future internet

MDPI

Article

Blockchain-Based Loyalty Management System

André F. Santos , José Marinho and Jorge Bernardino *

Polytechnic of Coimbra, Coimbra Institute of Engineering (ISEC), Rua Pedro Nunes, 3030-199 Coimbra, Portugal; a21280412@isec.pt (A.F.S.); fafe@isec.pt (J.M.)
* Correspondence: jorge@isec.pt

Abstract: Loyalty platforms are designed to increase customer loyalty and thus increase consumers' attraction to purchase. Although successful in increasing brand reach and sales, these platforms fail to meet their primary objective due to a lack of incentives and encouragement for customers to return. Along with the problem in originating sales, they bring excessive costs to brands due to the maintenance and infrastructure required to make the systems feasible. In that sense, recent blockchain technology can help to overcome some of these problems, providing capabilities such as smart contracts, which have the potential to reinvent the way loyalty systems work and solve current problems. Although blockchain is a relatively new technology, some brands are already investigating its usefulness and rebuilding their loyalty systems. However, these platforms are independent and linked directly to a brand. Thus, there is a need for a generic platform capable of creating and managing different loyalty programs, regardless of the size of the business. This paper explores the shortcomings of current loyalty programs identified through the literature review, and proposes a loyalty management system with blockchain integration that allows any retailer to create and manage their loyalty programs and have customers interact directly with multiple programs in a single application.

Keywords: loyalty programs; blockchain; smart contracts; loyalty management

check for updates

Citation: Santos, A.F.; Marinho, J.; Bernardino, J. Blockchain-Based Loyalty Management System. *Future Internet* **2023**, *15*, 161. https://doi.org/10.3390/fi15050161

Academic Editors: Christoph Stach and Clémentine Gritti

Received: 18 March 2023
Revised: 21 April 2023
Accepted: 25 April 2023
Published: 27 April 2023

1. Introduction

A loyalty program is a marketing strategy used by companies to reward their most loyal customers. Its other goals include attracting new customers, encouraging repeat purchases, and collecting information about customers. Currently, there are a wide range of loyalty programs; this is an issue because different applications are needed and the rewards do not suit all customers. Moreover, there is no interaction between the different programs, which makes it almost impossible to keep track of the status of all of them, and means users have to provide their information multiple times.

Some loyalty programs have found a way to operate with different businesses, but there are many difficulties in tracking information outside of their scope, for example, when a partner provides information for an external service.

A number of loyalty management systems have emerged as a way of overcoming these problems. These loyalty management systems provide new possibilities to build loyalty programs, and the tools to monitor them, faster. At the same time, they offer a way to create an ecosystem of loyalty programs that supports the user experience. While they solve drawbacks such as user experience, they still have problems related to third-party tools and incomplete connection between different systems.

Therefore, blockchain may be a viable option to address these challenges. The concept of blockchain consists of a chain of blocks that allows the storage of information. Each block stores a set of transactions, and these blocks are part of a network that any user can access. This technology introduces a new paradigm based on three core ideas: security, decentralization, and disintermediation [1].

For maximum security, this technology uses distributed consensus algorithms that ensure that every transaction is trustworthy. Additionally, it utilizes mechanisms of cryptography and digital signatures to ensure only the owner can access its data [1]. A final security-related concept is persistence, which means the data cannot be changed. Concerning decentralization and disintermediation, these two properties play a crucial role. Decentralization allows the user to control and to be the owner of the information it produces, as opposed to the current reality of large companies such as Meta and Google. Another property of a decentralized network is the trustless environment, which means no one needs to know or trust another entity, and each member has a copy of the data. This environment prevents anyone from corrupting the data and optimizes the distribution of resources [2].

Since blockchain does not use third-party tools to analyze and verify transactions, it uses the concept of smart contracts. Smart contracts are self-executing programs stored on the blockchain. These contracts are triggered when the established conditions are met. Usually, they are used to automate the execution of an agreement between entities so that all of them can be aware of the expected result. This mechanism brings benefits such as performance and efficiency, trust and transparency, and security, as it only works as long as the conditions are met and stored later in the immutable blockchain. Additionally, blockchain may allow people to save money as it avoids delays, fraud, and intermediaries [3].

This paper explores the shortcomings of current loyalty programs, identified through a literature review, and how the blockchain technologies might overcome them. Therefore, an innovative blockchain-based loyalty system is proposed, with the following contributions:

- A fully decentralized platform;
- Management of multiple loyalty programs of different types on a single platform;
- Integration of meta transactions without including personal information, providing a seamless experience for any level of customer expertise;
- The ability to handle multiple tokens within a single contract through the use of an ERC-1155 interface;
- Reduction of financial and maintenance costs for retailers by facilitating near real-time transactions and eliminating the need for intermediaries and proprietary infrastructures;
- Provision of better exposure to retailers and a better user experience for customers through a single marketplace in which all loyalty programs coexist.

The rest of this paper is structured as follows. Section 2 presents a literature review, which includes a background on loyalty programs, blockchain, smart contracts, and meta transactions; it also discusses related work. Section 3 describes and explains the proposed loyalty management system. Section 4 discusses the proposed system and compares it with other systems. Finally, Section 5 presents our conclusions and future work.

2. Literature Review

This section first provides an overview of the current state of loyalty programs and the challenges they face. In addition, it clarifies how blockchain technology has the potential to revolutionize these programs by introducing technical concepts that can facilitate a more effective approach. Second, this section presents a comprehensive analysis of the current state of the art regarding the use of blockchain in various domains and contexts associated with loyalty programs.

2.1. Background

Loyalty programs are one of the main options for brands to increase their accessibility and attractiveness. However, these systems have shortcomings that can influence interaction with users. Blockchain is an innovative technology capable of transforming the behavior of these systems and solving current problems. For better understanding and context, this subsection describes the issues of current loyalty programs and presents how blockchain technology complements these programs. The concepts of smart contracts and meta transactions are also explained.

2.1.1. Traditional Loyalty Programs

Currently, traditional loyalty programs are based on a variety of mechanisms, and it is possible to classify them based on several categories: stamp cards, points, cashback, tiers, and coalition [4]. However, despite being a dominant choice and offering several tools to attract new customers and retain them, these programs have problems related to customer and company perspectives [5].

Starting with the customer perspective, the main current difficulties are as follows:

- The first step in joining a new loyalty program is usually to provide the system with personal information, which nowadays, with unquestionable security concerns, may imply a constraint for some users. Additionally, customers feel that they belong to too many loyalty programs, since each brand has its own program [6];
- With traditional approaches, the customer does not own assets such as points, tiers, and others. Instead, the program controls and stores the assets inside its database. Additionally, the value of the assets is changeable due to the company having the power to decide when to inflate or deflate it. These programs naturally have limits due to the way they are implemented. An example is that rewards only have value within the scope of the application and cannot be transferred to another system [5];
- The customer experience of traditional loyalty programs is not satisfactory. Despite their popularity, customers are less motivated to continue interacting with the system due to the effort–reward relationship. Most programs set an expiration date for rewards, which indirectly requires the customer to spend them before the deadline, even if he/she does not intend to. Another factor that reduces the customer experience is the lack of customization in the rewards for customers, with their needs and desires not usually considered (each customer is different, but compensation is the same for all) [7].

From the perspective of companies, the main difficulties consist of the following:

- Companies are responsible for protecting all the information of customers. Moreover, to process the transactions, they rely on third party entities, which makes them very costly. Apart from the need for intermediaries, a company is also responsible for infrastructure management, which adds overwhelming complexity and increases costs;
- Building partnerships with the traditional approach is complex due to the nature of the programs. Usually, brands conceive of loyalty programs with the intention of expanding their own business, and therefore, said program becomes restricted to its original scope. Whenever a company decides to build a partnership with another company, the typical solution consists of creating a new loyalty program. Consequently, another issue arises: that of who owns the assets (since with the traditional approach, the customers do not own them).

2.1.2. Blockchain

Blockchain is a recent technology that has introduced a new paradigm for the secure storage and sharing of information. This technology essentially consists of a distributed database of a chain of blocks, where each block stores a transaction. To guarantee security, blockchain utilizes several mechanisms, such as cryptography, consensus algorithms, and digital signatures [1]. All transactions are performed in a decentralized way, removing the need for intermediaries, and once inside a block, the transaction cannot be deleted. This modern technology is based on the following core principles [8]:

- Decentralization—one of the main goals of blockchain is to remove an entity's ability to have absolute control of all data. Hence, this technology is based on a network in which every user has equal power and the users themselves validate the transactions [3];
- Immutability—every node in the network holds a copy of the digital ledger. To add a new transaction, all nodes must check the transaction, and most of them must validate it. Validated records become irreversible and cannot be changed. In a scenario of a sudden change in one node, all nodes check the validity of the new state. When all

nodes finish the verification, a voting process is started, and if the majority rejects the change, the new state is rejected. Then, a copy of a trusted digital ledger is sent to that node [8];
- Transparency—any user can access any record by accessing the node and consulting the timestamp [1];
- Security—when interacting with the blockchain, it is mandatory for a user to enter the key every time s/he wants to create a transaction. Each record is individually encrypted. The information inside the blockchain is hashed cryptographically, which means that each piece of data has its own unique identifier [8];
- Disintermediation—in order to avoid the use of third-party entities, blockchain takes advantage of smart contracts. These contracts are used to facilitate the negotiation process but still provide security and reliability to the process.

Although blockchain is a relatively new technology, there is growing interest in and research into its applicability across a range of sectors. In [1], the authors describe what distinguishes blockchain from currently known paradigms, and characterize their specifications from fundamentals to architecture. After specifying the capabilities of the technology, the authors present recent studies on the incorporation of blockchain in certain areas. The domains involved in these studies are financial activities, health, information systems, wireless networks, IoT (Internet of Things), smart grids, government and military services, and defense. For each specified domain, this survey presents some examples of applicability and their respective benefits, limits, and challenges.

2.1.3. Loyalty Programs Complemented with Blockchain

Blockchain integration will reshape the way current loyalty programs work and add some benefits. First, companies will not need to spend more money on infrastructures to manage the assets and the system. Furthermore, with blockchain integration, third parties are removed due to the characteristics of decentralization, disintermediation, and consensus, leading to financial savings. Data security is now guaranteed by blockchain mechanisms, leaving the company less burdened.

It will become simpler to create partnerships with other companies because they are in a shared ecosystem (blockchain) in which there are no restrictions, as any asset is valid.

For end users/customers, there are also some benefits. The first of these is the elimination of any physical support for managing rewards, such as stamp cards. Secondly, users do not need to provide any personal information to interact with the blockchain. The user just needs a virtual wallet. In addition, with the possibility for companies to partner with each other more easily, the number of loyalty programs the user belongs to will decrease. The user will also be able to trade assets between loyalty programs or even with other users. Finally, the user will be the owner of her/his assets in these loyalty programs, which is the opposite of the traditional approach.

