Banking app

This is a model banking app which can also be used for real purposes such as budget management and tracking spending. I created it by myself as a hobby and to expand my knowledge and demonstrate understanding in java class structures, GUI modeling, data structures, and use of APIs. This is version 2 of this app and includes several new features such as use of a database, better user experience, exporting of data, and timestamps for transactions.

This project might still have issues that haven't been found. The biggest difficulty in making this was creating a database to store data. The current database is a public txt file read by a scanner, which isn't secure or efficient. I plan on using better types of databases in future projects. Another major difficulty was using maven dependencies, it was my first time using them and I had to learn how to write the pom.xml file and read several API docs to figure out the suitable APIs to use.

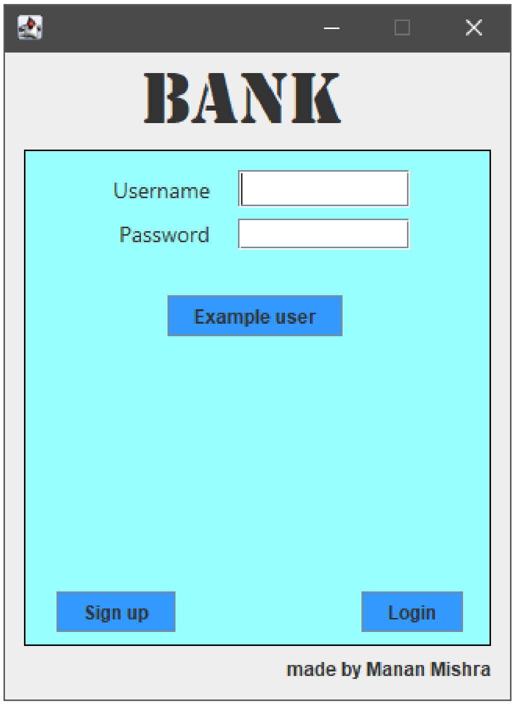
Overall, this project took about 20 hours over the course of 2 weeks and I learned a lot about several different java concepts and how they all work with each other when packaging a real application.

APIs

- The JFrame framework was used to create the GUI
- Mindrot jbcrypt-0.4 was used to encrypt passwords to be stored in the database
- Apache pdfbox-3.0.0 was used for exporting pdfs

```
 <dependencies>
    <dependency>
        <groupId>org.mindrot</groupId>
        <artifactId>jbcrypt</artifactId>
        <version>0.4</version>
    </dependency>
    <dependency>
        <groupId>org.apache.pdfbox</groupId>
        <artifactId>pdfbox</artifactId>
        <version>3.0.0</version>
    </dependency>
</dependencies>
```

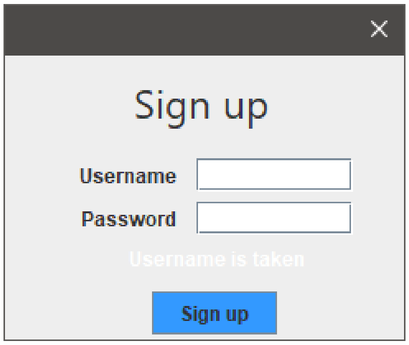
App walkthrough

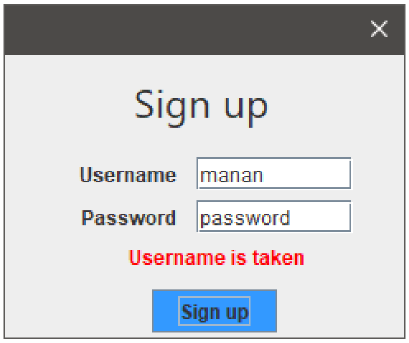


This is the login and screen and the first window shown after opening app

The example user button fills in the username and password fields with “manan” and “password” respectively. This user is a model user and has a long transaction history.

The sign up button opens the sign up window which is where new users can sign up with a username and password. The username must be unique.



