

Installation guide

Introduction

Hotel management application is a desktop application which could be utilized to reduce the management efforts of the hotel business operators. The application consists of several important features such as room status, customer personal information and booking's information of customers to help managers easily monitoring the hotel's state and digitalize documentations with lesser efforts.

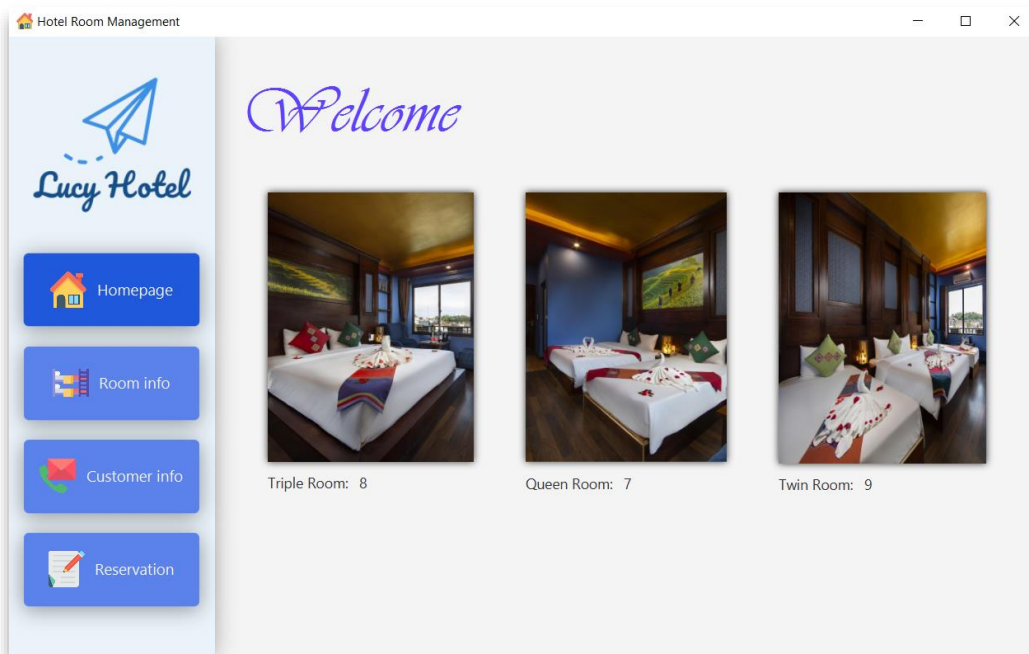


Figure 0-1-1: Homepage

The desktop application version is developed with clean-looking and user-friendly interface which could be helpful for multiple usage purposes. This version includes four main pages with different purposes as follows:

- Homepage provides a general looking of the hotel management by displaying the available rooms and some room previews according to the room's type.
- Room information page lists out the entire hotel's rooms with their current status. Additionally, on this page, the managers can also either edit, delete or updating a specific room.

- Customer information page shows all customer's registered personal information once booked a room in current hotel, together with their previous booking information.
- Booking page, allows receptionist to register a new customer by simply entering all of their required personal information. However, if the customer is already registered, their previous information can also be retrieved from the database.

Requirements

To launch the desktop application version, these requirements must be installed beforehand:

- Apache Maven 3.6.3, for managing the libraries utilized in the project, you can download [here](#).
- Java SE Runtime 11, to run the java application, you can download from [here](#).
- Java SDK 15 or above, for developments purpose (optional), download from [here](#).
- JavaFX SDK 15 or above, for the desktop UI of the application, download from [here](#).
- MySQL Server 8.0 or above, for hosting local MySQL database, download from [here](#).
- Tomcat 9.0, for deploying the web-server locally, download from [here](#).

Configuration

1. Install the dependencies as listed above.
2. Setup MySQL database
 - To set up the local database, please find and execute the SQL script named "**sqlConfig.sql**".

Run application

1. Deploy and run web-server
 - Go to server's folder named "**hotelServer**": `cd hotelServer`
 - Deploy with cargo plugin by: `mvn cargo:deploy`
 - Run Tomcat's server by running startup.bat from your Tomcat directory. By default, it locates at `C:\Program Files\Apache Software Foundation\Tomcat 9.0\bin\startup.bat`
 - The web server should be in "localhost:8080/hotelmanagement".
 - Further information: [click here](#)
2. Run application
 - Go to JavaFX client's folder: `cd hotelClient`

- Run JavaFX using the following command: `mvn javafx:run`

Optional (for developers):

- Build package by using command: `mvn package`
- Run ".jar" file by `"java --module-path <your-javafx-sdk-path> --add-modules javafx.controls javafx.fxml -jar target\hotelmanagement-1.0-SNAPSHOT.jar"`
 - o Download JavaFX SDK from [here](#).
 - o Place JavaFX SDK as your favourite
 - o Add your JavaFX SDK path to the command. **Example:** `"java --module-path E:\javafx-sdk-15.0.1\lib --add-modules javafx.controls,javafx.fxml -jar target\hotelmanagement-1.0-SNAPSHOT.jar"`.

Troubleshooting

1. Fixing *"UnaccessibleObjectException: Unable to make field private final java.time.LocalDate java.time.LocalDateTime.date accessible: module java.base does not "opens java.time" to module gson"*.
 - Java 13 might be not properly worked in this setup, since some updated classes affect the module "gson" which is implemented to communication with web-server via JSON (JavaScript Object Notation)
 - To fix this error, you must install the **Java Runtime 11** or changing the enviroment variables to **Java version 11** (if you have another Java version installed on your computer). Otherwise, it won't work.
2. Fixing *"NullPointerException"*
 - This error occurs when the database is not properly installed. To fix this error, you must execute the "sqlConfig.sql" before running the web-server to make sure that data is available to fetch.
3. Fixing *"Port 8080 required by Tomcat v8.0 Server at localhost is already in use. There may already be running in another process, or a system process may be using the port. To start this server you will need to stop the other process or change the port number(s)"*.

- These errors cause as port 8080 is being occupied by another process.
- To fix this error, you need to follow step-by-step:
 - o **Step 1:** Find the process id that is using the port 8080: `netstat -ano | findstr 8080`.
 - o **Step 2:** Open command as administrator and kill the process: `taskkill /F /pid <pid>`.

Details setup

A. MySQL installation

1. Get can get MySQL Workbench and Server from here, you can use any MySQL workbench as your favourite, however, for convenient MySQL Workbench 8.0 CE is installed in this tutorial.
2. Install MySQL pack including MySQL Workbench and MySQL Server.

