Xeni Rate Limiter Service

This project implements a Rate Limiter Backend Service in Java using Spring Boot. The service supports client registration and enforces rate-limiting using the Fixed Window strategy.

# Features

* Client Registration with rate-limiting configuration
* Fixed Window Rate Limiting Strategy
* API to verify if a request is allowed (Eligibility Check)
* In-Memory Data Store (No external DB required)
* Designed to easily extend for: Sliding Window Strategy, Token Bucket Algorithm, Quota-Based Models
* Unit and Integration tests using JUnit and Mockito

# Technologies

* Java 17+
* Spring Boot
* JUnit 4
* Mockito
* Maven

# Setup Instructions

1. Clone the repository:

git clone https://github.com/AmitKumar1996/Xeni-Rate-Limiter-Service.git

cd Xeni-Rate-Limiter-Service

2. Build the project:

mvn clean install

3. Run the application:

mvn spring-boot:run

The service will start on: http://localhost:8080

# API Endpoints

1. 1. Register Client

POST -> http://localhost:8080/api/register

JSON Body:  
{  
 "clientId": "client123",  
 "requestsPerWindow": 5,  
 "windowInSeconds": 60  
}

1. 2. Check if Request is Allowed

GET -> http://localhost:8080/api/request?clientId=client123

1. 3. Get Client Configuration

GET -> http://localhost:8080/api/client-config?clientId=client123

# Design

The system uses a fixed window counter algorithm to enforce rate-limiting. It maps each client to a rate configuration and tracks requests using Java's in-memory structures (ConcurrentHashMap).

## Extensibility

The architecture allows easy integration of additional rate limiting strategies like:

* Sliding Window
* Token Bucket
* Leaky Bucket

# Testing

The project includes:

* Unit tests for RateLimiterService
* Integration test using MockMvc for RateLimiterController

Run tests with:

mvn test

# Repository

GitHub: https://github.com/AmitKumar1996/Xeni-Rate-Limiter-Service