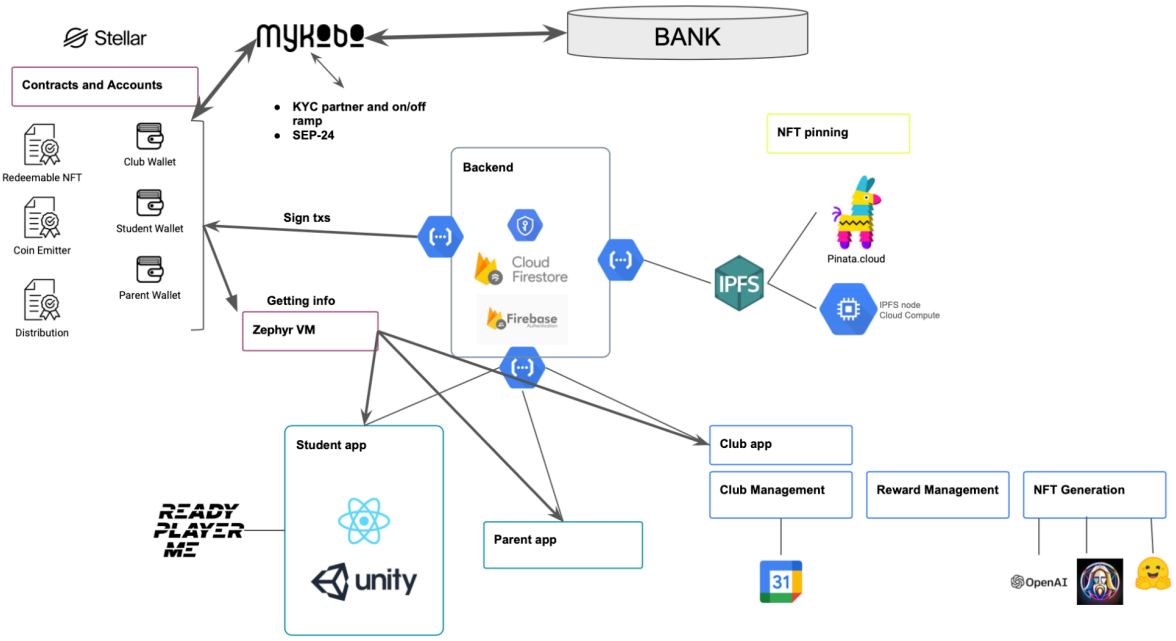
Gladius Technical Architecture - SCF#27



Frontend application

Club Application

The club application is divided into 3 core features sets, NFT generation, reward management , and club management. These features enable clubs to create and distribute Soroban based rewards in an easy to use interface. The frontend is mainly built in react leveraging Firebase for authentication, storage and generic computing.

- **NFT Generation** :- The NFT Generation module of the Gladius platform allows sports clubs to upload images or input thematic prompts.. This module integrates APIs such as OpenAI, Leonardo.ai, and Hugging Face models for dynamic image generation. Images, once generated, are stored in Firebase storage and made accessible by the reward management features.
- **Reward Management** :- In the Reward Management section of the Gladius platform, clubs can define specific rules for distributing rewards. For example, automated triggers can be set up whereby achieving certain milestones, such as a student collecting enough points, results in the automatic minting and allocation of a new NFT to their account. Similarly, new members can automatically receive a club badge NFT upon joining. These rules are save in Firebase and are executed by cloud functions which interact with the Soroban contracts
- **Club Management** :- In the Club Management section of the Gladius platform, the system architecture remains largely consistent with that of SCF#23. A key feature is course creation in the distribution contract, which handles the distribution of Gladius

coins from a common pool within the club. Integration of this contract is handled by cloud functions. In addition, Google Calendar is integrated to facilitate event organization and scheduling. This allows coaches and club owners to efficiently manage club activities such as recording attendance, distributing Gladius coins, and sending out invitations via cloud functions. Converting Gladius coins into EURC can also be managed here via an off ramp integration with mykobo (details below)

Parent Application
The parent application is built and hosted the sameway as the club application. The main feature of the parent app is to subscribe students to a club course using cloud function integrations with the Sorban contract. A key step in this process is KYC and payment handling via mykobo partner.

Student
The Student Application of the Gladius platform is specifically designed to allow students to view and interact with their NFT rewards in a visually engaging manner. The app uses Unity along with WebGL to render open standard 3D models that have been minted as NFTs by the club. Integration with Ready Player Me enables students to create and customize their own 3D avatars. These avatars serve as personal digital identities within the platform, allowing the students to interact with and showcase their NFTs in a virtual environment that mirrors real-world interactions and ownership.