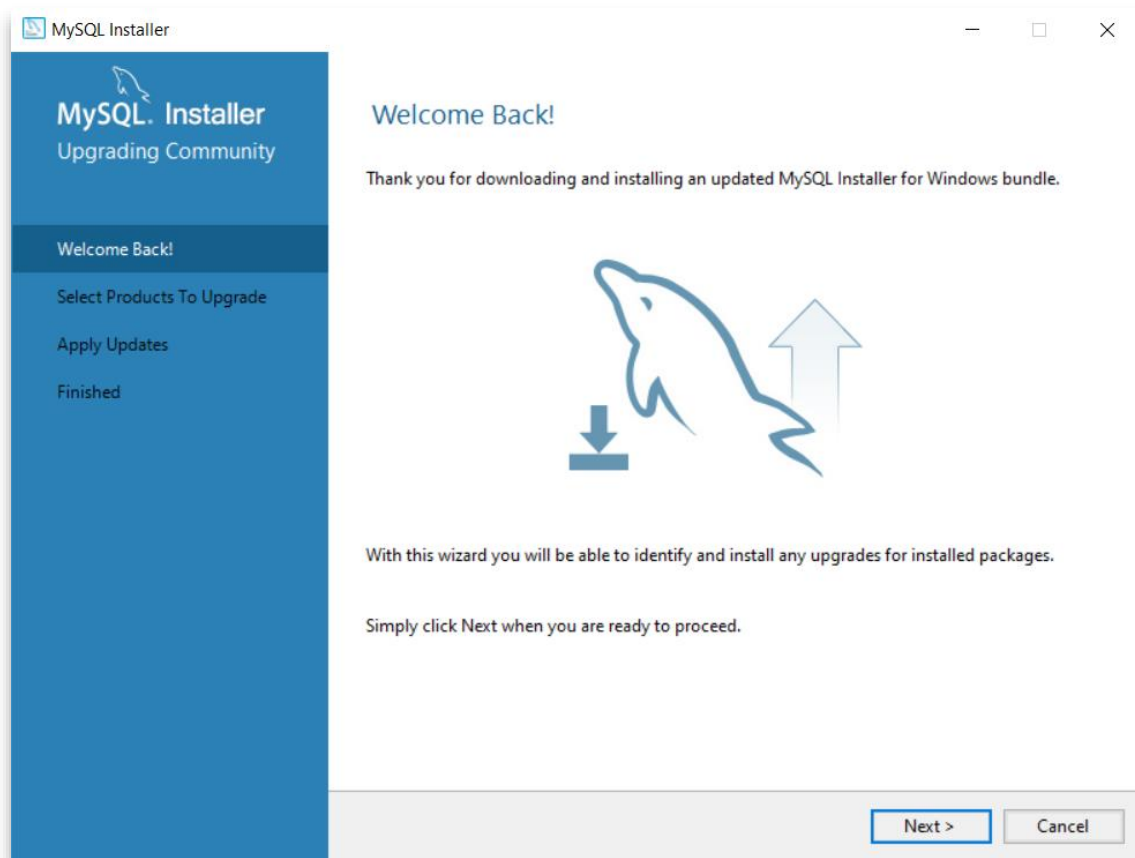


Figure 2-1: Welcome page

Step 1: Click the “Next” button to continue the installation

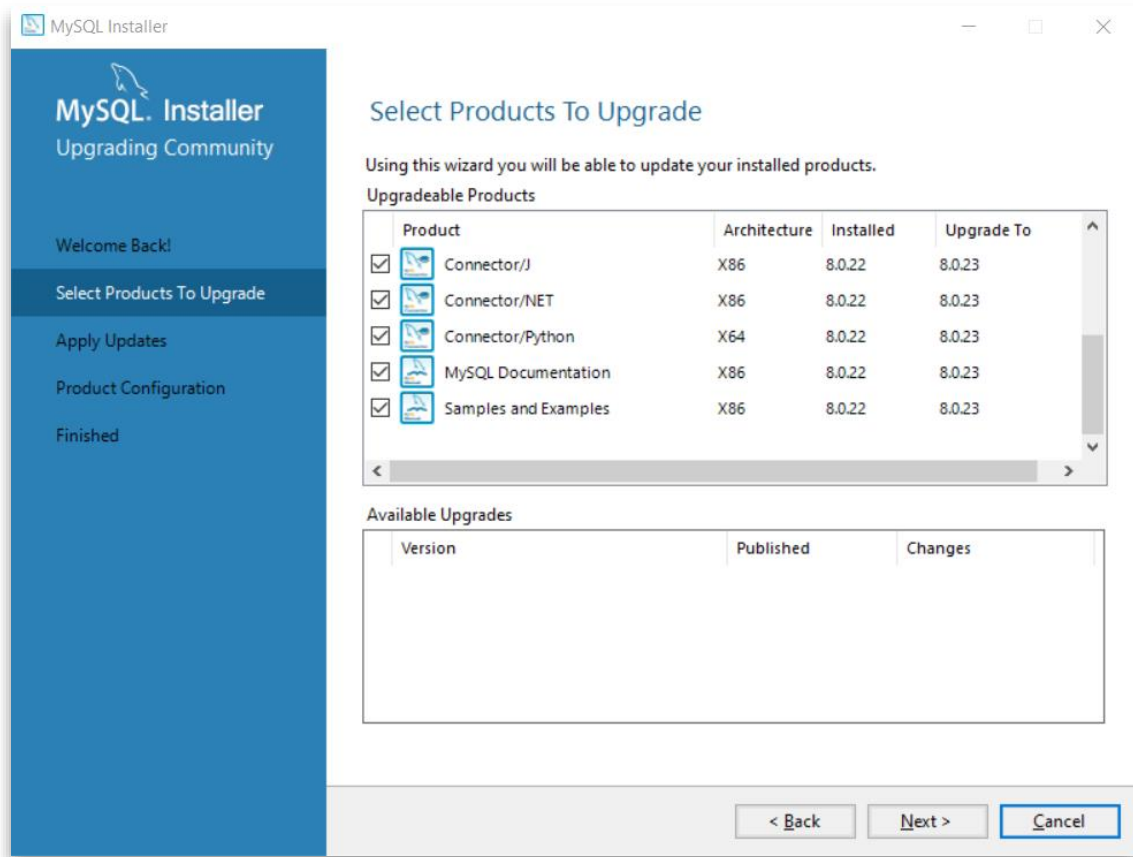


Figure 2-2: Select product to install

Step 2: Select the components that you would like to install. However, it's recommended that you should select all to make sure all components are installed for future usage. Then, click on the "Next" button to continue the installation.

