IoT Bank Automat Application

Name: Kirill Cheremisin, Andrei Bobrenko; TVT24SPL
Degree Programme in Information Technology, Option of Software Development for Embedded Systems

Introduction

This project aimed to develop an ATM application for simulating banking functions with virtual money, integrating a user interface, backend, and database. The team also focused on improving collaboration and technical skills.

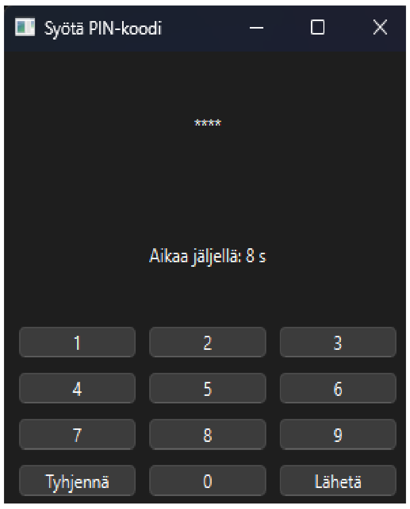
Objectives

The project's objectives were to:

•Create an ATM application with features such as user login, balance checks, deposits, withdrawals, and transaction history viewing.

•Integrate the frontend, developed in *Qt Creator* using C++, with a backend built using *Express.js* and *Node.js*, and a *MySQL* database.

•Enable RFID card login and virtual money transactions for debit and credit cards.



Syötä PIN-koodi

****

Aikaa jäljellä: 8 s

FIGURE 1. User Login Window

Software Development Application Project

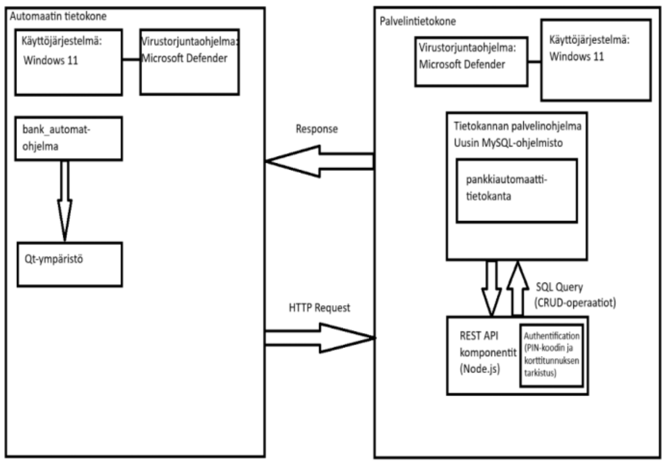
ECTS Credits: 15

Date of Publication: 2025 Spring

Instructors: Kari Jyrkkä

Methods

The team developed the application's interface using *Qt Creator* and *C++*, ensuring responsive user interactions (*Figure 1*). The backend was created in *Express.js* and *Node.js* to handle HTTP requests, with testing performed via *Postman*. The *MySQL* database, managed with *MySQL Workbench*, stored user and transaction data. The team used *GitHub* and *Git Bash* for version control, facilitating collaboration. A REST API enabled seamless communication between the frontend, backend, and database, as illustrated in the system's architecture (*Figure 2*).

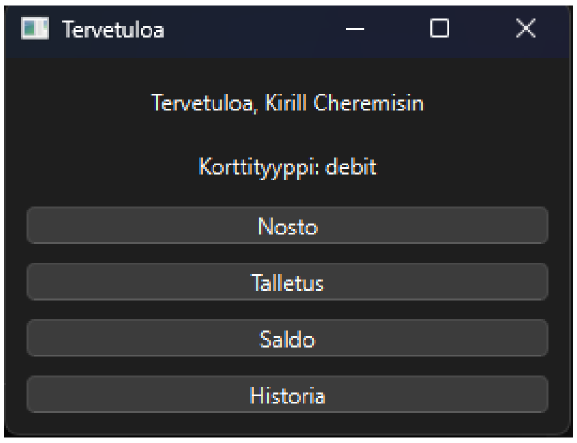


Automaatin tietokone
Käyttöjärjestelmä: Windows 11
Virustorjuntaohjelma: Microsoft Defender
bank_automat-ohjelma
Qt-ympäristö
Response
HTTP Request
Palvelintietokone
Virustorjuntaohjelma: Microsoft Defender
Käyttöjärjestelmä: Windows 11
Tietokannan palvelinohjelma Uusin MySQL-ohjelmisto
pankkiautomaatti-tietokanta
SQL Query (CRUD-operaatiot)
REST API komponentit (Node.js)
Authentification (PIN-koodin ja korttitunnuksen tarkistus)

FIGURE 2. System Architechture UML Diagram

Results

The team delivered a functional ATM application. Users can log in with RFID cards, check balances, deposit, withdraw, and view transaction history (*Figure 3*). *Git* version control and teamwork ensured success. The interface is intuitive and responsive.



Tervetuloa

Tervetuloa, Kirill Cheremisin

Korttityyppi: debit

FIGURE 3. Home Screen with Options

Conclusions

The project achieved its goals, delivering a working ATM application despite challenges in integrating the frontend, backend, and database. The team learned the importance of collaboration, the effective use of *Git* for version control, and the need for clear documentation. While the application's database design is functional, future improvements could enhance scalability. The project provided valuable experience in software integration and team dynamics.

References

1.Visual Studio Code Documentation. https://code.visualstudio.com/docs.

2. MySQL Documentation. https://dev.mysql.com/doc/.

3. Qt Creator Documentation. https://doc.qt.io/qtcreator/.

4. Postman Documentation. https://www.postman.com/.

5. Git Documentation. https://git-scm.com/doc.