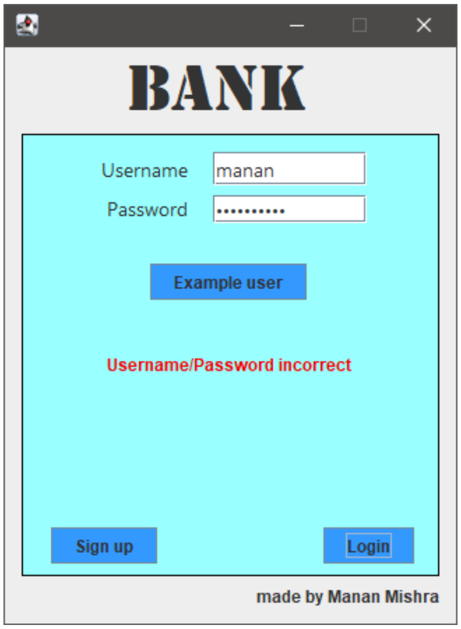


Username manan

Password password

Username is taken

The login button checks if the username and password combos are right, if not, it shows an error message



BANK

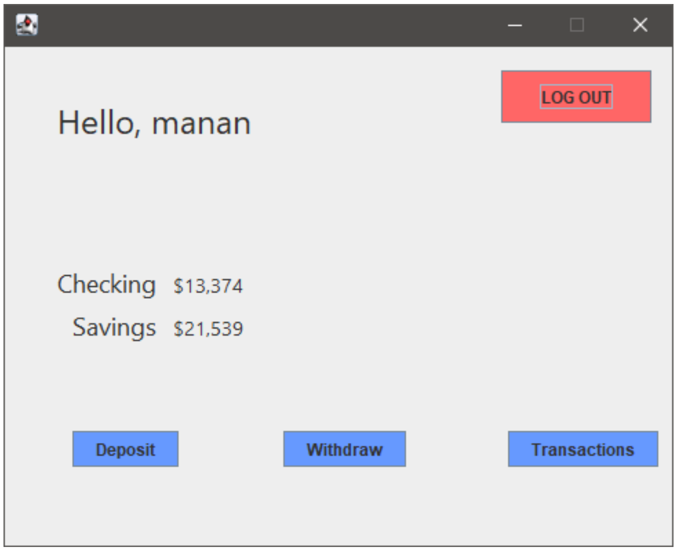
Username manan

Password ••••••••••

Username/Password incorrect

made by Manan Mishra

If the combo was right, a new account window opens



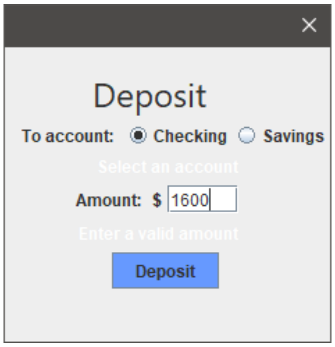
Hello, manan

Checking $13,374

Savings $21,539

This window shows the account balances, and allows the user to perform several actions.

The deposit button is used to deposit money into either one of the two accounts, the value given must be a valid double



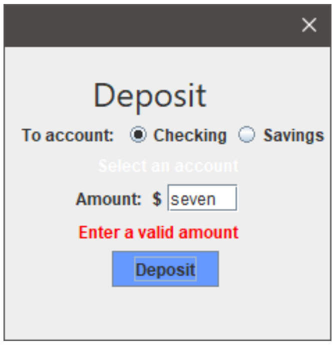
Deposit

To account: ● Checking ○ Savings

Select an account

Amount: $ 1600

Enter a valid amount



Deposit

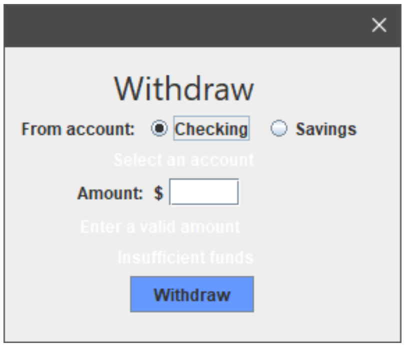
To account: ● Checking ○ Savings

Select an account

Amount: $ seven

Enter a valid amount

The withdraw button is used to withdraw money from either account, the value must be lower than or equal to balance and it must be a valid double