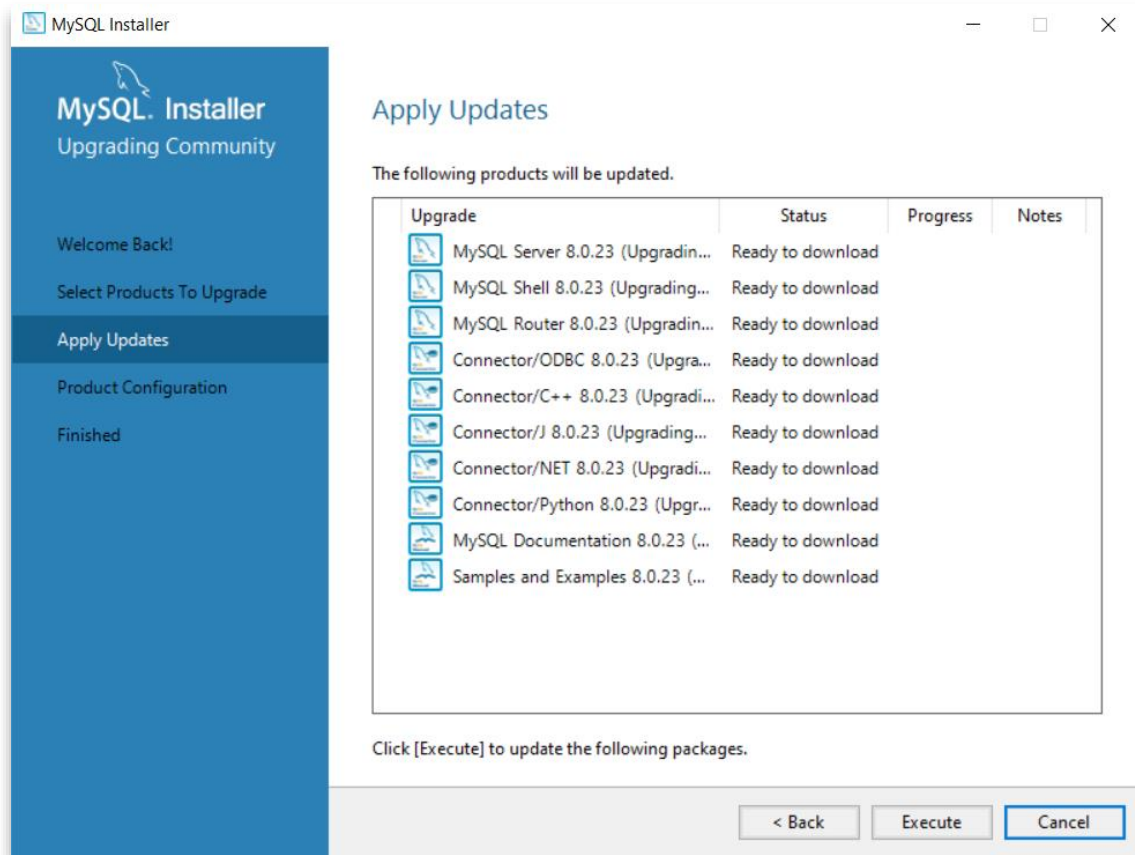


Figure 2-3: Apply updates & downloads

Step 3: You will have to wait for the installer to download all of the components. Click on the “Execute” button and wait for the downloading. It should take a while to complete the downloads.

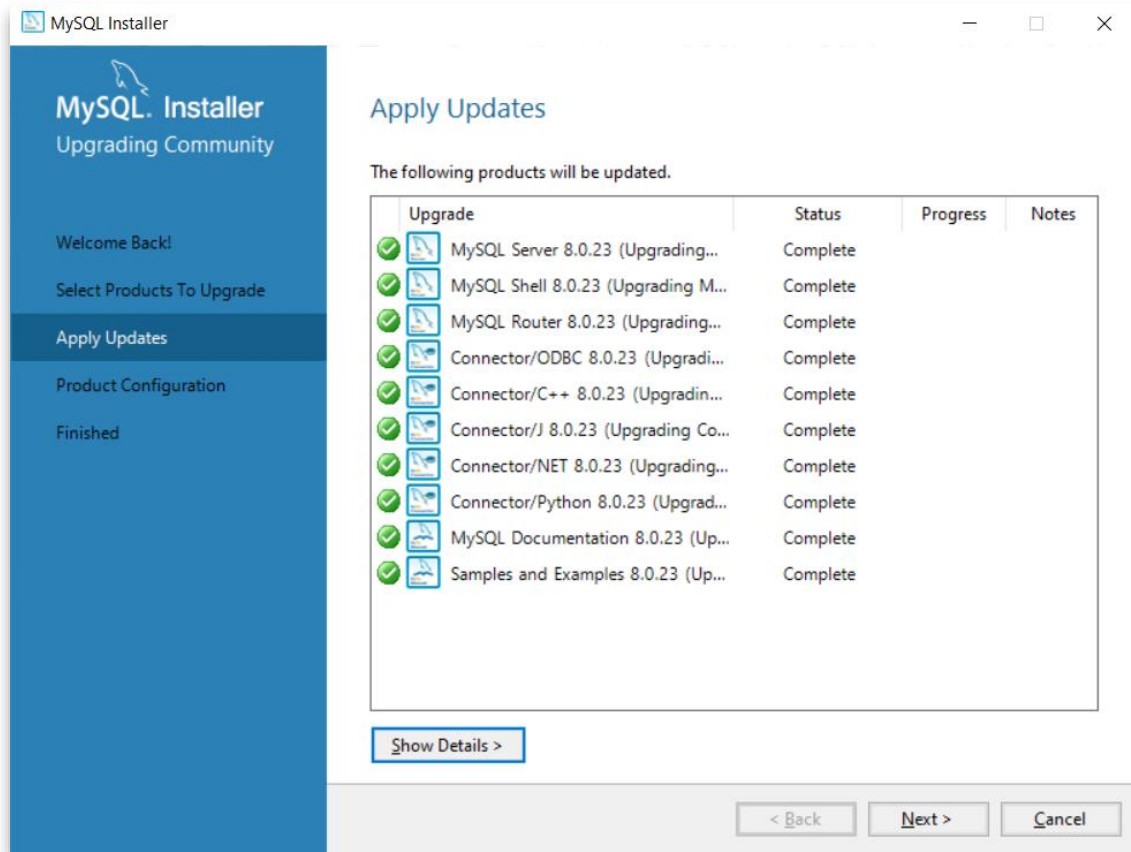


Figure 2-4: Apply Updates completed

Step 4: After downloading is complete, click on the “Next” button again to the next procedure

