

A Crypto-Payment Digital Gift Card Platform

A Project Submitted in Partial Fulfillment of the Requirements for the

Degree of

Bachelor of Science in Computer Science and Engineering

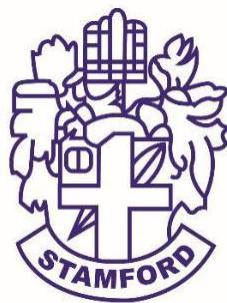
by

Yousuf Ali Shakib

CSE 07208257

Supervised by: Md Towhidul Islam Robin

Assistant Professor



Department of Computer Science and Engineering

STAMFORD UNIVERSITY BANGLADESH

November 2025

Abstract

This project introduces a secure and user-friendly digital gift card marketplace designed to make international digital purchases simple, fast, and accessible. The platform allows users to buy popular global gift cards—such as Google Play, Apple, Amazon, Netflix, Xbox, and Spotify—using cryptocurrency. Gift cards help users avoid many common problems, including payment failures, currency conversion issues, regional restrictions, and the need for international bank cards. As a result, users can access global digital services without relying on traditional financial systems.

One of the main goals of the platform is to bypass the challenges associated with conventional banking in regions like Bangladesh, where international card payments, dollar endorsements, and foreign currency regulations often create significant barriers. By using cryptocurrency as the exclusive payment method—specifically through Tron (TRX) and BNB Smart Chain (BNB)—the system offers low fees, high-speed transactions, and global accessibility. This provides a smooth and reliable alternative to credit cards, banks, and dollar-based payment methods.

The platform ensures instant delivery of gift card codes, secure encrypted code storage, and automated verification. It supports MetaMask, TronLink, and WalletConnect-enabled mobile wallets, offering a seamless checkout experience. An integrated admin panel allows efficient management of cards, orders, and system insights.

Declaration

I hereby declare that the project titled “Digital Gift Card Marketplace with Crypto Payments” is the result of my own work, carried out under the supervision of my department. This project has not been submitted, in whole or in part, to any other institution or university for the award of any degree, diploma, or certificate.

All the information, system design, implementation details, and analysis presented in this report are based on my own understanding, research, and development activities. Any external sources, references, or materials used in the project have been properly acknowledged.

I affirm that this work maintains academic integrity and follows the ethical guidelines set by the institution.

Signature and Date:

Yousuf Ali Shakib

.....

Student Name: Yousuf Ali Shakib

Department: Computer Science and Engineering

ID: CSE 072 08257

Date: 15-11-2025

Acknowledgements

First and foremost, praises and thanks to the Almighty, for His showers of blessings throughout my project work to complete the project successfully. He gave us the ability, courage and chance to complete this project.

I would like to express our deep and sincere gratitude to my project supervisor, Md Towhidul Islam Robin, Assistant Professor, Department of Computer Science & Engineering, Stamford University Bangladesh, for providing invaluable guidance throughout this project. His dynamism, vision, sincerity and motivation have deeply inspired me. It was a great privilege and honor to work and study under his guidance. Whenever I faced a problem he was there for me. I am extremely grateful for what he has offered us. Without his help and guidance I never would have been able to complete a project of this caliber.

I am also thankful to all of my teachers of the department as they have given me useful insights, various resources and moral support.

Last but not least, I am extremely grateful to my parents for their love, prayers, caring and sacrifices for educating and preparing me for my future. They have supported me unconditionally both financially and emotionally throughout my degree.

Table of Content

01. Introduction.....	7
1.2 History of Blockchain Technology.....	8
1.3 Traditional Payment System.....	8
1.4 Blockchain-Based Payment System.....	9
1.5 Future of Digital Payments.....	9
1.6 Motivation.....	9
1.7 Project Overview.....	10
1.7.1 Project Objectives.....	10
1.7.2 Aim of the Project.....	11
1.7.3 Project Description.....	11
1.7.4 Features of the Project.....	11
1.7.5 Scope of the Project.....	12
1.8 Chapter Summary.....	12
02. Literature Review.....	13
2.1 Background Study.....	13
2.2 Existing Gift Card Platforms.....	13
1. Bitrefill.....	13
2. CoinsBee.....	14
3. Raise.com.....	14
2.3 Drawbacks of Current Platforms.....	14
2.4 Our Platform.....	15
2.4.1 Limitations of Our Platform.....	15
2.5 Chapter Summary.....	15
03. Requirement Analysis.....	16
3.1 Requirement Analysis.....	16
3.2 Device Requirements.....	16
3.3 Hardware Requirements.....	17
3.4 Software Requirements.....	17
3.5 Functional Requirements.....	18
3.6 Non-Functional Requirements.....	18
3.7 Summary.....	18
04. Technical Overview.....	19
4.1 Tools.....	19
4.1.1 NodeJS.....	19
4.1.1 ReactJS.....	20
4.1.2 TypeScript.....	20

Table of Content

01. Introduction.....	7
1.2 History of Blockchain Technology.....	8
1.3 Traditional Payment System.....	8
1.4 Blockchain-Based Payment System.....	9
1.5 Future of Digital Payments.....	9
1.6 Motivation.....	9
1.7 Project Overview.....	10
1.7.1 Project Objectives.....	10
1.7.2 Aim of the Project.....	11
1.7.3 Project Description.....	11
1.7.4 Features of the Project.....	11
1.7.5 Scope of the Project.....	12
1.8 Chapter Summary.....	12
02. Literature Review.....	13
2.1 Background Study.....	13
2.2 Existing Gift Card Platforms.....	13
1. Bitrefill.....	13
2. CoinsBee.....	14
3. Raise.com.....	14
2.3 Drawbacks of Current Platforms.....	14
2.4 Our Platform.....	15
2.4.1 Limitations of Our Platform.....	15
2.5 Chapter Summary.....	15
03. Requirement Analysis.....	16
3.1 Requirement Analysis.....	16
3.2 Device Requirements.....	16
3.3 Hardware Requirements.....	17
3.4 Software Requirements.....	17
3.5 Functional Requirements.....	18
3.6 Non-Functional Requirements.....	18
3.7 Summary.....	18
04. Technical Overview.....	19
4.1 Tools.....	19
4.1.1 NodeJS.....	19
4.1.1 ReactJS.....	20
4.1.2 TypeScript.....	20

4.1.3 Tailwind CSS.....	21
4.1.4 MongoDB.....	21
4.1.5 Visual Studio Code.....	21
4.2 Summary.....	22
05. Design & Implementation.....	23
5.1 The need for System Analysis & Design.....	23
5.2 System Development Life Cycle.....	23
5.3 System Development Life Cycle Models.....	25
5.4 Project Methodology.....	25
5.4.1 Phases of Agile Methodology.....	26
5.4.2 Advantages of Agile Methodology.....	28
5.4.3 Disadvantages of Agile Methodology.....	28
5.4.4 Traditional vs Agile Methodology.....	28
5.5 Use Case Diagram.....	30
5.6 Data Flow Diagram (DFD).....	31
5.6.1 DFD Level 0 Diagram.....	31
5.6.2 DFD Level 1 Diagram.....	31
5.6.3 DFD Level-2 (User Order Management).....	33
5.6.4 DFD Level 2 (Crypto Payment Processing).....	34
5.6.5 DFD Level 2.....	35
5.7 Activity Diagram.....	35
5.7.1 The User Activity Diagram:.....	36
5.7.2 The Admin Activity Diagram:.....	37
5.8 Entity Relationship Diagram (E-R diagram).....	38
06. User Interface.....	39
6.1 Home Page.....	39
6.2 Card List Page.....	40
6.3 Blogs Overview Page.....	40
6.4 Blog Post Page.....	41
6.5 Wallet Select Modal.....	42
6.6 Mobile Wallet Connect.....	43
6.7 Payment Modal.....	43
6.8 QR-Code-Pay Modal.....	44
6.9 Admin Dashboard Page.....	45
6.9 Order's Overview and Revenue Page.....	46
6.10 Card's Overview Page.....	47
6.11 Add Card/Edit Card Page.....	47
References.....	49

01. Introduction

The rapid digital transformation of the global economy has changed the way individuals purchase and access online services. Among these changes, digital gift cards have become one of the most convenient and flexible payment instruments for consumers worldwide. At the same time, blockchain technology has emerged as a revolutionary financial infrastructure capable of removing the limitations of traditional banking. This project focuses on combining these two innovations by creating a crypto-powered digital gift card marketplace. The proposed system enables users to purchase international gift cards using cryptocurrency, avoiding the complexities of banking systems, card restrictions, and foreign currency regulations. This introduction provides historical context, technological background, and the motivation behind the development of this platform.

1.1 History of Gift Cards

Gift cards have evolved significantly since their introduction in the late 1990s. Initially launched as a replacement for paper-based vouchers, companies like Blockbuster and Starbucks were among the first to popularize physical gift cards, which users could purchase and redeem in stores. Over time, as e-commerce platforms grew, gift cards transitioned from physical plastic cards to digital codes that could be bought online and delivered instantly via email. This shift made gift cards not only a gifting choice but also a practical financial tool. Digital gift cards help users bypass issues such as payment gateway failure, international card restrictions, or foreign currency problems. They also eliminate the risks associated with carrying cash. Today, platforms like Google Play, Amazon, Netflix, Xbox, and Spotify heavily rely on digital gift cards to provide global access regardless of regional banking limitations. Their simplicity, security, and universal availability make them an essential component of the modern digital economy.

1.2 History of Blockchain Technology

Blockchain technology emerged in 2008 with the publication of the Bitcoin whitepaper by an individual or group using the pseudonym Satoshi Nakamoto. The original purpose of blockchain was to create a decentralized digital currency system that could operate without banks or centralized authorities. Over the years, this technology expanded far beyond Bitcoin. Ethereum introduced smart contracts, enabling decentralized applications (dApps) and automated financial logic. Later, networks like Tron and Binance Smart Chain (BNB Chain) accelerated blockchain adoption by offering high-speed, low-cost transactions suitable for everyday use. Today, blockchain serves as the foundation for decentralized finance, tokenized assets, cross-border payments, supply chain tracking, and many other applications. Its tamper-resistant ledger, transparency, and global accessibility have made it one of the most disruptive technologies of the

modern era. Blockchain's ability to operate without geographical or political restrictions makes it an ideal solution for users facing international payment barriers.

1.3 Traditional Payment System

Traditional payment systems rely heavily on banks, credit cards, and centralized financial institutions. While these systems have been widely used for decades, they come with several limitations that affect online and international transactions. Users often face issues such as high transaction fees, slow settlement times, and currency conversion charges. In many developing countries, including Bangladesh, international card availability is extremely limited. Users must go through complex processes such as dollar endorsement, obtaining dual-currency cards, or using restricted international gateways. Additionally, online payment failures are common due to card declines, insufficient limits, or regional restrictions by service providers. As a result, many consumers struggle to purchase digital products like apps, subscriptions, or online services directly. These drawbacks highlight the need for a more flexible, secure, and globally accessible method of payment that does not depend on traditional financial infrastructure.

1.4 Blockchain-Based Payment System

Blockchain-based payment systems offer a modern solution to overcome the restrictions of traditional banking. Unlike centralized systems, blockchain transactions are processed on decentralized networks, allowing users to send and receive funds without bank approval or card verification. Networks such as Tron (TRX) [1] and BNB Smart Chain (BNB) provide extremely low fees, often costing only a few cents per transaction, and offer confirmation speeds within seconds. This makes them ideal for digital purchases, micro-transactions, and international transfers. Users maintain full control of their funds through personal wallets like MetaMask, TronLink, or TrustWallet, ensuring privacy and security. Blockchain payments also eliminate the need for dollar endorsement, currency conversion, or compliance with international card rules. For regions with strict banking regulations, blockchain provides a practical alternative that is borderless, fast, and reliable. These advantages make blockchain a powerful tool for powering gift card marketplaces and other digital platforms.

1.5 Future of Digital Payments

The future of global payment systems is shifting toward decentralization, automation, and borderless digital transactions. With the rise of cryptocurrencies, stablecoins, and decentralized finance (DeFi), users are increasingly seeking faster, cheaper, and more accessible ways to handle digital payments. Global trends indicate that traditional banking alone will not be sufficient to support the rapidly growing digital economy. Technologies such as blockchain, QR-based wallets, contactless payments, and tokenized financial assets are becoming mainstream. In the near future [4], multi-wallet systems, multi-chain connectivity, and instant

cross-border settlements will become standard features of payment platforms. Digital gift card marketplaces are expected to leverage these technologies, enabling instant purchases without geographic or regulatory restrictions. As financial systems become more digital and decentralized, crypto-powered platforms will play a major role in shaping the next generation of online payments and global commerce.

1.6 Motivation

The primary motivation behind this project comes from real-world challenges faced by users attempting to purchase international digital products in countries like Bangladesh. Due to banking restrictions, foreign currency regulations, and strict card limitations, many users are unable to buy services such as Google Play, Netflix, Amazon, or gaming subscriptions [2]. Dollar endorsement requirements, limited availability of dual-currency cards, and frequent payment failures create significant barriers for consumers. Gift cards provide a practical alternative, but existing platforms are often expensive, unreliable, or dependent on centralized payment systems. By using cryptocurrency, these barriers can be bypassed entirely. Crypto payments offer speed, low fees, global accessibility, and independence from banks. This project aims to build a secure, modern, and scalable marketplace where users can easily purchase digital gift cards using blockchain payments. The goal is to solve a genuine market problem and provide a smooth, borderless purchasing experience for everyone.

1.7 Project Overview

The increasing demand for international digital services has highlighted the need for a payment system that is fast, borderless, and independent of traditional banking limitations. This project focuses on building a modern digital gift card marketplace where users can purchase global gift cards using cryptocurrency. By integrating blockchain payments, instant digital delivery, and secure backend systems, the platform offers a smooth and reliable experience for both users and administrators. The system is designed to eliminate regional banking restrictions, allow faster transactions, and provide secure handling of gift card codes through advanced encryption. This section outlines the objectives, goals, features, and scope of the project, providing a thorough understanding of what the system aims to achieve.

1.7.1 Project Objectives

- To develop a secure and user-friendly digital marketplace for purchasing international gift cards.
- To integrate cryptocurrency-based payment systems using Tron (TRX) and BNB Smart Chain (BNB).

- To eliminate traditional banking barriers such as card restrictions, dollar endorsement, and international payment limitations.
- To ensure fast and reliable delivery of gift card codes through automated email notifications.
- To implement encrypted gift card storage and automated code verification for enhanced security.
- To support multiple crypto wallets, including MetaMask, TronLink, and WalletConnect-enabled mobile wallets.
- To design a scalable backend architecture that can support more networks, wallets, and digital services in the future.
- To provide administrators with an efficient panel for managing cards, users, orders, and system insights.
- To create a platform that maintains transparency, reliability, and strong user experience across all processes.

1.7.2 Aim of the Project

- To create a decentralized, borderless payment environment for digital product purchases.
- To offer users a seamless method of buying international gift cards without relying on traditional banks.
- To build a trustworthy platform that provides instant, secure, and globally accessible digital goods.
- To encourage the adoption of blockchain technology in everyday digital transactions.
- To bridge the gap between digital service demand and payment limitations in regions with strict financial regulations.
- To establish a scalable system that can evolve into a full digital marketplace supporting multiple virtual services.

1.7.3 Project Description

This project is a comprehensive digital gift card marketplace designed to allow users to purchase a variety of international gift cards using cryptocurrency. The system supports popular brands such as Google Play, Apple, Amazon, Netflix, Xbox, and Spotify. By integrating blockchain-based payment methods, the platform overcomes traditional banking barriers that many users face, especially in regions where international card usage is restricted. The platform accepts cryptocurrency payments through Tron and BNB Smart Chain networks, both known for their low fees, high speed, and global accessibility. Once payment is confirmed on the blockchain, the system instantly delivers encrypted gift card codes to the user's email, ensuring convenience and speed. The backend includes secure code handling, automated verification, and a powerful admin panel that allows administrators to manage cards, orders, and system insights.

effectively. The architecture is designed to be scalable, making it possible to add new networks, wallets, or digital goods in the future. Overall, the project provides a modern, secure, and efficient solution for purchasing global digital products.

