

Setup and Execution Guide

Project Requirements

- **Java Version:** 23
 - **Maven Version:** 3.x
 - **Database:** MongoDB
 - **API Key Requirement:** Exchange Rate API (<https://app.exchangerate-api.com>)
-

1. Prerequisites

Ensure the following are installed on your system:

- Java Development Kit (JDK) 23
 - Apache Maven 3.x
 - MongoDB (Community Edition or higher)
-

2. Project Setup

Step 1: Clone the Repository

```
git clone < >
```

```
cd < >
```

Step 2: Install Project Dependencies

Clean and build the project using Maven:

```
mvn clean install
```

3. Configure MongoDB

Step 1: Install MongoDB

- Download and install MongoDB from [MongoDB Download Center](#).
- Start the MongoDB service:

- mongod

Step 2: Update MongoDB Connection Details

Edit the `src/main/resources/application.properties` file with your MongoDB connection string:

```
spring.data.mongodb.uri=mongodb://localhost:27017/<your-database-name>
```

4. Configure Currency Exchange API

- Register and get a **Free API Key** from [Exchange Rate API](#).
- Add the API Key to `application.properties`:

```
spring.exchangerate.key=YOUR_API_KEY
```

5. Run the Application

Start the Spring Boot application using Maven:

```
mvn spring-boot:run
```

The application will be available at:

```
http://localhost:8080
```

6. Unit Testing

Step 1: Configure MongoDB for Testing

Create a new properties file for testing:

```
src/test/resources/application-test.properties
```

```
spring.data.mongodb.uri=mongodb://localhost:27017/testdb
```

Step 2: Run Unit Tests

Execute unit tests with Maven:

```
mvn test
```

7. Integration Testing

Step 1: Add Integration Test Dependencies

Make sure the following dependencies are included in your pom.xml under <dependencies>:

```
<!-- Spring Boot Starter Test -->
```

```
<dependency>
```

```
    <groupId>org.springframework.boot</groupId>
```

```
    <artifactId>spring-boot-starter-test</artifactId>
```

```
    <scope>test</scope>
```

```
</dependency>
```

```
<!-- Embedded MongoDB for testing -->
```

```
<dependency>
```

```
    <groupId>de.flapdoodle.embed</groupId>
```

```
    <artifactId>de.flapdoodle.embed.mongo</artifactId>
```

```
    <version>3.0.0</version>
```

```
    <scope>test</scope>
```

```
</dependency>
```

```
<!-- RestAssured for API testing -->
```

```
<dependency>
```

```
    <groupId>io.rest-assured</groupId>
```

```
    <artifactId>rest-assured</artifactId>
```

```
    <version>5.3.0</version>
```

```
    <scope>test</scope>
```

</dependency>

Step 2: Run Integration Tests

Execute integration tests with Maven:

mvn verify

8. Dependencies Summary

Dependency	Purpose
Spring Boot Starter Web	Build RESTful APIs
Spring Data MongoDB	MongoDB integration
Mockito	Mocking framework for testing
JUnit	Unit testing framework
Spring Boot Starter Test	Testing support for Spring
Flapdoodle Embedded MongoDB	Embedded MongoDB for tests
RestAssured	REST API integration testing