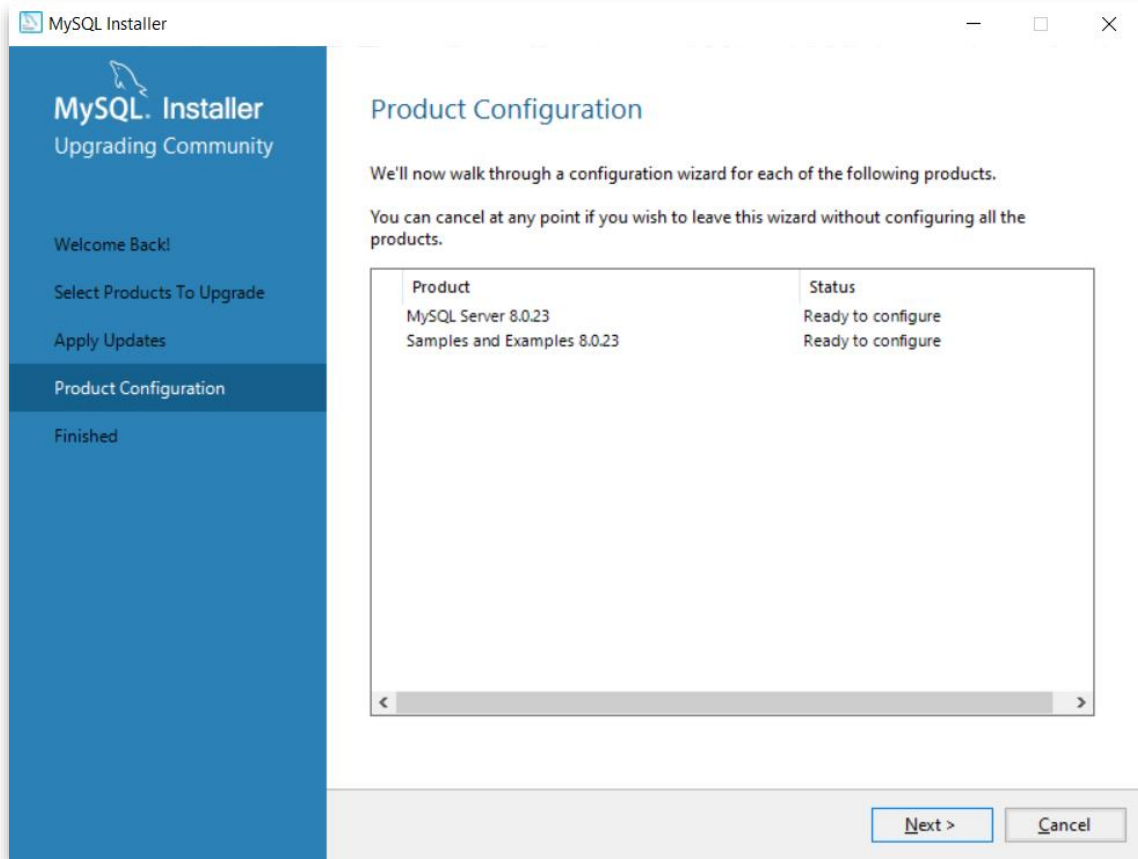


Figure 2-5: Product Configuration

Step 5: Click on the “Next” button to continue

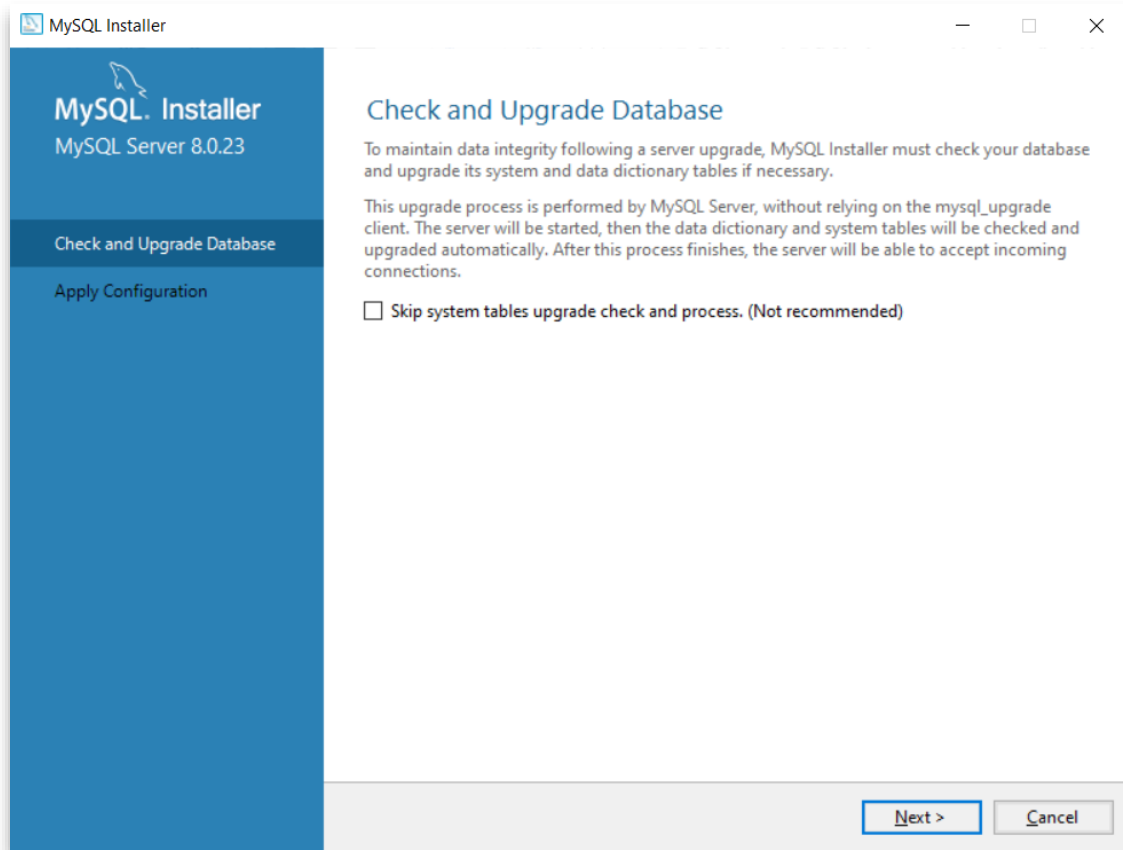


Figure 2-6: Check & Upgrade Database

Step 6 (Optional): You can skip this step. However, it does not make sure that the database configuration is up-to-date.