1.7.4 Features of the Project

- **Crypto Payment Support:** Users can pay using Tron (TRX) and BNB Smart Chain (BNB) for fast and low-cost transactions.
- **Instant Gift Card Delivery:** Codes are delivered immediately after blockchain confirmation.
- **Encrypted Code Storage:** All gift card codes are securely stored using advanced encryption.
- **Automated Verification:** The system verifies gift card codes before delivery to ensure authenticity.
- **Multi-Wallet Compatibility:** Supports MetaMask, TronLink, TrustWallet, and WalletConnect for mobile wallet connections.
- **User-Friendly Checkout:** Simplified steps ensure a smooth and hassle-free purchase experience.
- **Scalable Architecture:** Easy integration of additional networks, wallets, or digital products.
- **Admin Dashboard:** Comprehensive management of cards, orders, users, and system analytics.

1.7.5 Scope of the Project

- Development of a full-stack crypto-enabled digital marketplace.
- Integration of blockchain networks (Tron & BNB Chain) for payment processing.
- Secure backend for gift card storage, encryption, and delivery.
- Creation of a web-based UI for browsing, purchasing, and managing cards.
- Support for both desktop wallets and mobile wallets using WalletConnect.
- Implementation of a robust admin panel for platform management.
- Scalability for future expansion to more networks, cryptocurrencies, and digital services.
- Support for automated transaction validation and instant delivery workflows.

1.8 Chapter Summary

The introduction chapter presented the fundamental concepts and background necessary to understand the development of this crypto-based digital gift card marketplace. It explored the history of gift cards, the evolution of blockchain technology, and the limitations of traditional payment systems that inspired the need for a decentralized solution. The chapter also discussed the advantages of blockchain-based payments, the future landscape of digital transactions, and the strong motivation behind building a system that bypasses international banking restrictions. Finally, the chapter outlined the project's objectives, aims, features, and scope, offering a clear overview of what the platform intends to achieve. This foundation prepares the reader for the upcoming technical chapters, where the system design, methodologies, implementation details, and results will be presented.

02. Literature Review

2.1 Background Study

Digital gift cards and blockchain-based payment systems have become increasingly relevant in modern digital commerce. Gift cards were initially introduced as a convenient alternative to traditional vouchers and quickly evolved into a universal solution for accessing online services. Their rapid growth is closely tied to the expansion of international digital platforms such as Google Play, Amazon, Netflix, Xbox, and Spotify. At the same time, blockchain technology has revolutionized financial transactions by offering a decentralized, transparent, and borderless payment environment. Many users, especially in countries with strict banking regulations, face difficulties when making international purchases due to currency restrictions, card limitations, or dollar endorsement requirements. As a result, cryptocurrency has emerged as a practical alternative for purchasing digital goods without relying on traditional banking systems. Several international marketplaces have begun adopting crypto payments to simplify global transactions. This background highlights the need for a unified platform that merges the convenience of digital gift cards with the flexibility and reliability of blockchain payments.

2.2 Existing Gift Card Platforms

Below are three major gift-card or crypto-gift-card platforms, each described based on publicly available information.

1. Bitrefill

Bitrefill is one of the world's leading crypto-powered gift card marketplaces, allowing users to purchase thousands of gift cards with cryptocurrencies. It supports brands such as Amazon, Google Play, Netflix, and many others. Bitrefill operates globally and offers instant code delivery once a blockchain payment is confirmed. This eliminates the need for traditional banking, credit cards, or currency conversions. Bitrefill also provides mobile top-ups, bill payments, and prepaid phone credits, making it a versatile digital service hub. The platform stands out for its fast transactions, high availability, and ability to bypass international payment barriers. Its crypto-only model has gained popularity in regions where international card usage is limited, offering a practical solution for purchasing global digital services.

2. CoinsBee

CoinsBee is a global gift card platform based in Germany that supports cryptocurrency payments for over 5,000 brands. It operates in more than 195 countries and allows users to buy gaming cards, shopping vouchers, entertainment subscriptions, and even mobile top-ups. Payments can be made using Bitcoin, Ethereum, Litecoin, Binance Coin, and many other cryptocurrencies. The platform focuses on delivering fast, secure, and borderless transactions, making it suitable for users looking to avoid traditional banking obstacles. CoinsBee has positioned itself as a flexible and highly accessible marketplace for diverse digital products. Its international coverage, broad crypto payment options, and instant delivery system make it one of the most popular crypto-friendly gift card providers.

3. Raise.com

Raise is an online marketplace where users can buy, sell, and trade gift cards at discounted prices. Unlike crypto-based platforms, Raise functions as a peer-to-peer marketplace, allowing sellers to list unused or unwanted gift cards while buyers purchase them at reduced rates. With thousands of supported brands, Raise offers flexibility and cost savings for consumers. The platform verifies card balances and provides fraud protection, ensuring users receive valid and accurate gift card information. While Raise does not focus on crypto payments, it plays an important role in the digital gift card ecosystem by offering competitive pricing and strong buyer protection. Its model also highlights the importance of card verification, instant delivery, and reliable customer support—elements crucial for any gift card platform.

2.3 Drawbacks of Current Platforms

Although existing platforms are well-established, they still face several limitations:

Major Drawbacks

- **Limited or no crypto support:** Many popular platforms still rely on bank cards or PayPal, making them inaccessible to users without international payment methods.
- **High fees and pricing inconsistencies:** Some platforms charge higher service fees or offer minimal discounts, reducing user satisfaction.
- **Stock availability issues:** Popular gift cards often run out of stock, especially on crypto-exclusive platforms.
- **Delayed customer support:** Users sometimes experience slow response times, particularly when resolving disputes or incorrect card balances.
- **Verification challenges:** Some platforms cannot always guarantee card authenticity or balance, leading to user mistrust.
- **Regional restrictions:** Not all platforms support global delivery, making them unusable for certain countries.

These drawbacks indicate that although the market is active, there is significant room for improvement—especially in creating a crypto-driven, reliable, and user-friendly global marketplace.

2.4 Our Platform

Our platform aims to address many of the limitations found in existing systems by creating a fully crypto-enabled digital gift card marketplace. Users can purchase international gift cards using Tron (TRX) and BNB Smart Chain (BNB), two networks known for their low transaction fees, rapid confirmation times, and global adoption. The system ensures instant delivery of gift card codes upon payment confirmation, supported by strong encryption and automated verification to maximize security. The platform also includes multi-wallet compatibility, supporting MetaMask, TronLink, and WalletConnect for mobile wallet users. An integrated admin panel enables administrators to manage gift cards, orders, and system analytics efficiently. The architecture is built to be scalable, allowing easy integration of additional brands, tokens, blockchain networks, and wallets in the future.

Overall, our platform brings a modern, fast, and reliable solution to the digital gift card industry, particularly for users facing international banking restrictions.

2.4.1 Limitations of Our Platform

- Currently limited to Tron and BNB networks; additional chains are planned for future expansion.
- Cryptocurrency price volatility may affect the final purchase cost.
- Delivery through email may pose risks if the user's email account is not secure.
- Advanced scalability may require additional server resources and continuous maintenance.

2.5 Chapter Summary

This chapter reviewed the technological and commercial background of digital gift card platforms and blockchain-based payment systems. It analyzed three existing platforms—Bitrefill, CoinsBee, and Raise—highlighting their strengths and limitations. The discussion identified several drawbacks of current services, including limited crypto support, high fees, stock shortages, and verification challenges. Finally, the chapter presented the proposed platform, describing how it integrates secure blockchain payments, instant code delivery, strong encryption, and scalable system design. The chapter concludes that while existing platforms are useful, a dedicated crypto-first gift card marketplace can better address the needs of users—especially those facing international banking restrictions.

03. Requirement Analysis

3.1 Requirement Analysis

This project focuses on building a smooth, secure, and crypto-friendly digital gift card marketplace. To design the system properly, we analyzed the needs of users, devices, and the platform itself. The goal was to ensure that the system supports fast crypto payments, instant delivery, strong security, and an overall effortless buying experience. The analysis below highlights the expectations of different user roles, the technical requirements needed to run the system, and both functional and non-functional needs that shape the platform.

Customer:

- Should be able to browse, search, and buy gift cards.
- Must be able to connect wallets (MetaMask, TronLink) and make crypto payments.
- Should receive instant gift card codes after payment confirmation.

Admin:

- Should manage card inventory (add, update, remove).
- Must view system insights, orders, and platform activity.

- Should be able to verify transactions and handle user issues.

3.2 Device Requirements

- Modern web browser (Chrome, Firefox, Brave, Edge).
- Desktop or mobile device with internet access.
- Wallet extension support on desktop (MetaMask, TronLink).
- Mobile crypto wallet apps with WalletConnect support.

3.3 Hardware Requirements

Server side hardware

- Processor: Intel Pentium dual core or above.
- RAM: Recommended 4GB or above.
- Hard Disk: 200GB or above.

Client side hardware

- Hardware recommended by respective client's operating system and web
- RAM: 2GB or more.
- Communication hardware to communicate to the server.

3.4 Software Requirements

Server side software:

- Database tools
 - MongoDB DB client.
- Server side scripting tools:
 - NodeJS Runtime
- Front-end development tools:
 - Typescript
 - ReactJS
 - HTML
 - CSS
 - Javascript
- Integrated Development Environment (IDE):
 - Visual Studio Code

Client side software:

- Web browser supporting JavaScript, such as:
 - Google Chrome.
 - Mozilla Firefox.
 - Microsoft Edge.
 - Opera.
- Wallet
 - TrustWallet (Mobile Wallet)
 - TronLink (Browser Wallet Extension)
 - MetaMask (Browser Wallet Extension)

3.5 Functional Requirements

- User authentication and secure admin login.
- Wallet connection using MetaMask, TronLink, or WalletConnect.
- Crypto payment processing on Tron and BNB Smart Chain.
- Automatic gift card code delivery after payment confirmation.
- Admin dashboard to manage cards and monitor system data.
- Encrypted storage and automated verification of gift card codes.

3.6 Non-Functional Requirements

The system must maintain strong performance, security, and reliability to ensure a smooth gift-card purchasing experience. It should be designed to scale easily with future networks and wallet integrations.

Key points:

- High availability with minimal downtime.
- Fast response time and low-latency checkout flow.
- Strong encryption for sensitive data and codes.
- User-friendly UI with simple navigation.
- Scalable backend architecture for future expansion.

3.7 Summary

Overall, the requirement analysis outlines what the platform must achieve to offer a seamless crypto-based gift card marketplace. By combining user needs, device capabilities, and both functional and non-functional expectations, the system is designed to be secure, fast, and easy to use. This ensures a smooth experience for customers and efficient management for administrators.

04. Technical Overview

The platform is built using a modern, scalable, and developer-friendly technology stack designed for speed, security, and smooth user experience. The full system includes a responsive web interface, secure backend services, and reliable database management. Each component works together to support crypto payments, instant code delivery, and real-time wallet interaction. Tools like Node.js, React, TypeScript, Tailwind, and MongoDB create a flexible environment where new features, networks, or wallets can be added easily. The development workflow is managed through Visual Studio Code, ensuring efficient coding, debugging, and rapid iteration across the entire project lifecycle.

4.1 Tools

To build this platform, a carefully chosen set of tools and technologies were used to ensure performance, simplicity, and long-term scalability. Each tool plays a specific role—frontend rendering, API development, styling, type-safety, database operations, or development workflow. Together, they form a stable and efficient environment for both developers and users.

Name of the Tool	Purpose
Node.js	Backend runtime environment
React	Frontend UI library
TypeScript	Typed language for safer development
Tailwind CSS	Utility-based styling framework
MongoDB	NoSQL database for scalable data storage
Visual Studio Code	Development and debugging environment

4.1.1 NodeJS

Node.js is the foundation of the platform's backend and plays a crucial role in handling API requests, wallet communication, and transaction verification. Since Node.js uses a non-blocking, event-driven architecture, it is ideal for real-time systems like crypto payment applications. It allows the backend to process multiple operations simultaneously without slowing down, making the entire checkout and delivery process faster. Node.js also has a massive ecosystem of packages (via npm), which speeds up development and reduces the need for reinventing common functionality. For example, crypto utilities, JSON-RPC libraries, and security modules can be easily integrated. Its compatibility with JavaScript and TypeScript makes the workflow smoother across frontend and backend, ensuring the entire project stays consistent. Node.js is also highly scalable, meaning the service can handle thousands of users buying gift cards without significant performance issues. Whether it's processing a payment, verifying a transaction hash, or sending an email with the gift card code, Node.js ensures everything runs reliably and efficiently.

4.1.1 ReactJS

React powers the user interface of the platform and ensures a fast, interactive, and smooth experience for customers. Its component-based architecture makes it easy to reuse UI elements like wallet modals, payment alerts, or card listings. This reduces code duplication and increases efficiency. React's virtual DOM allows the website to update only the parts that need changes instead of reloading entire pages—this makes actions like selecting a card or connecting a wallet feel instant. React also integrates beautifully with modern UI libraries and API systems, making it simple to handle dynamic data such as payment status, wallet connection, or card availability. With React Hooks, the platform manages complex states—like user authentication, wallet state, and crypto transaction tracking—without complicated code. The ecosystem is huge, giving developers access to thousands of helpful libraries. Overall, React gives users a smooth, responsive, app-like experience even though everything runs inside a browser.

4.1.2 TypeScript

TypeScript is used across the project to ensure type-safety, cleaner code, and fewer runtime errors. Since JavaScript alone can be unpredictable, TypeScript helps catch mistakes before the code runs—saving time and reducing bugs in critical features like payment processing or API communication. TypeScript's strong typing ensures that wallet addresses, transaction responses, card data, and user inputs are always validated. This reduces accidental errors that could break functionality or create vulnerabilities. It also improves developer productivity by offering intelligent autocomplete, easier refactoring, and better documentation. In large projects with many modules, TypeScript keeps everything consistent and easier to maintain. When handling

crypto transactions or integrating networks such as Tron or BNB Chain, type definitions help ensure API calls, RPC responses, and blockchain interactions are handled correctly. Its compatibility with both Node.js and React makes TypeScript a perfect fit for full-stack development, ensuring smooth communication between frontend and backend layers. Ultimately, TypeScript provides reliability, safety, and long-term stability for the entire system.

4.1.3 Tailwind CSS

Tailwind CSS is used to design the user interface quickly and consistently without writing large amounts of custom CSS. Tailwind's utility-first approach allows developers to style elements directly within HTML or JSX using predefined classes. This dramatically speeds up development and makes the UI easier to maintain. It ensures pixel-perfect responsiveness across devices, from desktop browsers to mobile screens. Features like wallet modals, card grids, QR prompts, and admin dashboards are all styled efficiently using Tailwind utilities. Instead of switching between multiple CSS files, developers can handle layout, spacing, typography, colors, and animations directly while building components. Tailwind's customization options also allow the design system (colors, fonts, shadows) to remain consistent across the entire platform. Since it generates only the required CSS, the website stays lightweight and fast. For a product where user experience matters—especially during checkout or wallet connection—Tailwind ensures everything looks clean, modern, and user-friendly.