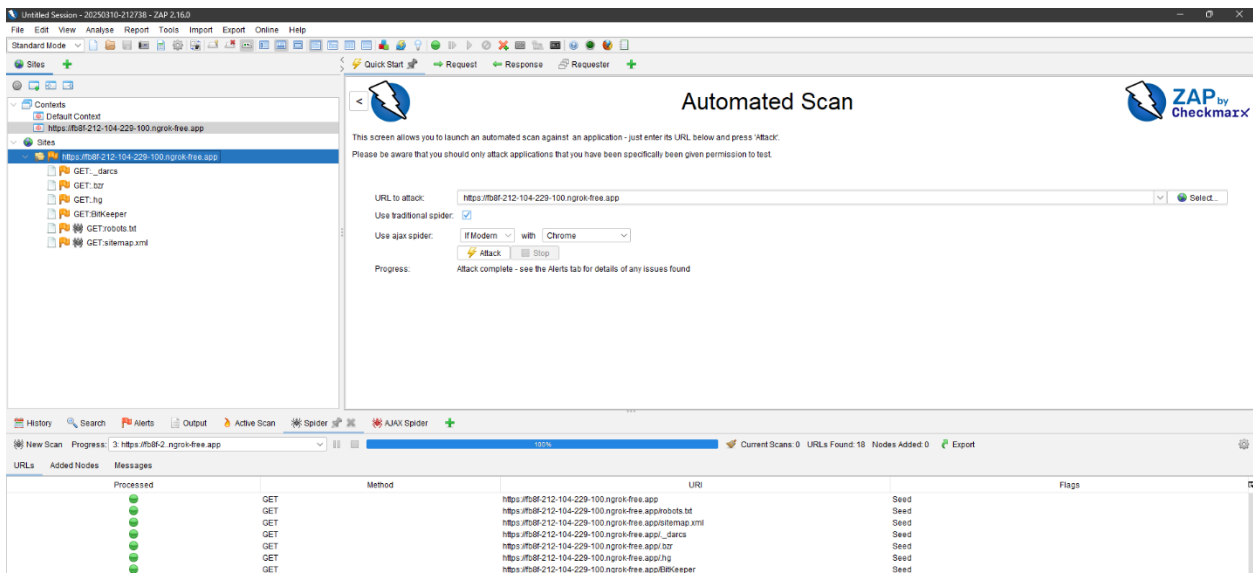
9. Security Testing

Install ngrok

Expose Spring Boot app: ngrok http 8080

The screenshot shows the ZAP (Zed Attack Proxy) interface. The main window displays the 'Automated Scan' configuration. The URL to attack is set to 'https://b8f212-104-229-100.ngrok-free.app'. The scan progress is 100% complete. The bottom section shows a table of sent messages with columns: ID, Req. Timestamp, Resp. Timestamp, Method, URL, Code, Reason, RTT, Size Resp. Header, and Size Resp. Body.

ID	Req. Timestamp	Resp. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Header	Size Resp. Body
407	10/03/2025, 21:43:53	10/03/2025, 21:43:53	GET	https://b8f212-104-229-100.ngrok-free.app/7163732646703	200	OK	180 ms	325 bytes	2,916 bytes
408	10/03/2025, 21:43:53	10/03/2025, 21:43:54	GET	https://b8f212-104-229-100.ngrok-free.app/1587530458233	200	OK	167 ms	325 bytes	2,916 bytes
410	10/03/2025, 21:43:54	10/03/2025, 21:43:54	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/web.xml	200	OK	97 ms	325 bytes	2,916 bytes
411	10/03/2025, 21:43:54	10/03/2025, 21:43:54	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/applica...	200	OK	204 ms	325 bytes	2,916 bytes
412	10/03/2025, 21:43:54	10/03/2025, 21:43:55	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	855 ms	325 bytes	2,916 bytes
413	10/03/2025, 21:43:55	10/03/2025, 21:43:55	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	617 ms	325 bytes	2,916 bytes
414	10/03/2025, 21:43:55	10/03/2025, 21:43:56	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	1,113 s	325 bytes	2,916 bytes
415	10/03/2025, 21:43:56	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	183 ms	325 bytes	2,916 bytes
416	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	177 ms	325 bytes	2,916 bytes
417	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	102 ms	325 bytes	2,916 bytes
418	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	92 ms	325 bytes	2,916 bytes
419	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	104 ms	325 bytes	2,916 bytes
420	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	88 ms	325 bytes	2,916 bytes
421	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	89 ms	325 bytes	2,916 bytes
422	10/03/2025, 21:43:57	10/03/2025, 21:43:57	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	84 ms	325 bytes	2,916 bytes
423	10/03/2025, 21:43:57	10/03/2025, 21:43:58	GET	https://b8f212-104-229-100.ngrok-free.app/WEB-INF/dasse...	200	OK	89 ms	325 bytes	2,916 bytes
424	10/03/2025, 21:43:58	10/03/2025, 21:43:58	POST	https://b8f212-104-229-100.ngrok-free.app/_darcy-d-alo...	400	Bad Request	357 ms	280 bytes	0 bytes



10. Performance Testing

Tools -

- Apache JMeter
- Postman Collection Runner

Extract and run - Windows: jmeter.bat

Add headers like - Authorization: Bearer <your-jwt-token>

Content-Type: application/json

11. Useful Links

- [Java 23 Download](#)
 - [Maven Download](#)
 - [MongoDB Installation Guide](#)
 - [Exchange Rate API](#)
-

Author Mannapperuma MVN / IT22125866