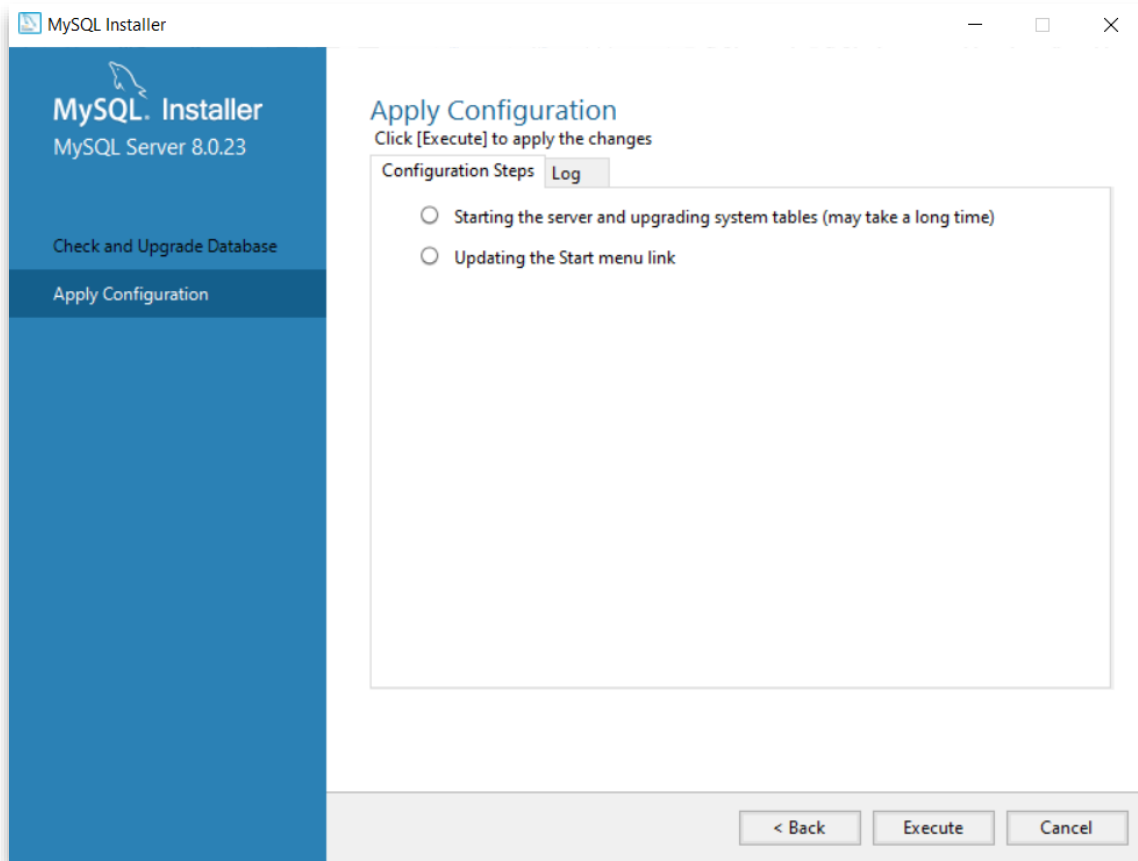


Figure 2-7: Apply the configuration

Step 7: Click on the “Execute” button to apply the configuration.

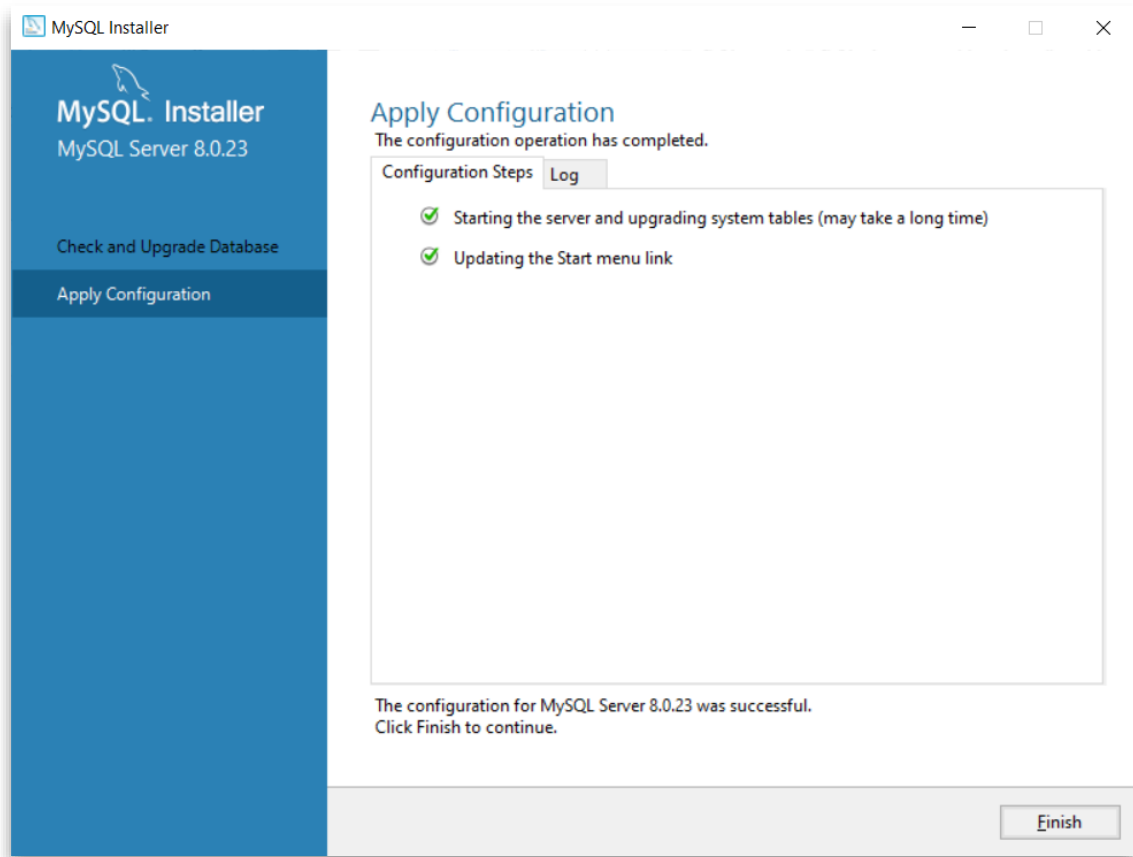


Figure 2-8: Apply configuration

Step 8: Click on the “Finish” button to finish the upgrade.

