones. It also allows users to check the status of their requests to block a phone and unblock the device once it's found. CEIR is a central system that allows network operators to share a list of blacklisted mobile devices. This prevents a reported and blocked phone from being used on other networks, even if the SIM card is changed. CEIR is free to use, but users need to keep their phone's IMEI number handy. The IMEI number can be found on the box of the smartphone.

LIST OF ABBREVIATIONS

UI: User Interface

JDK: Java Development Kit

MVC: Model View Controller

POM: Project Object Model

XML: Extensible Markup Language

JSON: JavaScript Object Notation

YAML: Yet Another Markup Language

REST: Representational State Transfer

API: Application Programming Interface

LIST OF FIGURES

Figure Number	Figure Caption	Page No
1	Spring MVC Architecture	3
2	Ceir Architecture	5
3	Swagger-UI	7
4	Posting block request through Swagger-UI	8
5	Response of blocking request	8
6	Request added in database	8
7	Posting unblock request through Swagger-UI	9
8	Response of unblocking request	9
9	Device information deleted from database	9

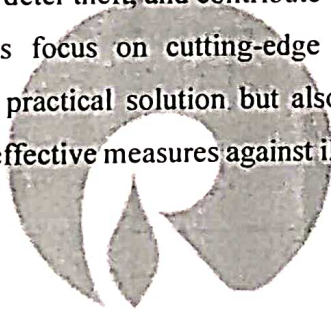
INTRODUCTION

1.1 Introduction

In response to the escalating challenges posed by mobile device theft and fraud, this project introduces a replication of the Central Equipment Identity Register (CEIR). The CEIR system, functioning as a centralized database using International Mobile Equipment Identity (IMEI) numbers, plays a pivotal role in deterring illicit activities by rendering stolen or lost devices inoperable on mobile networks.

1.2 Motivation

The project is motivated by the pressing need to address the rising challenges of mobile device theft and fraud. By replicating a Central Equipment Identity Register (CEIR), the aim is to enhance overall security, deter theft, and contribute valuable insights to the mobile security landscape. The project's focus on cutting-edge technologies and real-world applications provides not only a practical solution but also an educational resource for understanding and implementing effective measures against illicit activities related to stolen or lost mobile devices.



1.3 Problem Statement

The ubiquity of smartphones has led to an alarming surge in mobile device theft and fraudulent activities, posing a significant threat to user data security and personal safety. Existing measures to combat this issue are often insufficient. Law enforcement agencies and telecommunication providers lack a centralized and robust system for tracking and rendering stolen or lost devices inoperable. Consequently, there is a pressing need for an effective solution that replicates the functionality of a Central Equipment Identity Register (CEIR). This project addresses the gap in current mobile security infrastructure by proposing and implementing a CEIR replication, aiming to provide a comprehensive system for blacklisting stolen devices, enhancing overall mobile security.

1.4 Organization of report

- **Chapter 1 - Introduction:**

This chapter highlights the problem statement, the motivation behind it and objectives.

- **Chapter 2 - Proposed System:**

This chapter elaborates on the proposed system and covers architecture/framework and process design details. Also states the necessary software and hardware details.

- **Chapter 3 - Design and Methodology:**

This chapter explains the design and methodology.

- **Chapter 4 – Results and Discussion**

This chapter illustrates the implementation of the project and goes over the results obtained.

- **Chapter 5 – Conclusion and Future Scope**

This chapter contains the conclusions that we have reached through research and implementation of the project. Also, it describes the planned developments for the project.

PROPOSED SYSTEM

2.1 Proposed System

The proposed system aims to replicate the Central Equipment Identity Register (CEIR) functionality with a streamlined approach for blacklisting and unblocking stolen or lost mobile devices. The core components include a secure internal request database for efficient tracking, a blocking controller to initiate blacklisting based on theft reports, and an unblocking controller for legitimate device reactivation. Users can enter the blocking details such as the device information, owner information and lost information. A request id will be generated upon a successful block request which user will have to submit during unblocking.