2.1.4. Smart Contracts

Smart contracts are self-executing computer programs stored on the blockchain. The primary purpose of these contracts is to provide a more secure way to produce transactions without having to rely on third parties and question whether the information has been altered for personal benefit. These contracts contain the logic to proceed with transactions, with this logic being defined by programmers. Each contract, when stored in the blockchain, receives a unique address, which is the entry point for the interaction.

To interact with smart contracts, every user needs a virtual wallet, as every interaction with the blockchain requires a fee payment due to processing effort. The hosted blockchain's native cryptocurrency is used to pay this fee. During the interaction with a smart contract, the user sends data to the contract that will be processed based on its logic.

The benefits of smart contracts are speed and efficiency due to automatization processes. Once a given condition is met, the execution of the associated contract is triggered, and the transaction is performed without errors or any kind of bureaucracy. Associated with automatization, these contracts offer trust and transparency, since there are no third-party entities, and the blockchain encrypts every transaction and shares it across the nodes. Additionally, with a consensus mechanism it is nearly impossible to modify transactions for personal benefit. As a result of their nature, smart contracts help entities save money and time by eliminating intermediaries and the associated delays and expenses.

Currently, the adaptability and reusability of smart contracts is an open research issue. In [9], the author proposed a design pattern to implement the reusability of verification rules across multiple contracts, which allows reconfigurability at runtime. The study was conducted using permissioned blockchains and Java as the programming language. The proposed design pattern exploits polymorphism and manages inheritance through sealed classes, and includes two layers: abstract and concrete. The abstract layer includes an abstract contract class with a single abstract method ("checkSC"), as well as an interface for rule implementation. The concrete layer contains all the contracts, which are classes that extend the abstract contract class and therefore implement its single abstract method (i.e., the contract behaviour). Furthermore, the concrete layer includes the definition of rules, where each rule is encapsulated by a class that implements the rule interface. Although a simple pattern, the proposal in [9] reduces code redundancy and allows configuration at runtime.

In the same context, the paper [10] details a design pattern, called the proxy pattern, that allows smart contracts to be updated without losing data. OpenZeppelin provides an extensible plugin based on this pattern. It comes with extensive documentation explaining its functionality [11]. Unlike a single contract that combines both data and business logic, the proxy pattern involves splitting the contracts into a proxy contract and a business logic contract. The proxy contract acts as a dedicated storage and delegates calls to the business logic contract, while the business logic contract handles all the business logic operations. In the proposed design pattern, the read and write operations performed by the delegated contract have an impact on the storage of the proxy contract.

2.1.5. Meta Transactions

Every day, thousands of smart contracts are deployed on existing blockchains. Smart contracts can offer significant advantages, but they can also present challenges for new users in the blockchain space. The primary issue is related to user onboarding, which requires the use of native cryptocurrencies to interact with the contracts and can have an impact on the overall user experience. Users who are not familiar with the blockchain environment will need to pre-purchase cryptocurrency to use the loyalty application.

Meta transactions are a mechanism that allows anyone to interact with a blockchain regardless of their level of knowledge on this topic, and to make transactions without fee payments. This is achieved by decoupling the owner of the data to be sent and the entity that will pay the fee [12].

The meta transactions' flow consists of the following three steps [12]:

1. The user who wants to interact with a smart contract creates a transaction request. This transaction is equivalent to a usual smart contract transaction, except that it will be signed with the user's private key and additional parameters to "craft" the transaction;
2. Following the transaction request creation and signature phase, the request is sent to an entity that acts as a relayer and validates the transactions according to a specification (whitelist for example). Once validated, the relayer sends the transaction to the blockchain network in which the targeted smart contract is hosted;
3. Finally, the smart contract receives the transaction and performs the intended action.

Meta transactions are a design pattern that offers a seamless experience to users, in which they do not have to spend money to engage with the blockchain.

2.2. Related Work

This section presents an analysis of the current state of the art regarding the use of blockchain in various domains and contexts associated with loyalty programs.

In [13], the authors propose and describe an implemented loyalty program that is based on blockchain. The authors identified issues related to the mechanisms of customer dissatisfaction and loyalty. Afterward, they explain the solution they propose, which consists of a smart contract created in the Ethereum network [14] and capable of producing tokens. These tokens are called TECH tokens (the name given by the authors) and are similar to cryptocurrencies. TECH tokens are stored in Ethereum wallets and can be transferred between wallets to perform payments or exchange goods in the system. Since the Ethereum network requires a fee for each transaction and the cost is expressed in the ether cryptocurrency, the authors implemented and tested the system on Rinkeby, an Ethereum-compatible test network. The implementation consisted of three distinct phases. First, the smart contract is written and deployed on the blockchain. Then, the token is created according to ERC20 standards [15], and finally, the frontend is developed in React.js for user interaction. The system has integrated Web3.js as a way to allow the frontend to communicate with the blockchain.

The authors in [5] propose a universal loyalty platform. Before presenting the solution, they detail how blockchain can improve a loyalty program from both the point of view of customers and companies. The proposed solution consists of an application in which customers can use points to redeem rewards, and a website for companies to interact with and manage their business. The system uses React Native and React for mobile and website frontend, respectively. The system also uses Stellar and Hyperledger Fabric Blockchains to manage users' information and assets. At the customer level, upon registration, a Stellar account is created. This account allows the user to trade any asset type, being able to exchange points between loyalty programs. At the company level, upon registration, a Stellar account is created. During the registration process, the company can choose other companies to collaborate with. Once this entire process is over, the smart contract allocated in the Hyperledger blockchain is invoked to create and connect the inherent token to that company and link it to the Stellar account.

Two studies from 2019 [16,17] aimed to explore how blockchain can impact loyalty programs and understand their relation. Both studies presented four theoretical foundations: (1) self-determination theory (SDT) [18]; (2) customer perceived value; (3) loyalty programs' design in loyalty program efficiency; and (4) blockchain application. The authors use SDT to define four dimensions of customer motivation: economy, autonomy, competence, and relatedness. To better understand the link between SDT and customer perceived value, the authors assess customers' perceived value in three domains that they proposed: economic utility, psychological self-fulfillment, and social interaction. For the last two theoretical foundations, the authors focus more on loyalty programs, specifically on design effectiveness and blockchain infrastructure. Both papers chose BubiChain as the target blockchain infrastructure, due to its success in the loyalty program market, to undertake the evaluation. They also chose an exploratory case study on a loyalty points-based system. Although the works are analogous, the depth of the study differs between them.

The authors of the first paper [16] only assess the main characteristics of blockchain integration and its impact on each SDT domain. The second paper [17] evaluates the BubiChain infrastructure itself and how it works, the main elements of the loyalty points-based system, how the code is structured, and, finally, the impact of BubiChain on participation in loyalty programs. While the exploratory case with blockchain has proven to be more effective than existing traditional schemes to satisfy customers, both papers state that more future research is needed to explore the relationship between blockchain design for loyalty programs and customer behaviors.

The work described in [7] aims to review the potential and challenges of blockchain in marketing areas for building brand loyalty. The authors first describe blockchain technology and its unique features. Then, an overview of the characteristics of brand loyalty and current failures is presented. The authors of the paper also explain how integration with blockchain infrastructure can eventually solve and improve the quality of loyalty programs, making them more effective and affordable.

The authors in [19] propose an implementation of a loyalty program for a mobile platform based on the Waves blockchain. The choice for this blockchain platform is due to its lightweight and fast features when compared to Ethereum, and to the premise that the cell phone is an essential commodity today. The proposed solution consists of three key components: a token creator in Waves, a web API, and a database to store accounts' information.

In [6], the authors give a simple introduction to blockchain and aim to analyze its impact on the retail industry and customer loyalty. Although the authors focus both on the supply chain and loyalty areas, their emphasis is on the need for a loyalty program as the best solution to retain customers. According to them, customers remain dissatisfied even though numerous companies are now investing in loyalty programs. To support this claim, the authors provide statistics from a survey in which they list several problems pointed out by the respondents. They also suggest that blockchain integration could potentially address these issues and provide a lower cost system with better security and transparency.

In [20], the author studies the importance of blockchain technology in enhancing loyalty programs. The scope of this paper is restricted to the airline business. Regardless of the study domain, the author explains how blockchain can impact and revolutionize a company's experience. Integrating blockchain technology from a business perspective could reduce maintenance costs and enhance customer service. Blockchain can foster better customer relationships and captivate customers by providing the benefits of traditional loyalty programs and better customer service. From the point of view of the transaction process, blockchain also makes all the validation, storage and sharing of transactions easier, since intermediaries would no longer be needed. For better context, the author explains the five basic principles of blockchain. This study consists of a survey that was carried out using two sources of information: the primary source, using 450 questionnaires, and the secondary source, the literature. The 450 questionnaires were distributed among customers and airline managers, and a high-efficiency response rate was achieved. The use of blockchain technology was emphasized by the surveyed airlines. They concluded that the degree of effective use of customer loyalty programs is related to blockchain technology, and the results showed that loyalty systems were effectively improved by blockchain integration.

In [21], the authors propose a blockchain-based framework to enhance loyalty programs in the fast-moving consumer goods (FMCG) industry. This framework aims to revolutionize the conventional process of exchanging coupons by introducing a token called promotion asset exchange (PAX). Instead of customers purchasing products and retaining coupons for later trading with merchants, followed by the cumbersome process of merchants trading numerous coupons with the manufacturing company, which can lead to counting errors and result in confusion, customers will scan a QR code on the product packaging, and the promotion (PAX token) will be stored in their wallet. Once the customer has accumulated enough PAX tokens, they can be traded with the merchant, which will trigger a transfer to the merchant's wallet. To receive payment, the merchant can return the PAX token to the manufacturing company, which will refund it to the customer based on the number of tokens. The authors emphasize that blockchain technology enables the secure tracking of all transactions, enabling manufacturing companies to better understand customer needs while ensuring no coupons or payments are lost due to human error.

The authors of [22] propose a blockchain-based platform for coalition loyalty program management. The primary objective of the platform is to provide customers with the capability to exchange their points across distinct existing blockchain loyalty programs. The system consists of different companies that operate their loyalty programs, customers who accumulate points through these programs, and a platform that facilitates token exchange. The token exchange platform incorporates smart contracts that define the exchange rates between the tokens of each company. To ease this process, the companies are required to negotiate the rates and establish smart contracts with the rates. To enable interoperability, the authors utilize sidechain technology for conducting transactions across different blockchains. In order to encourage customer engagement, the authors allow customers to participate in the blockchain consensus protocol and receive rewards in return. The authors have selected the proof of stake (PoS) protocol due to its low computational requirements, minimal delay, and potential to attract more customers. The proposed system ensures that customers can effortlessly exchange their points with other companies, and companies can easily join the coalition/partnership without the need for a system overhaul.

The authors of [23] propose a blockchain-based platform that facilitates the exchange of loyalty points among customers that is similar to the work described in [22]. The proposed system leverages a many-to-many matching method based on the call auction approach used in the stock exchange market, thereby eliminating the need for exchange rate negotiations. The call auction method increases the likelihood of orders being executed, prevents points from going unused, and incentivizes customers to accumulate more points. In contrast to the system proposed in [22], in which companies are responsible for handling rate negotiations and defining the corresponding smart contracts, this system [23] uses the call auction method to simplify the points exchange process, making it more efficient and eliminating the need for partnerships and negotiations. Both systems aim to enhance the current blockchain-based loyalty programs and improve customer experience.

