Development Documentation

This document provides comprehensive details about the development process, including Javadocs, source code directory structure, build process, compiler time dependencies, development standards, database setup for development, and instructions on how to retrieve the source code from the repository.

Table of Contents

1. Javadocs

2. Source Code Directory Structure

3. Build Process

4. Compiler Time Dependencies

5. Development Standards

6. Database Setup for Development

7. Retrieving Source Code from the Repository

1. Javadocs

Javadocs are automatically generated documentation from the source code comments. They provide detailed descriptions of classes, methods, and fields.

To generate Javadocs:

javadoc -d docs $(find src -name "\*.java")

The generated documentation will be located in the `docs` directory.

2. Source Code Directory Structure

The source code is organized in the following structure:

project-root/

│

├── src/

│ ├── main/

│ │ ├── java/

│ │ │ └── com/

│ │ │ └── example/

│ │ │ └── project/

│ │ │ ├── MainClass.java

│ │ │ └── utils/

│ │ │ └── Helper.java

│ │ └── resources/

│ │ └── application.properties

│ └── test/

│ └── java/

│ └── com/

│ └── example/

│ └── project/

│ ├── MainClassTest.java

│ └── utils/

│ └── HelperTest.java

│

├── build/

│

├── docs/

│

├── lib/

│

└── README.md

3. Build Process

Prerequisites

- Java Development Kit (JDK) 8 or higher

- Apache Maven 3.6.0 or higher

Steps to Build

1. Navigate to the project root directory:

cd project-root

2. Compile the project:

mvn clean compile

3. To package the project:

mvn package

The compiled and packaged files will be in the “target” directory.

4. Compiler Time Dependencies

The project uses the following dependencies, which are managed by Maven in the `pom.xml` file:

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>5.3.8</version>

</dependency>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

<!-- Additional dependencies here -->

</dependencies>

5. Development Standards

- Code Style: Follow the Google Java Style Guide.

- Version Control: Use Git for version control. Follow the Git Flow branching model.

- Testing: Write unit tests for all new features and bug fixes. Use JUnit for testing.

- Documentation: Ensure all public methods and classes are documented using Javadocs.

6. Database Setup for Development

Prerequisites

MySQL or PostgreSQL database server

Steps to Set Up

1. Create a new database:

CREATE DATABASE project\_db;

2. Set up the database schema:

mysql -u username -p project\_db < schema.sql

3. Update the `application.properties` file with your database configuration:

properties

spring.datasource.url=jdbc:mysql://localhost:3306/project\_db

spring.datasource.username=db\_username

spring.datasource.password=db\_password

7. Retrieving Source Code from the Repository

Prerequisites

Git client installed

Steps to Retrieve

1. Clone the repository:

git clone https://github.com/yourusername/yourproject.git

2. Navigate to the project directory:

cd yourproject

3. Pull the latest changes:

git pull origin main

By following this documentation, developers should be able to set up their development environment, understand the project structure, build the project, and adhere to the development standards.