4.1.4 MongoDB

MongoDB serves as the primary database for storing all platform data, including card information, orders, delivery logs, admin details, and system analytics. Being a NoSQL database, MongoDB stores data in flexible JSON-like documents, making it ideal for evolving systems like digital marketplaces. Adding new card types, networks, or features becomes easier because MongoDB does not require strict table structures like SQL databases. It also provides excellent performance for read/write operations, which is important when users are browsing through many gift cards or when the system is storing instant delivery data. MongoDB's horizontal scalability means the database can grow alongside the business without rewriting architecture. Its indexing and aggregation features help generate analytics quickly—useful for the admin panel and system insights page. MongoDB also integrates smoothly with Node.js through libraries like Mongoose, enabling developers to create clean and structured data models. Overall, MongoDB ensures flexible storage, fast performance, and future-proof scalability.

4.1.5 Visual Studio Code

Visual Studio Code is the primary development environment used to build and manage the entire project. It provides powerful features like IntelliSense, debugging tools, Git integration, and extension support, making the development process smoother and more efficient. VS Code's support for TypeScript, JavaScript, React, and Node.js allows developers to work seamlessly across the full stack within a single editor. Extensions like ESLint, Prettier, Tailwind IntelliSense, and MongoDB tools help maintain clean code, consistent styling, and easy database inspection. VS Code's terminal integration allows developers to run servers, watch processes, compile TypeScript, and interact with package managers without switching tools. Its debugging system makes it easy to identify issues in both frontend and backend logic—especially helpful when dealing with wallet connections, RPC calls, or complex crypto operations. The editor is lightweight, customizable, and extremely developer-friendly. Overall, VS Code boosts productivity and creates a smooth workflow for the entire development team.

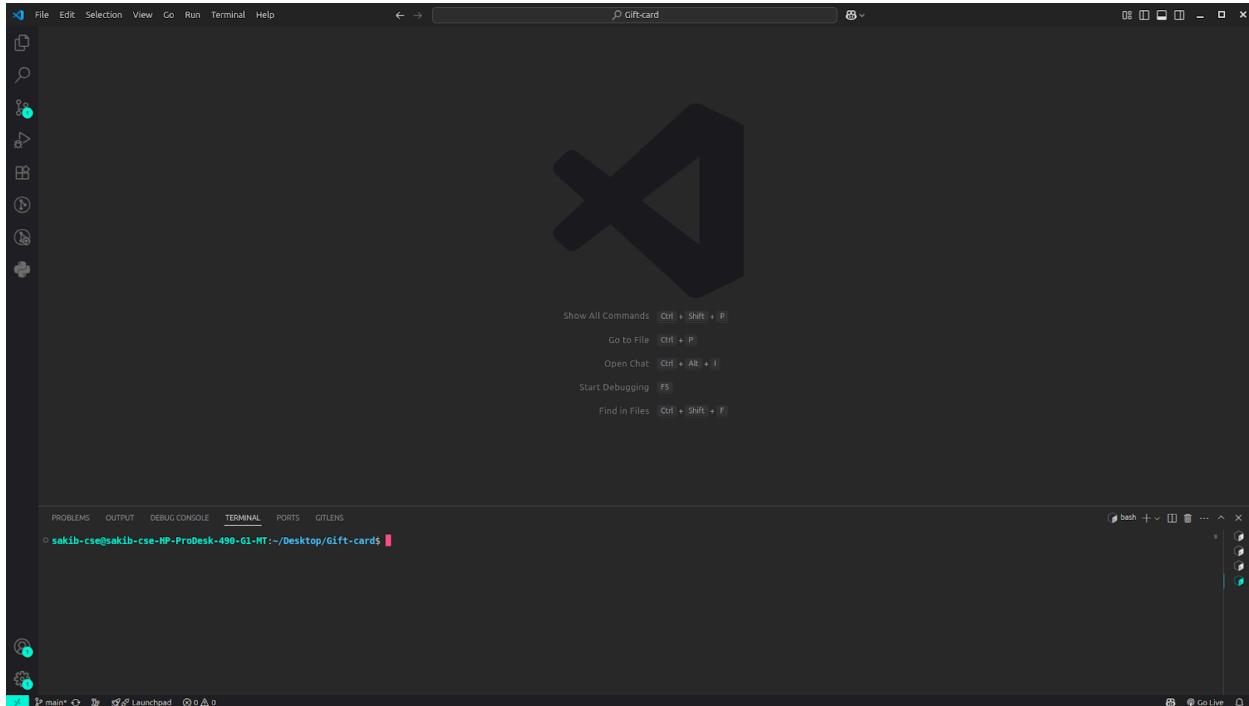


Figure 4.1 Visual Studio Code

4.2 Summary

In summary, the platform is built using a powerful and modern technology stack designed for performance, scalability, and smooth user experience. Each tool contributes to a specific layer—Node.js handles backend logic, React powers the UI, TypeScript ensures code safety, Tailwind enhances design efficiency, MongoDB manages data, and Visual Studio Code

streamlines development. Together, these technologies create a secure, fast, and future-ready digital gift card marketplace.

05. Design & Implementation

Systems Analysis and Design (SAD) is a broad term for describing methodologies for developing a high quality Information System which combines Information Technology, people and Data to support business requirements. The SAD technique is not only limited to IT systems and can be used to create just about anything, from a family house to the international space station. But there is no silver bullet in simplifying the development of computer systems. This principle is still true today. In other words, there is no single, simple technique that developers can use to ensure successful Information Technology (IT) projects. However, there are development methodologies that can be followed which will greatly assist an IT professional in developing and enhancing systems. A methodology is essentially a procedure to get something done. A development methodology can be thought of as a roadmap. While a roadmap for a traveler will provide the details from driving from point A to point B, a development methodology will provide the IT professional with guidelines for taking a system from conception through implementation and beyond.

5.1 The need for System Analysis & Design

The software projects have a poor track record and the U.S. The National Institute of Standards and Technology stated that in 2003, software bugs cost domestic companies \$59.5 billion (0.6% of GNP). A 2002 Standish Group study (CHAOS) found that only 34% of IT projects were completed on budget and 17% were complete failures. In 2005, the FBI abandoned their Virtual Case File (VCF) project after spending over \$100,000,000 on code that will never be used. Even so, the need for IT projects keeps increasing. In 2000, there were 300,000 new IT projects, and in 2001, over 500,000 new IT projects were initiated. So, understanding SAD is the first step in developing a successful project. However, the proper application of SAD principles are the key to making a project successful. In order to properly apply these principles, an analyst must be able to lead, communicate and sell the SAD principles to the project team.

5.2 System Development *Life* Cycle

The Systems Development Life Cycle (SDLC) is a structured conceptual model used in project management to describe, organize, and manage the different stages involved in creating an information system. It provides a clear roadmap that begins with an initial feasibility study and continues all the way through the development, deployment, and long-term maintenance of the completed application. Although SDLC is most commonly associated with technical projects, especially in the fields of software and hardware development, it can also be applied to non-technical systems where organized planning and structured processes are required.

In real-world scenarios, an SDLC usually deals with IT systems such as computer hardware, software applications, databases, and network systems. Project managers and program managers use this framework to monitor progress and ensure that the project stays on schedule, while system analysts, software engineers, developers, and end-users also contribute important input during different phases. Each system—whether hardware, software, or a combination of both—goes through a series of carefully planned steps. These steps are often iterative, meaning they may be repeated or revisited to improve the final product or to fix issues discovered during development.

The purpose of using SDLC is to establish a clear and rigid structure that guides the entire system development process. This structure helps reduce risks, improve communication among stakeholders, and increase the overall quality of the final system. It also ensures that each phase is completed properly before moving on to the next one, creating a sense of order and predictability.

SDLC is also known by other meanings, such as Synchronous Data Link Control, but in the context of project management, it primarily refers to the Systems Development Life Cycle. A closely related term is the Software Development Life Cycle, which follows a similar process but focuses only on the development and maintenance of software applications instead of entire systems.

7 Stage of Software Development Life Cycle

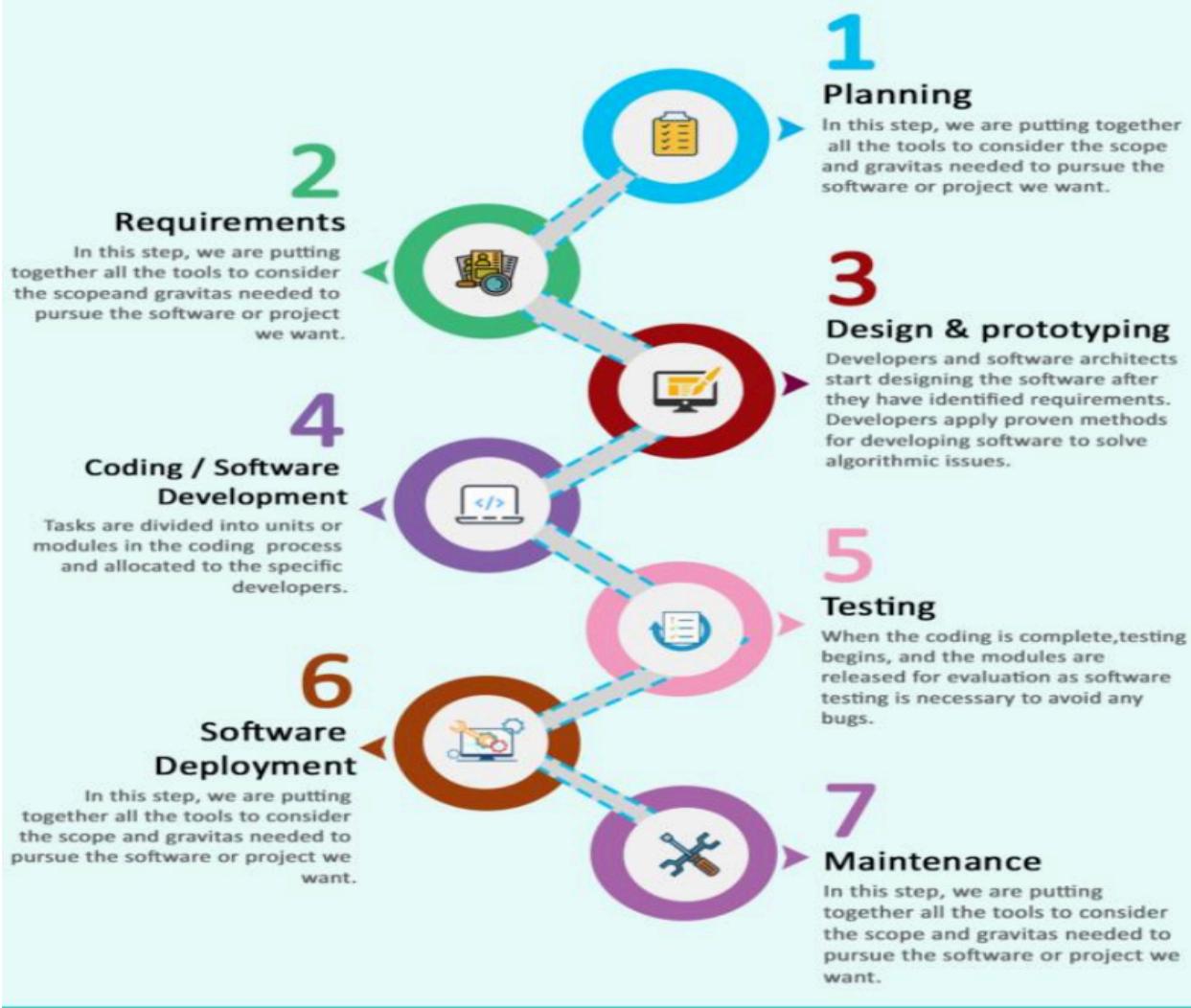


Figure 5-1 Phases of SDLC

5.3 System Development Life Cycle Models

There are different software development life cycle models specified and designed, which are followed during the software development phase. These models are also called Software

Development Process Models. Each process model follows a series of phases unique to its type to ensure success in the step of software development. Here are some of the important SDLC models-

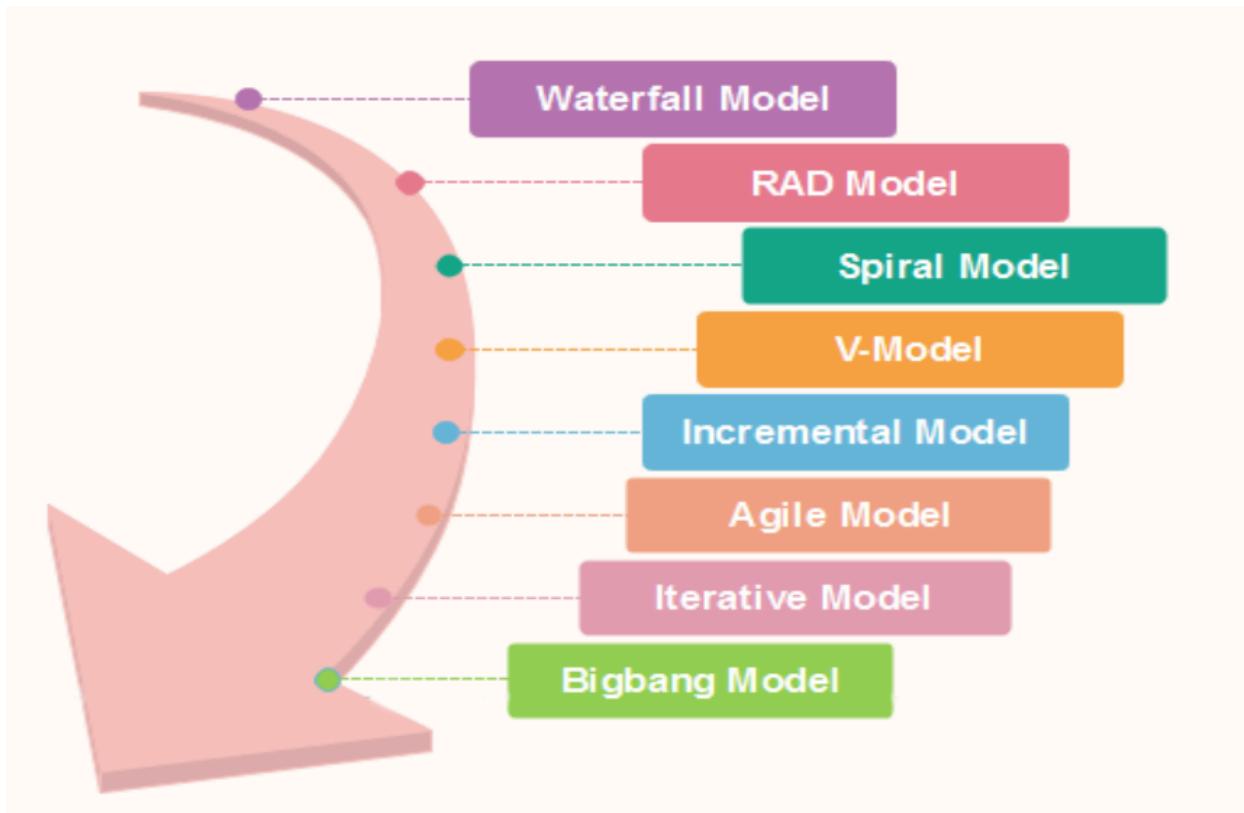


Figure 5-2 Various SDLC Models

5.4 Project Methodology

Choosing the right model for developing the software product or system is very important. We have found that the Agile Model best suits our needs so we have used it to develop our system. The main aim of the Agile model is to facilitate quick project completion. To accomplish this task agility is required. Agility is achieved by fitting the process to the project, removing activities that may not be essential for a specific project. Also, anything that is a wastage of time and effort is avoided. In the Agile model, the requirements are decomposed into many small parts that can be incrementally developed. The Agile model adopts Iterative development. Each incremental part is developed over an iteration. Each iteration is intended to be small and easily manageable and that can be completed within a couple of weeks only. At a time one iteration is planned, developed and deployed to the customers. Long-term plans are not made.