A study on the impact of a blockchain-based loyalty program on customer loyalty in the renewable energy sector was conducted by the authors in [24]. More specifically, this paper addresses the problem of green electricity tariffs (GETs), where companies often fail to generate the promised amount of renewable energy, leading to dissatisfaction, mistrust, and customers considering switching to competitors. First, the authors present a theoretical study assessing how blockchain can increase institutional trust and decrease institutional distrust. They then propose a loyalty program that allows users to track electricity generation data, monitor their electricity consumption, and optimize their consumption by setting rules for their smart devices. The proposed loyalty program will provide customers with tokens that can be redeemed to reduce the price of GETs, donate to charity, use other utility services such as electric scooters or car sharing, or reinvest in renewable energy shares. While affordability is a system requirement in [24], it is not clear whether customers are responsible for paying blockchain fees, which may be impractical due to high transaction prices on the Ethereum network. This problem could be solved by introducing meta transactions. In addition, the smart contract architecture of the proposed system is not demonstrated, and it is unclear whether the system utilizes multiple contracts or any interface for token creation. Although the authors have achieved the desired outcome, the results are specific to the renewable energy sector. Furthermore, the proposal in [24] is supported by a qualitative analysis that includes interviews and a literature review, but lacks quantitative analysis.

Although there are already some proposals for blockchain integration, such as [5,13,19,21], they present a poor user experience with respect to blockchain onboarding, mainly due to the need for user knowledge, the authentication process, the input of personal information, and the lack of meta transactions. Moreover, they contradict the premise of blockchain decentralization using centralized database systems. Systems such as [22,23] propose innovative concepts and solutions to address the challenges faced by existing loyalty programs that rely on blockchain technology. All these systems use blockchain to manage the loyalty programs' infrastructure.

3. Proposed Loyalty Management System

In order to properly design a loyalty management system, first of all, we have to understand the problem and consider some factors. These factors include understanding blockchain capabilities and how they fit into the proposed system, addressing current customer and retailer concerns, and providing a better overall user experience.

The proposed system is a mobile application capable of managing several loyalty programs through interaction with a blockchain. Regardless of the knowledge and role of the user (i.e., retailer or customer), the application is designed to completely abstract users from the underlying blockchain.

3.1. System Description and Features

The system is a self-managed loyalty platform with which retailers of all sizes can create loyalty programs in which their customers can participate. The platform has two distinct users (customers and retailers), and the loyalty programs can include a variety of mechanisms. Despite the system requiring the use of blockchain technology, it abstracts and helps users in onboarding by providing each one with a wallet. These wallets are connected to the users themselves and are private. Furthermore, customers are not required to pay any transaction fee generated by the blockchain interaction, since the resulting costs are intended to be supported by the retailers.

This system includes two main components: a mobile application and smart contracts. Most blockchain applications are web applications, which limits interaction and provides an entry barrier for some less tech-savvy users. On the contrary, a mobile solution improves accessibility, taking advantage of the ubiquity of smartphones in everyday life. Despite having two types of users, the mobile application is the same for both of these types. The objective of smart contracts is to manage the loyalties’ mechanisms and to ensure that the assets of the users are secure.

One of the most critical aspects of loyalty programs is the need for security measures, which is especially relevant today, since security is a major global concern. With the objective of ensuring customer privacy, the proposed system does not store any personal information. On the other hand, it uses Web 2.0 authentication methods, which allow users to use their previously created accounts, thus preventing them from providing any personal information whenever they access the loyalty management system. Although the data stored on the used blockchain are publicly accessible and can be viewed by anyone, the underlying consensus protocols make it possible to validate transactions and identify any potential risk, which makes the blockchain a useful technology for ensuring data security. Furthermore, by implementing appropriate smart contracts, addresses not whitelisted can be excluded, and any malicious changes detected during transactions can be immediately reversed, thus ensuring data integrity.

Despite the mobile application being unique, both types of users have different use cases. Customers have access to a marketplace in which all the loyalty programs are within the scope of the application. For each loyalty program, a customer can view the information related to the business and a description of the loyalty program itself. Additionally, the customer can enroll in loyalty programs to start participating and earning rewards. Once the customer subscribes to a program, he/she can begin to spend and earn rewards. The system helps customers to track all the assets related to the various loyalty programs so that they can check and be aware of their progress.

Retailers have different use cases, as their goal is to attract more customers, thus increasing their business. Retailers can create a loyalty program by providing their business information and by choosing a loyalty mechanism along with a brief description. Additionally, during this process, the retailer can upload images to give a better perception of the loyalty program or business. Since the main goal is to increase brand reach, the retailer can access the analysis and statistics data of the loyalty program. As the retailer must pay transaction fees, he/she can also buy the required cryptocurrency in the app.

The mobile application provides a transaction system based on a QR code approach to trading assets with retailers/customers, thereby enhancing the overall user experience. All transactions require the existence of two individuals, and the rewards are defined by the retailers during the process. The two individuals must be one retailer and one customer, since customer-to-customer and retailer-to-retailer transactions are not possible.

Figure 1 gives an overview of the proposed loyalty management system, illustrating the main components and interactions. Customers and retailers, which are the only actors in the system, use their personal smartphones running the mobile application to access the system.

A customer accesses the system to interact and discover new retailers, while a retailer must create its loyalty program and exchange rewards with customers. The mobile application employs Web 2.0 credentials for user login. To enable customer interaction with the underlying blockchain network at no financial cost, each transaction originated by a customer first passes through a relayer, which then redirects it to the respective smart contract. On the contrary, a retailer interacts directly with the smart contracts. For the execution of asset transfers or rewards, the receiving entity displays the QR code while the other agent reads it. Upon completion of the QR code-reading process, the transfer of assets from the QR code reader to the receiver is triggered through a transaction sent to the smart contract. Depending on whether the transfer is being performed by the retailer or the customer, it will go directly to the smart contract or through the relayer, respectively.

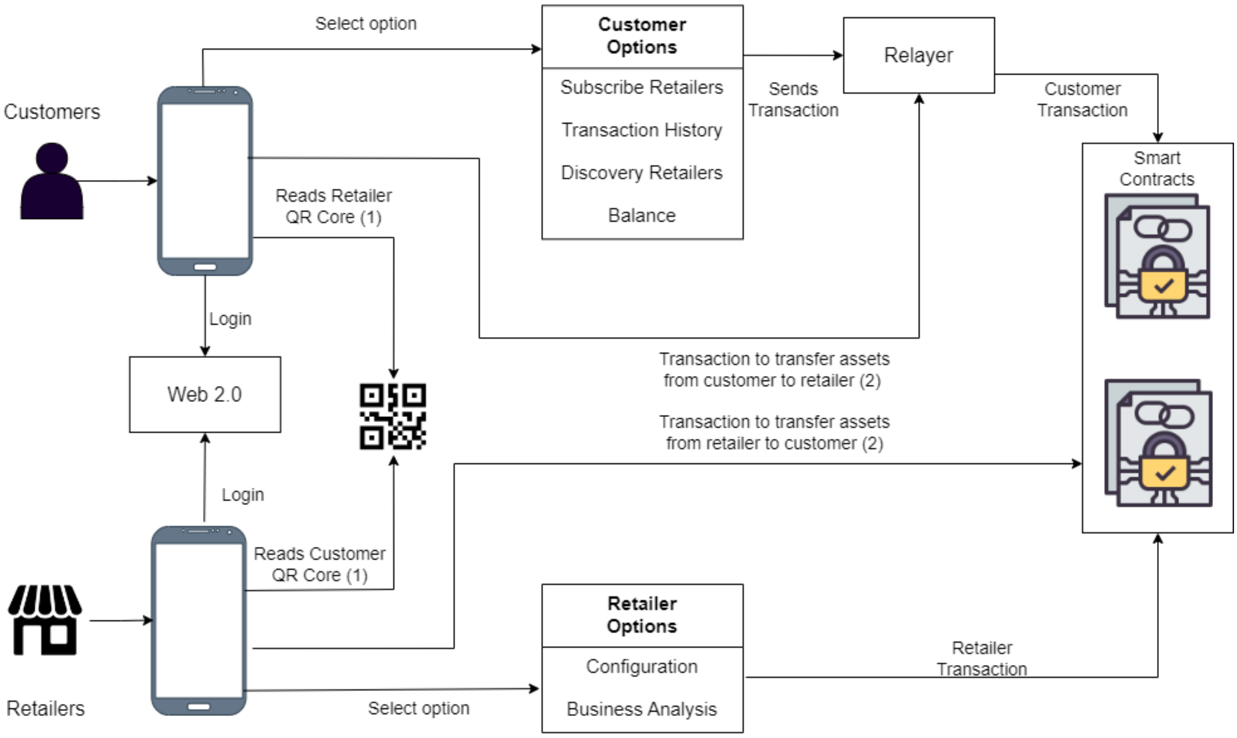
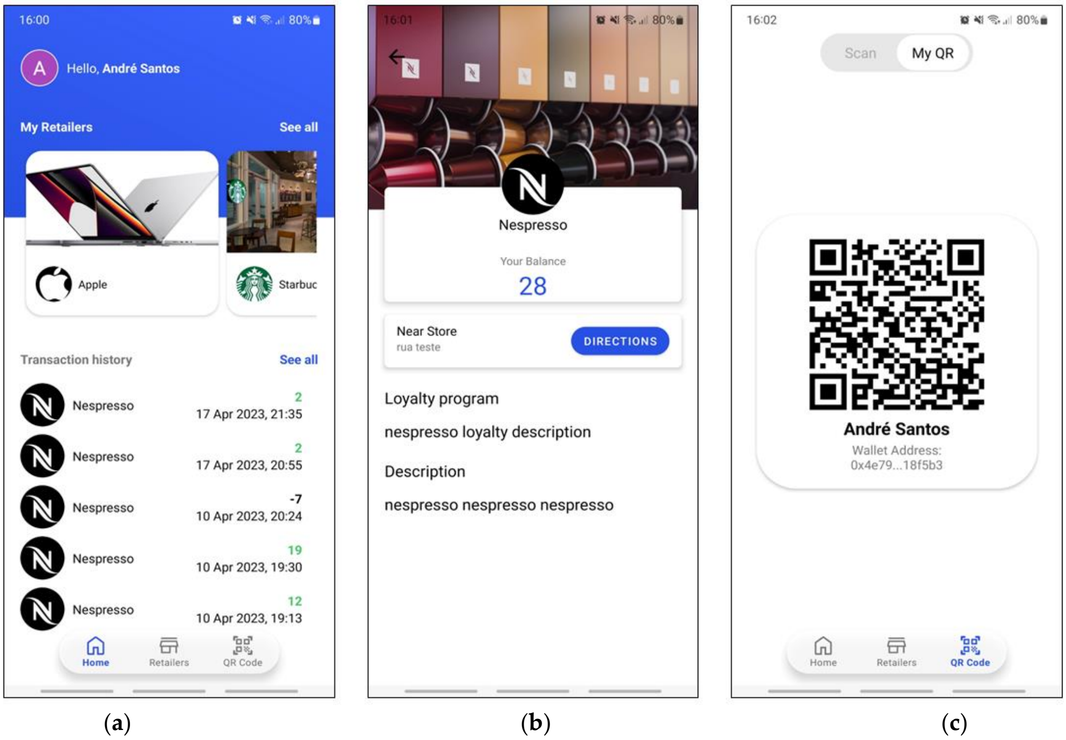


Figure 1. Proposed loyalty management system overview.