Withdraw

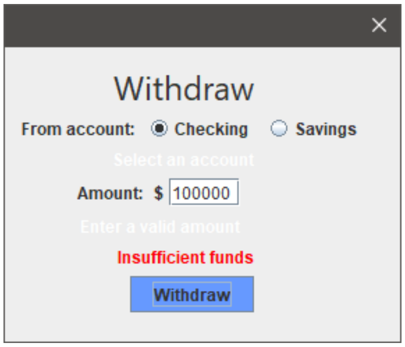
From account: ● Checking ○ Savings

Select an account

Amount: $

Enter a valid amount

Insufficient funds



Withdraw

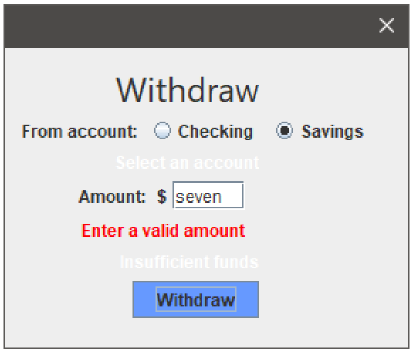
From account: ● Checking ○ Savings

Select an account

Amount: $ 100000

Enter a valid amount

Insufficient funds



Withdraw

From account: ○ Checking ● Savings

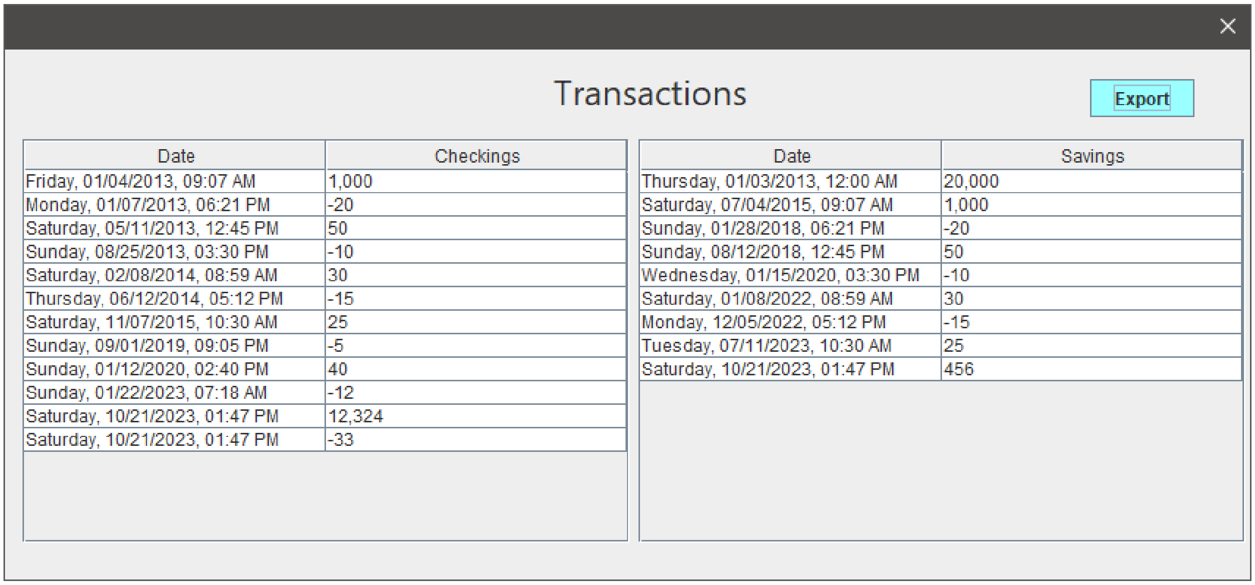
Select an account

Amount: $ seven

Enter a valid amount

Insufficient funds

The transactions button is used to show all transaction made in both accounts, it also shows the date and time of transactions.