2.2 Architecture/Framework

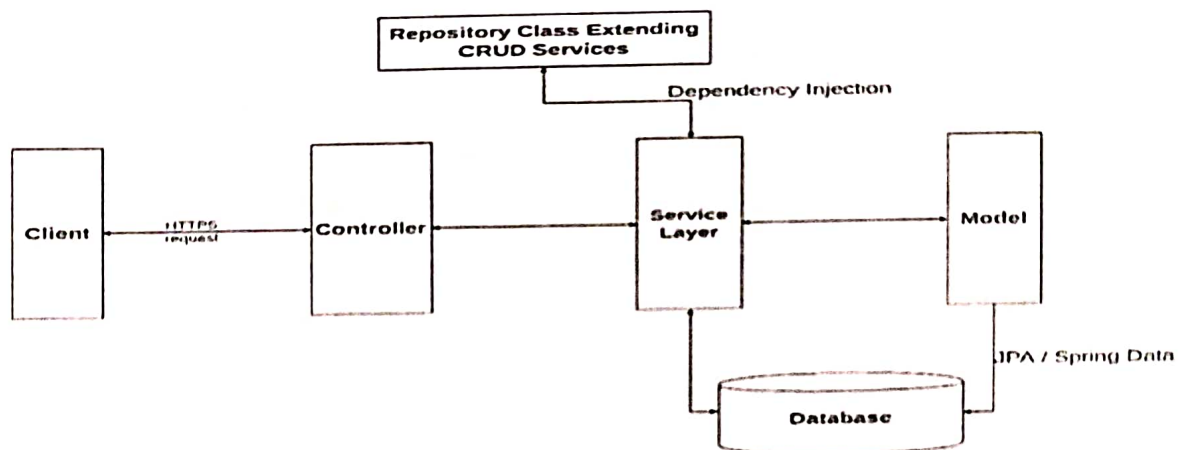


Figure 2.1

Spring MVC is a web module within the broader Spring Framework that facilitates the development of web applications based on the Model-View-Controller architectural pattern. In the context of our CEIR replication project, Spring MVC plays a crucial role in handling HTTP requests, managing the business logic, and rendering views.

- **Model:** Represents the data and business logic of the application. In the CEIR system, the model would involve entities representing mobile devices and their associated information.

- **View:** Defines how the data is presented to the user. In a Spring MVC application, views are typically implemented using templates (e.g. Thymeleaf) or other technologies.
- **Controller:** Manages the flow of the application, handling incoming HTTP requests, invoking the necessary business logic. In the CEIR system, controllers would handle actions like blocking and unblocking devices.

Spring MVC promotes a modular and loosely coupled design, making it easier to maintain and extend the application. It also integrates well with other Spring components, such as Spring Boot, which simplifies the configuration and deployment of web applications.

2.3 Data Collection:

1. Data Collection: Block Request JSON Structure

The data collection process involves gathering information to initiate a block request, aiming to blacklist a mobile device reported as lost or stolen. The block request is encapsulated in a JSON structure named block request models, consisting of three primary data transfer objects (DTOs): device information, owner information, and lost information models.