Figure 2 presents three screenshots of the mobile user interface. The first screenshot shows the customer's home page, which provides a concise summary of the retailers that the customer is subscribed to, along with their five most recent transactions. The second screenshot shows the retailer's page, which provides comprehensive information about the retailer and its loyalty program, including the rewards available. Finally, the third screenshot shows the QR code that the user will need to scan in order to redeem their rewards.



16:00 80%

A Hello, **André Santos**

My Retailers **See all**

Apple

Starbuc

Transaction history **See all**

Nespresso 2 17 Apr 2023, 21:35

Nespresso 2 17 Apr 2023, 20:55

Nespresso -7 10 Apr 2023, 20:24

Nespresso 19 10 Apr 2023, 19:30

Nespresso 12 10 Apr 2023, 19:13

(**a**)

16:01 80%

Nespresso

Your Balance

28

Near Store
rua teste

Loyalty program

nespresso loyalty description

Description

nespresso nespresso nespresso

(**b**)

16:02 80%

André Santos

Wallet Address:
0x4e79...18f5b3

(**c**)

Figure 2. Mobile app screenshots. (**a**) customer profile; (**b**) retailer page; (**c**) QR code.

Following an agile methodology, Tables 1–3 summarize the system user stories mentioned previously.

Table 1. Common user stories.

User Story	Description	Name
US-1	As a user, I need to login so that I can use the app	Login
US-2	As a user, I need to have a QR code identifier so that I can receive transactions	Display QR code
US-3	As a user, I need to scan QR codes so that I can send transactions	Read QR code

Table 2. Customer user stories.

User Story	Description	Name
US-4	As a customer, I need to check my loyalty balance so that I can be aware of my progress	Check loyalty balance
US-5	As a customer, I need to access my transaction history so that I can track all my transactions	Transaction history
US-6	As a customer, I need a profile page so that I can see my system information	Profile page
US-7	As a customer, I need a discovery page so that I find new businesses to subscribe to	Discovery page
US-8	As a customer, I need to subscribe to a retailer so that I can start earning rewards	Subscribe retailers

Table 3. Retailer user stories.

User Story	Description	Name
US-9	As a retailer, I need to configure my profile so that I can create my loyalty program	Retailer configuration
US-10	As a retailer, I need to check my business status so that I can gain insight into my success	Analysis of the business
US-11	As a retailer, I need to exchange money into cryptocurrency so that I can pay for blockchain interactions	Currency exchange

3.2. Blockchain Selection

One of the most critical components within the proposed system is the choice of the underlying blockchain platform. Currently, there are numerous blockchain networks with distinct characteristics and purposes. For this specific type of loyalty management system, some key factors are essential, such as the number of transactions per second, cost per transaction, ability to support smart contracts, scalability, and a good source of information and documentation. Since this system will support several loyalty programs and each customer can subscribe to several programs, high transaction volumes per second are essential to ensure good performance. Considering the expected number of transactions, the cost of each transaction must be minimal so that retailers are not penalized and the platform becomes more affordable.

Since the purpose of the system is to support and manage different types of loyalty schemas and loyalty programs, smart contracts are mandatory. Therefore, the chosen blockchain must support them. Due to the need for the system to handle platform growth, scalability is required. Therefore, the blockchain network must permit scaling. The blockchain system must also have good documentation to ensure the best use of blockchain resources and capabilities.

The choice of a set of blockchain network candidates took all the mentioned factors into account. The chosen networks were Avalanche, Ethereum, Hyperledger Fabric, Polygon, and Solana. These networks are currently the most popular and are distinct from each other. Table 4 summarizes their main characteristics for comparison. The information presented was obtained from the available documentation, except for the transaction price. Transaction prices for Polygon, Ethereum, and Avalanche have been determined utilizing the following specialized websites, which are designed for scanning transactions provided by the blockchains: PolygonScan [25], EthereumIo [26], and Avascan [27] for Avalanche. Solana's price is available on their official website [28]. It is important to note that these prices are subject to change, as they depend on several factors, such as the value of the cryptocurrency at any given time and the volume of transactions processed over the course of a day. Concerning Hyperledger, the transaction price is zero, as it is a private network.

Table 4. Blockchain network comparison.

	Avalanche	Ethereum	Hyperledger	Polygon	Solana
Environment execution	EVM [1]-AVM [2]	EVM [1]	Docker	EVM [1]	LLVM
Smart contract language	Solidity	Solidity	Java/GO	Solidity	C/Rust
Transaction per second	4500	15–27	3500	10,000	65,000
Permission type	Public/Private	Public	Private	Public	Public
Transaction price	<$0.01	≈$4	$0	<$0.01	$0.00025
Layer	Layer 1	Layer 1	Layer 1	Layer 2	Layer 1

[1] EVM—Ethereum virtual machine; [2] AVM—Avalanche virtual machine.

Despite Ethereum being the most popular among the five solutions, it presents a weakness regarding the number of supported transactions per second. Additionally, the cost for each transaction is too high, which can lead to platform rejection. Hyperledger Fabric (designated Hyperledger in Table 4) is a private network with a smart contract environment and language distinct from the usual Ethereum Virtual Machine (EVM) and Solidity language, which can cause implementation problems. Even though Solana supports a higher number of transactions per second and results in a lower cost, it suffers from the same problems as Hyperledger.

The Avalanche and Polygon networks are both compatible with smart contracts developed for the Ethereum environment, since they use EVM. The main differences between these two solutions are the level of popularity, the supported volume of transactions per second, and the type of layer. Polygon is more popular and allows more transactions per second. It is also a layer-two network, while Avalanche is a layer-one. This means Avalanche is a network that works independently, while Polygon works in parallel with a main network (i.e., a layer one network such as Ethereum and Avalanche). In the Polygon case, the main network is Ethereum, but in the future, it will also be compatible with other main networks.

In conclusion, the preferred blockchain network for the proposed loyalty system is Polygon, due to its increasing popularity and capability of providing a fair price and a good volume of transactions per second. In addition, Polygon is compatible with Ethereum, which allows the transfer of smart contracts to the Ethereum network at any time.

3.3. Polygon Blockchain

Polygon is a layer-two scaling solution designed to improve existing blockchains. Polygon's objective is to solve scalability issues while not compromising on decentralization. This is accomplished by using sidechains and leveraging the developer community and the existing ecosystem. Each blockchain network is made up of multiple networks. These networks consist of two distinct environments, one for testing and another for production. The production network is entitled the mainnet. Sidechains are "copies" of the mainnet that support asset transfers between them and the mainnet. They are an alternative to a mainnet network that create a new blockchain with its own mechanisms. Currently, Polygon is only compatible with Ethereum. Its key features are the following [25]:

- Scalability—fast, low-cost, and secure transactions;
- High throughput—it offers up to 10,000 transactions per second on a single sidechain. It is possible to add more sidechains to increase the horizontal scaling;
- Security—Polygon validators are also stakeholders in the proof-of-stake system, required to hold and lock a certain amount of cryptocurrency as a security deposit. This ensures that validators have a vested interest in maintaining network security and integrity. Validators who act maliciously or make errors that result in a loss of funds will be penalized by losing their security deposit. As a result, validators have a strong incentive to follow the rules and act in the best interests of the network;
- Public sidechains—sidechains are naturally public, permissionless, and capable of supporting multiple protocols.

3.4. System Architecture

After taking into consideration all factors mentioned previously, it is essential to define a detailed architecture with all the different software components, the interaction with external services, and the communication protocols. The architecture is composed of three major modules: (i) the mobile application the users will interact with; (ii) external services for facilitating user integration with blockchain and providing a set of required features, namely InterPlanetary File System (IPFS), Torus Web3Auth, Ramp Exchange, and Openzeppelin; and (iii) smart contracts that contain the system's business logic. Figure 3 represents the system architecture.

As the system allows for different loyalty mechanisms, to facilitate the entire loyalty program management process, each smart contract represents a specific loyalty mechanism. Since numerous retailers with a specific loyalty mechanism can exist, each smart contract will inherit the ERC-1155 interface [29] through the contract available through Openzeppelin. ERC-1155 is a token standard interface that facilitates the smart contract representing multiple tokens simultaneously, and includes management functions that efficiently help in reducing the fees. This interface, along with the already pre-built ERC-1155 contract, simplifies the whole management process and diminishes the effort needed to create smart contracts from scratch. Furthermore, since ERC-1155 allows multiple tokens to be managed within a single contract, the proposed loyalty management system can significantly reduce the number of smart contracts required, unlike current systems that require separate ERC-20 contracts for each token. Given the large number of tokens involved in our system, the use of ERC-1155 thus results in a more efficient and scalable solution that requires a minimum number of contracts: a single smart contract is required for delegating calls and storing data unrelated to loyalty programs, and a separate smart contract inheriting the ERC-1155 contract is required for each loyalty mechanism.

The proposed loyalty system includes an intelligent contract manager to simplify blockchain calls, which is responsible for receiving transaction requests and delegating their execution to a contract running a specific loyalty mechanic. The manager is also responsible for storing all generic data, such as which retailers are defined in the system, customer subscriptions, and smart contract addresses for loyalty mechanics. Through this approach, calls to interact and obtain information from blockchain are simplified. Furthermore, in order to prevent any unauthorized data manipulation, all data are directly associated with the user's unique wallet address. This approach guarantees that only the users themselves can modify their data, thus preventing any malicious user from tampering with other users' information.

In order to streamline the development of smart contracts for each mechanism and store them in the loyalty manager component of the proposed system, a loyalty mechanism interface was created (see Figure 3). This interface defines four methods that each mechanism contract must implement to properly define its behavior. The loyalty manager stores all the mechanism contracts with a map, upon which the keys are an "enum" and the values are references to the loyalty mechanism interface type that corresponds to contracts responsible for delegating the received calls to the corresponding contracts.

Torus Web3Auth is a pluggable authentication infrastructure for Web3 wallets and applications. It provides seamless onboarding by providing web 2.0 login flows, such as Facebook, Gmail, and Twitter. Despite the fact that the user is abstracted from wallet creation, he/she is always in control of ownership. Torus Web3Auth also manages the key infrastructure of the wallets (thus taking the responsibility away from the system) and can perform direct integration through a software development kit (SDK) for native Android, which provides a group of tools to enable the implementation of authentication. This authentication infrastructure allows for better user integration with blockchain, with all abstraction layers provided.