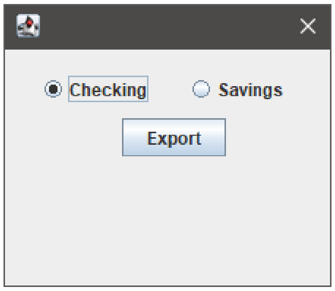


Transactions

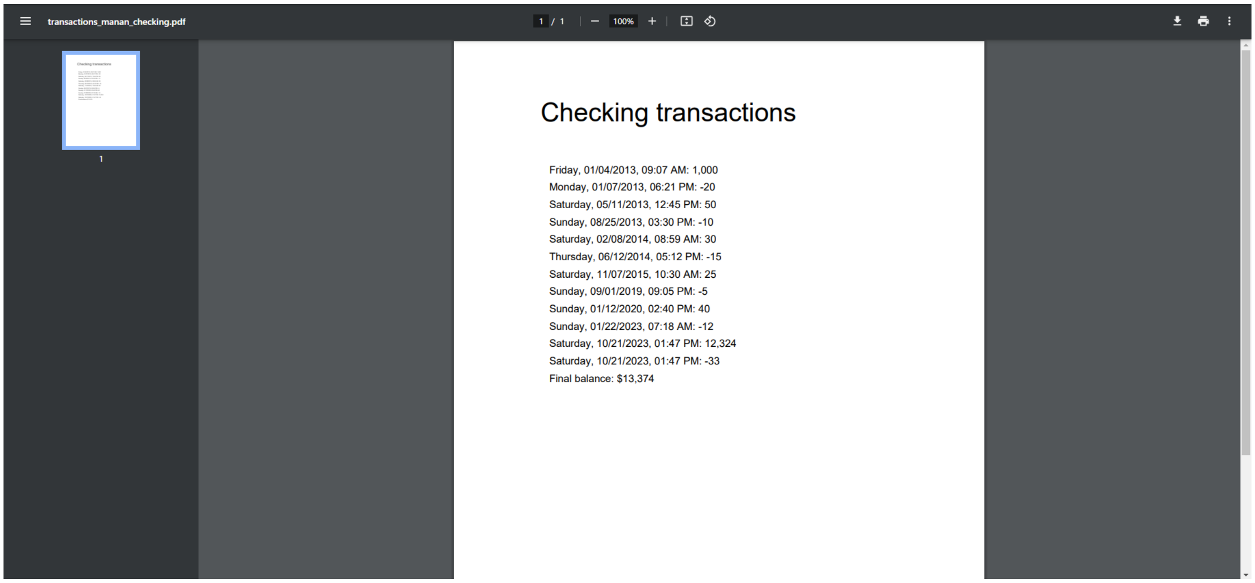
Date	Checkings
Friday, 01/04/2013, 09:07 AM	1,000
Monday, 01/07/2013, 06:21 PM	-20
Saturday, 05/11/2013, 12:45 PM	50
Sunday, 08/25/2013, 03:30 PM	-10
Saturday, 02/08/2014, 08:59 AM	30
Thursday, 06/12/2014, 05:12 PM	-15
Saturday, 11/07/2015, 10:30 AM	25
Sunday, 09/01/2019, 09:05 PM	-5
Sunday, 01/12/2020, 02:40 PM	40
Sunday, 01/22/2023, 07:18 AM	-12
Saturday, 10/21/2023, 01:47 PM	12,324
Saturday, 10/21/2023, 01:47 PM	-33

Date	Savings
Thursday, 01/03/2013, 12:00 AM	20,000
Saturday, 07/04/2015, 09:07 AM	1,000
Sunday, 01/28/2018, 06:21 PM	-20
Sunday, 08/12/2018, 12:45 PM	50
Wednesday, 01/15/2020, 03:30 PM	-10
Saturday, 01/08/2022, 08:59 AM	30
Monday, 12/05/2022, 05:12 PM	-15
Tuesday, 07/11/2023, 10:30 AM	25
Saturday, 10/21/2023, 01:47 PM	456

The export button shows a dialog for choosing an account



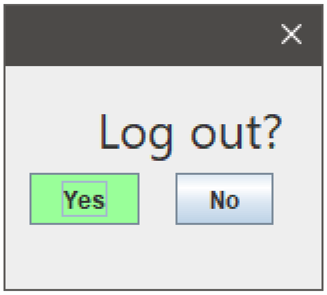
The export button then opens a pdf with the selected account transactions, which is saved in the same directory as the app



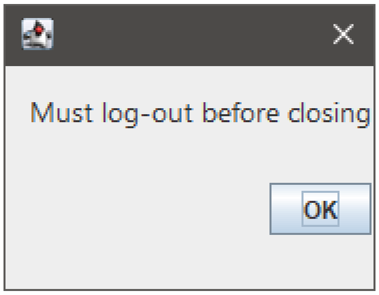
Checking transactions

Friday, 01/04/2013, 09:07 AM: 1,000
Monday, 01/07/2013, 06:21 PM: -20
Saturday, 05/11/2013, 12:45 PM: 50
Sunday, 08/25/2013, 03:30 PM: -10
Saturday, 02/08/2014, 08:59 AM: 30
Thursday, 06/12/2014, 05:12 PM: -15
Saturday, 11/07/2015, 10:30 AM: 25
Sunday, 09/01/2019, 09:05 PM: -5
Sunday, 01/12/2020, 02:40 PM: 40
Sunday, 01/22/2023, 07:18 AM: -12
Saturday, 10/21/2023, 01:47 PM: 12,324
Saturday, 10/21/2023, 01:47 PM: -33
Final balance: $13,374

The logout button shows a confirmation and logs the user out


Log out?

Closing the window without logging out shows an error message


Must log-out before closing