HOW TO RUN THE PROJECT –ZIDIO CONNECT

Complete step-by-step guide to run the Zidio Connect Job Finding Platform.

1.System Requirements

Minimum Requirements:

- **Operating System:** Windows 10/11, macOS, or Linux
- **RAM:** 4GB minimum (8GB recommended)
- **Storage:** 500MB free space
- **Internet:** Required for downloading dependencies

Required Software:

- **Java JDK 24.0.2** or higher
- **Apache Maven 3.9+**
- **MySQL 8.0+**
- **Eclipse IDE** (or IntelliJ IDEA)

2.Pre-Installation Setup

Step 1: Install Java JDK

Download and Install:

1. Visit: https://www.oracle.com/java/technologies/downloads/
2. Download **Java JDK 24** for your OS
3. Install to default location: C:\\Program Files\\Java\\jdk-24

Verify Installation:

```
cmd
java -version
```

Expected Output:

```
java version "24.0.2" 2025-10-15
Java(TM) SE Runtime Environment (build 24.0.2+7)
Java HotSpot(TM) 64-Bit Server VM (build 24.0.2+7, mixed mode)
```

Set JAVA_HOME (Windows):

```
setx JAVA_HOME "C:\\Program Files\\Java\\jdk-24"

setx PATH "%JAVA_HOME%\\bin;%PATH%"
```

Step 2: Install Apache Maven

Download and Install:

1. Visit: https://maven.apache.org/download.cgi
2. Download **Binary zip archive**
3. Extract to: `C:\\Program Files\\Apache\\maven`

Add to PATH (Windows):

```
setx PATH "C:\\Program Files\\Apache\\maven\\bin;%PATH%"
```

Verify Installation:

```
mvn -version
```

Expected Output:

```
Apache Maven 3.9.x

Maven home: C:\\Program Files\\Apache\\maven

Java version: 24.0.2
```

Step 3: Install and Configure MySQL

Download and Install:

1. Visit: https://dev.mysql.com/downloads/installer/
2. Download **MySQL Installer for Windows**
3. Run installer and select:
    - MySQL Server 8.0
    - MySQL Workbench (optional)
4. Set root password: raghu1234 (or your choice)
5. Complete installation

Verify MySQL is Running:

```
net start MySQL80
```

Create Database:

```
mysql -u root -p
```

Enter your password, then run:

```
CREATE DATABASE job_find;

USE job_find;

EXIT;
```

Step 4: Download Project Files

Option A: Download ZIP

1. Download project ZIP file
2. Extract to: C:\\JOBFIND\\

Option B: Clone from Git

bash

```
cd C:\\
git clone https://github.com/yourusername/zidio-connect.git
cd zidio-connect
```

Step 5: Configure Database Connection

1. **Navigate to:**

C:\\JOBFIND\\src\\main\\resources\\application.properties

2. **Open with Notepad**
3. **Update these lines:**

properties

```
spring.datasource.url=jdbc:mysql://localhost:3306/job_find
spring.datasource.username=root
spring.datasource.password=raghu1234
```

Replace raghu1234 with your MySQL password.

4. **Save the file** (Ctrl + S)

Method 1: Running from Eclipse IDE

Step 1: Import Project

1. **Open Eclipse IDE**

2. **Import Maven Project:**
    - File → Import
    - Maven → Existing Maven Projects
    - Click **Next**
3. **Select Project:**
    - Root Directory: Browse to C:\\JOBFIND
    - Click **Finish**
4. **Wait for Dependencies:**
    - Eclipse will download all dependencies
    - Check progress in bottom-right corner
    - Wait until "Building workspace" completes

Step 2: Update Maven Project

1. **Right-click** on project **JOBFIND**
2. Select **Maven → Update Project**
3. Check **Force Update of Snapshots/Releases**
4. Click **OK**
5. Wait for completion

Step 3: Run the Application

1. **Navigate to Main Class:**

```
src/main/java
└── com.JobFindingPlatform
    └── JobfindApplication.java
```

2. **Right-click** on JobfindApplication.java
3. **Select:** Run As → Java Application
4. **Check Console Tab** for output

Step 4: Verify Application Started

Look for these messages in Console:

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v2.7.15)
```

Started JobfindApplication in 5.234 seconds

Tomcat started on port(s): 8080 (http)

Adding welcome page: class path resource [static/index.html]

Success! Application is running.

Step 5: Access the Application

1. **Open any web browser**
2. **Go to:**

http://localhost:8080/

3. **You should see:**
    - Beautiful purple gradient homepage
    - "Find Your Dream Job Today" heading
    - Search bar and navigation buttons

Method 2: Running from Command Prompt

Step 1: Open Command Prompt

1. **Press** Windows + R
2. **Type:** cmd
3. **Press** Enter

Step 2: Navigate to Project

cmd

cd C:\\JOBFIND

Step 3: Verify Project Structure

cmd

dir

You should see:

pom.xml
src
target
README.md

Step 4: Start MySQL (if not running)

cmd

net start MySQL80

Step 5: Clean and Build Project

cmd

mvn clean install

This will:

- Clean previous builds
- Download dependencies
- Compile code
- Create JAR file

Wait for:

[INFO] BUILD SUCCESS
[INFO] Total time: XX.XXX s

Step 6: Run the Application

cmd

mvn spring-boot:run

Wait for:

Started JobfindApplication in X.XXX seconds
Tomcat started on port(s): 8080

Step 7: Access Application

Open browser:

http://localhost:8080/

Method 3: Running from IntelliJ IDEA

Step 1: Open Project

1. **Launch IntelliJ IDEA**
2. **Click:** Open
3. **Navigate to:** C:\\JOBFIND
4. **Select:** pom.xml
5. **Click:** Open as Project

Step 2: Wait for Indexing

- IntelliJ will index project files
- Wait for progress bar at bottom
- Maven will auto-download dependencies

Step 3: Configure JDK

1. **File → Project Structure**
2. **Project Settings → Project**
3. **SDK:** Select Java 24
4. **Language Level:** 24
5. **Click:** OK

Step 4: Run Application

1. **Locate:** JobfindApplication.java

src/main/java/com/JobFindingPlatform/JobfindApplication.java

2. **Click** the green play button (▶) next to class name
3. **Or:** Right-click → Run 'JobfindApplication.main()'

Step 5: Check Console

Look for startup messages in Run console:

Started JobfindApplication in X.XXX seconds

First-Time Setup

After Application Starts Successfully:

1. Create Admin Account (Optional)

Open MySQL Workbench or Command Line:

sql

```
USE job_find;
```

```
INSERT INTO users (name, email, password, role, is_active)
VALUES ('Admin', 'admin@zidio.com',
    '$2a$10$encrypted_password_here', 'ADMIN', true);
```

---

2. Test Basic Functionality

A. Register as Job Seeker:

1. Go to: http://localhost:8080/
2. Click **"Get Started"**
3. Fill form:
    - Name: John Doe
    - Email: john@example.com
    - Password: password123
    - Role: Job Seeker
4. Click **"Create Account"**

B. Login:

1. Click **"Login"**
2. Enter credentials
3. Click **"Login to Account"**

C. Browse Jobs:

1. Click **"Browse Jobs"**
2. View available positions

Testing the Application

Method 1: Browser Testing

Test Homepage:

http://localhost:8080/

✅Should show beautiful landing page

Test API Health:

http://localhost:8080/health

✅Should return: {"status":"UP"}

Test Job Listings:

http://localhost:8080/api/jobposts/all

✅Should return JSON array of jobs

Method 2: Postman Testing

1. Download Postman:

https://www.postman.com/downloads/

2. Test Register Endpoint:

POST http://localhost:8080/api/auth/register
Content-Type: application/json

Body:

```
{
  "name": "Test User",
  "email": "test@example.com",
  "password": "password123",
  "role": "JOBSEEKER"
}
```

3. Test Login:

POST http://localhost:8080/api/auth/login
Content-Type: application/json

Body:

```
{
  "email": "test@example.com",
  "password": "password123"
}
```

□ Stopping the Application

From Eclipse:

1. Go to **Console** tab
2. Click **Red Square** button (Terminate)
3. Or press Ctrl + F2

From Command Prompt:

1. Press Ctrl + C in the CMD window
2. Type Y when asked to terminate
3. Press Enter

From IntelliJ:

1. Click **Red Square** in Run toolbar
2. Or press Ctrl + F2

Troubleshooting

Issue 1: Port 8080 Already in Use

Error:

Web server failed to start. Port 8080 was already in use.

Solution - Windows:

cmd