Despite the proposed loyalty management system not being intended to store personal information, retailers can publish images about their business and other business information. All these data are stored in the IPFS module in a JSON-like file, in which the file is also encrypted and assigned a hash in order to access it. The hash is a set of randomly generated characters that is created based on the content of the file. IPFS is a distributed system for storing and accessing files and a peer-to-peer hypermedia protocol. Since inserting information into the blockchain has an economic cost, non-relevant and sensitive information related to users and the system can be left off the chain. IPFS is appropriate for this scenario. The only way of accessing the data is through the unique hash code, which will be stored on the blockchain and linked to the retailer's information.

Ramp is a service that allows exchange between cryptocurrency and physical money. The need for this service comes from the retailer use case, as retailers need to pay transaction fees to interact with the blockchain. So, to simplify this activity, the platform directly supports the required money exchange option.

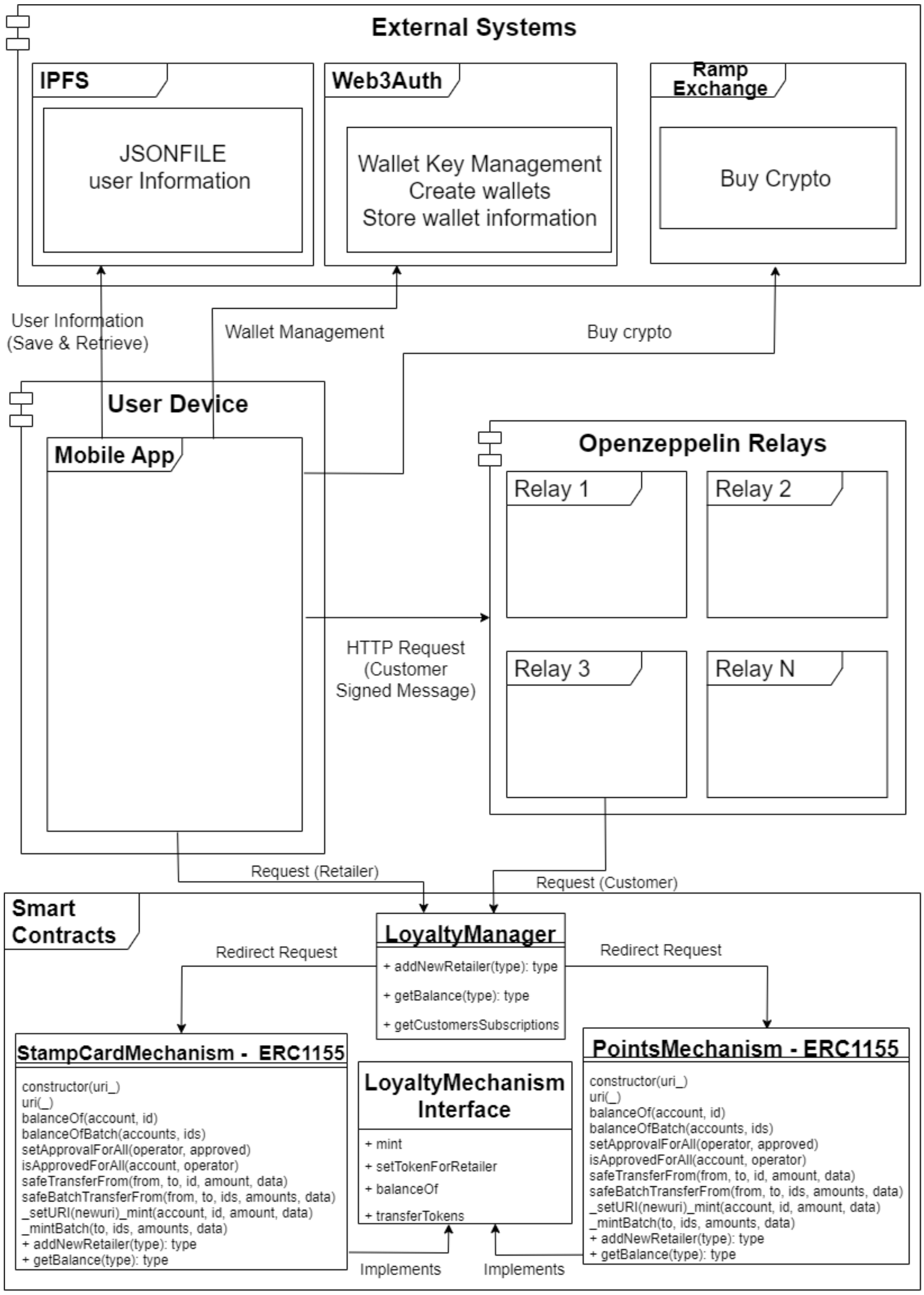

External Systems
IPFS
JSONFILE
user Information
Web3Auth
Wallet Key Management
Create wallets
Store wallet information
Ramp Exchange
Buy Crypto
User Information (Save & Retrieve)
Wallet Management
Buy crypto
User Device
Mobile App
Openzeppelin Relays
Relay 1
Relay 2
Relay 3
Relay N
HTTP Request (Customer Signed Message)
Request (Retailer)
Request (Customer)
Smart Contracts
LoyaltyManager
+ addNewRetailer(type): type
+ getBalance(type): type
+ getCustomersSubscriptions
Redirect Request
Redirect Request
StampCardMechanism - ERC1155
constructor(uri_)
uri(_)
balanceOf(account, id)
balanceOfBatch(accounts, ids)
setApprovalForAll(operator, approved)
isApprovedForAll(account, operator)
safeTransferFrom(from, to, id, amount, data)
safeBatchTransferFrom(from, to, ids, amounts, data)
_setURI(newuri)_mint(account, id, amount, data)
_mintBatch(to, ids, amounts, data)
+ addNewRetailer(type): type
+ getBalance(type): type
LoyaltyMechanism Interface
+ mint
+ setTokenForRetailer
+ balanceOf
+ transferTokens
PointsMechanism - ERC1155
constructor(uri_)
uri(_)
balanceOf(account, id)
balanceOfBatch(accounts, ids)
setApprovalForAll(operator, approved)
isApprovedForAll(account, operator)
safeTransferFrom(from, to, id, amount, data)
safeBatchTransferFrom(from, to, ids, amounts, data)
_setURI(newuri)_mint(account, id, amount, data)
_mintBatch(to, ids, amounts, data)
+ addNewRetailer(type): type
+ getBalance(type): type
Implements
Implements

Figure 3. Loyalty system architecture.

Meta transactions are essential to provide a more satisfactory onboarding process for any user, regardless of their degree of expertise. Openzeppelin relays will be employed to implement meta transactions, acting as intermediaries that will receive transactions from customers, thus preventing them from paying fees. These transactions will be sent via HTTPS and signed by the client that created them. In the current system architecture, all requests are routed to the loyalty manager or, in the case of customer requests, to the relay, which are then routed to the loyalty manager contract. However, as the number of customer requests may increase in the future, it may be necessary to assign multiple relays to the loyalty manager to prevent any relayer from being overwhelmed with requests and causing system blockages, and to ensure a smoother flow of operations.

Summarizing the aforementioned set of external services, Web3Auth is used for wallet/key infrastructure management, IPFS is used to store any data in off-chain mode, Ramp is used by retailers to purchase cryptocurrency to pay fees, and Openzeppelin relays are for meta transactions.

The mobile application is the interaction center in which users will execute their requests. In the case of retailers, the application communicates directly with the blockchain and external services. In the case of the customers, an HTTPS request is redirected to a smart contract, and is sent to a relayer whenever it is necessary to interact with the blockchain.

Figure 4 illustrates a temporal sequence diagram of a blockchain transaction in the proposed loyalty management system. Both types of users, i.e., customer and retailer, must first log in to interact with the system, as shown in the diagram. Then, if the user is a retailer, invoked transactions are directly sent to the blockchain, whereas if it is a customer, transactions go through a relay to avoid fees. After completing the login process, the retailer queries the blockchain to retrieve their data, such as the IPFS hash. Once the data are returned, a second request is sent to IPFS to retrieve the remaining information associated with the hash. When the transaction reaches the smart contracts, it firstly contacts the loyalty manager contract that is responsible for delegating the transaction to the right contract loyalty type.

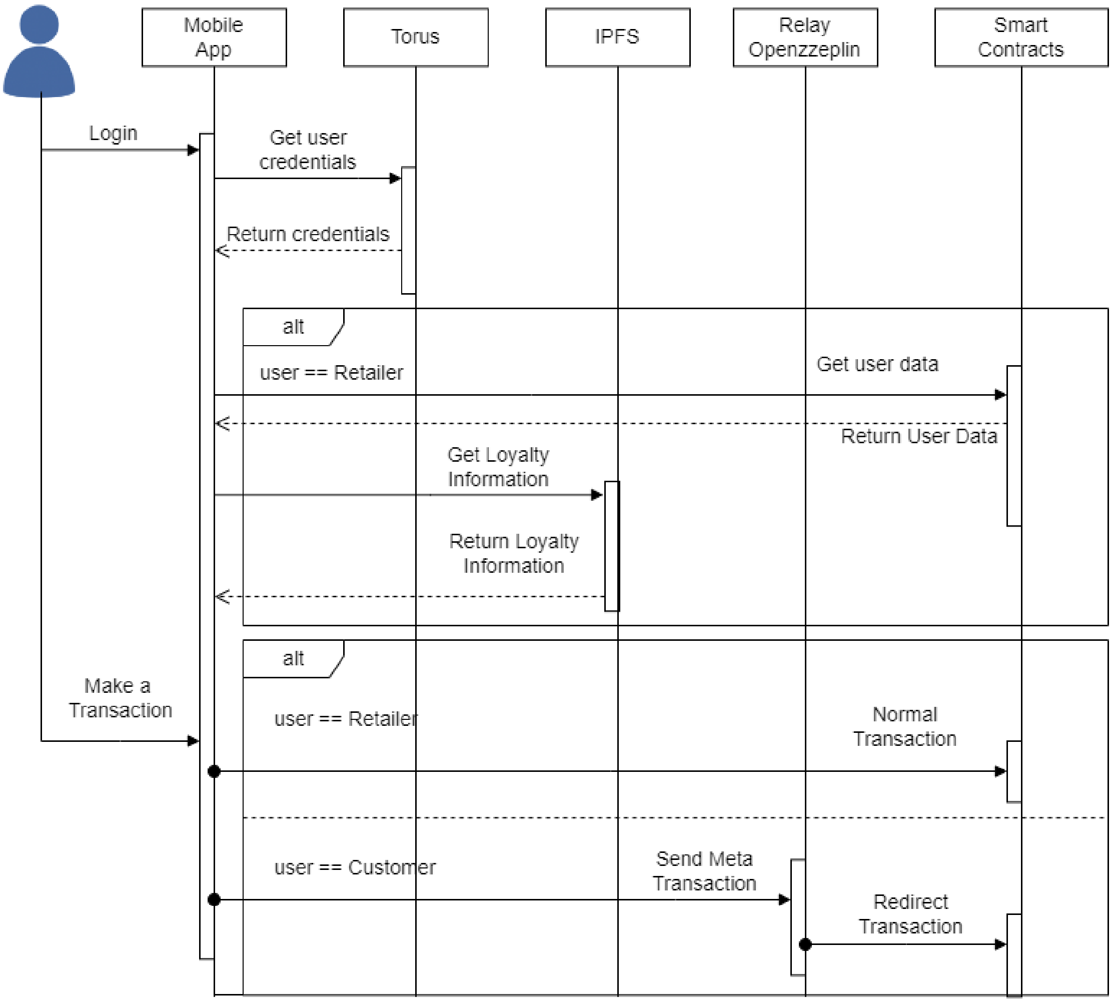
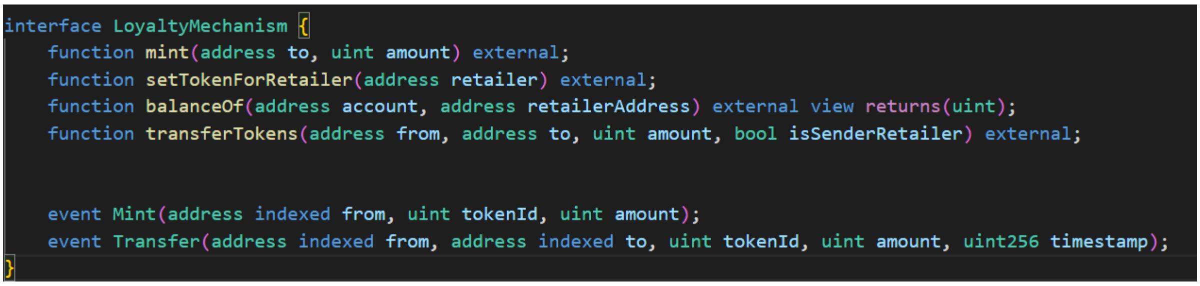


Figure 4. Sequence diagram.

3.5. Smart Contract Design

Smart contract design is a critical aspect of a fully decentralized system. As the sole source of project execution for data-related operations, smart contract design plays a pivotal role in determining the efficiency and cost-effectiveness of the system.

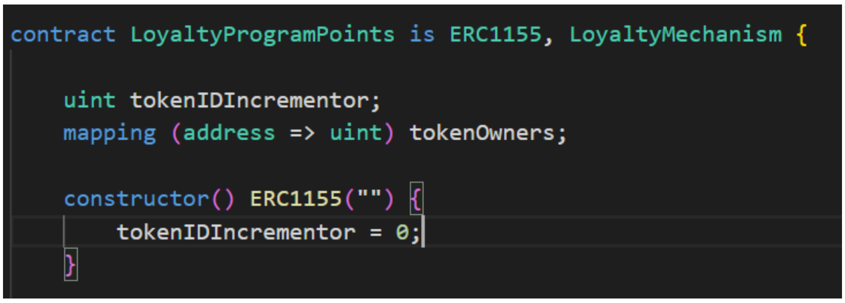
In order to streamline the implementation of multiple loyalty mechanisms, an interface has been created that must be implemented by each contract representing such mechanisms. Figure 5 shows the loyalty mechanism interface, which defines four core methods that form the basis of each contract, as they inherit the capabilities of the ERC-1155 contract. In addition, the interface defines two standard events that must be included in each mechanism contract to ensure consistency and standardization across the system.



```
interface LoyaltyMechanism {
    function mint(address to, uint amount) external;
    function setTokenForRetailer(address retailer) external;
    function balanceOf(address account, address retailerAddress) external view returns(uint);
    function transferTokens(address from, address to, uint amount, bool isSenderRetailer) external;


    event Mint(address indexed from, uint tokenId, uint amount);
    event Transfer(address indexed from, address indexed to, uint tokenId, uint amount, uint256 timestamp);
}
```

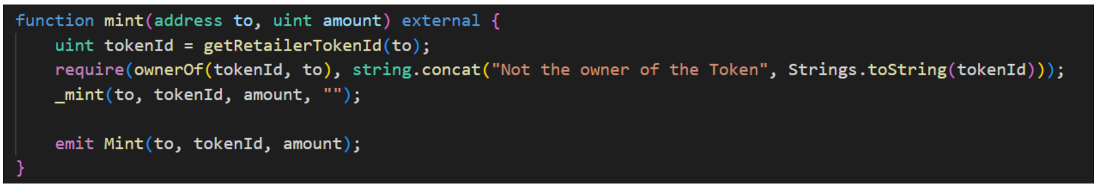
Figure 5. Loyalty mechanism interface.

Figure 6 highlights the points mechanism contract, which is responsible for managing the entire loyalty programs for this specific mechanism. The main functions of this contract include assigning a unique token ID to each merchant, allowing only the owner to mint tokens for their respective merchants, and performing transfers. Figure 7 shows the mint function, which is responsible for creating tokens for the retailer. To prevent unauthorized minting by any entity, only the token owner has the ability to generate tokens for their designated loyalty program. In addition, when a new retailer registers its loyalty points program, the smart contract assigns a token ID and stores it within a hash table (see Figure 8). This combined mechanism of token ID assignment and owner-restricted minting capability ensures secure and controlled token generation.



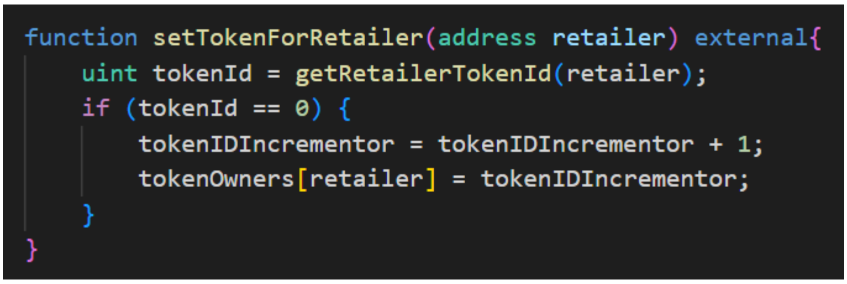
```
contract LoyaltyProgramPoints is ERC1155, LoyaltyMechanism {

    uint tokenIDIncrementor;
    mapping (address => uint) tokenOwners;

    constructor() ERC1155("") {
        tokenIDIncrementor = 0;
    }
```

Figure 6. Loyalty program points contract.



```
function mint(address to, uint amount) external {
    uint tokenId = getRetailerTokenId(to);
    require(ownerOf(tokenId, to), string.concat("Not the owner of the Token", Strings.toString(tokenId)));
    _mint(to, tokenId, amount, "");

    emit Mint(to, tokenId, amount);
}
```

Figure 7. Loyalty program points—function mint.

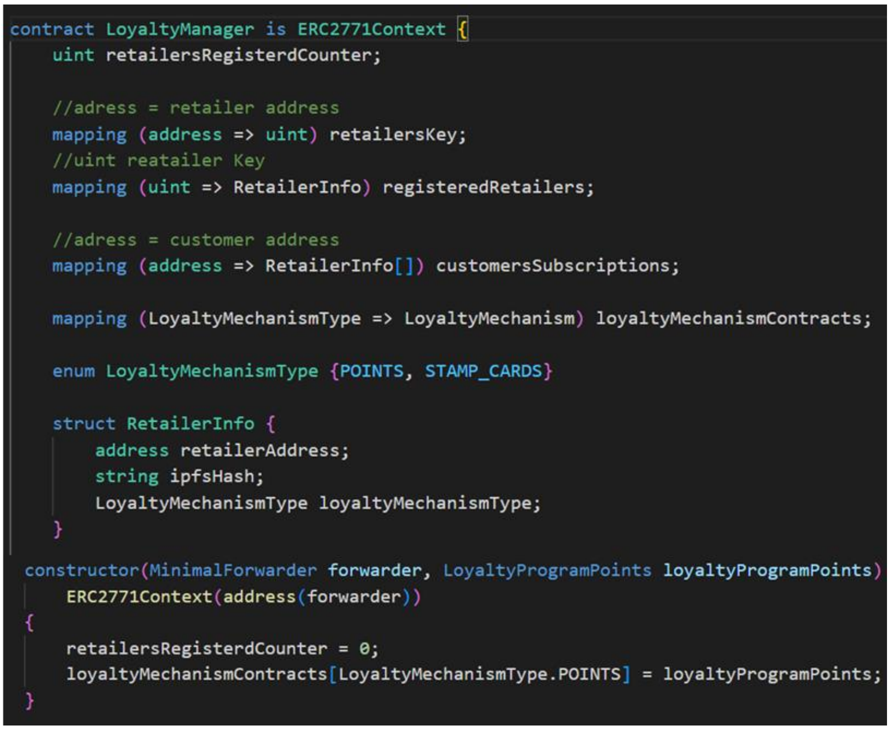


```
function setTokenForRetailer(address retailer) external{
    uint tokenId = getRetailerTokenId(retailer);
    if (tokenId == 0) {
        tokenIDIncrementor = tokenIDIncrementor + 1;
        tokenOwners[retailer] = tokenIDIncrementor;
    }
}
```

Figure 8. Loyalty program points—function setTokenForRetailer.

Finally, the loyalty manager contract, shown in Figure 9, acts as a central hub for all incoming requests in the loyalty system. It is also designed to support meta transactions, allowing customers to transact without incurring transaction fees. To fulfil this requirement, the contract inherits from the ERC2771Context contract and requires a forward contract as a parameter to validate whether a transaction should be executed or ignored. This forward contract and the ERC2771Context contract are standard contracts provided by OpenZeppelin and have not been implemented by us.

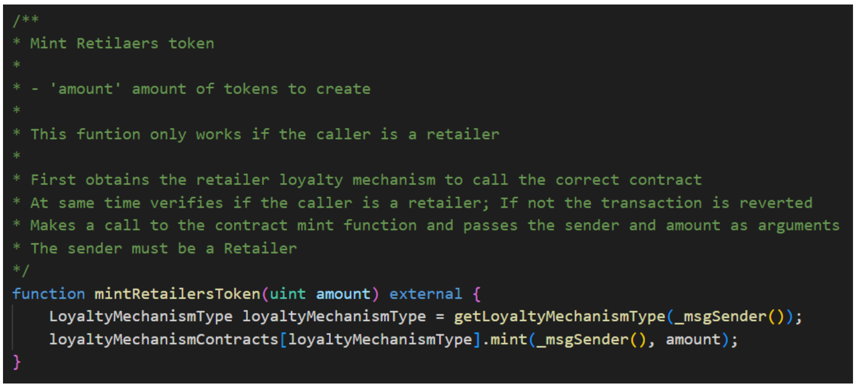
Similar to the approach presented in [9], where contracts store rules with references to interfaces, our Loyalty Manager contract follows a similar pattern. This allows for a more dynamic code implementation and the ability to add new contracts during runtime. To achieve this, the contract uses mapping, as shown in Figure 9, with an *enum* as the key, which is converted to integers at compilation time, and uses the contract references as the values. In the future, the *enum* can be removed, and contracts can be defined directly with dynamic incrementing integers.



```
contract LoyaltyManager is ERC2771Context {
    uint retailersRegisterdCounter;

    //adress = retailer address
    mapping (address => uint) retailersKey;
    //uint reatailer Key
    mapping (uint => RetailerInfo) registeredRetailers;

    //adress = customer address
    mapping (address => RetailerInfo[]) customersSubscriptions;

    mapping (LoyaltyMechanismType => LoyaltyMechanism) loyaltyMechanismContracts;

    enum LoyaltyMechanismType {POINTS, STAMP_CARDS}

    struct RetailerInfo {
        address retailerAddress;
        string ipfsHash;
        LoyaltyMechanismType loyaltyMechanismType;
    }
 constructor(MinimalForwarder forwarder, LoyaltyProgramPoints loyaltyProgramPoints)
     ERC2771Context(address(forwarder))
 {
     retailersRegisterdCounter = 0;
     loyaltyMechanismContracts[LoyaltyMechanismType.POINTS] = loyaltyProgramPoints;
 }
```

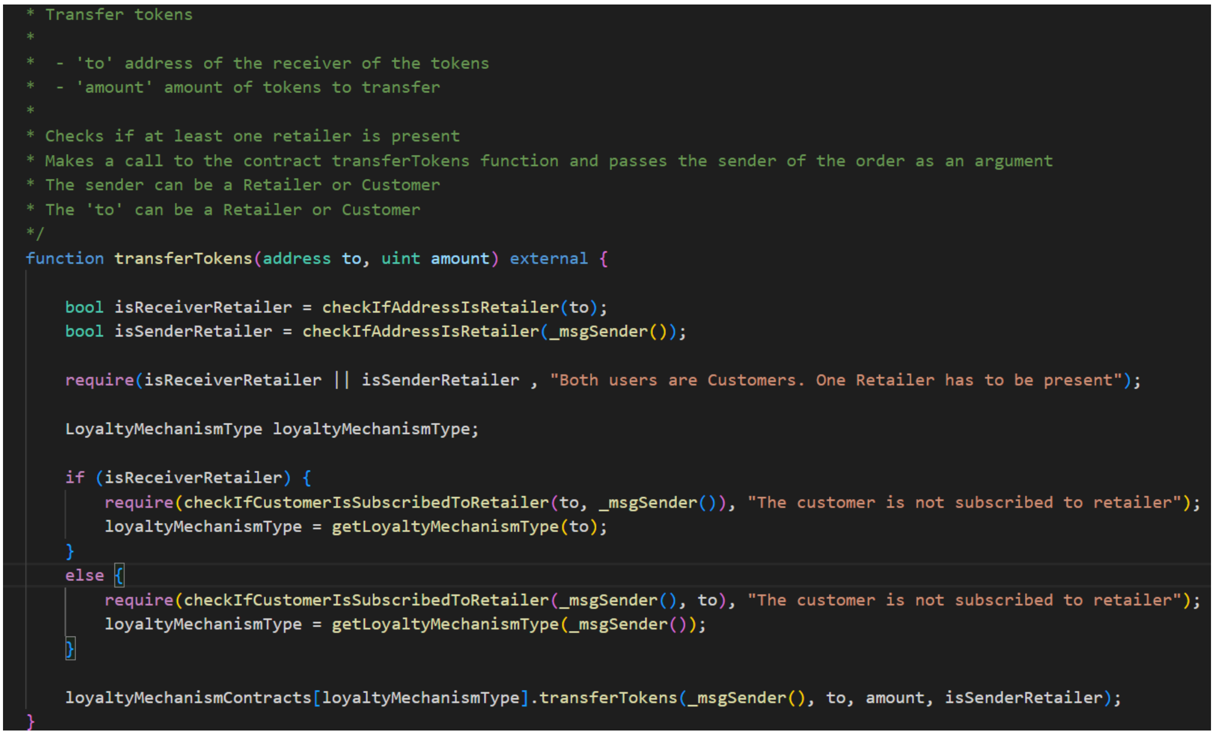
Figure 9. Loyalty manager contract.

By using references to the loyalty mechanism interface, the loyalty manager contract only recognizes the four defined methods, providing enhanced security and preventing access to other functions that may be defined by the contracts themselves. This approach also makes the code more dynamic, eliminating the need for nested 'if' statements to determine the correct contract to invoke. Figures 10 and 11 show how requests can be redirected to the appropriate contract in a straightforward manner.



```
/**
* Mint Retilaers token
*
* - 'amount' amount of tokens to create
*
* This funtion only works if the caller is a retailer
*
* First obtains the retailer loyalty mechanism to call the correct contract
* At same time verifies if the caller is a retailer; If not the transaction is reverted
* Makes a call to the contract mint function and passes the sender and amount as arguments
* The sender must be a Retailer
*/
function mintRetailersToken(uint amount) external {
    LoyaltyMechanismType loyaltyMechanismType = getLoyaltyMechanismType(_msgSender());
    loyaltyMechanismContracts[loyaltyMechanismType].mint(_msgSender(), amount);
}
```

Figure 10. Loyalty manager—function mint.



```
* Transfer tokens
*
*  - 'to' address of the receiver of the tokens
*  - 'amount' amount of tokens to transfer
*
* Checks if at least one retailer is present
* Makes a call to the contract transferTokens function and passes the sender of the order as an argument
* The sender can be a Retailer or Customer
* The 'to' can be a Retailer or Customer
*/
function transferTokens(address to, uint amount) external {

    bool isReceiverRetailer = checkIfAddressIsRetailer(to);
    bool isSenderRetailer = checkIfAddressIsRetailer(_msgSender());

    require(isReceiverRetailer || isSenderRetailer , "Both users are Customers. One Retailer has to be present");

    LoyaltyMechanismType loyaltyMechanismType;

    if (isReceiverRetailer) {
        require(checkIfCustomerIsSubscribedToRetailer(to, _msgSender()), "The customer is not subscribed to retailer");
        loyaltyMechanismType = getLoyaltyMechanismType(to);
    }
    else {
        require(checkIfCustomerIsSubscribedToRetailer(_msgSender(), to), "The customer is not subscribed to retailer");
        loyaltyMechanismType = getLoyaltyMechanismType(_msgSender());
    }

    loyaltyMechanismContracts[loyaltyMechanismType].transferTokens(_msgSender(), to, amount, isSenderRetailer);
}
```

Figure 11. Loyalty manager- function transferTokens.

As discussed in Section 2.1.4, Openzeppelin provides a plugin that allows the proxy pattern to be used without having to implement it manually. This plugin allows the use of the proxy pattern and the modification of the business logic of the contracts while preserving the data. All of the above contracts are publicly available in the GitHub repository [30].

4. Discussion

The proposed system aims to manage various loyalty programs with different mechanisms. Even though the blockchain solves significant problems, it is impossible to address concerns related to rewards and personal preferences due to inherent human characteristics.

The proposed loyalty management system offers limitless opportunities and can reshape how traditional loyalty systems are implemented. It provides new paradigms and possibilities to users with any level of knowledge about technology to create and interact with the blockchain without having the inconvenience of learning it. This way, users can easily participate and enjoy the loyalty programs of their favorite brands.

The system works on the Polygon chain, which implies the use of Solidity as the smart contract programming language. Polygon also includes a network for testing purposes (which is useful while developing the system) and offers ways to track transactions.

A mobile application is available to interact with the blockchain. This application abstracts its users by setting up all the requirements for them. Additionally, the way smart contracts are designed allows for a more manageable infrastructure of loyalty programs.

With the ERC-1155 token interface standard, each retailer can mint its own tokens and be responsible for their generation, thus gaining better control and ownership. Whenever a customer purchases something, the retailer mints the desired number of tokens and gives them to the client. In order for the customer to be able to purchase goods or services from a retailer, he/she needs to have a certain number of tokens from that retailer.

The designated external services are intended to produce a better user experience. Web3Auth is used to grant the user the ability to join the system with any credentials they have from Web 2.0, while managing the wallet and keys for user interaction. Ramp service is used to accommodate users when they need to buy cryptocurrencies. It offers an easy access point for currency exchange without leaving the application. Relays are used to diminish entry barriers for customers, as blockchain requires mastery, experience, and cryptocurrency to interact with. Since customers do not adhere to applications for which constant payment is required, the system acceptance would fail without the use of relays.

With the proposed loyalty management system, the users apply their previous Web 2.0 credentials and do not have to enter new information. Furthermore, the system does not include any database, which means that personal data will not be stored.

4.1. Comparison with Other Systems

There are several advantages that can be highlighted when comparing the proposed loyalty system with the related works discussed in Section 2. The universal blockchain system [5] supports multiple loyalty programs and partnerships, which our system can incorporate. However, this system limits the loyalty mechanism to points and uses a backend server for the users' application to communicate. Additionally, it stores user data in a database. In contrast, the proposed system eliminates the need to store personal information. Moreover, it does not rely on any backend system and achieves a more decentralized approach.

The system proposed in paper [13] has similarities with the universal blockchain system, since both employ the points mechanism using tokens. However, the proposed system incorporates additional types of loyalty mechanics and abstracts the users from the concept of tokens, enabling them to interact with points in a manner similar to non-blockchain-based platforms. Another distinguishing factor is that the author in [13] employs the ERC 20 interface for token creation, which requires that each token exists as a smart contract, whereas the proposed system uses ERC-1155 for managing multiple tokens. Additionally, the expectation that users possess knowledge of Ethereum wallets in [13] may present challenges for novice users.

The loyalty program that utilizes the Waves blockchain and a mobile application [19] employs a token creator that generates a unique token for each retailer. However, this system also stores personal information and potentially creates a token contract for each retailer, similar to the aforementioned system [13].

The reward system in [21] uses tokens that can be converted into points, and the platform is specifically designed for the FMCG industry, limiting its scope to this domain. However, this system resembles our transaction process, in that it employs QR codes to initiate the transfer of assets.

Although the systems proposed in [22,23] do not directly aim to manage loyalty programs, they improve these programs by facilitating the exchange of assets between different loyalty programs. Currently, our loyalty management system lacks features that allow for partnerships between retailers; we plan to implement these features in the future.

Since the existing loyalty systems discussed in Section 2 do not specify their authentication process, it is assumed that they have created one and currently manage it. This results in an additional step for user onboarding, as they have to enter new information to create a new account. In contrast, the proposed system employs Web3Auth, which uses Web 2.0 credentials to streamline the authentication process. Additionally, a significant advantage of the proposed system is the use of meta transactions, which provide a superior user experience by eliminating the need for users to pay blockchain fees.

4.2. System Benefits

The designed system architecture offers countless benefits for loyalty program ecosystems; it helps to reshape the traditional approach for a more modern, intuitive, and secure model.

