

Developer Pre-Requisites

1. Ensure you have Maven installed (either locally or embedded on and IDE [e.g., Eclipse]).
 - a. Maven 3.5.3 (Development work done using this version)
2. Ensure you have Java 11+ installed.
 - a. JRE System Library used (in Eclipse) is JavaSE-11
3. Ensure you have a MySQL server running.
 - a. MySQL Community Server 8.0.23 (Development work done using this version)
 - b. Create a database and note down the URL. (Suggested: Use [jdbc:mysql://localhost:3306/ticketssystem])
4. Ensure you have a SonarQube server running (if generating Sonar reports)
 - a. Sonarqube-8.7.0.41497(Development work done using this version)

Eclipse Setup

Importing the Project

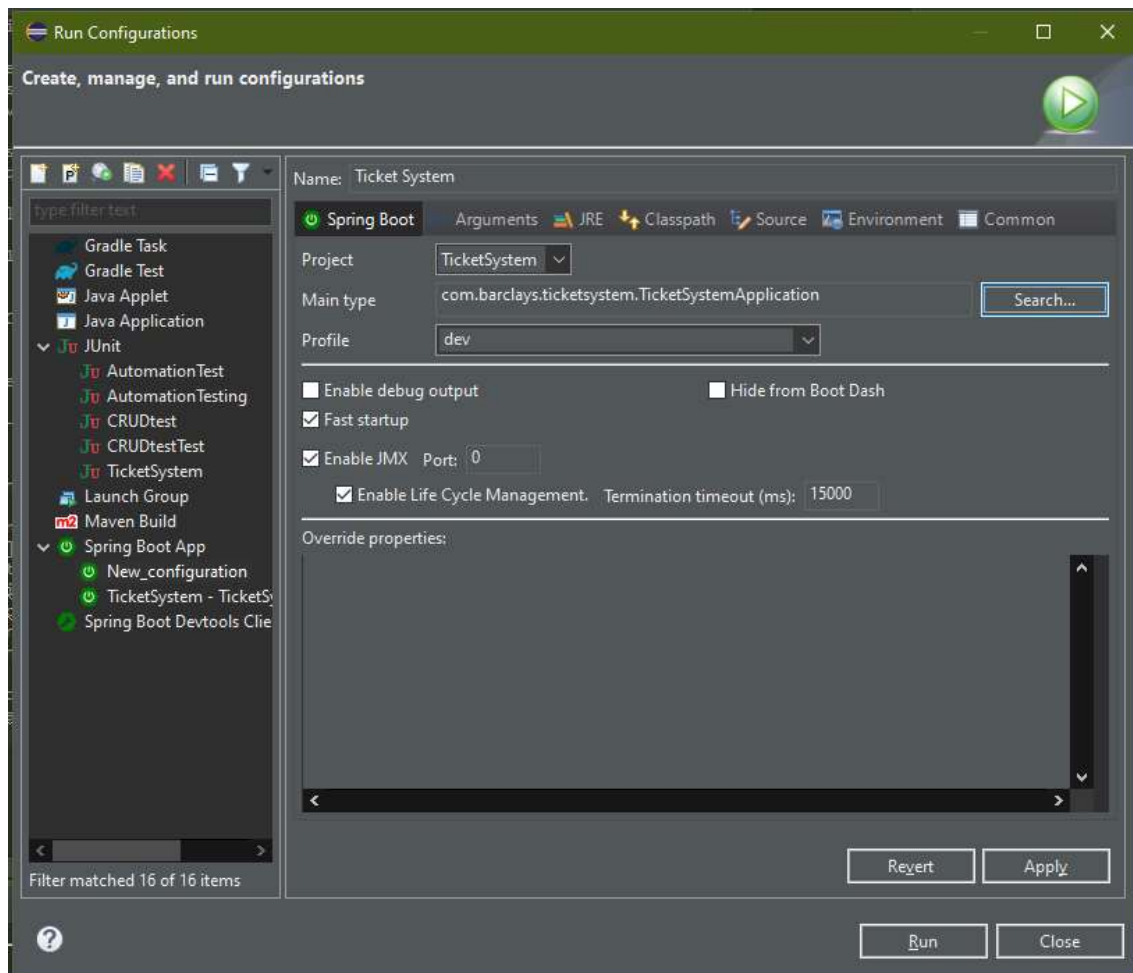
1. Create a new workspace and empty project.
2. Go to File->Import->Maven->Existing Maven Projects->Click 'Next'.
3. Select the Root Directory Folder (e.g., C:\Users\muham\Desktop\Ticket-System-Back-End if you clone the repository in Desktop)
4. Select both pom.xml files and import the project.

