Development Documentation

This document explains how to work on the project, including details about the code, how to build it, and set up your computer for development.

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. Open 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. Dependencies Needed to Compile

The project uses certain tools to work correctly. These are listed in a file called `pom.xml`.

5. Rules for Writing Code

When writing code, follow these rules:

- Style: Use the Google Java Style Guide.

- Version Control: Use Git and follow the Git Flow method.

- Testing: Always test your code with JUnit.

- Documentation: Explain all public parts of your code using Javadocs.

6. Setting Up the Database

Before you start, make sure you have a database like MySQL or PostgreSQL Steps to Set Up

1. Create a new database:

CREATE DATABASE project\_db;

2. Set up the database structure:

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. Getting the Code from GitHub

To get the code for your project, follow these steps:

1. Open your terminal and type:

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

This will copy the project from GitHub to your computer.

2. Go into the project folder

cd yourproject

3. To get any new changes others have made, type:

Git pull origin main

*By following this guide, you can set up your computer for development, understand how the code is organized, build the project, and follow the rules for writing good code.*