5.4.1 Phases of Agile Methodology

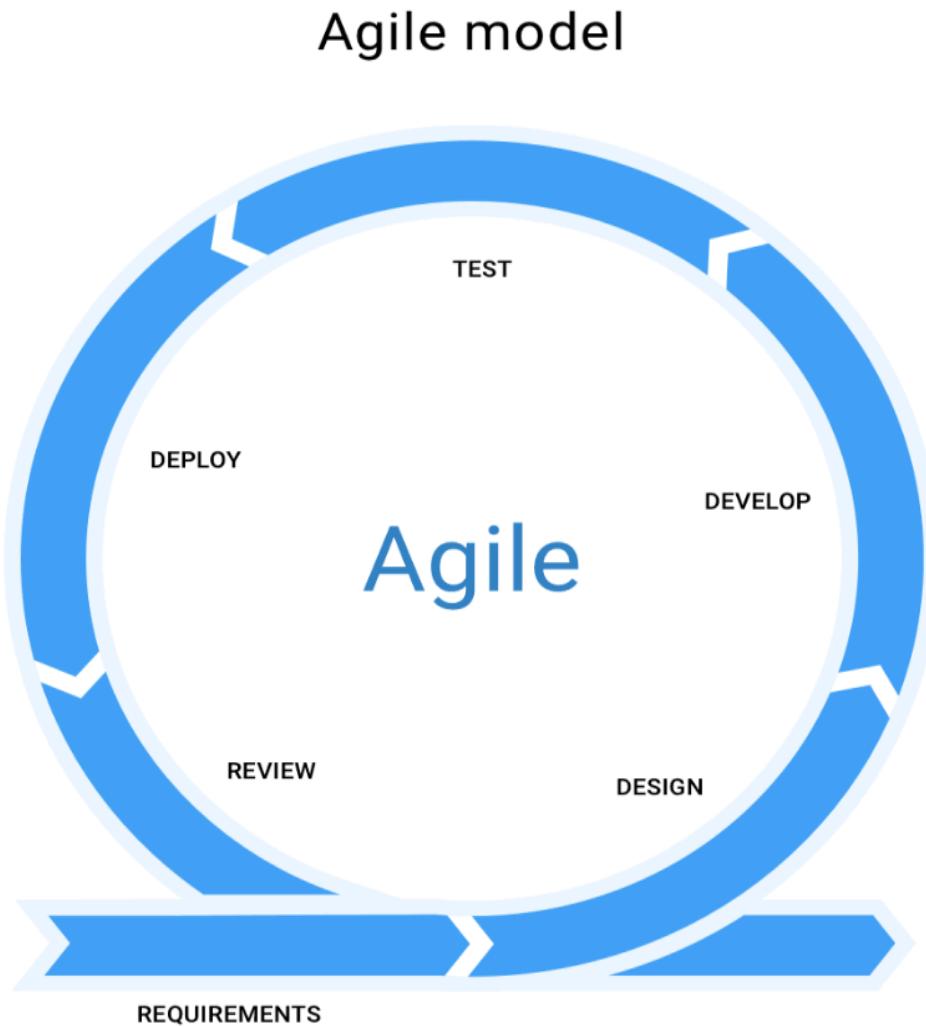


Figure 5-3 Phases of Agile Methodology

- Phase 1: Requirements

Before a Product Owner can even start designing a project, they need to create the initial documentation that will list the initial requirements. They are:

- The end result the project is going to achieve. For example, a text editor.
- The features that it will support. For example, different font sizes.
- The features that it will not initially support. For example, adding

- animations to the text or ability to embed video

- Phase 2: Design

There are two ways to approach design in the software development — one is the visual design and the other is the architectural structure of the system.

- Software Design: During the first iteration, the Product Owner assembles their development team and introduces the requirements created during the previous stage. The team then discusses how to tackle these requirements, and proposes the tools needed to achieve the best result. For example, the team defines the programming language, frameworks, and libraries that the project is going to be using.
- UI/UX Design: During this SDLC stage, the designers create a rough mock-up of the UI. If the product is consumer-grade, the user interface and user experience are most important. So it's generally a good idea to review the possible competitors to see what they are doing right — and especially what they are doing wrong. Further iterations are spent refining the initial design and/or reworking it to suit the new features.

- Phase 3. Development and Coding

The development phase is about writing code and converting design documentation into the actual software within the software development process. This stage of SDLC is generally the longest as it's the backbone of the whole process. There aren't many changes between the iterations here.

- Phase 4. Integration and Testing

This stage is spent on making sure that the software is bug-free and compatible with everything else that the developers have written before. The Quality Assurance team conducts a series of tests in order to ensure the code is clean and business goals of the solution are met.

- Phase 5. Implementation and Deployment

The application is deployed on the servers and provided to the customers — either for the demo or the actual use. Further iterations update the already installed software, introducing new features and resolving bugs.

- Phase 6. Review

Once all previous development phases are complete, the Product Owner gathers the Development Team once again and reviews the progress made towards completing the requirements. The team introduces their ideas toward resolving the problems that arose during the previous phases and the Product Owner takes their propositions into consideration. Afterwards, the Agile

software development lifecycle phases start anew — either with a new iteration or by moving toward the next stage and scaling Agile.

5.4.2 Advantages of Agile Methodology

Here are some advantages of Agile model:

- It allows for changes to be made after the initial planning stage. It follows the client's requirements changes.
- In Agile methodology the delivery of software is unremitting.
- It is easier to add features that will keep the product up to date with the latest developments in the industry.
- At the end of each sprint, project priorities are evaluated. This allows clients to add their feedback, so that they ultimately get the product they desire.
- The testing at the end of each sprint ensures that the errors are caught in each cycle.
- An Agile/Scrum approach can improve organizational synergy by breaking down organizational barriers and developing a spirit of trust and partnership around organizational goals.

5.4.3 Disadvantages of Agile Methodology

- This dynamic methodology is not suitable for processes that require a complex decision making of formal planning such as construction, manufacturing, military, health care system among others.
- As the initial project does not have a definitive plan, the final product can be grossly different than what was initially intended.
- Because of the ever-evolving features, there is always a risk of the ever-lasting project.
- In few of the projects at the starting of the software development life cycle it's difficult to estimate the actual effort required.

5.4.4 Traditional vs Agile Methodology

Unlike traditional SDLC like Waterfall, agile frameworks have a collaborative and cross-functional approach to planning, analysis, design, and delivery. This involves the whole team. According to the CHAOS Report by the Standish Group (2020), agile projects are more successful than Waterfall projects, which have fewer challenges and fewer failures.

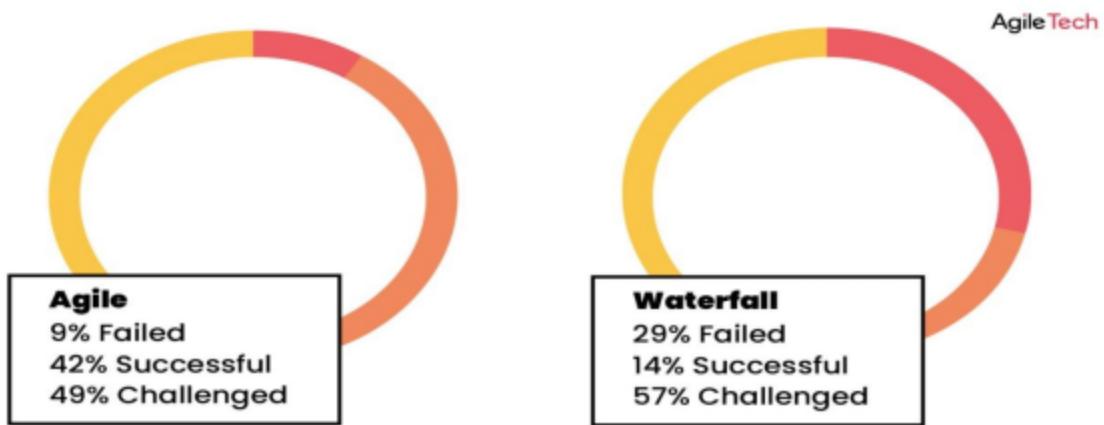


Figure 5-4 Traditional vs Agile Methodology

With Agile methodology, big projects are broken down into small. Before moving into design and development, team members worked simultaneously toward the same goal and followed an iterative process with regular checkpoints. Each Sprint ends with a review and test from actual users and customers. If a result is under standard, you will jump back to requirement analysis. This is the process with every sprint. Now, look at the table below and see the key difference between traditional SDLC and Agile.

Traditional SDLC	Agile SDLC
Clear plan at first and won't change after the project starts	Adaptability and may change at any stage in the development process
Top-down responsibility	Shared responsibility and ownership
Initial research and planning	Ongoing research, planning and testing
Different teams work in different stages	Ongoing collaboration
Customers are involved from the time work is being performed	Customers get involved early in the project but not when the execution has started
Suitable for small and medium projects	Suitable for large projects

Table 5-1 Traditional SDLC vs Agile SDLC

5.5 Use Case Diagram

A use case diagram is a simple visual tool used in software development to show how users interact with a system. It helps developers understand what the system should do from the user's point of view. In a use case diagram, there are two main elements: actors and use cases. Actors are the users or other systems that interact with the application. Here is the use case diagram for the Gift card platform:

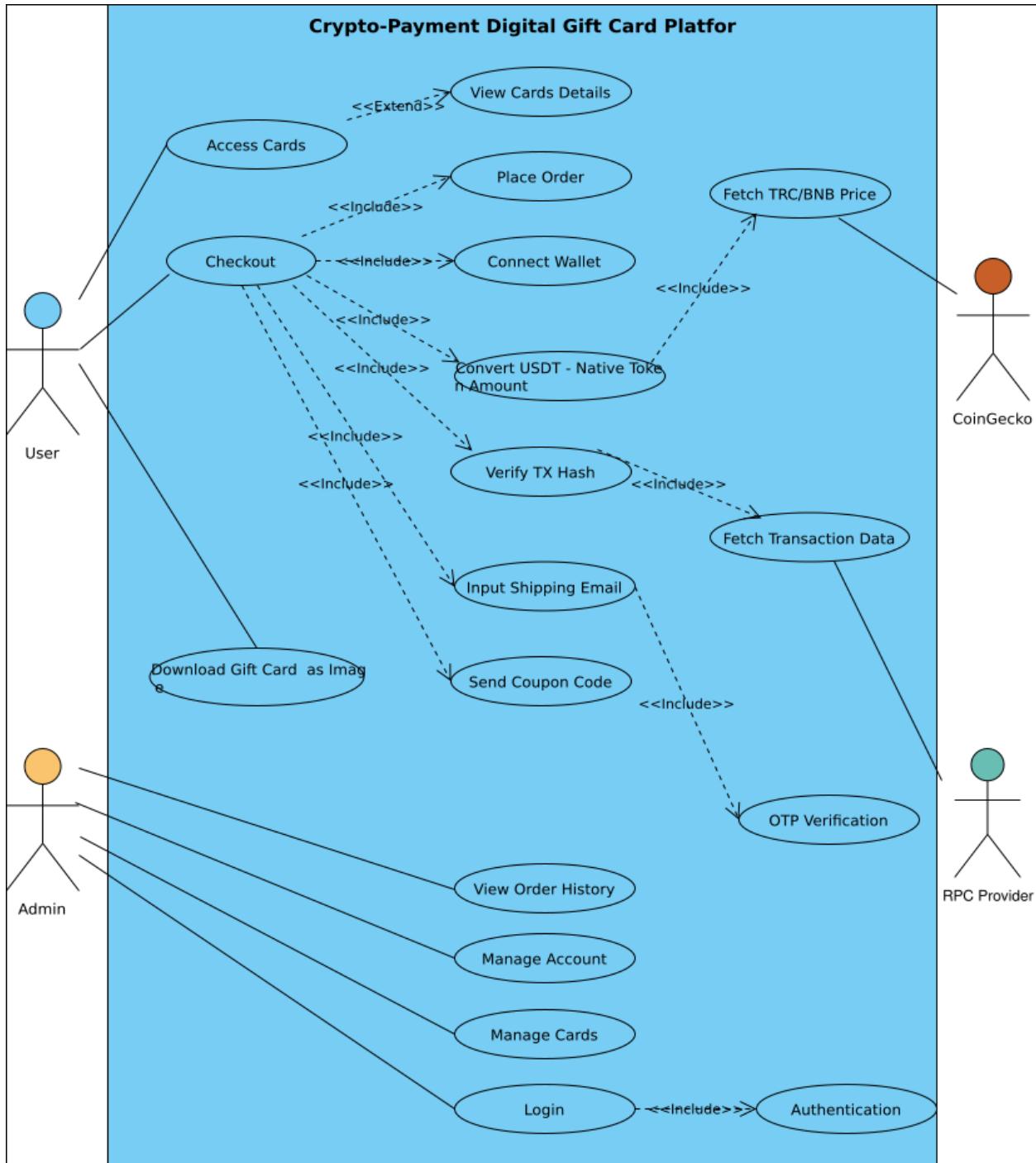


Fig 5-5 Use Case Diagram

5.6 Data Flow Diagram (DFD)

A Data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

5.6.1 DFD Level 0 Diagram

This DFD represent the data flow at context level

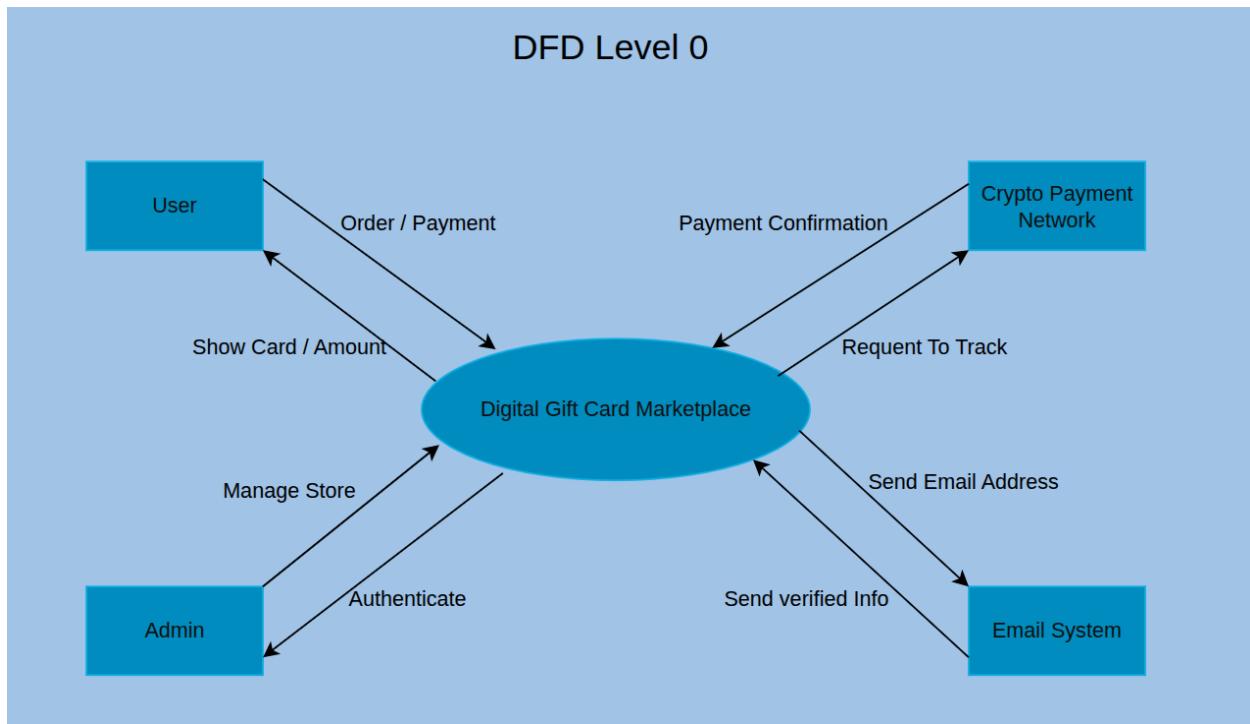


Fig 5-6 DFD Level 0 Diagram

5.6.2 DFD Level 1 Diagram

A Level 1 Data Flow Diagram (DFD) breaks down a system's main process into several sub-processes for clearer understanding. It shows how data moves between these sub-processes, external entities, and data stores. By adding more detail than a Level 0 DFD, it helps analysts identify specific functions and data interactions. Each process in a Level 1 DFD is labeled and connected with data flows that describe the information being transferred. This diagram is useful for analyzing system requirements and improving overall system design.

