Introduction

Loyalty programs keep customers engaged, but most reward points are locked to a single brand, limiting their usefulness. A shared rewards system would give customers more flexibility, but businesses may resist, fearing loss of exclusivity.

This document explores ways to make a cross-platform rewards system beneficial for both customers and sellers by designing incentives, ensuring smooth technical integration, and maintaining financial sustainability. By aligning business goals with customer needs, this system could transform loyalty programs into something far more valuable for everyone.

### Unified Reward Wallet System Proposal

We propose the development of a new digital wallet system that seamlessly integrates with sellers through APIs, enabling a reward-based transaction model similar to UPI. This system will function as a web and app-based platform, allowing buyers to register using their phone numbers for a smooth onboarding experience.

Getting Started:

To encourage early adoption, both sellers and buyers will receive **f**ree reward coins upon registration.

* Sellers get the flexibility to integrate the system with no initial costs.
* Buyersimmediately receive coins, giving them a reason to engage with the system right away.

Earning and Spending:

* Buyers earn reward coins through purchases, special offers, and interactions with sellers.
* Sellers have the ability to reward buyers with coins based on their purchases, boosting loyalty and increasing sales.
* To redeem the reward coins the user have to enter the unique ID which it will when he will sign-up.
  + From the backend a request will be send to the users wallet will it have to approve.

A screenshot of a coupon

Description automatically generated

Flexibility in Reward Redemption:

* Unlike fixed-point systems, we don’t set a fixed conversion rate between reward coins and monetary value. This is because every seller has different profit margins and pricing strategies.
* By not fixing the rate, we give sellers the freedom to offer varying rewards based on their own business models. For example: One electronics shop might offer 10% off for 500 coins, while another might offer 15% off for the same 500 coins, this creates healthy competition among the sellers and gives sellers the flexibility to design offers that attract customers, driving more sales and increasing market competition.

Coin Validity and Redemption:

Reward coins will have a validity of 3 months. To encourage timely redemption and keep the system fresh, we have a dynamic validity extension process, 60% of total coins will automatically be redeemed when validity expires, helping users get the most out of their rewards while encouraging more regular purchases.

Aggregated Analytics for Sellers:

* Sellers will have access to aggregated, anonymized analytics on cross-platform spending patterns. This will allow businesses to see where their customers are spending and fine-tune their offers.
* Sellers can also purchase targeted campaigns, attracting users who frequently redeem points in their specific product category. This adds a layer of personalized marketing, enabling businesses to reach the right customers at the right time.

Seller Incentives for Adopting the Reward Wallet System

* Increased Sales & Customer Retention: Boost sales and foster customer loyalty with flexible reward offerings.
* Customizable Offers to Drive Business: Set tailored reward redemption rates to align with your profit margins and attract more customers.
* Cost-Effective Marketing: Run targeted campaigns that reach customers most likely to redeem points, maximizing marketing ROI.
* Simple Integration & Low Operational Overhead: Seamlessly integrate the system with minimal setup and ongoing operational costs.

### Technologies Planned for Wallet Platform**:**

#### API: The API will access customers to:

* + Access their wallets.
  + Perform transactions.
  + Check coin balances.

Request Parameters**:** API Endpoint: The application or website initiates a transaction by sending a request to a designated API endpoint

* Sender's Wallet ID: The unique identifier of the user initiating the transaction.
* Receiver's Wallet ID: The unique identifier of the recipient (or the business).
* Transaction Amount: The number of points or coins to be transferred.
* Transaction Type: E.g., "redeem," "add points," or "transfer."
* To ensure security and prevent fraud:
  + Use authentication methods like OTPs before making any query to the consumer's wallet.
  + Implement short-lived access tokens instead of long-lived API keys to minimise the risk of API key leakage.

Authentication**:** OAuth 2.0: Implement OAuth 2.0 for secure delegated access and authorization Or OpenID Connect: Use OpenID Connect for user authentication and identity verification.

Database**:**

For the databases we are planning to use Relational Database: MySQL for structured data storage, such as user accounts and transaction history.

Why are we using:

* Ideal for structured data with defined relationships (e.g., users, businesses, transactions).
* Provides ACID compliance (Atomicity, Consistency, Isolation, Durability), which ensures reliable financial transactions.
* Supports complex queries for analytics and reporting.
* Encryption: Apply Transport Layer Security (TLS) for encrypted communication between the database and the application.

Use Cases:

* Storing user accounts, wallet balances, and transaction histories.
* Managing business-specific reward configurations.

#### 

#### Frontend**:**

The front end is critical to ensuring a smooth, intuitive, and responsive user experience for your wallet platform. Since the platform must cater to end-users (customers) and businesses (vendors), the design and technology stack should emphasize performance, accessibility, and security.

Frameworks:

* React.js or Next.js: For a seamless and modern web interface that is fast, scalable, and user-friendly.
* Utility-first CSS framework for rapid UI development with minimal custom CSS.

Front-End Features:

* Customers Dashboard
  1. Display wallet balance, reward points, and transaction history.
  2. Option to redeem points directly at partner stores or websites.
  3. Notifications for expiring points or offers.
* Brand Dashboard
  1. Track issued rewards and redemptions.
  2. Create and manage promotional campaigns.
  3. View analytics for user behaviour (e.g., popular redemption patterns).

#### Cost-Effective Operations**:**

* Hosting: Use cloud services like **AWS**, **Google Cloud**, or **Azure**, prioritizing serverless options like **AWS Lambda** for scalability and cost reduction.
* **CDN**: Use **Cloudflare** to ensure faster content delivery and reduced server costs.

Security Measures

1. API Key Leakage: Use short-lived access tokens for API authentication.
2. Denial-of-Service (DoS) Attacks: Implement Web Application Firewalls (WAFs) for protection.
3. Denial-of-Service (DoS) Attacks: Use CAPTCHA to prevent bots from overwhelming endpoints.
4. Denial-of-Service (DoS) Attacks: Utilize AI/ML tools to detect and block bot-like behavior in real-time.
5. Sensitive Data Exposure: Encrypt sensitive data at rest using AES-256 encryption.
6. Sensitive Data Exposure: Encrypt sensitive data in transit using TLS.
7. Sensitive Data Exposure: Ensure compliance with data protection regulations such as GDPR and CCPA.
8. User Security Enhancement: Implement Multi-Factor Authentication (MFA) for added security.
9. User Security Enhancement: Generate one-time passwords (OTPs) for high security.
10. Integration Issues: Use an API Gateway like Kong, Apigee, or AWS API Gateway for managing API calls.
11. Transaction Failures: Implement caching layers (e.g., Redis) for quick balance checks.
12. Security Vulnerabilities: Use TLS for encrypting all communication.
13. High Load and Scalability Challenges: Utilize cloud-based infrastructure with autoscaling capabilities.
14. User Management Problems: Implement multi-factor authentication (MFA) for enhanced security.

Additional Feature**:**

#### **Cross-Platform Analytics:**

* Provide aggregated and anonymized analytics on user spending patterns across platforms.
* These insights can help sellers understand trends and behaviors.
* Enable sellers to purchase targeted campaigns designed to attract users who frequently redeem points in their category.
* Use analytics to create effective, personalized marketing strategies.

In conclusion, the Unified Reward Wallet System aims to revolutionize loyalty programs by offering unparalleled flexibility, seamless integration, and value for both customers and businesses. By fostering collaboration, enhancing customer experiences, and empowering sellers with actionable insights, this system has the potential to redefine how rewards are earned, redeemed, and managed creating a win-win ecosystem for all stakeholders.