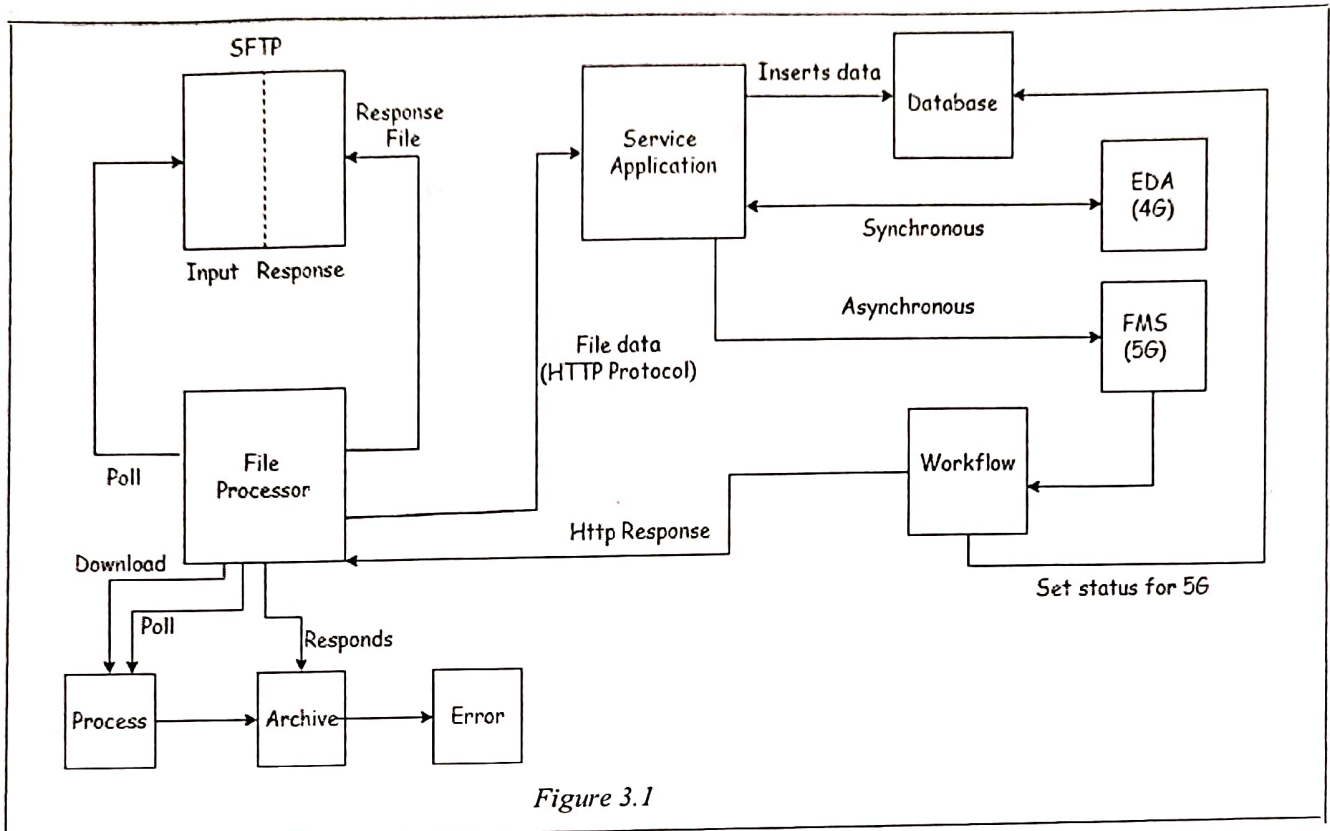
2. Data Collection: Unblock Request JSON Structure

The unblock request process involves gathering specific information to facilitate the reactivation of a previously blocked mobile device. The unblock request is encapsulated in a JSON structure named unblock request model, which includes the following attributes: Request Id, mobile number for OTP and reason for unblocking.

DESIGN AND METHODOLOGY

3.1 Design:

The following diagrams represents the logical and navigational flow of the application:



The above flowchart gives an overview of the flow and functionality/process of the entire application. The statements in the flow lines represent the events taking place in the background/behind the scenes.

3.2 Methodology:

Following steps were taken to implement the project successfully:

Building this application involves several steps that need to be properly implemented to the high-level methodology given below.

- **Research:**

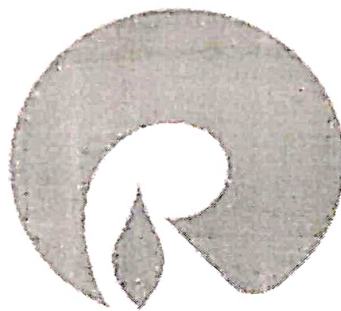
The first step was to conduct exhaustive research on Spring boot API development and OpenAPI specification to autogenerate API's using yaml. Research on all the dependencies required in the pom.xml file of the Spring boot project was done.

- **Project Setup:**

Installed IntelliJ Idea and set up the development environment for a new Springboot 3.0 Project. The autogenerated APIs were implemented with the controller and necessary business logic was written.

- **Data Storage:**

Implementation of data storage for block requests was done in H2 database which is also known as in memory database.



RESULTS AND DISCUSSION

4.1 Implementation

The implementation and execution of our project is discussed below in detail:

4.1.1 Enabling Swagger-UI

Swagger UI is an open-source tool that provides an interactive user interface for visualizing and interacting with APIs (Application Programming Interfaces) that are described using the OpenAPI Specification (formerly known as Swagger Specification). The OpenAPI Specification defines a standard way to describe RESTful APIs, including their endpoints, request/response formats, authentication methods, and other relevant details. Swagger UI generates a web-based documentation interface that allows developers to explore, test, and understand APIs without the need to consult external documentation. It provides a user-friendly and interactive way to discover and consume API endpoints.