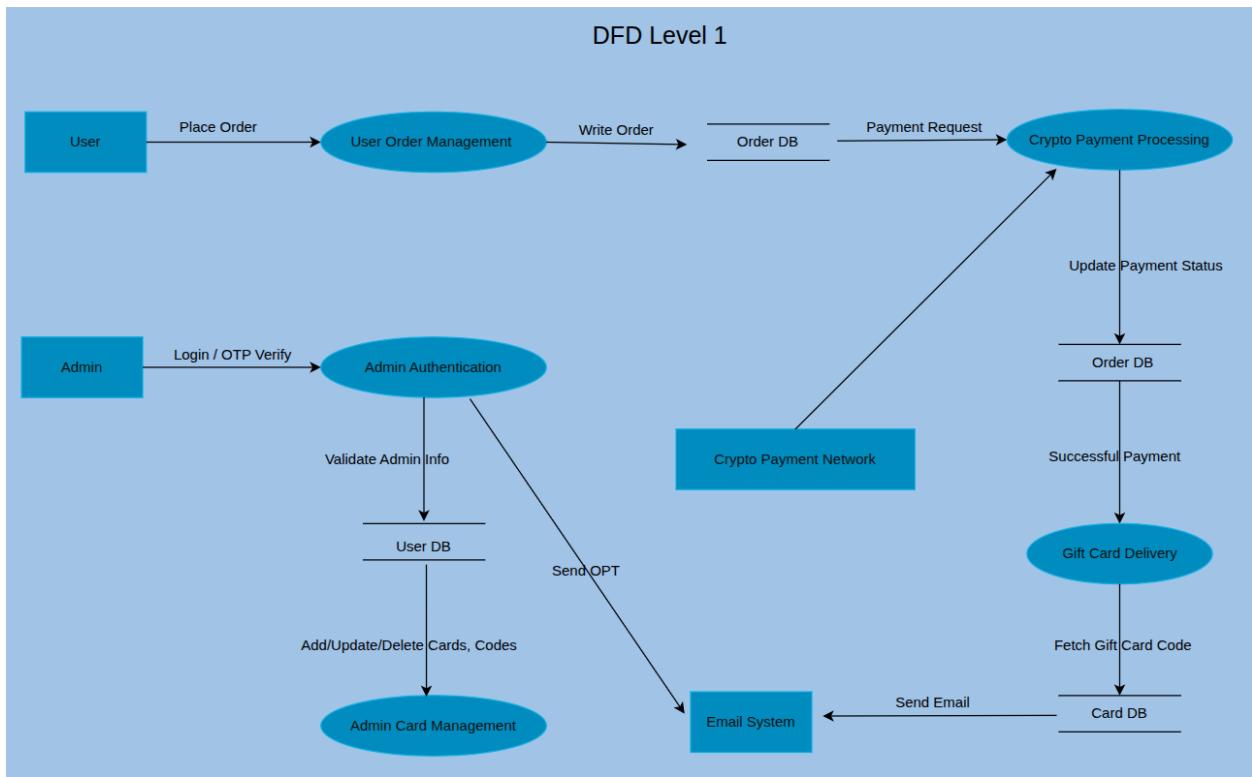


Fig 5-7 DFD Level 1 Diagram

5.6.3 DFD Level-2 (User Order Management)

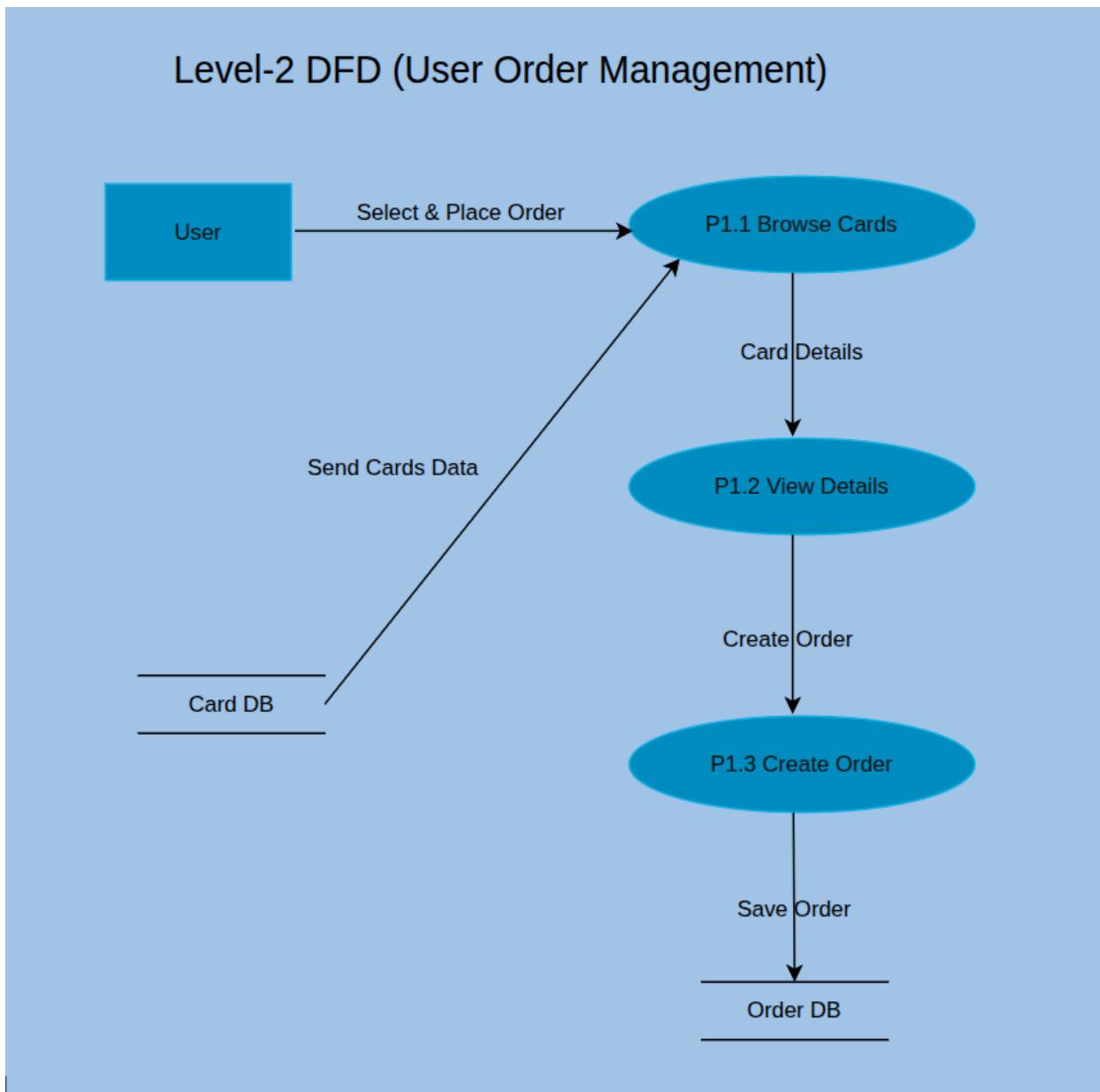


Fig 5-8 DFD Level 2 (User Order Management)

5.6.4 DFD Level 2 (Crypto Payment Processing)

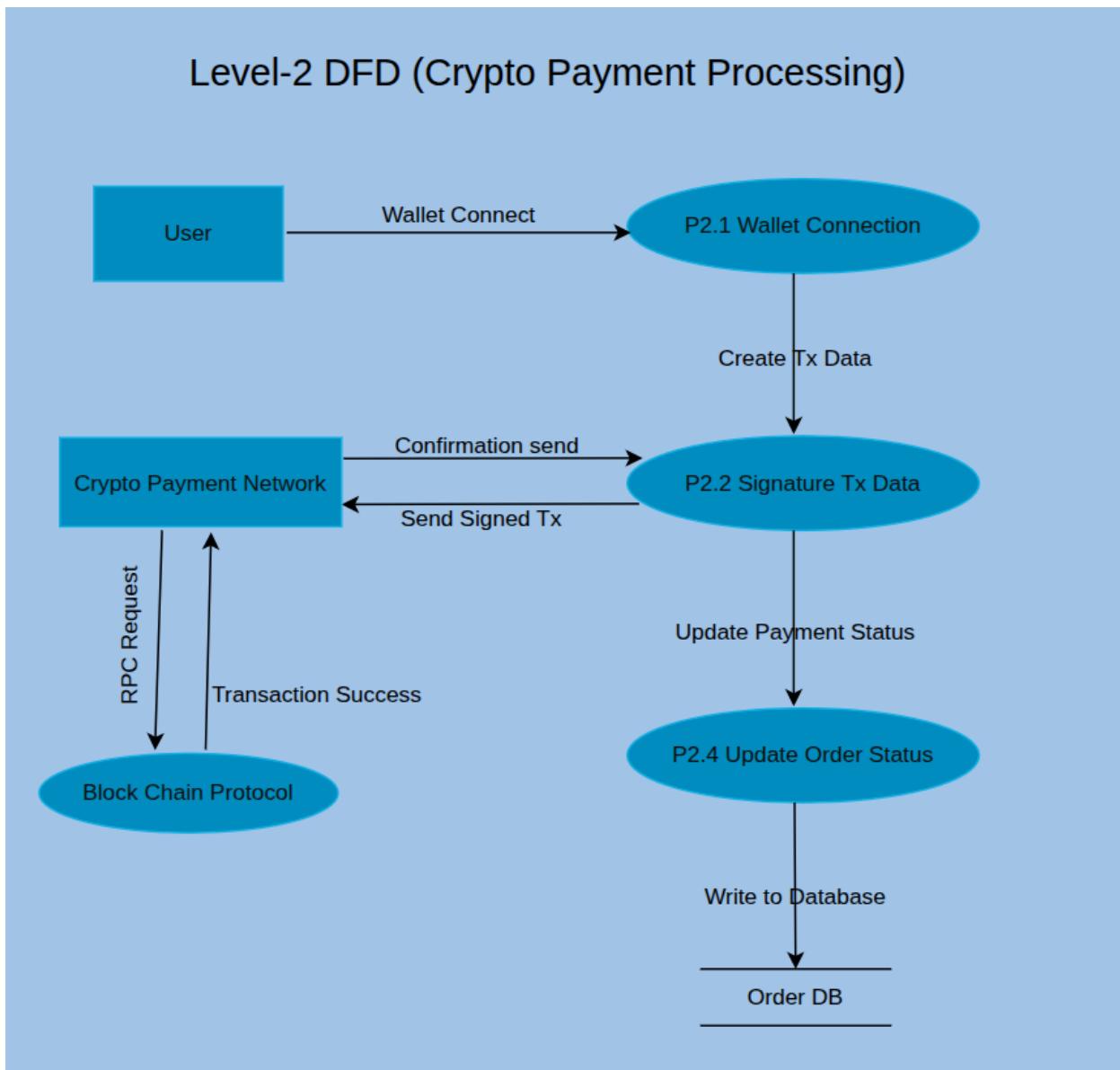


Fig 5-9 DFD Level 2 (Crypto Payment Processing)

5.6.5 DFD Level 2

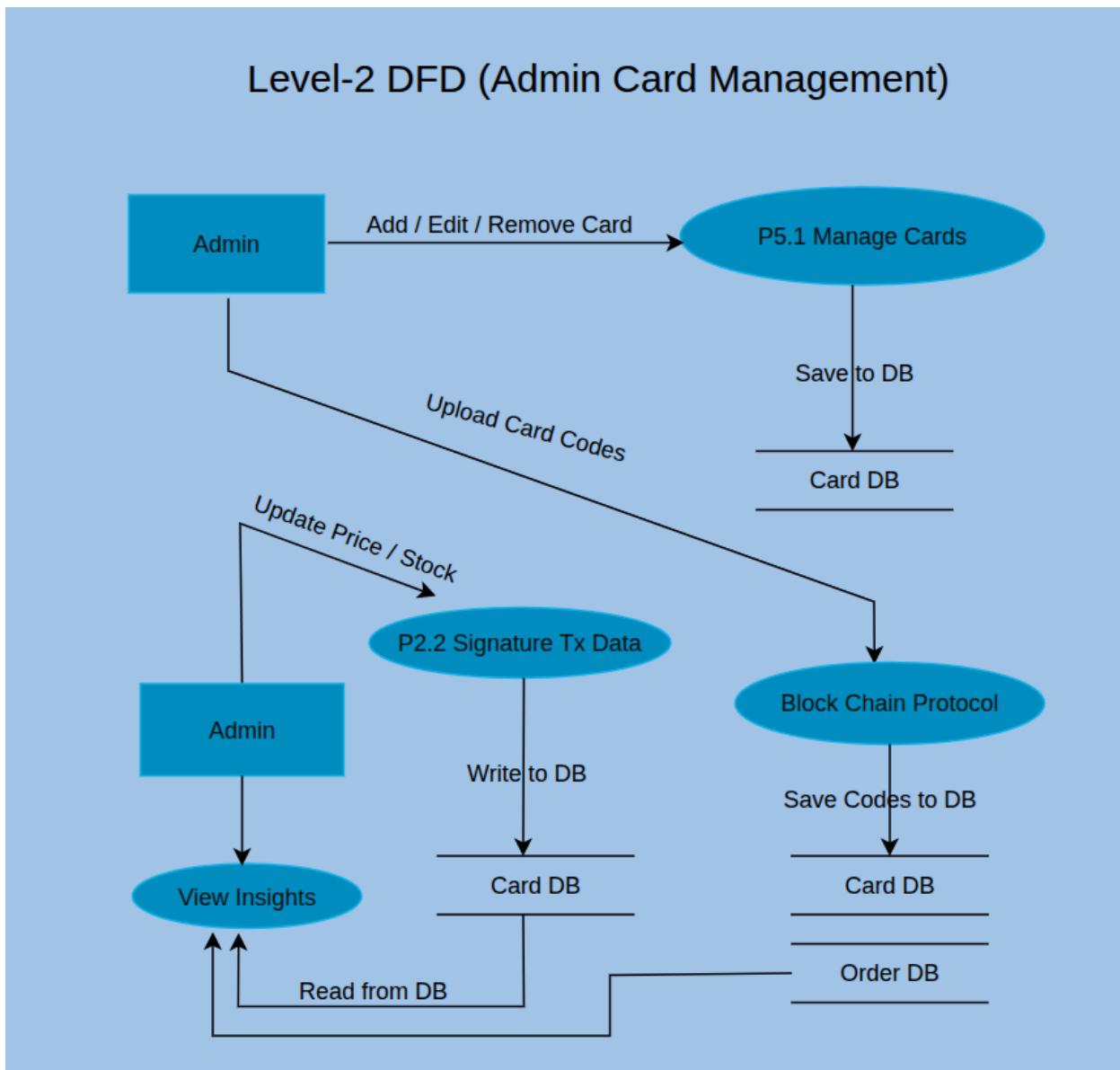


Fig 5-10 DFD Level 2 (Admin Card Management)

5.7 Activity Diagram

An activity diagram is a type of UML diagram used to model the flow of activities or actions within a system or process. It visually represents the sequence of steps, decisions, parallel processes, and the overall workflow from start to finish. Activity diagrams use symbols such as action nodes, decision diamonds, swimlanes, and synchronization bars to show how tasks are organized and who performs them. They are especially useful for analyzing business processes,

system behaviors, and complex use-case scenarios. Overall, an activity diagram helps clarify how a process works and assists in identifying improvements or potential issues.

