

The ATM Project using Modern Technologies

Niko Pippuri, TVT24KMO
Iiro Toivari, TVT24KMO
Jan-Markus Jaakola, TVT24KMO
Mika Wallenius, TVT24KMO
Information Technology, Option of Software Design

Introduction

The aim of the project was to make an ATM system using modern technologies, including a REST API made with Node.js, MySQL database, and a Qt-based user interface and combine all of these together.

Objectives

The objective of this project was to create a secure ATM system that allows customers to login using the account ID and their password.

The system supports three types of cards: Debit, Credit and a combo card.

Users can withdraw money using predefined amounts or enter a custom amount using button-based interactions.

Users can check their account balances, and all the transactions are logged and stored in database.

Security is a key focus, with PIN authentication ensuring that only authorized users can access their accounts, and all the PIN codes are stored encrypted as shown in the Figure 1.

Too many failed login attempts will result in the card being locked, and an automatic logout function prevents access after prolonged inactivity.

card_id	pin
1	\$2b\$10\$KTKGTyhtKDyQgPV0ZhrEOv
2	\$2b\$10\$WRaKiPR3IDffYJxo.vgL3u2V
3	\$2a\$10\$7cYv9scd/aALNIEGBVF22uOr
* NULL	NULL

FIGURE 1. Encrypted PIN codes

Methods

The backend is built with Node.js and with a Express.js framework and it handles many security features, including checking user credentials and granting access if authentication is successful. Backend is responsible for encrypting passwords with bcrypt. JWT tokens are used for secure session handling, as shown in Figure 2.

The database follows Entity-Relationship (ER) model for storing customer, account, card, transactions and rest of the data.

The withdrawals are processed in the database using SQL procedure for most optimized performance.

The frontend of the software is designed for intuitive user interaction, and it also supports ATM operations via touchscreen or physical buttons as shown in the Figure 3.

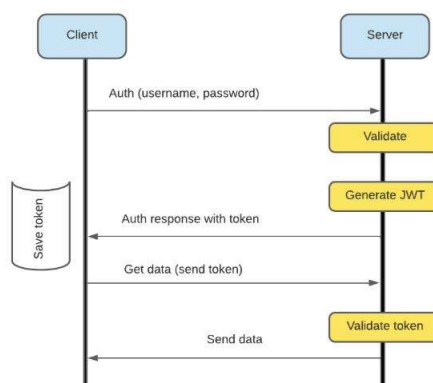


FIGURE 2. JSON Web Token (JWT)

Results

The application is working as intended. A user can log in using their credentials and perform bank automat actions such as withdraw money, check their balance or view recent transactions.



FIGURE 3. Application interface

Conclusions

In summary, all intended features were successfully implemented in the application, and all parts of the project we communicated effectively with each other.

This project demonstrates that using modern software development technologies is an efficient and secure way to build a banking application.

In the future, there are possibilities to enhance the program with features, such as facial recognition or fingerprint authentication

References

1. Tutorials about Computer Programming by Pekka Alaluukas <http://peatutor.com>