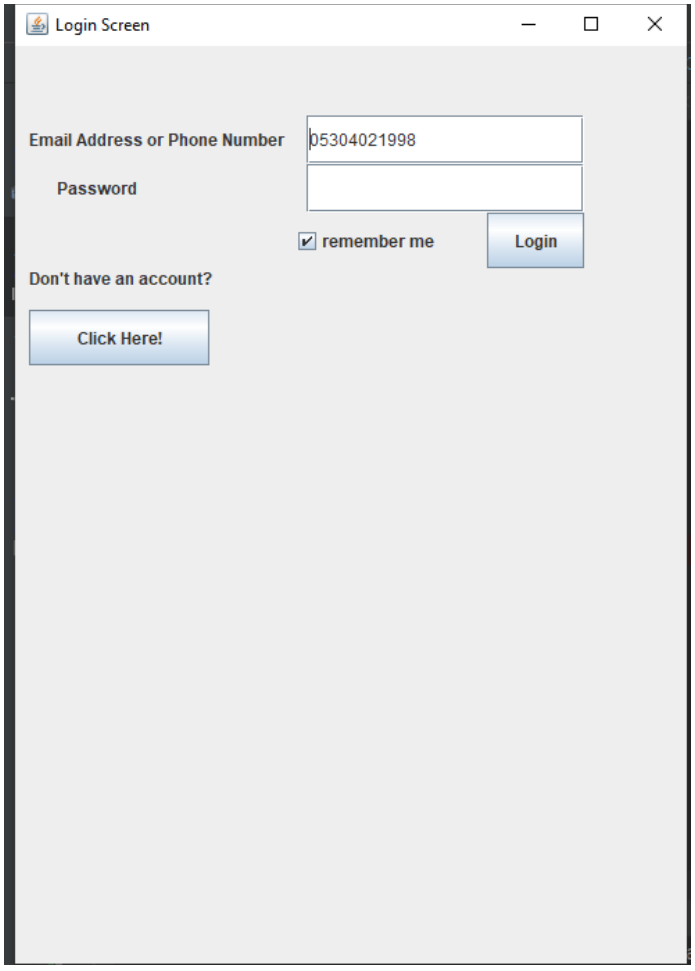
Java Programming Project Report

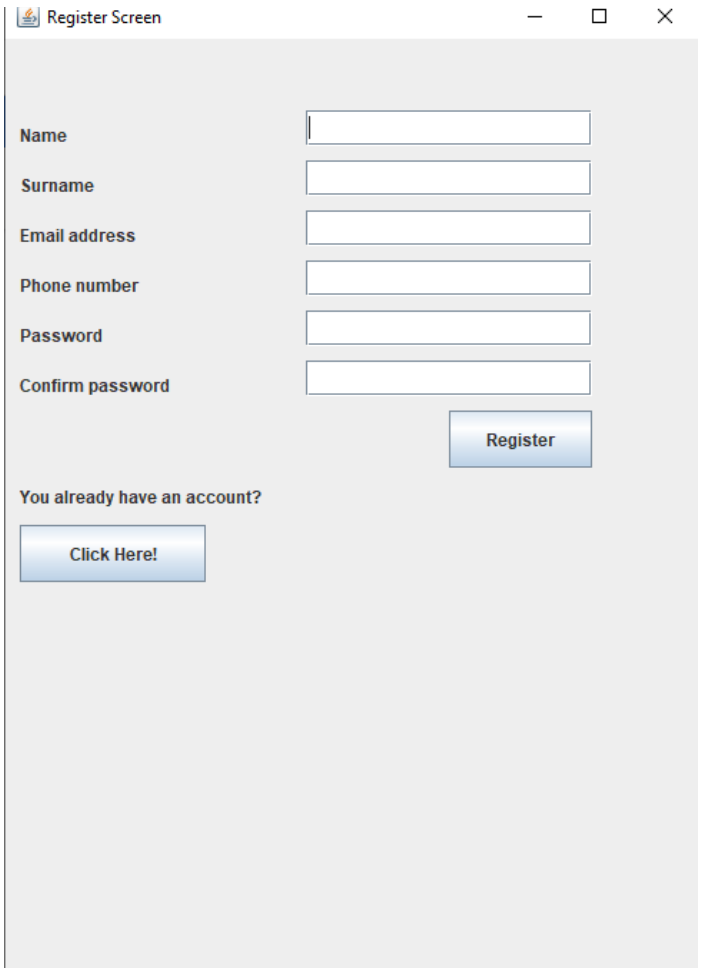
This report details the development of a banking desktop application. The project involved the use of the Swing framework, as well as the Jsoup and Connector/J libraries.

The login screen is the initial interface presented to the user. Upon login, users can access their account, with a remember me feature providing the ability to easily log in again after initial access. The register screen can be accessed via a button located at the bottom left of the screen, enabling users to create an account if they do not already have one.