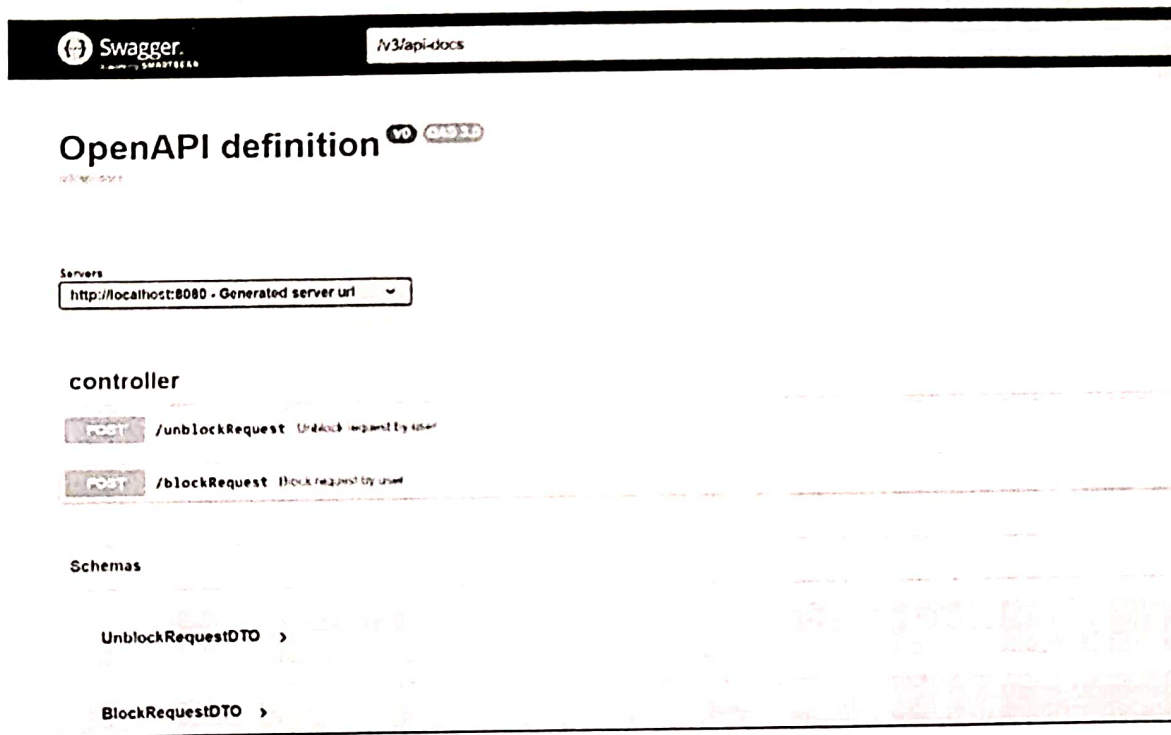


Figure 4.1

4.1.2 Block Request Working

Posting Block request through Swagger-UI

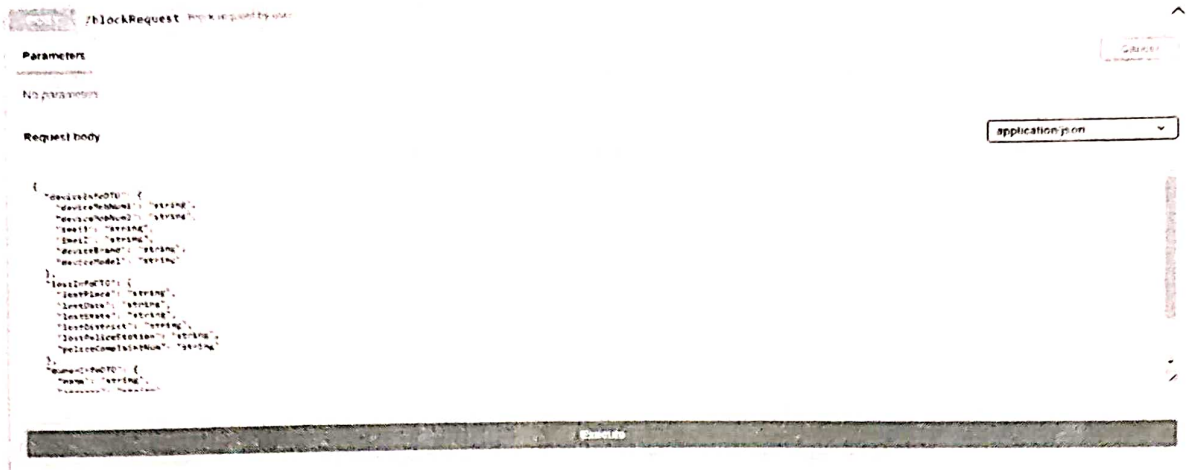


Figure 4.2

Response of blocking request:

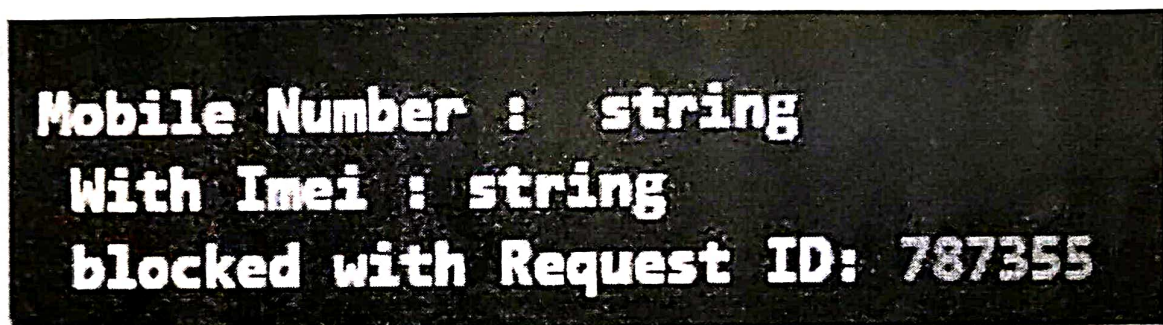


Figure 4.3

Database:

```
SELECT * FROM INTERNAL_REQUEST;
```

REQUEST_ID	DEVICE_MOB_NUM1	IMEI1	MOB_NUM_FOR_OTP
53978	1234	1234	1234
787355	string	string	string

(2 rows, 0 ms)

Figure 4.4

4.1.3 Unblock Request Working

Posting unblock request.

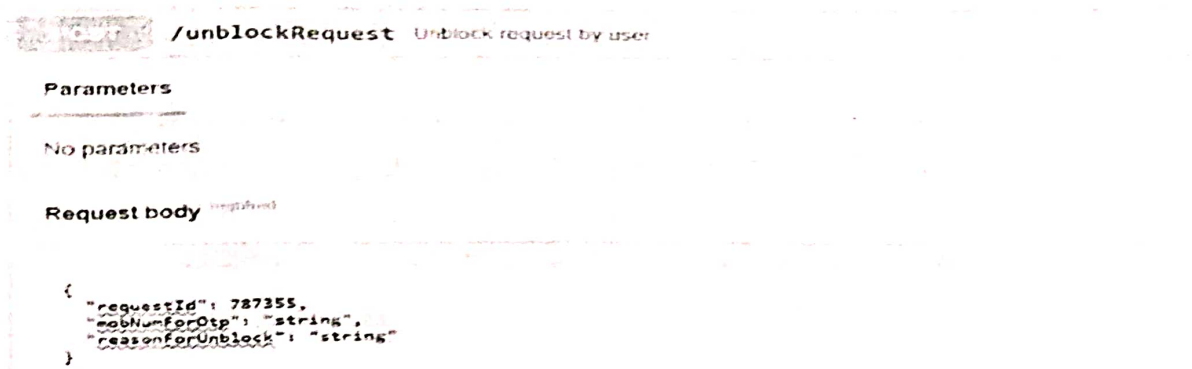


Figure 4.5

Response of unblocking request.



Figure 4.6

Database:

SELECT * FROM INTERNAL_REQUEST;

REQUEST_ID	DEVICE_MOB_NUM1	IMEI1	MOB_NUM_FOR_OTP
53978	1234	1234	1234

(1 row, 0 ms)

Figure 4.7

CONCLUSION

Conclusion

The Central Equipment Identity Register has achieved its primary objective of blocking and unblocking mobile devices based on the information provided by the user. In retrospect, the "Central Equipment Identity Register" project has been a journey of immense learning and growth. This endeavor not only met its objectives in creating a secure mobile device tracking system but also served as a catalyst for personal and professional development. Throughout the project lifecycle, I had the opportunity to enhance my Java programming skills significantly. The hands-on experience allowed me to delve deeper into object-oriented programming (OOP) principles, applying them in a real-world scenario. A pivotal aspect of this project was the integration of Spring Boot, which provided a solid foundation for building robust and scalable applications. Exploring Spring Boot's capabilities not only expanded my technical toolkit but also deepened my understanding of modern application development practices. There were a lot of challenges faced in the integration of swagger-UI in this project, all the solutions and hyperlinks important to important documentation are mentioned in the GitHub repository of the project. I express my gratitude to my mentors and colleagues whose guidance and support played a crucial role in the success of this project and my professional development.