# Design and Implementation of a Web-Based Currency Exchange Automation System

Sertaç Özdemir (231307039)

Semih Gökmen (231307070)

Eren Ceylan (231307090)

Department of Information Systems Engineering

Kocaeli University

Kocaeli, Türkiye

**Abstract**

This study presents a web-based currency exchange automation system that allows users to perform instant foreign exchange transactions and monitor historical exchange rates. Developed using PHP and MySQL, the system offers secure user management, transaction logs, and currency-specific wallets. API technology was used to fetch live exchange rates. Additional features such as index usage for performance, triggers for balance updates, and views for easier data access were also integrated.

**Keywords**

currency transactions, web automation, database systems, PHP, MySQL, phpMyAdmin, JS, API

## I. Introduction

This project was developed to facilitate foreign exchange transactions and help users manage their financial operations securely. In today's dynamic financial environment, rapid changes and fluctuations in currency rates have made it more complex to track transactions and manage accounts. The proposed web-based system enables users to view their currency wallets, balances, and transaction histories, while also allowing them to perform new transactions.

The project uses a user-friendly interface combined with a database structure that prioritizes data integrity. Additionally, various filtering options are offered on transaction history and user data, helping users analyze past operations more effectively. The main goal is to provide individual users with an efficient way to manage currency operations and offer system administrators the tools to control and monitor the platform through an admin panel.

## II. Problem Statement

The main objective of this project is to develop a system that allows users to manage their money securely and carry out online foreign exchange operations. The system should also monitor Turkish Lira (TRY) and foreign currency balances, automatically update values after transactions, and prevent data inconsistencies via database constraints.

## III. Research Conducted

At the start of the project, technologies required for a web-based currency automation system were thoroughly researched. To design a user-friendly and functional interface, HTML, CSS, and JavaScript technologies were explored, and Chart.js was selected for graphical data visualization.

For integrating live currency rates into the system, several public APIs (such as ExchangeRate-API, exchangeratesapi.io) were evaluated, and a reliable, JSON-compatible source was selected.