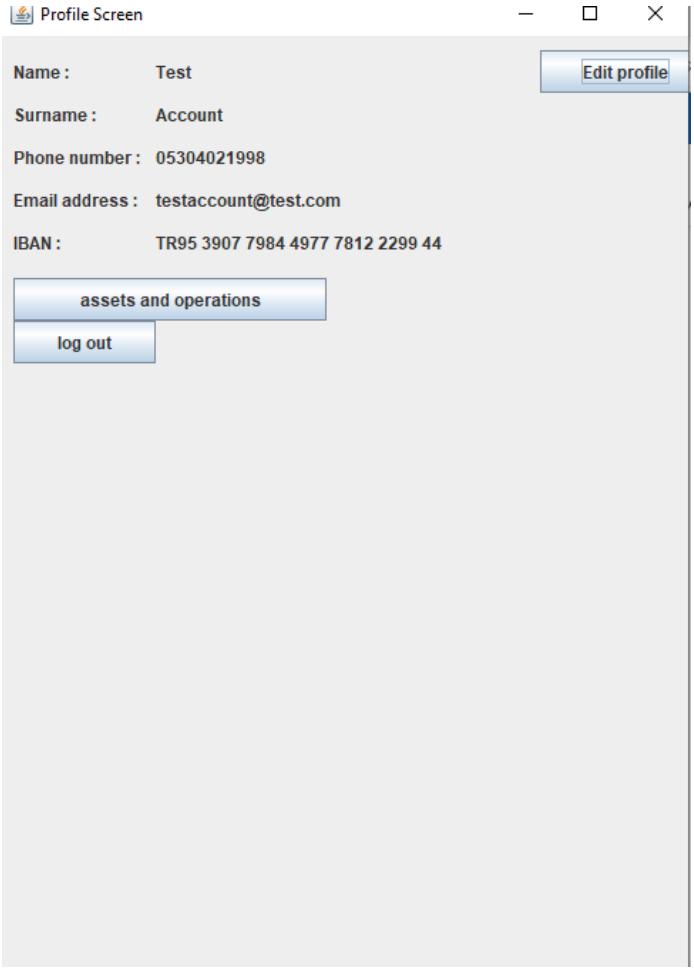


Upon clicking the button user The registration process includes a series of checks to ensure the data entered meets specific requirements. These checks include verifying that names and surnames do not include illegal characters, that passwords are longer than 10 characters and contain digits, and that email and phone numbers are correctly formatted. Additionally, the code confirms that the data entered is not already registered in the database.

If all checks pass, the data is registered to the dataset and the user is logged in automatically.

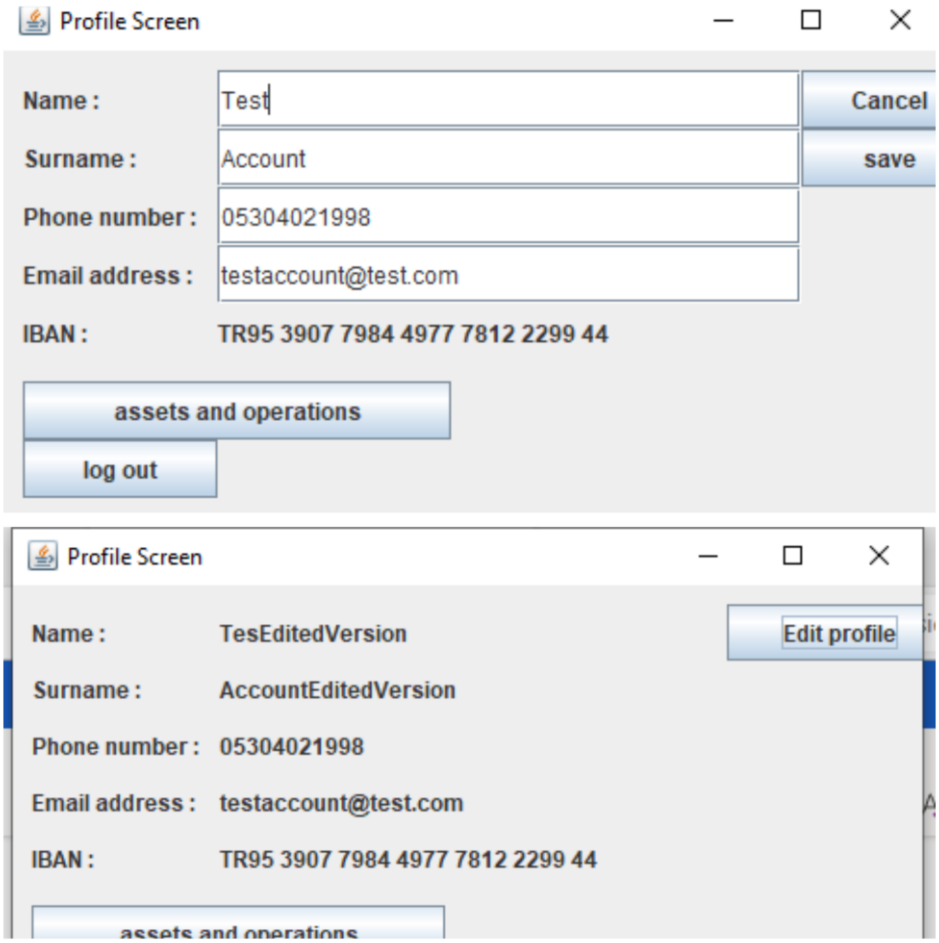


Following login or registration, the user can access their account information, including an IBAN address generated through a recursive function. This function creates a random IBAN number and verifies that it does not already exist in the database before assigning it to the user.



