**Task Master Application Setup**

This guide will walk you through the process of setting up Task Master, a desktop application for managing employees, tasks and projects. Task Master was built using NetBeans and uses Swing for its graphical user interface (GUI). It also uses the following additional libraries:

* **FlatLaf**: A library that sets the look and feel of the application
* **JCalendar**: A library that provides the JDateChooser component for selecting dates
* **MySQL Connector/J**: A library that provides connectivity with a MySQL database

**Prerequisites**

Before you can set up Task Master, make sure you have the following installed on your system:

* Java SE Development Kit (JDK) 8 or higher
* NetBeans IDE 14.0 or higher
* MySQL Server 5.6 or higher

**Installation**

To install Task Master, follow these steps:

1. Clone or download the Task Master repository to your local machine.
2. Open NetBeans IDE and select **Open Project** from the **File** menu.
3. Navigate to the location of the Task Master project and select the **TaskMaster** folder. Click **Open**.
4. Right-click on the project and select **Resolve Project Problems**. This will download any missing libraries and dependencies.
5. In the **Files** tab of the project, navigate to the **config** folder and open the **dbconfig.properties** file. Here, you will need to enter the URL, username, and password for your MySQL database.
6. In the **Files** tab of the project, navigate to the **ddl** folder and run the **ddl.sql** script to create the necessary tables in your MySQL database.

**Usage**

To run Task Master, follow these steps:

1. Open the **Main** class located in the **main** package of the project.
2. Run the **Main** class by clicking the **Run Project** button or pressing **F6** on your keyboard.
3. Once the application is launched, you can start managing tasks and projects.

**Overview**

Task Master is a desktop application that allows users to manage employees, tasks and projects. The application uses a MySQL database to store task and project information, and provides a user-friendly interface for creating, assigning, and tracking tasks.

The application is comprised of the following classes:

* **Employee**: Represents an employee with information such as full name, email, phone number, date of birth, monthly salary and tasks.
* **Task**: Represents a task with information such as title, description, assigned employee, due date, and project.
* **Project**: Represents a project with information such as name, description, start date, end date, budget, and tasks.

Note that a task may or may not have an assigned employee, but it must be associated with a project.

By following the installation and usage instructions provided above, you should be able to set up and use Task Master for your task and project management needs.

**Folder structure overview :**

**controllers**: This directory contains three controllers: EmployeeController, TaskController, and ProjectController. These controllers are responsible for handling the business logic of the application and for communicating between the views and the model.

**images**: This directory contains images used in the application.

**main**: This directory contains the Main class, which is the entry point for the application. This class is responsible for setting up the application and starting the user interface.

**model**: This directory contains the entity classes that represent the data of the application. These classes define the structure and behavior of the data, such as its attributes and relationships.

**persistence.connection**: This directory contains the DBConnectionFactory class for establishing a database connection.

**persistence.dao**: This directory contains DAO interfaces that define operations that can be performed on entities.

**persistence.dao.impl**: This directory contains implementations of the DAO interfaces. These classes provide the actual implementation of the DAO methods, using the database connection to read and write data.

**session**: This directory contains session classes for mapping entities (Employee) .

**uI.form**: This directory contains form classes that display and collect data from the user.

**uI.form.table.model**: This directory contains model classes for displaying tabular data in the user interface.