Zephyr vm
To efficiently manage the dynamic nature of NFT metadata in the Gladius platform, particularly as students' avatars level up and acquire new items, we employ a blockchain indexer. This indexer is specifically designed to facilitate fast and efficient retrieval of updated data from the blockchain, where all NFT transactions and metadata changes are immutably recorded.

Hosting and Deployment
All apps are currently React applications hosted on Firebase.

IPFS, Pinata, and IPFS node

To enhance the robustness and efficiency of the NFT storage solution within the Gladius platform, we plan to establish our own dedicated IPFS node. This node will serve as the primary host for NFT assets, ensuring decentralized storage and swift access to data. By hosting our own IPFS node, we gain greater control over data availability and network performance.
In addition to our IPFS node, we will integrate with Piñata as a secondary storage solution to pin NFT assets. Piñata will serve as a backup to our primary IPFS storage, providing an additional layer of reliability and redundancy.
All NFT assets, once created by clubs, will be pinned on our IPFS node and backed up to Piñata.This setup enables "lazy minting" of NFTs, where assets are only minted when they are sent to students.

For high-performance needs, such as rapid loading of 3D assets in the Student Application, a copy of these assets will also be stored in Google Cloud Platform (GCP). This dual-storage approach ensures that large files are accessible with minimal latency, enhancing the user experience, particularly when interacting with complex 3D visuals in the Unity-based Student Application.

KYC, On/Off Ramps, Stellar anchors

KYC

KYC will be handled through a rest api through a partner mykobo following SEP12. Specially for parent and club wallets.

Stellar Anchors

Parents will be able to pay from their bank directly on the app using the mykobo Stellar Anchors. Payment is converted to EURC stable coins with a percentage also being converted to Gladius coins for student rewards.
Students are able to spend Glaidus coins and Clubs are allowed to convert spent gladius coins back into EURC stable coins. Clubs can then utilize the mykobo off ramp to transfer fiat currency to their bank

SCF#23 Below

Integration with Web3, Stellar and Soroban

Wallets

Each student and club is automatically assigned their own wallet when they sign up for Gladius. These wallets will contain any gladius coin or minted NFTs owned by that user, but won't contain any native chain currency. Additionally a Gladius wallet owned by Gladius is used to help facilitate transactions by providing the native currency for transactions via the TxnCoordinator contract.

Soroban Smart Contracts:
We think that we'll need 4 smart contracts. However, the detailed mechanism will be defined after Deliverable 1: Tokenomics and Smart Contracts specifications. The Smart Contracts that we think will need to be developed are:

1. **Gladius Coin Token Contract:**
   A standard token contract for keeping record of which clubs and students own Gladius coins. Premium clubs will have the option of creating their own branded coin, which will be provided using the gladius token coin contract as a template.

   This may not be equal to the token contract available in the Stellar Github, because depending on the Tokenomics model we might block Clubs to spend these coins so they will be forced to send them to students. Also, students will be forced to either hold their points, or spend them in the Club physical store or NFT's.

2. **Token Distribution Smart Contract**
   This contract allows Clubs to send the Gladius Coins to Students depending on their progress. In order to the Token to have value, this contract will also serve as a Liquidity Pool, where part of the fees payed by Parents will be in this pool.

3. **NFT smart contract**
   NFT's can only be minted by spending a Gladius coin or a club's own coin.

4. **Token Distribution & Coin Factory Contract**
   The Factory Contract facilitates Clubs in generating personalized Token Distribution Smart Contracts, enabling them to issue branded coins and NFTs to students, receiving premium payments and deploying both a Premium Branded Coin and Token Distribution Smart Contract. This contract will deploy a branded Token Distribution Contract and a branded Coin for each Premium Club

IPFS and Pinata

Soroban and implement Pinata for IPFS descentralize image storage. Implementation of a lazy minting technique in Soroban. In the proof of concept stage NFT data is stored on Pinata and is made accessible via a public IPFS gateway. Additionally we will host a small IPFS gateway and node using a Google cloud platform EC2 instance to account for any downtime with Pinata.

Backend Infrastructure

Firebase

Firebase is used to handle authentication, general user and club account data, and managing serverless cloud functions. User and club data is stored in a No-SQL database which is optimized to enable fast reads from the frontend.

The data within the Gladius platform is organized into three main categories:

- Users: Information about individual users, including athletes and parents.
- Clubs: Data pertaining to sports clubs, their management, and memberships.
- