Overview

Project Interfaces

DisasterVictimManagenment Interface

The DisasterVictimManagement interface facilitates data entry and management for disaster victims within the relief system. It enables relief workers to:

- Add New DisasterVictims:
  Input and store essential victim details such as name, date of birth/approximate age.

- Manage Relationships:
  Establish and remove relationships between disaster victims (e.g., family ties).

- Add Medical Records:
  Record and maintain medical treatment details and dates for each victim.

- Search and View Information:
  Retrieve and display comprehensive information about specific disaster victims, including personal data, relationships, and medical history.

Please refer to the project documentation or source code for detailed information on the methods and implementations associated with this interface.

ReliefWorker Interface

The ReliefWorker interface enables both local and central relief workers to access and manage relief operations. Local workers can add disaster victims, manage supplies, and maintain individual records specific to their location. Central workers, on the other hand, can log inquirer queries, search for disaster victims centrally, and perform various operations related to central relief tasks like log management and coordination.

In order to access the interface you will have to enter the username and password.
For testing you can use the following:

- **For central worker interface**   - **For location worker interface**
  Username: central_worker1   Username: location_worker1
  password: password1   password: password1

- **Central Relief Worker Interface**:

  - Provides authentication and access for central relief workers.

  - Allows central relief workers to log inquirer queries and centrally search for disaster victims.

  - Enables display of logs and execution of various operations related to central relief tasks.

- **Location-Based Relief Worker Interface**:

  - Facilitates authentication and access for location-based relief workers.

  - Empowers location-based relief workers to add new disaster victims, manage supplies, and maintain records for individuals at specific locations.

  - Supports supply management tasks including adding new supply items, updating quantities, and viewing the supplies list.

Please refer to the project documentation or source code for detailed information on the methods and implementations associated with this interface.

Instructions for Users

This part provides instructions for setting up the environment to run the project. Please follow the steps below to ensure the smooth execution of the project.

Pre-requisites

Before proceeding, ensure you have the following:

- Java Development Kit (JDK) installed on your system.

- Access to the internet to download PostgreSQL and the PostgreSQL JDBC driver.

- Command line terminal or shell access.

Steps to Follow

Step 1: Download and Install PostgreSQL

1. Go to the [PostgreSQL download page](#).

2. Choose the appropriate version for your operating system (e.g., Windows, macOS, Linux) and download the installer (version 16 was used for the development of the project).

3. Run the downloaded installer and follow the on-screen instructions to install PostgreSQL on your system.

4. During the installation process, make sure to remember the password you set for the default PostgreSQL user (`postgres`).

Step 2: Install PostgreSQL JDBC Driver

1. Download the latest PostgreSQL JDBC driver from the [official PostgreSQL JDBC website](#).

2. Extract the downloaded ZIP file to obtain the JAR file (`postgresql-X.X.X.jar`, where `X.X.X` represents the version number).
   NOTE- The one that was used during the building of project was postgresql-42.7.3.jar

Step 3: Set Up the Project

1. Open the project in the IDE.

2. Ensure that the PostgreSQL JDBC driver JAR file is included in the project's build path.

3. Also make sure that you have hamcrest-core-1.3.jar and junit-4.13.2.jar installed and included in the project.

Step 4: Grant Privileges to User

1. Open a command line terminal or shell.

2. Navigate to the `bin` directory in the PostgreSQL installation directory.

3. Enter the following command to log in to the PostgreSQL server:

```
- psql -U username
```

Replace `username` with your PostgreSQL username.

4. Enter your password when prompted.

5. Once logged in, enter the following command to connect to ensf380project database and execute the SQL commands from the provided project SQL file:

```
- \i path/to/your/project.sql
```

Replace `path/to/your/project.sql` with the actual path to your project SQL file (Example would be: \i C:/Users/username/Downloads/project.sql).

6. Next, execute the following SQL commands to grant privileges to the `oop` user for smooth database operation:

```
GRANT CREATE, CONNECT, TEMPORARY ON DATABASE ensf380project TO oop;
GRANT USAGE ON SCHEMA public TO oop;
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA public TO oop;
GRANT USAGE, SELECT ON ALL SEQUENCES IN SCHEMA public TO oop;
```

After running these commands you will be able to smoothly run the interfaces and make modifications to the database as oop.

Step 5: Run the Application

1. Compile and run the rest of the project (java files and test files) using the IDE of your own choice.

   1. **IntelliJ IDEA Documentation:**
      - Official IntelliJ IDEA documentation covers various aspects of Java development, including project setup, compilation, running tests, and more.
      - Link: [IntelliJ IDEA Documentation](#)
   2. **Visual Studio Code - Java Development Guide:**
      - The official Visual Studio Code documentation for Java development includes setup instructions, using extensions, and running tests.
      - Link: [Java in Visual Studio Code](#)
   3. **FreeCodeCamp: How to Run Java in Command Prompt**
      - FreeCodeCamp offers useful resources for learning Java programming, including tutorials on compiling and running Java programs using command-line tools.
      - Link: [FreeCodeCamp - How to Execute and Run Java Code](#)

Java Documentation Command (Windows)

Step 1: Open Command Prompt

- **Open Command Prompt**: Press `Win + R`, type `cmd`, and press `Enter` to open Command Prompt.

Step 2: Navigate to Your Source Code Directory

- **Change Directory**: Use `cd` command to navigate to your Java source code directory.

```
  cd path/to/your/source/code

Exmaple:
cd C:\Users\ami_s\Downloads\AI3\AI_3\src
```

Step 3: Executing JavaDoc command

- Since the source code directory also contains the test files you need to use the following command for documentation:

```
Command: javadoc -d docs -sourcepath . -classpath ".;path/to/your/junit-x.x.x.jar" your.package.name

Example:
javadoc -d docs -sourcepath C:\Users\ami_s\Downloads\AI3\AI_3\src -classpath ".;C:\Users\ami_s\Downloads\AI3\AI_3\src\edu\ucalgary\oop\junit-4.13.2.jar" edu.ucalgary.oop
```

This ensures that all dependencies are included and Java documentation is generated smoothly.

Contact Information

If you encounter any issues or have questions regarding the setup process, please contact amina.sagheer@ucalgary.ca.