The main benefits of the system are the following:

- The platform fully embraces decentralization and leverages the power of the ERC-1155 token to efficiently manage tokens and reduce the number of contracts required.
- The system provides near real-time transaction speed, allowing users to redeem their rewards faster without having to wait too long;
- Due to the behavior of smart contracts, third-party tools and intermediaries are no longer necessary;
- The system helps to reduce costs. As intermediaries and third-parties are abolished, retailers can save money. Furthermore, the infrastructure needed to manage the loyalty program is transferred to the blockchain;
- Blockchain offers a secure infrastructure;
- A single digital wallet can be used to store all retailers' tokens, eliminating the problem of managing rewards from multiple brands. It also gives complete control to the user;
- The platform accommodates all distinct levels of user knowledge, allowing more experienced users to use their wallet providers, and inexperienced people to have a smoother first contact with it;
- The platform is capable of managing several types of loyalty mechanisms, increasing the diversity of loyalty programs. In addition, the platform welcomes any retailer of any business size. This makes it easier for new retailers to join and customers with more options to choose from;
- It provides a marketplace with all available loyalty programs. On the one hand, the marketplace helps retailers gain greater exposure to attract more customers. On the other hand, customers have a better access point to all available loyalty programs, meaning they are able to discover new businesses;
- The system takes advantage of the fact that smartphones are increasingly present in people's lives, thus enhancing the overall experience.

4.3. System Limitations

The proposed system has great advantages. However, it also has some limitations:

- Due to human nature, it is almost impossible to solve concerns about rewards and personal preferences;
- Even though the system does not require any personal information, some information is required at the retailer's business level;
- The blockchain is a new subject, and this may affect outcomes. Because it is so new, it may be subject to impactful changes at any time, which can affect the system as it is currently designed, making it unsuitable for the current purpose;
- Exchanging of tokens between two customers is not allowed;
- Retailers cannot form any partnerships, which limits the possibility of trading retailers' assets;
- The transaction process requires the physical presence of the retailer and the customer. The application does not have an item or service available for purchase;

- Transactions can only be executed in person between a retailer and a customer;
- Remote transactions are not currently possible within this system.

5. Conclusions and Future Work

In this paper, a proposed loyalty management system that aims to manage different types of loyalty programs and mechanisms is described. This proposal includes several challenges in the technical domain, since the blockchain introduces new paradigms.

The system aims to overhaul the way in which traditional loyalty programs operate, and attempts to eliminate several entry barriers and infrastructure issues. It aims to replace conventional physical cards, instead adopting the strategy of using the smartphone, an increasingly ubiquitous object in everyday life. The platform also includes a universal system that aggregates loyalty programs, thereby simplifying the user experience.

The usability of this system allows users to have better interaction with the brands that they appreciate the most, alongside a better idea of and ownership over their rewards. Additionally, the system provides a marketplace with all available loyalty programs in a single space, removing the need to commit to having multiple applications and allowing management of all loyalty schemes at once. The system allows retailers to customize their own loyalty program without restrictions on the type of mechanism implemented. Furthermore, it exposes brands to new audiences, increasing the accessibility and reachability of the business.

The proposed loyalty management system reduces the problem of managing the loyalty program. It removes the responsibilities from the retailers' side, giving them more freedom to focus on satisfying customers and improving their loyalty program and business. It also removes concerns about the privacy of personal information, since the system does not require it. From the customers' perspective, it offers a new way to explore and know new businesses without the burden of having to download new apps and repeatedly introduce their data. It offers an opportunity to have a first encounter with the blockchain without worrying about learning and understanding the process. Overall, the platform prioritizes the user experience more.

Although the system solves significant problems, it has also some limitations. It cannot control human aspects, such as preferences for rewards. The blockchain is still extremely new and is constantly changing, which can impact the platform. The platform presents limitations about partnerships among retailers and the transaction process. The transaction process only considers trades between a retailer and a customer, and needs both to be present in the same room.

In general, the use of the proposed system is intuitive and simple. Its primary objective is to design a platform capable of managing different loyalty programs with distinct mechanisms. It presents a scalable solution, such that it is possible to add and expand its functionalities.

In terms of future work, we will be continuing with the implementation of the proposed system, particularly with regard to the remainder of the loyalty mechanism and improvements to the mobile app. It is also relevant to continue studying new blockchain networks and technologies, as they could potentially be better options. Furthermore, it is necessary to expand transaction possibilities in order to give users better control of assets. Regarding partnerships, it is crucial to review the architecture of smart contracts. It might be beneficial to make the application more interactive through the addition of a functionality that allows the purchase of products and services without the necessity of physically going to the store.

Author Contributions: Conceptualization, J.B.; Methodology, A.F.S. and J.B.; Software, A.F.S.; Validation, A.F.S., J.M. and J.B.; Formal analysis, A.F.S., J.M. and J.B.; Investigation, A.F.S.; Resources, A.F.S.; Data curation, A.F.S.; Writing—original draft preparation, A.F.S.; Writing—review and editing, A.F.S., J.M. and J.B.; Supervision, J.B. and J.M.; Project administration, J.B. and J.M.; Funding acquisition, J.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are openly available in GitHub, reference number [30].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krichen, M.; Ammi, M.; Mihoub, A.; Almutiq, M. Blockchain for Modern Applications: A Survey. *Sensors* **2022**, *22*, 5274. [CrossRef] [PubMed]
2. AWS—Amazon. What Is Decentralization in Blockchain? Available online: https://aws.amazon.com/blockchain/decentralization-in-blockchain/ (accessed on 30 October 2022).
3. IBM. What Are Smart Contracts on Blockchain? Available online: https://www.ibm.com/topics/smart-contracts (accessed on 30 October 2022).
4. Agrawal, D.; Jureczek, N.; Gopalakrishnan, G.; Guzman, M.N.; McDonald, M.; Kim, H. Loyalty Points on the Blockchain. *Bus. Manag. Stud.* **2018**, *4*, 80–92. [CrossRef]
5. Agrawal, M.; Amin, D.; Dalvi, H.; Gala, R. Blockchain-based Universal Loyalty Platform. In Proceedings of the 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, 20–21 December 2019; pp. 1–6. [CrossRef]
6. Hader, M.; Elmhamedi, A.; Abouabdellah, A. Blockchain technology in supply chain management and loyalty programs: Toward blockchain implementation in retail market. In Proceedings of the 2020 IEEE 13th International Colloquium of Logistics and Supply Chain Management (LOGISTIQUA), Fez, Morocco, 2–4 December 2020; pp. 1–6. [CrossRef]
7. Antoniadis, I.E.; Kontsas, S.; Spinthiropoulos, K. Blockchain and Brand Loyalty Programs: A Short Review of Applications and Challenges. *Procedia Econ. Bus. Adm.* **2019**, *5*, 8–16. [CrossRef]
8. GeeksforGeeks. Features of Blockchain. Available online: https://www.geeksforgeeks.org/features-of-blockchain/ (accessed on 7 December 2022).
9. Górski, T. Reconfigurable Smart Contracts for Renewable Energy Exchange with Re-Use of Verification Rules. *Appl. Sci.* **2022**, *12*, 5339. [CrossRef]
10. Kim, K.; Ryu, J.; Lee, H.; Lee, Y.; Won, D. Distributed and Federated Authentication Schemes Based on Updatable Smart Contracts. *Electronics* **2023**, *12*, 1217. [CrossRef]
11. OpenZeppelin. Proxy Upgrade Pattern. OpenZeppelin Docs. 2023. Available online: https://docs.openzeppelin.com/upgrades-plugins/1.x/proxies#upgrading-via-the-proxy-pattern (accessed on 18 April 2023).
12. Polygon. Meta Transactions. 2022. Available online: https://wiki.polygon.technology/docs/develop/meta-transactions/meta-transactions/ (accessed on 6 January 2023).
13. Sönmeztürk, O.; Ayav, T.; Erten, Y.M. Loyalty Program using Blockchain. In Proceedings of the 2020 IEEE International Conference on Blockchain (Blockchain), Rhodes, Greece, 2–6 November 2020; pp. 509–516. [CrossRef]
14. Ethereum Network. 2015. Available online: https://ethereum.org/ (accessed on 15 January 2023).
15. Vogelsteller, F.; Buterin, V. ERC-20 Token Standard. 19 November 2015. Available online: https://eips.ethereum.org/EIPS/eip-20 (accessed on 15 January 2023).
16. Wang, L.; Luo, X.; Hua, Y.; Wang, J. Exploring how blockchain impacts loyalty program participation behaviors: An exploratory case study. In Proceedings of the 52nd Hawaii International Conference on System Sciences, Maui, HI, USA, 8–9 January 2019; pp. 1–10. [CrossRef]
17. Wang, L.; Luo, X.; Lee, F. Unveiling the interplay between blockchain and loyalty program participation: A qualitative approach based on Bubichain. *Int. J. Inf. Manag.* **2019**, *49*, 397–410. [CrossRef]
18. Ryan, R.M.; Deci, E.L. Overview of self-determination theory: An organismic dialectical perspective. In *Handbook of Self-Determination Research*; University of Rochester Press: Rochester, NY, USA, 2002; Volume 2, pp. 3–33.
19. Perez, L.J.D.; Ibarra, L.; Alejandro, G.-F.; Rumayor, A.; Lara-Alvarez, C. A loyalty program based on Waves blockchain and mobile phone interactions. *Knowl. Eng. Rev.* **2020**, *35*, e12. [CrossRef]
20. Udegbe, S. Impact of blockchain technology in enhancing customer loyalty programs in airline business. *Int. J. Innov. Res. Adv. Stud.* **2017**, *4*, 257–263.
21. Bülbül, Ş.; İnce, G. Blockchain-based Framework for Customer Loyalty Program. In Proceedings of the 2018 3rd International Conference on Computer Science and Engineering (UBMK), Sarajevo, Bosnia and Herzegovina, 20–23 September 2018; pp. 342–346. [CrossRef]
22. Nguyen, C.T.; Hoang, D.T.; Nguyen, D.N.; Pham, H.-A.; Tuong, N.H.; Dutkiewicz, E. Blockchain-based Secure Platform for Coalition Loyalty Program Management. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021; pp. 1–6. [CrossRef]
23. Tu, S.-F.; Hsu, C.-S.; Wu, Y.-T. A Loyalty System Incorporated with Blockchain and Call Auction. *J. Theor. Appl. Electron. Commer. Res.* **2022**, *17*, 56. [CrossRef]
24. Utz, M.; Johanning, S.; Roth, T.; Bruckner, T.; Strüker, J. From ambivalence to trust: Using blockchain in customer loyalty programs. *Int. J. Inf. Manag.* **2023**, *68*, 102496. [CrossRef]

25. PolygonScan. PolygonScan. 2023. Available online: https://polygonscan.com/ (accessed on 8 April 2023).
26. EthereumIo. EthereumIo. 2023. Available online: https://etherscan.io/ (accessed on 8 April 2023).
27. Avascan. AvalancheScan. 2023. Available online: https://avascan.info/ (accessed on 8 April 2023).
28. Solana. Solana Website. 2023. Available online: https://solana.com/ (accessed on 8 April 2023).
29. Radomski, W.; Cooke, A.; Castonguay, P.; Therien, J.; Binet, E.; Sandford, R. ERC-1155: Multi Token Standard. Ethereum Improvement Proposals, 17 June 2018. Available online: https://eips.ethereum.org/EIPS/eip-1155 (accessed on 18 April 2023).
30. Smart Contracts Repository. Available online: https://github.com/AndreFSantos00/loyalty-smart-contracts (accessed on 21 April 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.