Lagerbank-App

Übersicht

Installation der Lagerbank App (Windows 10)

Die nachfolgenden Schritte der Reihe nach ausführen um die Lagerbank-App zu installieren.

Vorbereitung

1. Chocolatey installieren.

Chocolatey ist Windows Paketmanager. Über Chocolatey werden nachfolgend weitere Programme installiert.

Für die Installatoin der offiziellen Anleitung folgen. Am einfachsten geht es über die Powershell (als Admin): https://docs.chocolatey.org/en-us/choco/setup

2. Java installieren.

Das Programm ist in Java geschrieben und benötigt daher ein installiertes JDK (Java Development Kit), mit welchem das Program später gebaut und ausgeführt wird.

Die Installation erfolgt über Chocolatey. Dazu eine Powershell (als Admin) öffnen und folgenden Befehl ausführen:

```
choco install openjdk –version=11.0
```

Anschließend die Installation mit einen weiteren Befehl testen:

```
java --version
```

3. Maven installieren.

Maven ist ein sogenanntes Build-Tool, welches genutzt wird um aus dem Quellcode das ausführbare Programm zu bauen.

Die Installation erfolgt wieder über Chocolatey. Dazu eine Powershell (als Admin) öffnen und folgenden Befehl ausführen:

```
choco install maven
```

Anschließend die Installation mit einem weiteren Befehl testen:

```
mvn --version
```

Programm bauen und ausführen

Die Installation benötigt die in *Vorbereitung* beschriebenen Programme.

1. Quellcode **herunterladen**: https://github.com/joshuaju/ljw-applications/archive/refs/heads/master.zip
2. Heruntergeladene ZIP-Datei in einem beliebigen Ordner **entpacken** (z.B. im Downloads Ordner).
3. Anschließend im entpackten Ordner *ljw-applications-master* (in diesem Ordner liegt die Datei *pom.xml)* eine **Powershell öffnen**.

4. Dann einen Maven-Befehl ausführen um das Programm zu **bauen**

```
mvn clean install -DskipTests
```

5. Mit einem weiteren Befehl das Programm **ausführen**

```
java -jar client/target/client.jar
```

Alternativ kann man im Datei-Browser in den Ordner *ljw-applications-master/client/target* navigieren und dann **mit Doppelklick auf *client.jar* das Porgramm ausführen**. Die Datei kann an einen beliebigen Ort kopiert und dort ausgeführt werden (z.B. auf den Desktop).

Funktionen

Übersicht

- Kontoverwaltung
  - Erstellen
  - Bearbeiten
- Transaktionen
  - Einzahlung
  - Auszahlung
  - Überweisung
- Aktionen
  - Kassensturz: Summe aller Kontostände. Der berechnete Betrag sollte in Bar in der Kasse vorhanden sein.
  - Guthaben prüfen: Listet alle Konten mit negativem Guthaben auf.
  - Konten importieren: Erstellen von neuen Konten mit einem Einzahlungsbetrag (ausführliche Beschreibung siehe unten)
- konfigurierbarer Datenablageort (ausführliche Beschreibungim im Abschnitt *Daten*)

Konten importieren per CSV-Import

Der CSV-Import ermöglicht es mehrere Konten mit einem initialem Guthaben auf einmal zu importieren. Dieses Feature ist hilfreich zu Beginn der Freizeit, um einmalig das Taschengeld aller Teilnehmer*innen und Betreuer*innen in die Lagerbank App zu übertragen.

Der Import erzeugt neue Konten und zahlt die jeweiligen Beträge darauf ein. Sofern ein Konto bereits existiert ist der Import nicht möglich und kann stattdessen über die Kontoverwaltung gemacht werden.

Das Feature ist zugänglich über *Aktionen > Importiere Konten.*

Anleitung

1. Zuerst die Importdatei auswählen.
2. Eine Beschreibung eingeben, welche als Betreff für die Einzahlungen verwendet wird.
3. Auf *Anwenden* klicken um den Import zu starten.
4. Das Protokoll *Ergebnisse* beachten um eventuelle Fehler mitzubekommen.

Format-Beschreibung

- Trennzeichen der Spalten ist ein Komma

- die Spalten der CSV-Datei sind festgelegt auf: first name, last name, balance
- das Dezimaltrennzeichen für das Guthaben (balance) ist Punkt. Die Verwendung von Komma als Dezimaltrennzeichen führt zu einem Fehler.
    - gültige Beispiele: 22.50 oder 22 oder 0.00
    - ungültige Beispiele: 22,50 oder 0,00

Beispieldatei:

first name, last name, balance
Anna Marie, Neubauer, 75.50
Anton, Hilger, 25
Lisa-Marie, Franken, 0

Daten

Datenhaltung

Die Daten die bei der Benutzung der Applikation entstehen werden in CSV-Dateien gespeichert. Insgesamt werden zwei Dateien erzeugt:

- Benutzerkonten: Die Datei *accounts.csv* beinhaltet die erstellten Konten. Gespeichert werden eine erzeugte ID, sowie Vor- (first name) und Nachname (last name).
- Transaktionen: Die Datei *transactions.csv* beinhaltet die getätigten Transaktionen. Gespeichert werden das Ausführungsdatum (date), die ID des Sender-Kontos (source), die ID des Empfänger-Kontos (target), der Betrag (amount) und die Beschreibung/der Betreff (description). Bei Einzahlungen ist das Sender-Konto mit einem Platzhalter versehen (DEPOSIT), bei Auszahlung ist das Empfänger-Konto (WITHDRAWAL).

Der Ablageort der Datei ist standardmäßig im Verzeichnis aus dem das Programm ausgeführt wird. Jedoch ist der Pfad überschreibbar, in dem bei Programmstart die Pfade angegeben werden. Dabei wird als erstes Argument die Datei für die Konten und als zweites Argument die Datei für Transaktionen angegeben.

```
java –jar client.jar <accounts> <transactions>
```

Zum Beispiel also:

```
java –jar client.jar C:/accounts.csv C:/transactions.csv
```

Datensicherung

Die Applikation speichert Daten unmittelbar nach der Ausführung von Aktionen (z.B. Kontoerstellung, Überweisung getätigt) in den entsprechenden Dateien. Somit sind Daten bei einem eventuellen Programmabsturz bereits gespeichert. Daher ist es **nicht nötig manuell zu speichern**.

Es empfiehlt sich die erzeugten Dateien separat zu sichern (z.B. Cloud, USB-Stick, externe Festplatte), um bei einem Geräteschaden oder -verlust die Daten nicht zu verlieren. Die Dateien können auch auf einem anderen Computer verwendet werden, sofern sie im ausführenden Verzeichnis (meisten neben *client.jar*) abgelegt werden. Bei Programmstart werden die Datei eingelesen anstatt neu erzeugt.