Building the Spring Boot Server

On Package Explorer right click on the root folder for the Ticket-System -> Run as -> "4. Maven build.."

Running Within Eclipse

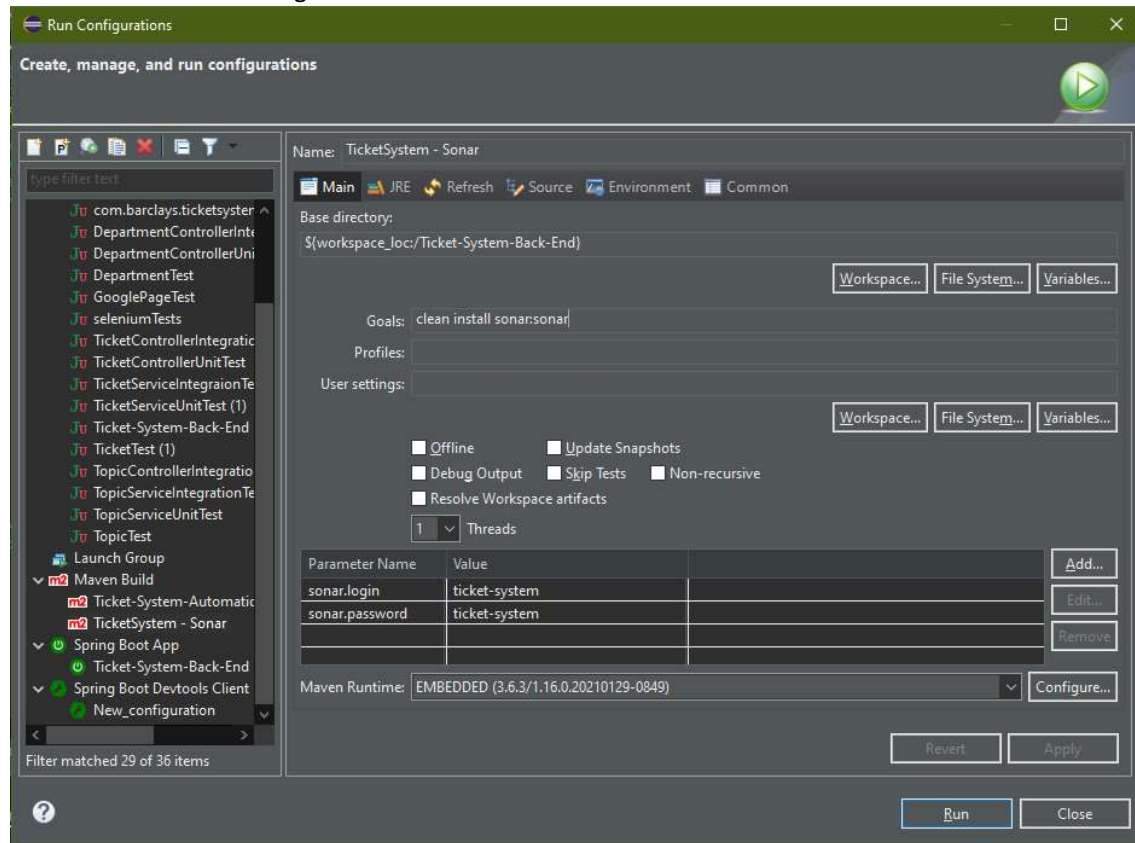
Create a Run Configuration for the Ticket-System project as follows:



In the field “Profile” select “dev” to run the spring boot application using an H2 database with dummy data. For connecting to production database select “prod” instead of dev. Navigate to “Ticket-System-Back-End/Spring Boot Server/src/main/resource/application-prod.properties” and make sure to connect to the correct datasource URL. The current value in there is “spring.datasource.url = jdbc:mysql://localhost:3306/ticketssystem?useSSL=false”.

Generating a Sonar Report

Create a Maven run configuration as follows:



Make sure to set the parameters sonar.login and sonar.password to appropriate values for your Sonar server. Navigate to "Ticket-System-Back-End\Spring Boot Server" and open "sonar.properties" to make sure the correct Sonar server is referenced.

Generating Eclipse Coverage Report (Unit Tests)

On Package Explorer right click on the root folder for the Ticket-System -> Coverage as -> JUnit Test

Running Selenium Automation Tests

On Package Explorer right click on the root folder for the Automation Testing Project -> Run as -> JUnit Test

Setup Issues

When the project is initially imported into Eclipse you might see some errors on the unit tests. If you see this issue, make sure you have the correct version of Java setup with your Eclipse. To check this right-click the TicketSystem project -> Build Path -> Configure Build Path -> Libraries and under "Modulepath" make sure it says "JRE System Library [JavaSE-11]").