```
netstat -ano | findstr :8080
taskkill /PID <PID_NUMBER> /F
```

Solution - Change Port: Edit application.properties:

properties

server.port=8081

Issue 2: MySQL Connection Failed

Error:

Failed to configure a DataSource

Communications link failure

Solution:

1. Start MySQL:

cmd

net start MySQL80

2. Verify credentials in application.properties
3. Test connection:

cmd

mysql -u root -p

Issue 3: Build Failed

Error:

[ERROR] Failed to execute goal

Solution:

cmd

mvn clean install -U

Or skip tests:

cmd

mvn clean install -DskipTests

Issue 4: Java Version Mismatch

Error:

java.lang.UnsupportedClassVersionError

Solution:

1. Check Java version:

cmd

java -version

2. Should be Java 24 or compatible
3. If wrong version, install correct JDK

Issue 5: Dependencies Not Downloading

Error:

Could not resolve dependencies

Solution:

cmd

mvn clean install -U

Check internet connection and proxy settings.

Issue 6: Page Shows 404

Error:

Whitelabel Error Page

This application has no explicit mapping for /error

Solution:

1. Verify index.html exists:

src/main/resources/static/index.html

2. Check console for:

Adding welcome page: class path resource [static/index.html]

3. Clear browser cache: Ctrl + Shift + Delete

Quick Start Scripts

Windows Batch Script

Create: run.bat in project root

batch

```
@echo off
title Zidio Connect - Starting...

echo ============================================
echo   ZIDIO CONNECT - JOB FINDING PLATFORM
echo ============================================
echo.

echo [1/3] Starting MySQL...
net start MySQL80
echo.

echo [2/3] Building project...
```

```
cd C:\\JOBFIND
call mvn clean install -DskipTests
echo.

echo [3/3] Starting application...
echo.
echo Server will be available at: http://localhost:8080
echo Press Ctrl+C to stop
echo ============================================
echo.

call mvn spring-boot:run

pause
```

Usage: Double-click run.bat to start application

Linux/Mac Shell Script

Create: run.sh in project root

bash

```
#!/bin/bash

echo "============================================"
echo "  ZIDIO CONNECT - JOB FINDING PLATFORM"
echo "============================================"
echo ""

echo "[1/3] Starting MySQL..."
sudo service mysql start
```

```
echo ""

echo "[2/3] Building project..."
cd ~/JOBFIND
mvn clean install -DskipTests
echo ""

echo "[3/3] Starting application..."
echo ""
echo "Server will be available at: http://localhost:8080"
echo "Press Ctrl+C to stop"
echo "============================================="
echo ""

mvn spring-boot:run
```

Make executable:

bash

```
chmod +x run.sh
./run.sh
```

Success Indicators

Application Running Successfully When You See:

1. **Console Output:**

   Started JobfindApplication in X.XXX seconds

   Tomcat started on port(s): 8080

2. **Browser Access:**
   - http://localhost:8080/ shows homepage
   - No 404 or 500 errors
3. **API Responses:**

- http://localhost:8080/api/jobposts/all returns data

4. **Database Connection:**

HikariPool-1 - Start completed

5. **No Errors:**
   - Console shows INFO/DEBUG only
   - No red ERROR messages

Getting Help

If application still won't run:

1. **Check Prerequisites:**
   - Java 24+ installed
   - Maven installed
   - MySQL running
   - Port 8080 free
2. **Review Logs:**
   - Check Eclipse/IntelliJ console
   - Look for red ERROR messages
   - Note exact error text
3. **Verify Configuration:**
   - application.properties correct
   - Database credentials valid
   - pom.xml not corrupted
4. **Try Clean Start:**

cmd

mvn clean

mvn install -U

mvn spring-boot:run

Success!

If you see the homepage at http://localhost:8080/, congratulations!

Your Zidio Connect application is now running!

Next Steps:

1. Create user accounts
2. Post some jobs
3. Test applications
4. Explore all features

Enjoy using Zidio Connect! □

Quick Reference

Essential Commands:

cmd

Start MySQL
net start MySQL80

Navigate to project
cd C:\\JOBFIND

Build project
mvn clean install

Run application
mvn spring-boot:run

Stop application
Ctrl + C

Essential URLs:

Homepage: http://localhost:8080/
API Docs: http://localhost:8080/api/
Jobs List: http://localhost:8080/api/jobposts/all
Health: http://localhost:8080/health