

# Hibernate CRUD Project Documentation

## Introduction

This project is a basic Hibernate-based CRUD application for managing user data. It demonstrates Create, Read, Update, and Delete (CRUD) operations using Hibernate ORM and Java. The database used is MySQL, and the application follows a layered architecture for scalability and modularity.

---

## Project Setup Instructions

### 1. Prerequisites

Ensure you have the following software installed:

- **Java Development Kit (JDK):** Version 8 or above  
Download: [JDK Download](#)
  - **Eclipse IDE:** Latest version recommended  
Download: [Eclipse Download](#)
  - **MySQL Database Server:** Version 5.7 or above  
Download: [MySQL Download](#)
- 

### 2. Required JAR Files

Download and include the following JAR files in the project's build path:

1. **Hibernate Core JAR**  
Download: [Hibernate ORM 5.x](#)
2. **MySQL Connector JAR**  
Download: [MySQL Connector](#)
3. **Hibernate Commons Annotations JAR**  
Download: [Hibernate Commons](#)
4. **Hibernate JPA API JAR**  
Download: [JPA API](#)
5. **Javax Persistence API JAR**  
Download: [Javax Persistence](#)

Download the JAR files, and copy them to a folder named `lib` within the project directory. Add them to the project's build path by:

1. Right-click the project in Eclipse.
2. Navigate to **Build Path > Configure Build Path > Libraries > Add JARs**.
3. Select the required JARs from the `lib` folder.

---

### 3. Database Setup

1. Open your MySQL client or MySQL Workbench.
2. Create a new database named `hibernate_crud` using the following command:

```
sql
Copy code
CREATE DATABASE hibernate_crud;
```

3. Use the following script to create the `users` table:

```
sql
Copy code
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    mobile VARCHAR(15),
    email VARCHAR(100),
    age INT
);
```

4. Note down your MySQL credentials (username and password) for the next step.
- 

### 4. Configure Hibernate

1. Open the `hibernate.cfg.xml` file in the project.
2. Update the database connection properties:

```
xml
Copy code
<property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/hibernate_crud</property>
<property name="hibernate.connection.username">[Your MySQL Username]</property>
<property name="hibernate.connection.password">[Your MySQL Password]</property>
```

3. Save the changes.
- 

### 5. Run the Project

1. Open the project in Eclipse.
2. Locate the `Main.java` file in the project.
3. Right-click the `Main.java` file and select **Run As > Java Application**.
4. Follow the menu options displayed in the console to interact with the CRUD operations.

---

## 6. CRUD Operations

- **Create User:** Add a new user by entering their name, mobile, email, and age.
  - **View Users:** Display all users with serial numbers and their details.
  - **Update User:** Update a specific user's details by entering their ID.
  - **Delete User:** Remove a user by entering their ID.
- 

## Project Structure

The project is organized into the following layers:

1. **Model Layer:** Contains the `User` class representing the database entity.
  2. **DAO Layer:** Handles direct interaction with the database using Hibernate.
  3. **Service Layer:** Contains business logic for CRUD operations.
  4. **Controller:** Provides a menu-driven interface for user interaction.
- 

## Notes

- Ensure MySQL service is running before executing the project.
  - If there are connection errors, verify the `hibernate.cfg.xml` settings and ensure the database exists.
- 

## FAQs

1. **Q: How do I add another entity?**  
A: Create a new entity class, update the `hibernate.cfg.xml` file, and replicate the DAO and Service layers for the new entity.
  2. **Q: What should I do if I encounter a `ClassNotFoundException`?**  
A: Ensure all required JAR files are added to the build path.
  3. **Q: Can I use a different database?**  
A: Yes, update the `hibernate.cfg.xml` with the appropriate database URL and credentials.
- 

## Contact Information

If you encounter any issues, feel free to reach out:

**Tejas Hande** – [Portfolio](#)

Email: [tejasamolhande@gmail.com](mailto:tejasamolhande@gmail.com)