5.7.1 The User Activity Diagram:

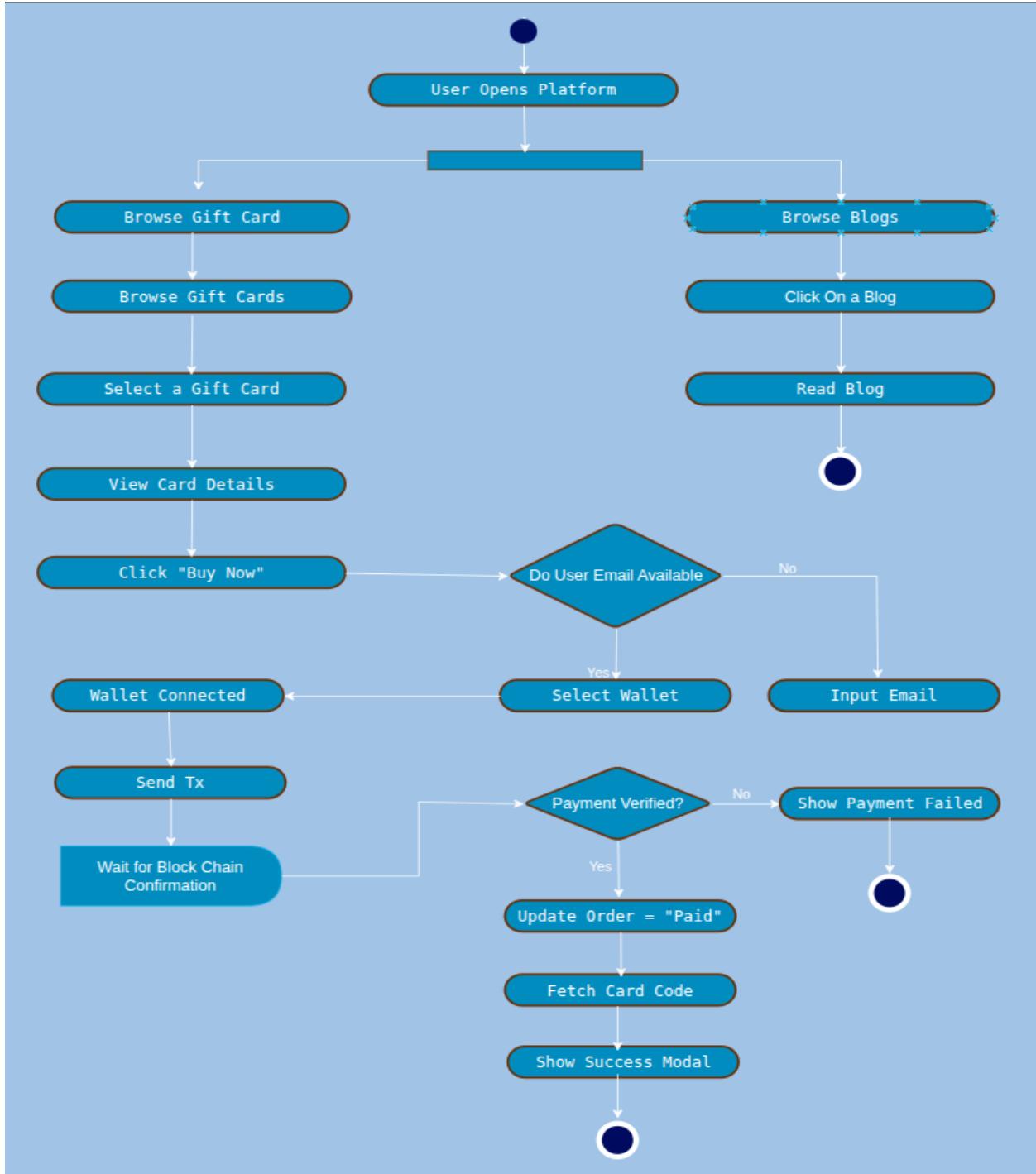


Fig 5-11 User Activity Diagram

5.7.2 The Admin Activity Diagram:

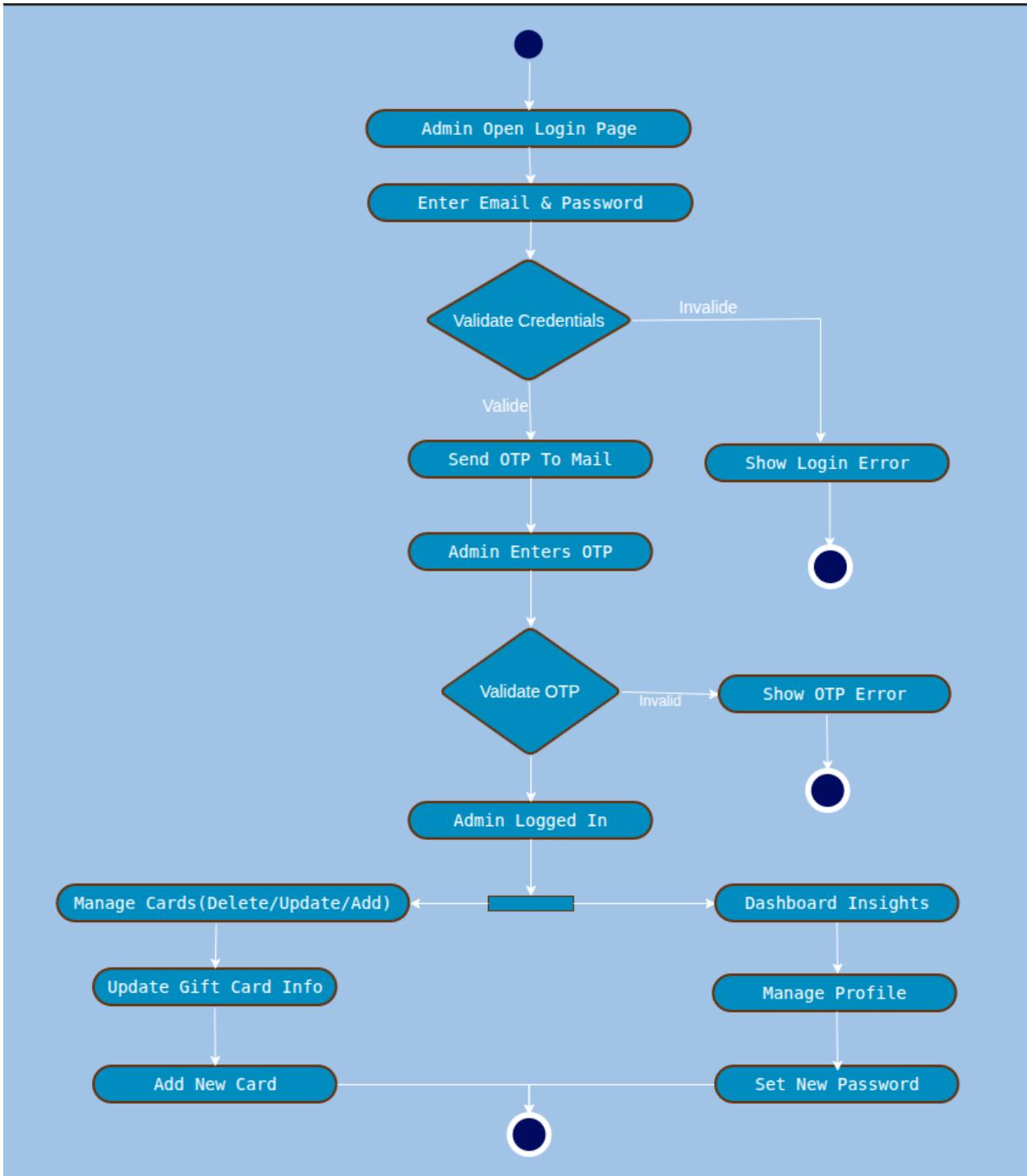


Fig 5-12 Admin Activity Diagram

5.8 Entity Relationship Diagram (E-R diagram)

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

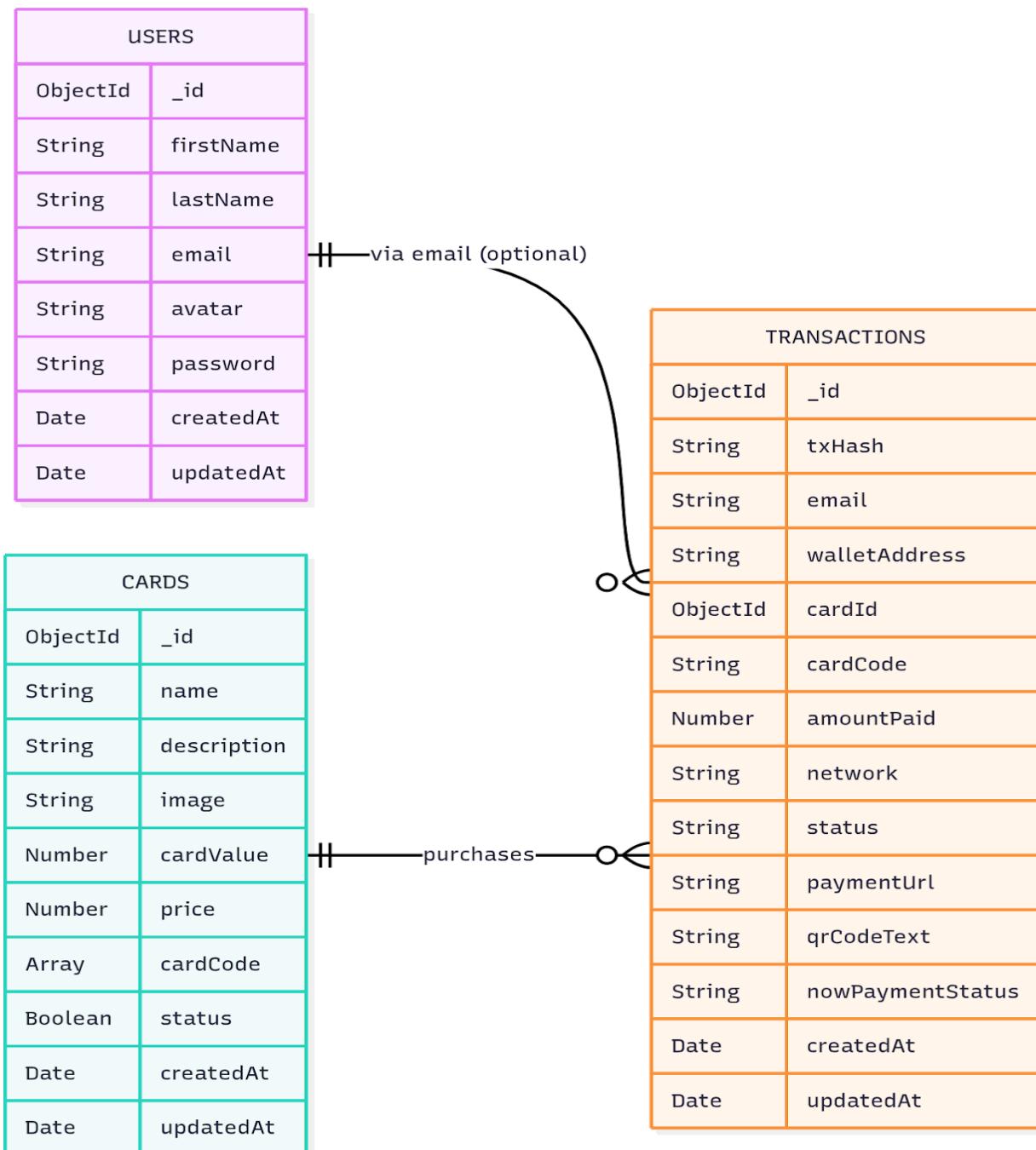


Fig 5-13 ER-Diagram

5.8.1 Schema Overview Users

The **USERS** entity stores essential information about individuals registered in the system. Its attributes—such as `firstName`, `lastName`, `email`, and `avatar`—help uniquely identify users and represent their profile details. Security-related fields like `password` ensure protected access, while timestamps such as `createdAt` and `updatedAt` help track account activity over time. Users can perform different actions in the platform, including initiating card purchases, which are later tracked in the transactions table.

5.8.2 Schema Overview Cards

The **CARDS** entity represents all card-related assets available in the system, such as gift cards, digital cards, or similar items. Attributes like `name`, `description`, `image`, `cardValue`, and `price` define the content and market value of each card. The `cardCode` attribute is stored as an array, indicating that a card may contain multiple redeemable codes. A `status` field helps determine whether a card is active, inactive, or out of stock. Like the **USERS** entity, the **CARDS** entity also includes timestamps to maintain accurate historical records.

5.8.2 Schema Overview Transactions

The **TRANSACTIONS** entity is central to system operations, as it records all purchase activity performed by users. Each transaction contains a unique `txHash`, the user's `email`, and the purchaser's `walletAddress`. It also stores references to related card information through `cardId` and `cardCode`, creating a direct link to the **CARDS** entity. Additional fields such as `amountPaid`, `network`, `paymentUrl`, `qrCodeText`, and `nowPaymentStatus` help track the payment process across different stages. This table is essential for reporting, auditing, and validating completed or pending transactions.

The relationships defined in the ERD clarify how the system components interact. The **USERS** entity has a **one-to-many** relationship with **TRANSACTIONS**, represented as **USERS** ||--o{ **TRANSACTIONS**. This means a single user may have multiple transactions, but a transaction belongs to only one user. Similarly, the **CARDS** entity has a **one-to-many** relationship with **TRANSACTIONS**, indicating that many transactions can involve the same card. These relationships form the logical backbone of the system, enabling reliable data flow and ensuring the platform can efficiently support user purchases, card management, and transaction tracking.

06. User Interface

The user interface, in the industrial-design field of human machine interaction, is the space where interaction between humans and machines occurs. The goal of this interaction is effective

operation and control of the machine on the user's end, and feedback from the machine, which aids the operator in making operational decisions. Examples of this broad concept of user interfaces include the interactive aspects of computer operating systems, hand tools, heavy machinery operator controls, and process controls. The design considerations applicable when creating user interfaces are related to or involve such disciplines as ergonomics and psychology. Now, we see the user interfaces of our project:

6.1 Home Page

This is the first page users see when they visit the platform. The home page gives a clean and simple introduction to the service, showing the main categories of gift cards and highlighting the key features of the system. The goal of this page is to help users quickly understand what the platform offers and guide them toward exploring or purchasing gift cards.