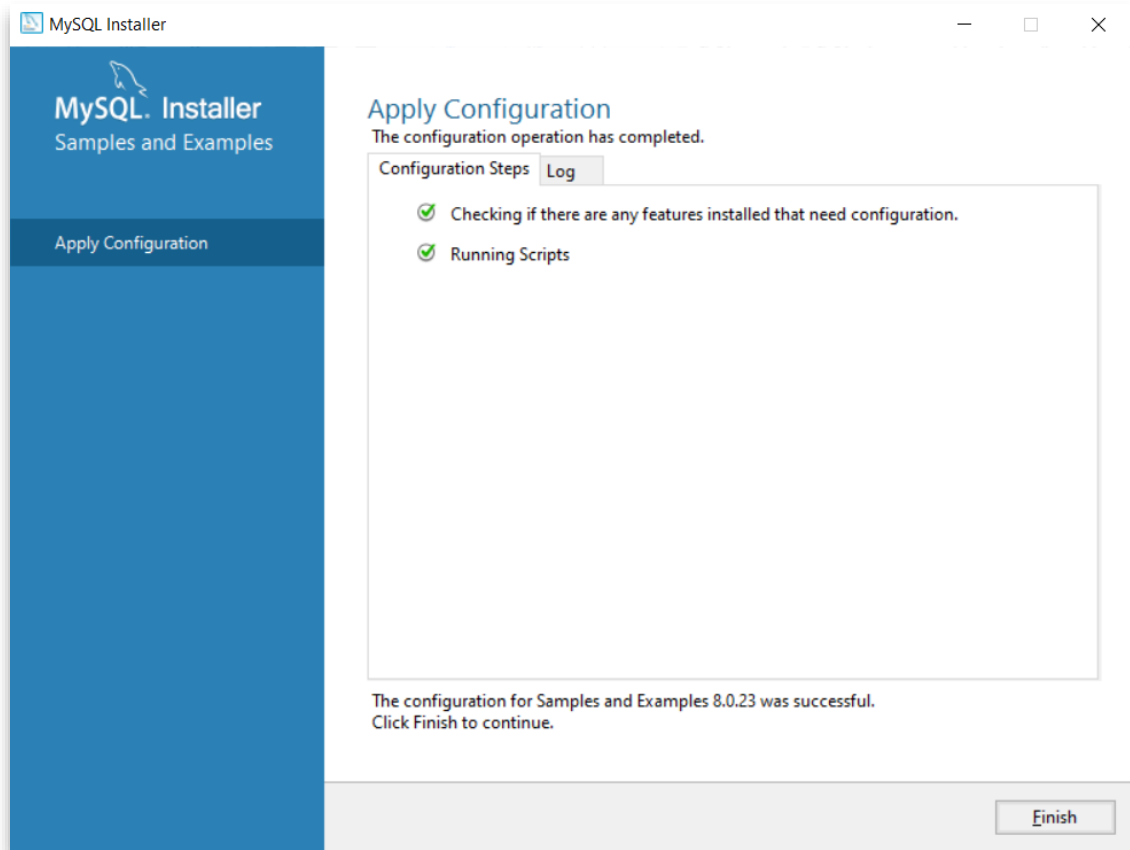


Figure 2-9: Apply Configuration

Step 9: Click “Finish” button to confirm the applied configuration.

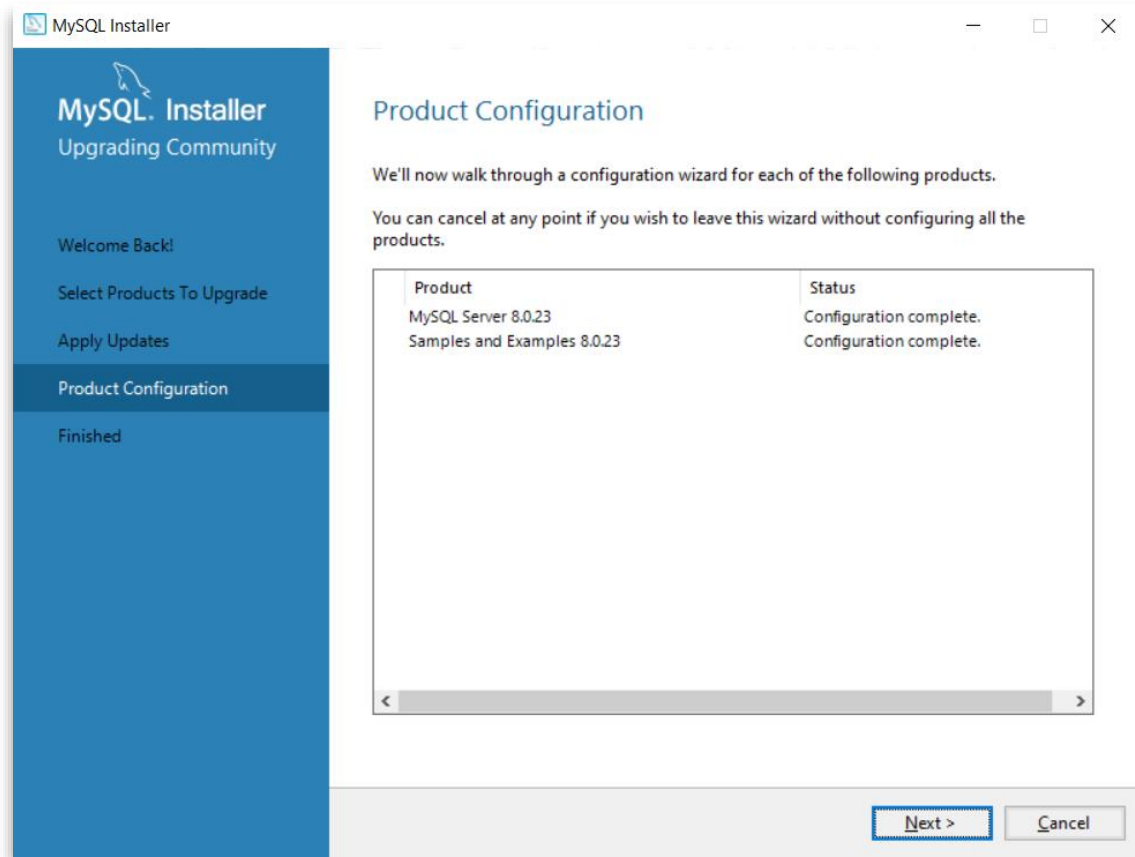


Figure 2-10: Product Configuration

Step 10: Click on the “Next” button to confirm the configuration.