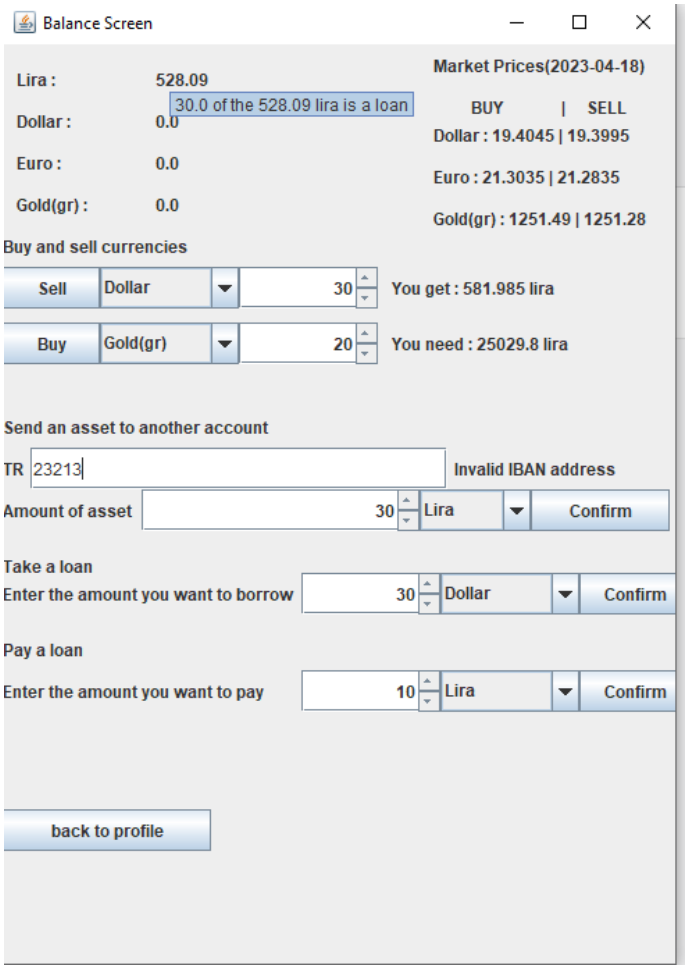
After clicking the edit profile buttons all labels turn into textfields and user can edit the data. If user clicks cancel nothing changes and if user clicks save button it checks if the name and surname doesn't contain any illegal chracters and ifsomeone else owns the edited phone number and email address if they are free to use they are saved to database and updated on the screen.

To edit their account information, users can click on the "edit profile" button, which converts all labels to text fields. If the user chooses to save changes, the code again checks for illegal characters in the name and surname, and confirms that the edited phone number and email address are not already owned by another user. If the data passes these checks, it is updated in the database and reflected on the user's screen.



When the user clicks the "Assets and Operations" button, their available currencies are displayed at the top left of the screen. On the right side, current market values are obtained from a website using the Jsoup library. Below the currency display, the user can buy foreign currencies using Turkish Lira or sell foreign currencies for Turkish Lira. The amount required to complete the transaction is calculated using the current market prices and is visually displayed to the user, indicating the final amount that they will receive. If the user does not have enough currency to complete the operation, an error message will be displayed and the transaction will not be processed. Below this section, users can send currency to another user's IBAN address. The program

verifies the existence of the IBAN address and checks if the user has sufficient currency to send. The IBAN address is then displayed to the user, and the money is transferred to the user with the specified IBAN account if all checks are passed. Lastly, users can borrow currency or repay their existing loans. The program checks if the user's debt exceeds a specified amount of Lira. If the user borrows currency in multiple forms, such as 10 dollars, 10 euros, and 10 Lira, the program will convert them all to Lira using the market prices displayed at the top left of the screen. If the user's debt is below the threshold, the transaction is processed. Users can hover their cursor over the Lira, Euro, and Dollar amounts to view their current debt .



In conclusion, the banking desktop app described in this report is a comprehensive solution for managing user accounts, financial assets, and transactions. It utilizes the Swing framework, Jsoup, and Connector/J libraries to provide a secure and user-friendly interface for logging in, registering, and managing user profiles. The app also allows users to buy and sell foreign currencies, send currency to other users' accounts, and borrow or repay loans. Here is my github link that contains source code and the database files for the project :
https://github.com/UlasTanErsoyak/banking_desktop_app