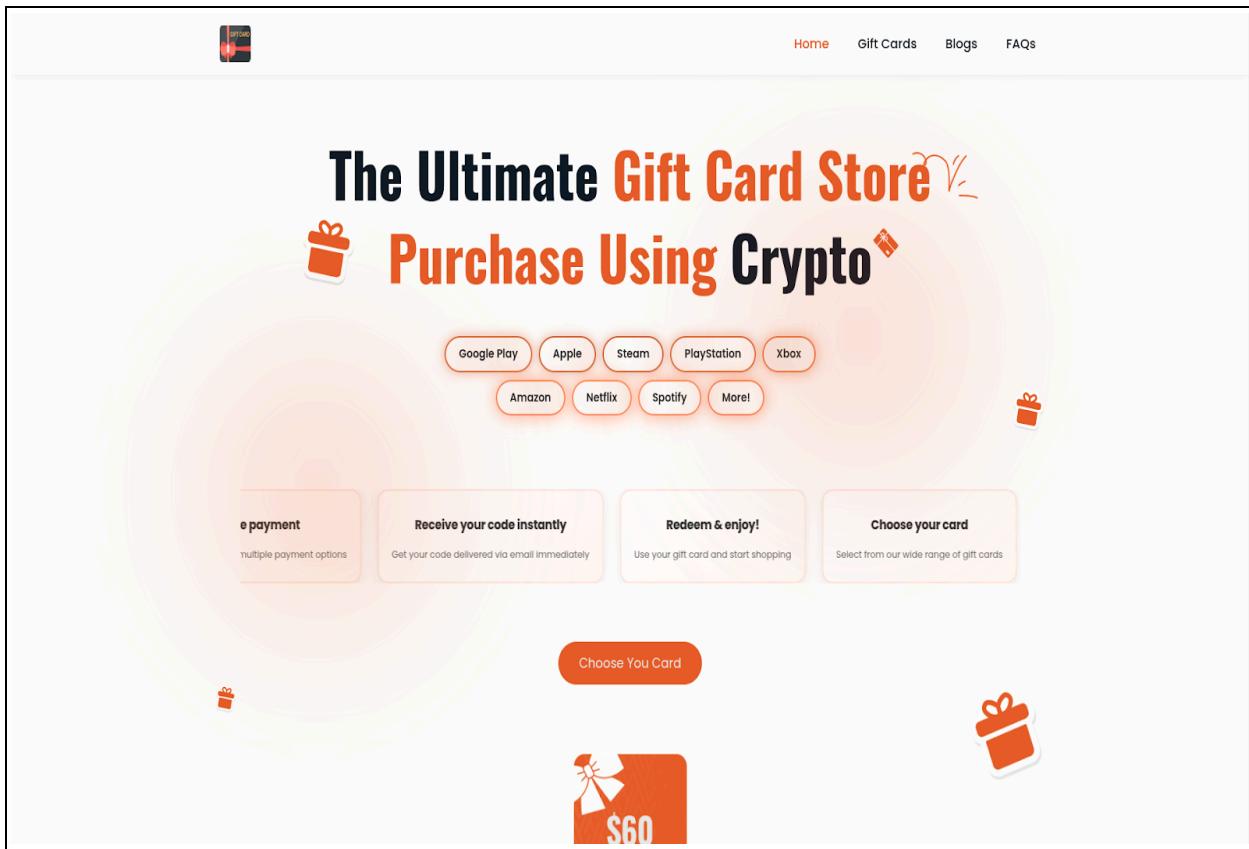


Fig 6.1 UI - Home Page

6.2 Card List Page

This page displays all available gift cards in a structured grid layout. Each card item shows its brand logo, price,. Users can scroll through the list, and click any card to see more details or proceed to purchase.

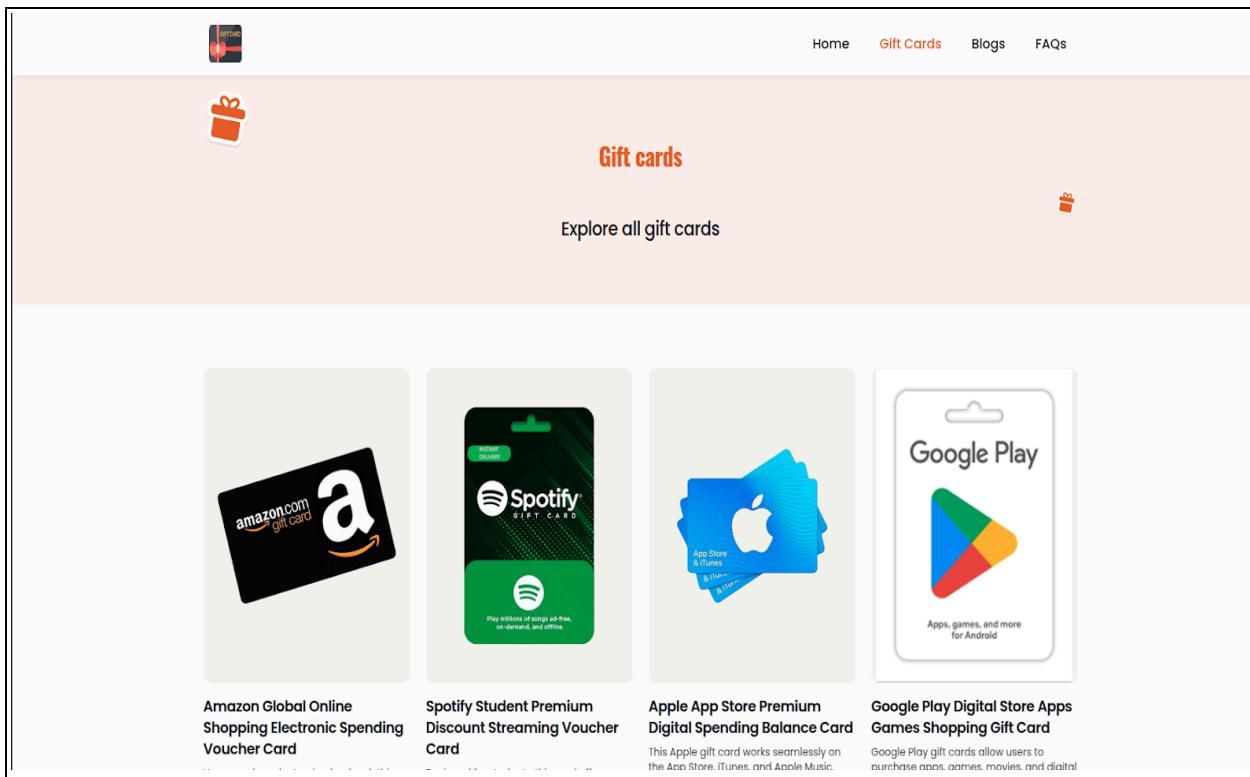


Fig 6.2 UI - Card List Page

6.3 Blogs Overview Page

The blogs overview page shows all published articles in a clean and readable layout. Each blog card includes a title, thumbnail, short description, and publish date. The purpose of this page is to help users learn more about crypto payments, digital gift cards, and related topics.

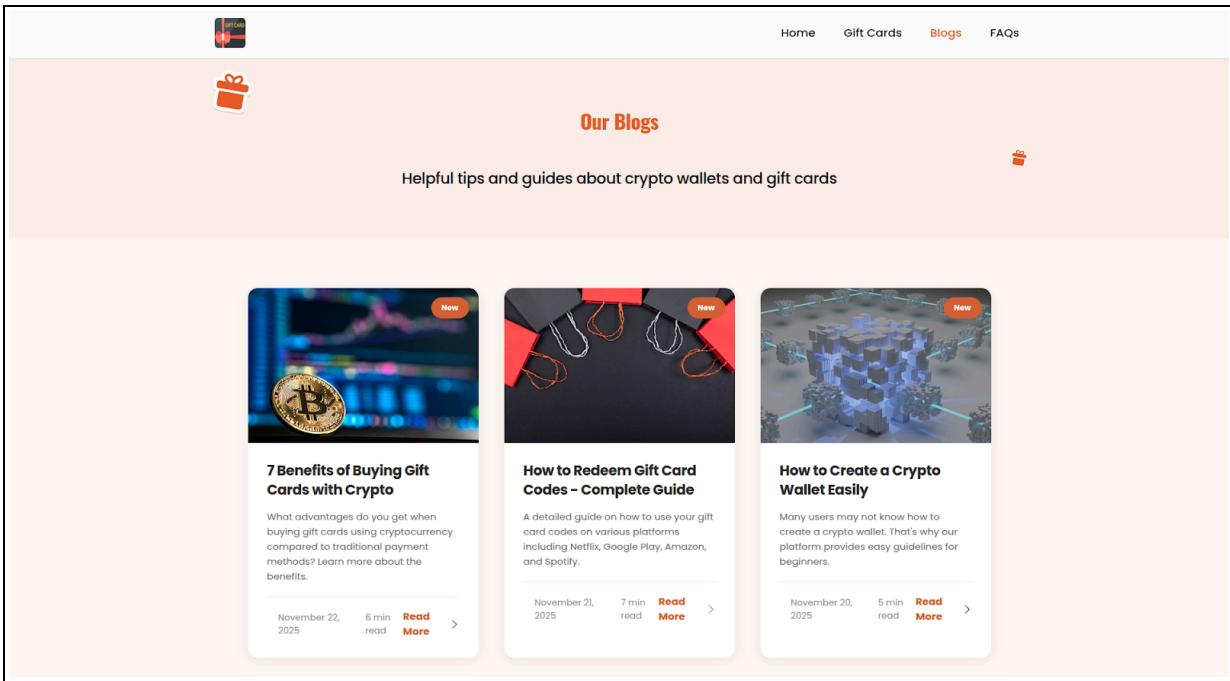


Fig 6.3 UI - Blogs Overview Page

6.4 Blog Post Page

This page displays a single blog article in full detail. It contains the blog title, publish date, main content, and related images. The page is designed for comfortable reading, allowing users to scroll smoothly and explore educational content about digital payments or gift card usage.

Fig 6.4 UI - Blog Post Page

6.5 Wallet Select Modal

This modal appears when the user tries to purchase a gift card. It lets the user choose their preferred crypto wallet—such as MetaMask or TronLink. The interface is simple and focuses on clarity so that users can connect their wallet easily and securely.

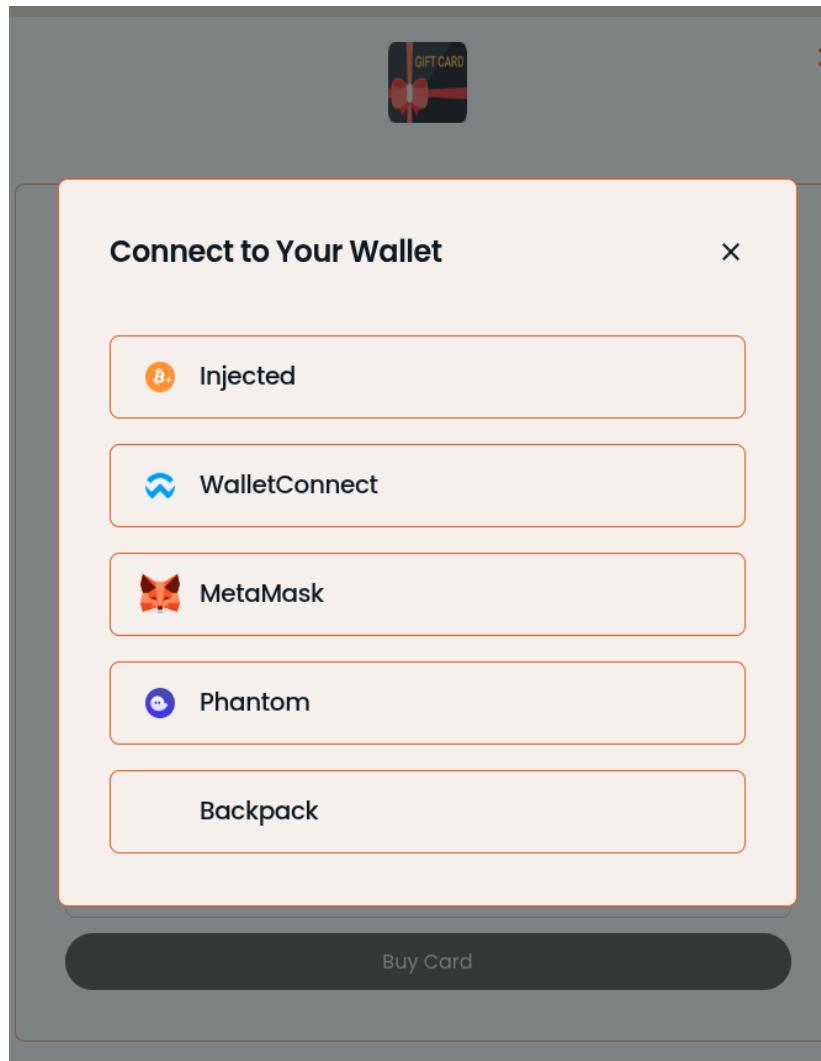


Fig 6.5 UI - Wallet Select Modal

6.6 Mobile Wallet Connect

This modal is used when the user wants to connect a mobile crypto wallet using the WalletConnect protocol. It shows a dynamic QR code that can be scanned from mobile wallets like TrustWallet or MetaMask Mobile. The modal explains that an encrypted session will be created to allow safe and secure transactions.

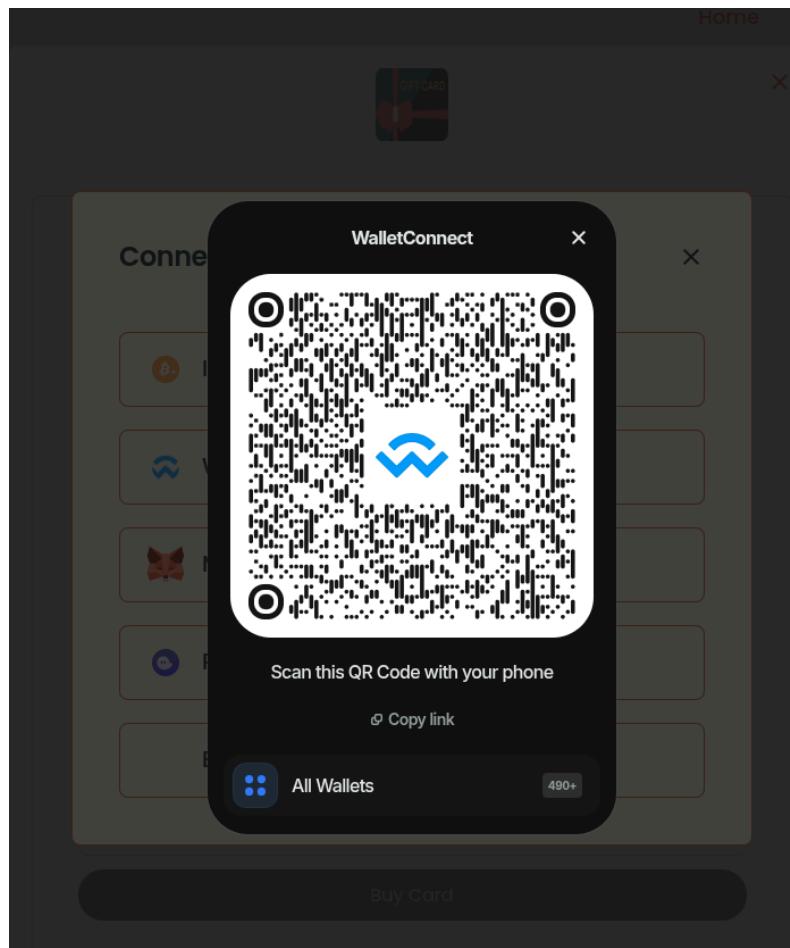


Fig 6.6 UI - Mobile Wallet Connect

6.7 Payment Modal

This modal appears when the user is ready to make the payment for a selected gift card. It provides two options: pay using a browser wallet extension or pay by scanning a QR code. The modal shows the amount, selected network, and confirmation options so users can complete payment easily [6].