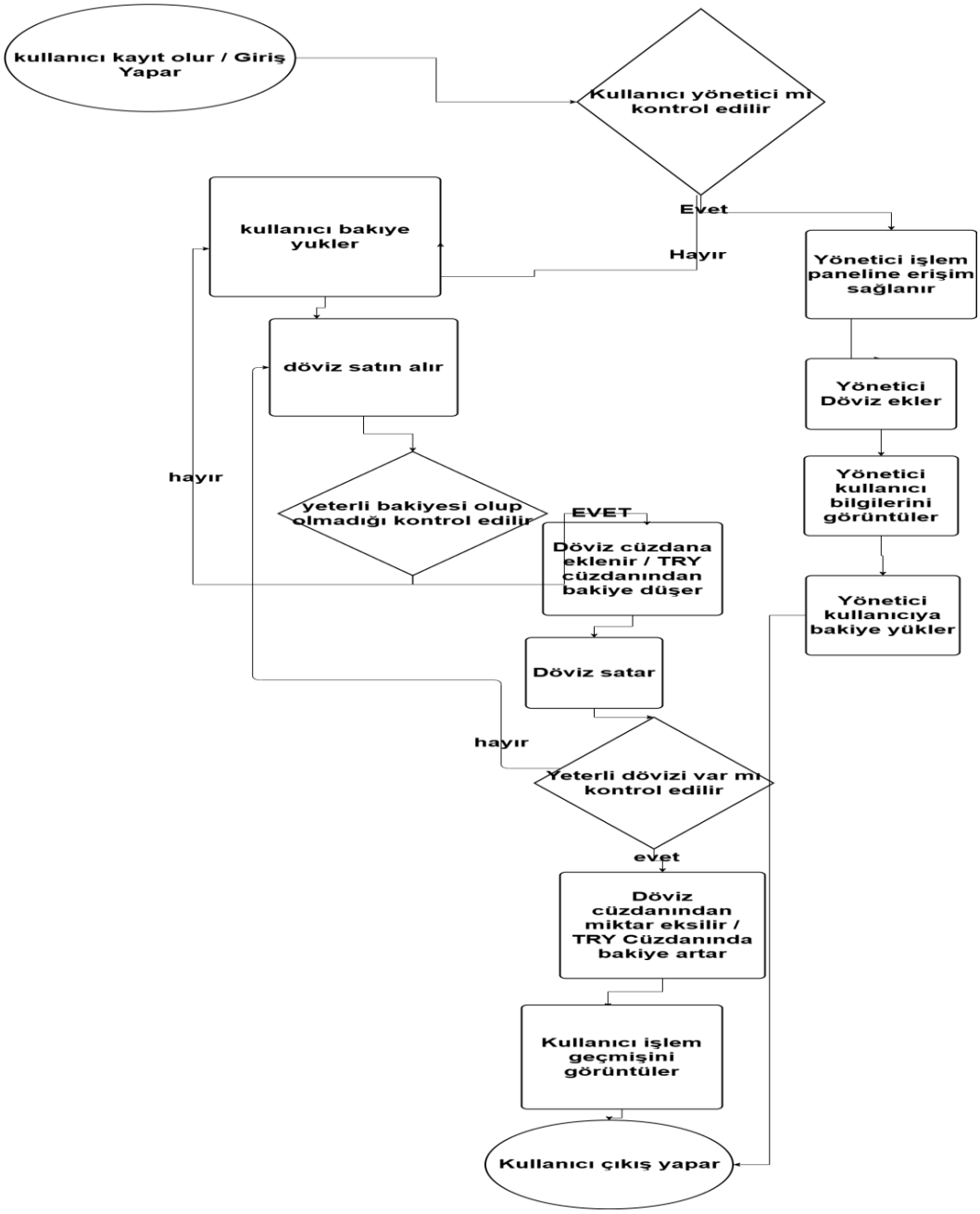
In terms of database optimization, indexing strategies were implemented to improve query performance, triggers ensured data consistency, and views simplified data presentation. These operations were carried out using phpMyAdmin, facilitating database design and management.

PHP was used to connect the interface and the database, ensuring secure handling of user transactions.

## IV. Flowchart

The system follows these steps:

1. The user registers/logs in.
2. The system checks if the user is an administrator.
    2.1. Admin accesses the admin panel
    2.2. Admin adds new currencies
    2.3. Admin views user information
    2.4. Admin adds TRY to user accounts
3. User performs deposit, buy or sell operations
4. System checks for sufficient balance/currency
5. Input is validated, price is calculated
6. Wallet and balance are updated
7. Transaction is recorded and shown in history
8. User logs out




## V. Software Architecture

The system is based on a three-layer architecture:

- **Presentation Layer:** HTML + CSS + JS interface
- **Business Logic Layer:** PHP scripts
- **Data Layer:** MySQL + phpMyAdmin