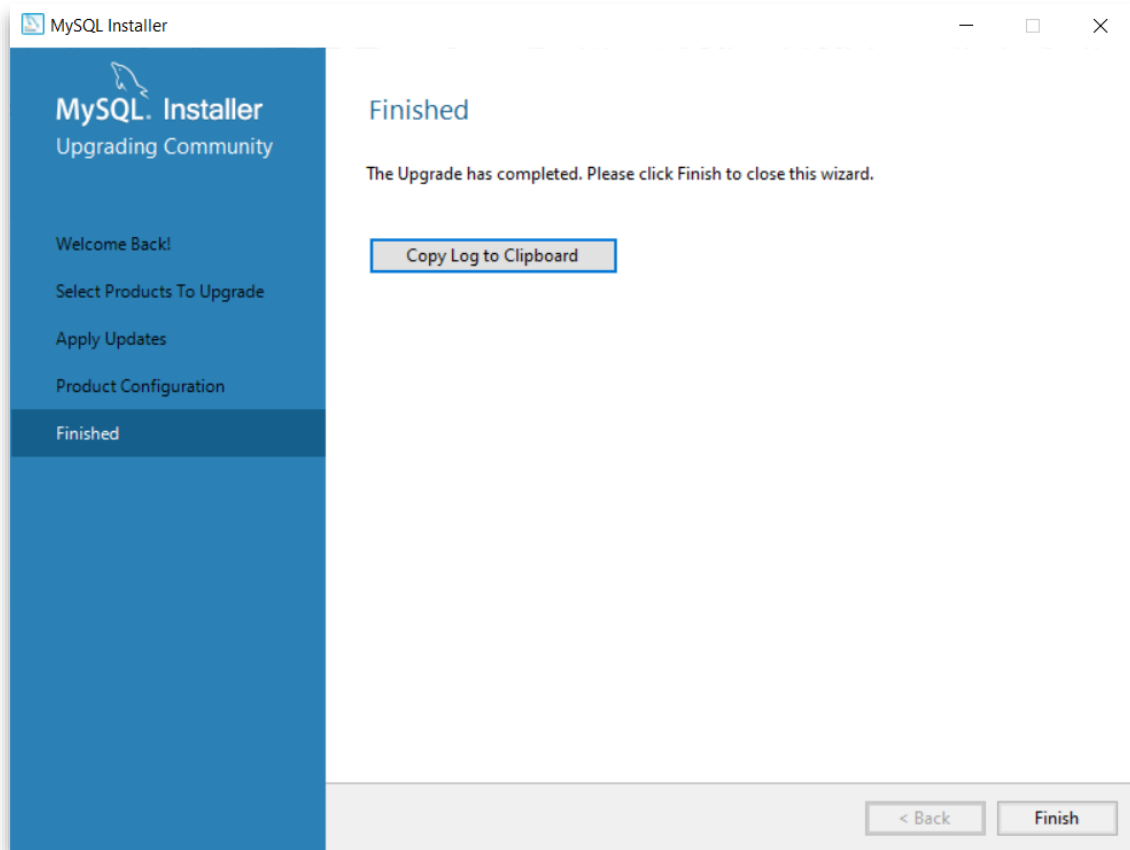


Figure 2-11: Finished

Step 11: Click on “Finish” button to complete the installation.

B. Tomcat installation

Step 1: Download Apache Tomcat

- Navigate [here](#)
- Choose the file named **32-bit/64-bit Windows Service Installer (pgp, sha512)** to download