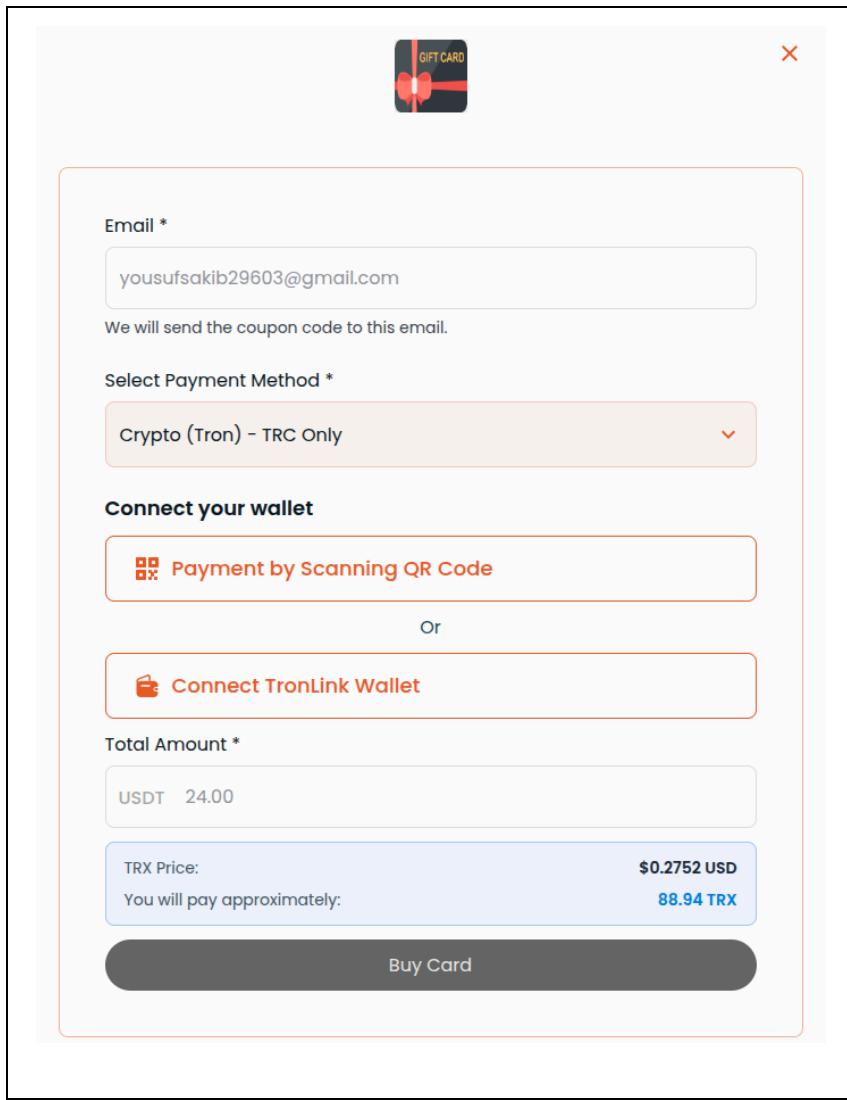


Fig 6.7 UI - Payment Modal

6.8 QR-Code-Pay Modal

This modal displays a generated QR code for users who want to pay from their mobile wallet. The user simply scans the QR code with a wallet app like TrustWallet or Binance Wallet. The modal ensures that the payment request is clear, safe, and quick to complete.

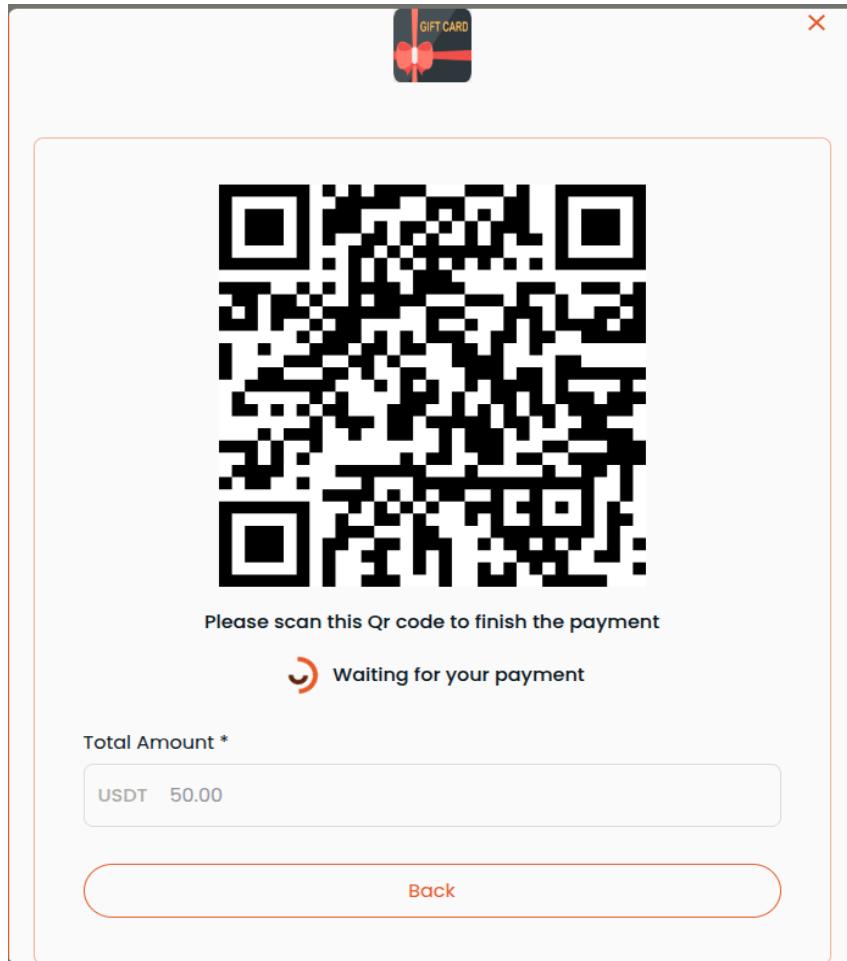


Fig 6.8 UI - QR-Code-Pay Modal

6.9 Admin Dashboard Page

This is the main control panel for administrators. The page shows total orders, total revenue, recently added cards, and revenue charts for weekly and monthly performance. It also includes a quick overview of recent orders. The dashboard helps the admin understand business performance at a glance.

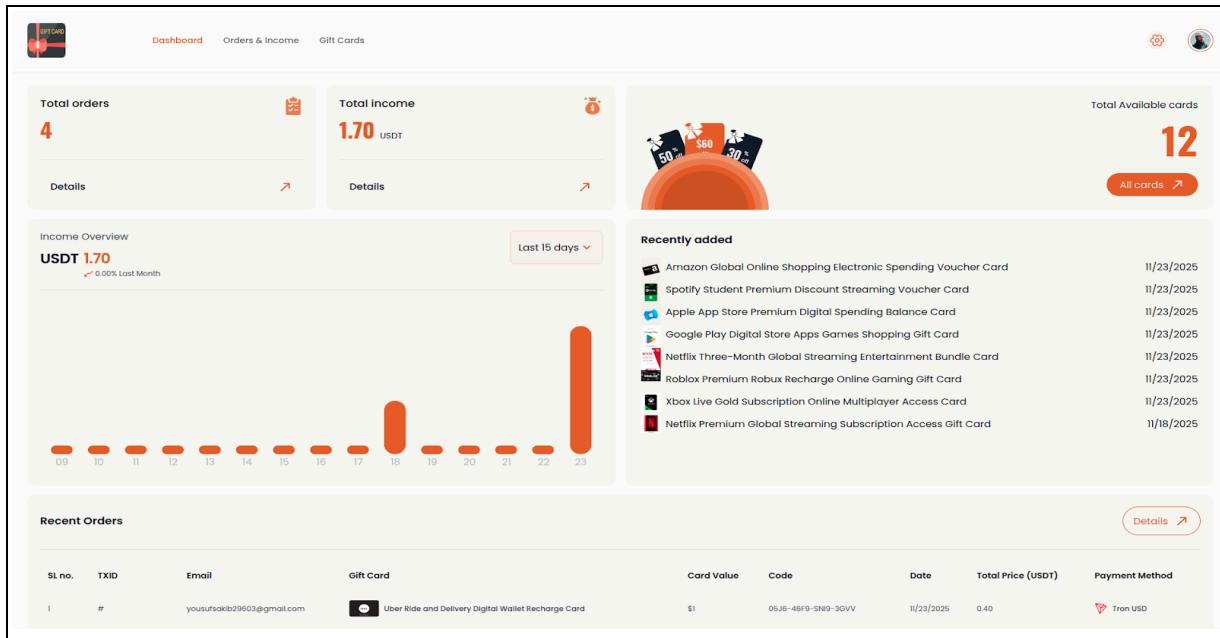


Fig 6.9 UI - Admin Dashboard Page

6.9 Order's Overview and Revenue Page

This page shows detailed insights about all orders. It includes total order count, revenue summary, and a paginated table listing every order. Admins can view customer details, payment status, and order timestamps. The layout is designed for fast tracking and data monitoring.

The screenshot displays the Order's Overview and Revenue Page with the following key metrics:

- Total orders:** 4
- Total Income:** 1.70 USDT

All Orders:

SL no.	TXID	Email	Gift Card	Card Value	Code	Date	Total Price (USDT)	Payment Method
1	#	yousufsakib29803@gmail.com	Uber Ride and Delivery Digital Wallet Recharge Card	\$1	05J8-46F9-SN19-30VV	11/23/2025	0.40	Tron USD
2	#d18b0_b2485	yousufsakib29803@gmail.com	Uber Ride and Delivery Digital Wallet Recharge Card	\$1	05J8-46F9-SN19-30VV	11/23/2025	0.40	Tron USD
3	#ela79_52520	yousufsakib29803@gmail.com	Uber Ride and Delivery Digital Wallet Recharge Card	\$1	315Y-BQJR-RLW0-BOTC	11/23/2025	0.40	Tron USD
4	#4d25d_cf478	yousufsakib29803@gmail.com	Google Play Digital Apps Movies Games Purchase Gift Card	\$1	N4R0-WGND-5HPC-W33Q	11/18/2025	0.50	Tron USD

1-4 of 4 items

Fig 6-9 Order Overview and Revenue Page

6.10 Card's Overview Page

This page shows all gift cards stored in the system. Each card is listed in a paginated table showing its brand, price, active/inactive status, stock status, and buttons to view or edit the card. This allows the admin to maintain and manage cards efficiently.

The screenshot displays a dashboard for managing gift cards. At the top, there are navigation links: Dashboard, Orders & Income, and Gift Cards. On the right side, there are user profile icons and a gear icon. Below the navigation, a large orange circle contains three gift card icons with values \$50, \$60, and \$30. To the right of the circle, it says "Total Available cards" and "12". A red button labeled "+ Add new card" is located at the bottom right of the main area. The main content is a table with the following columns: Product Code, Gift Card, Card Value, Date, Price (USD), Stock Count, Status, and Actions. The table lists 12 gift cards, each with a small thumbnail image, the product code, the card type, its value, the purchase date, the price in USD, the current stock count, its status (all marked as ACTIVE), and two action buttons (one with a magnifying glass and one with a pencil).

Product Code	Gift Card	Card Value	Date	Price (USD)	Stock Count	Status	Actions
#692343e72ef7bd39d521141d	Amazon Global Online Shopping Electronic Spending Voucher Card	\$40	11/23/2025	39.00	2	ACTIVE	
#6923437c2ef7bd39d52113dd	Spotify Student Premium Discount Streaming Voucher Card	\$28	11/23/2025	26.00	3	ACTIVE	
#692342de2ef7bd39d521130d	Apple App Store Premium Digital Spending Balance Card	\$52	11/23/2025	50.00	3	ACTIVE	
#6923427b2ef7bd39d52112d1	Google Play Digital Store Apps Games Shopping Gift Card	\$25	11/23/2025	24.00	3	ACTIVE	
#6923420a2ef7bd39d5211298	Netflix Three-Month Global Streaming Entertainment Bundle Card	\$30	11/23/2025	28.00	3	ACTIVE	
#6923419b2ef7bd39d5211264	Roblox Premium Robux Recharge Online Gaming Gift Card	\$20	11/23/2025	19.00	3	ACTIVE	
#692340df2ef7bd39d5211201	Xbox Live Gold Subscription Online Multiplayer Access Card	\$30	11/23/2025	28.00	4	ACTIVE	
#691fbabb2781b2357597c552	Netflix Premium Global Streaming Subscription Access Gift Card	\$60	11/18/2025	57.00	1	ACTIVE	
af6f03a4674fc1070fca455b4075	Linear Wireless Headphones Bluetooth Stereo Headset Card	41	11/23/2025	N/A	11	ACTIVE	

Fig 6-10 Card's Overview Page

6.11 Add Card/Edit Card Page

This page allows the administrator to add a new gift card or update an existing one. The form includes fields for uploading an image, entering coupon codes, setting the price, writing the description, and choosing active or inactive status. It ensures that card management is simple, flexible, and fully controlled by the admin.

The screenshot shows a web-based application interface for managing gift cards. At the top, there's a navigation bar with links for 'Dashboard', 'Orders & Income', 'Gift Cards', and a user profile icon. Below the navigation, the current location is shown as 'Dashboard > Gift Cards > Edit Card'. The main area is titled 'Edit Card' and contains two main sections: 'Basic Information' and 'Additional Information'.
Basic Information:
- A placeholder for 'Upload Image' with a small thumbnail of an Amazon gift card.
- 'Gift Card Name *': 'Amazon Global Online Shopping Electronic Spending Voucher Card'
- 'Gift Card Description *': A text box containing the description: 'Users can buy electronics, books, clothing, and groceries using this Amazon card. The balance applies instantly to their Amazon account. It works for millions of products worldwide. The card has no expiry, offering full spending freedom. A versatile choice for all types of online shoppers.'
- 'Card Value *': '\$ 40'
- 'Price *': 'USDT 39'
Additional Information:
- 'Gift card code *': Two input fields containing codes: 'N4RO-WGND-5HPC-W33Q' and 'OC5M-Q00Z-58HM-0IKX'. Each field has a minus sign to its right.
- '+ Add another code': A link to add more codes.
- 'Status': A dropdown menu set to 'Active' (indicated by a green dot).
At the bottom are two buttons: 'Save' (in red) and 'Cancel'.

Fig 6-11 Add Card/Edit Card Page

References

- [1] Outposts.io, "JM Gift Launches Crypto Gift Card Service on Tron and BNB Chain," 2024. Available:
<https://outposts.io/article/jm-gift-launches-crypto-gift-card-service-on-tron-and-bnb-2f321123-a315-413e-ba53-d04659abc984>
- [2] FinanceFeeds, "CoinsBee surpasses 5,000 gift card brands, becoming the largest crypto gift card platform worldwide," 2024. Available:
<https://financefeeds.com/bn/coinsbee-surpasses-5000-gift-card-brands-becoming-the-largest-crypto-gift-card-platform-worldwide/>
- [3] BanglaDiary, "ক্রিপ্টোকারেন্সি কী? বিনিয়োগ এবং ক্রিপ্টো কেনার পূর্ণ গাইড," 2023. Available:
<https://www.bangladiary.com/technology/ক্রিপ্টোকারেন্সি-কী-বিনিয়োগ-এবং-ক্রিপ্টো-কেনার-পূর্ণ-গাইড/>
- [4] Jugantor, "ভার্চুয়াল মুদ্রার প্রচলন অট্টরেই শুরু হবে," 2025. Available:
<https://www.jugantor.com/eid-shopping2025/933221>
- [5] M. Harishankar, D.-G. Akestoridis, S. V. Iyer, A. Laszka, C. Joe-Wong, and P. Tague, "PayPlace: Secure and Flexible Operator-Mediated Payments in Blockchain Marketplaces at Scale," arXiv, 2020. Available: <https://arxiv.org/abs/2003.06197>
- [6] A. K. Shrestha, S. Joshi, and J. Vassileva, "Customer Data Sharing Platform: A Blockchain-Based Shopping Cart," arXiv, 2020. Available: <https://arxiv.org/abs/2004.13890>
- [7] P. Weijland, "The Giving Game: A Game-Theoretic View on Gift Economics in Blockchain Systems," arXiv, 2021. Available: <https://arxiv.org/abs/2105.11761>