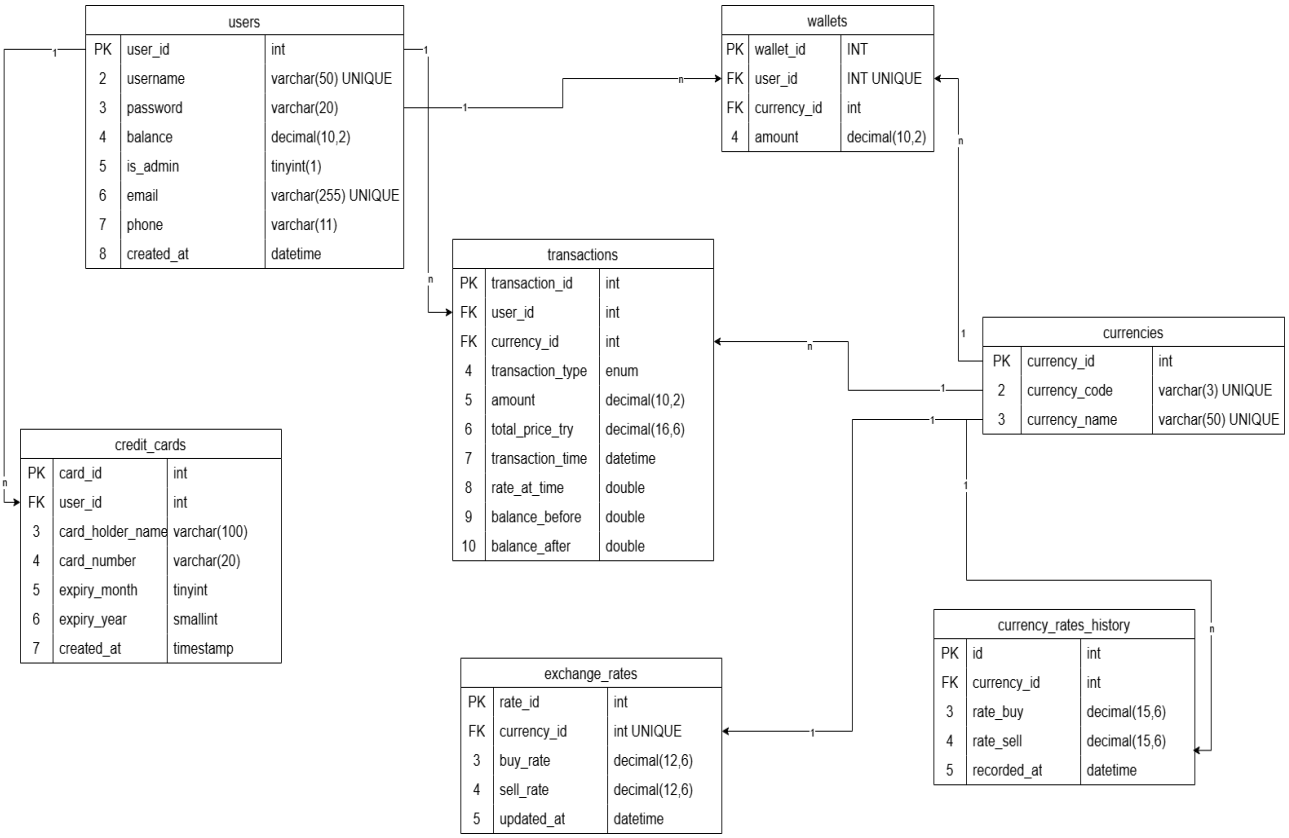
The database consists of the following tables:

- `users`: user information and balances
- `transactions`: all transaction logs
- `currencies`: currency metadata
- `wallets`: user-specific currency balances
- `exchange_rates` & `currency_rates_history`: live and historical exchange rates

## VI. ER Diagram

Relationships displayed in the ER diagram:

- 1 user → many transactions
- 1 user → many wallets
- 1 currency → many transactions
- 1 currency → many wallets
- 1 currency → 1 current exchange rate
- 1 currency → many historical exchange rates




## VII. System Overview

**Users can:**

- Register / log in
- View account information, change password, delete account
- Deposit TRY using card
- View current and historical exchange rates
- Buy / sell currencies
- Automatically receive a wallet for each purchased currency
- View wallet and transaction history

**Admins can:**

- Add TRY to users
- Add new currencies
- View users' transaction histories
- View overall system balances

**Technical Stack:**

- Backend: PHP
- Database: MySQL
- Frontend: HTML, CSS, and JavaScript (JS)
- Currency graphs visualized using Chart.js
- JS used for dynamic features like form validation and real-time updates

## VIII. References


[1] phpMyAdmin Documentation - https://www.phpmyadmin.net/docs/

[2] W3Schools PHP and MySQL Tutorials - https://www.w3schools.com/php/

[3] IEEE Template Guide - https://www.ieee.org/conferences/publishing/templates.html

[4] ChatGPT - OpenAI, 2025

[5] DeepSeek - 2025

[6] GitHub Pages Documentation - https://docs.github.com/en/pages