Step 2: Setup

- Choose the **full** option after download the **Step 1** file like below image

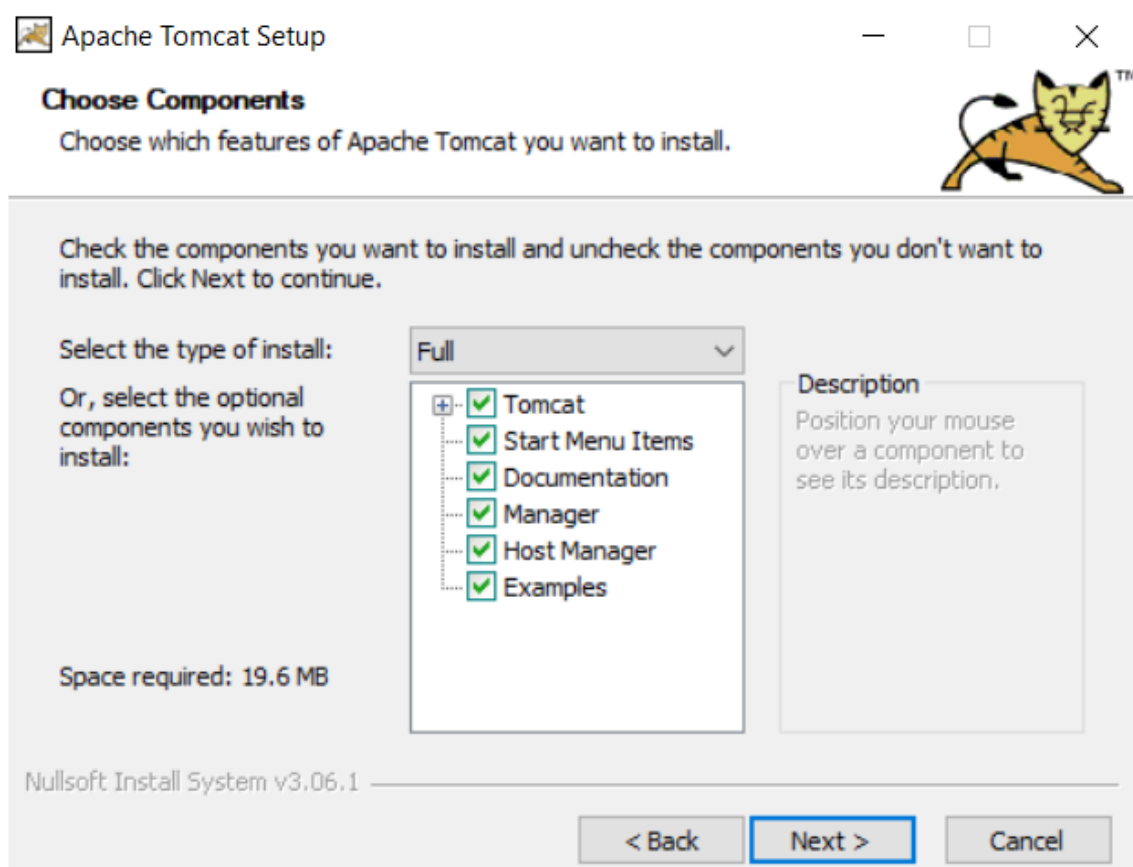


Figure 3-1: Apache Tomcat Setup

- Fill in the **user** and **password** form like below image

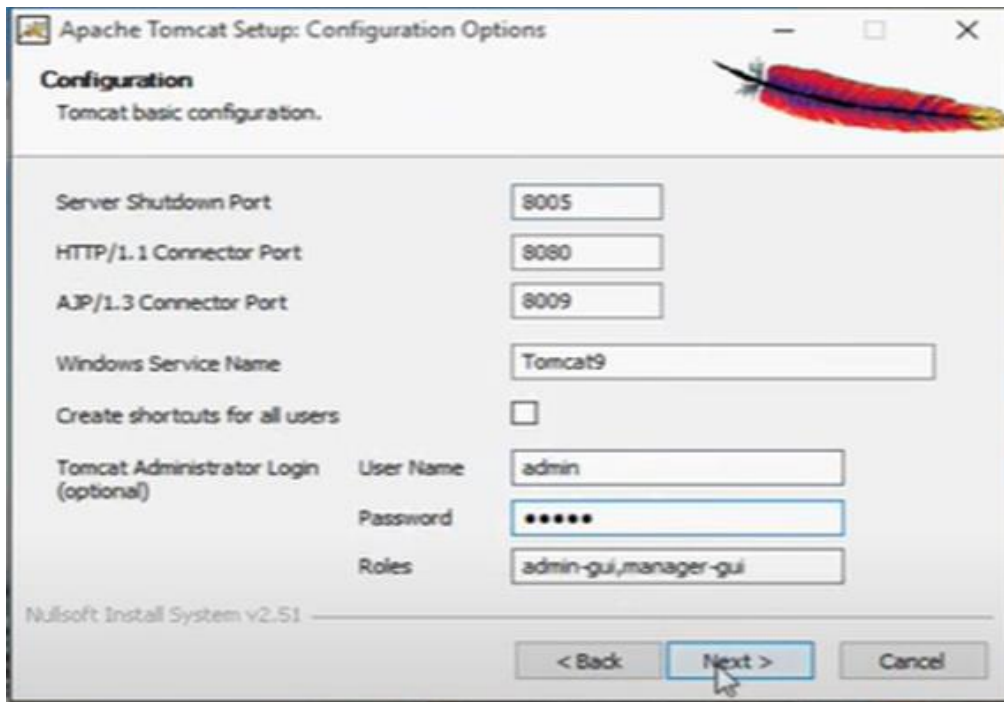


Figure 3-2: Tomcat configuration

Step 3: Check if success

- Navigate to **localhost:8080** on your web browser
- The below image indicates you have successfully installed Apache Tomcat 9 to your computer

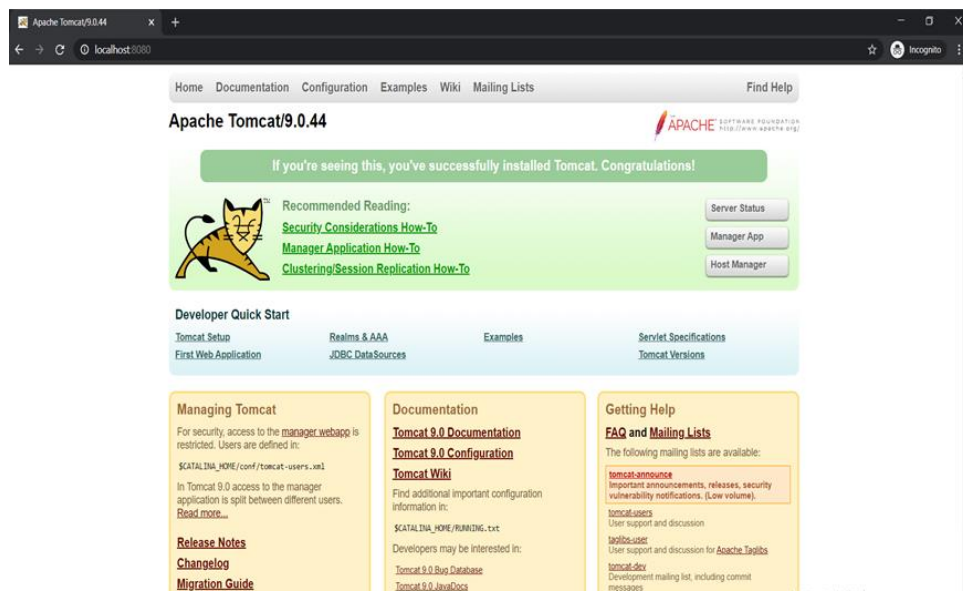


Figure 3-3: Tomcat successfully install

C. Maven installation

Step 1: Make sure Java has already been downloaded

- Download **Java-JDK** (the latest version), you can download it from [here](#)

Step 2: Create a user environment variable with following figures

- Name: **JAVA_HOME**
- Value: the directory to the path of **Java-JDK** stored in your computer

Step 3: Create a system environment variable (Path variable) with following figures

- Variable: **Path**
- Value: the directory path to the **bin** folder of the **Java-JDK** stored in your computer

Step 4: Download Apache Maven

- Navigate to [here](#) to download Apache Maven

Step 5: Create 2 user environment variables with following figures

- Name: **MAVEN_HOME** and **M2_HOME**
- Value: the directory path to the **Apache Maven** version you just downloaded

Step 6: Create a system environment variable (Path variable) with following figures

- Variable: **Path**
- Value: the directory path to the **bin** folder of the **Apache Maven** version you just downloaded

Step 7: Check whether you have successfully downloaded Maven

- Type command **mvn -version** to the command prompt
- The below picture indicates that Maven has been successfully installed

```
C:\Users\Ha Huy>mvn -version
Apache Maven 3.6.3 (cecedd343002696d0abb50b32b541b8a6ba2883f)
Maven home: C:\Program Files\Maven\apache-maven-3.6.3\bin\..
Java version: 15.0.1, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-15.0.1
Default locale: en_US, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

Figure 4-1: Maven successfully install

D. JavaFX SDK setup

Step 1: Download the JavaFX SDK package from [here](#) and exact to the folder you wish.

Step 2: Note your JavaFX SDK. It could be useful later on.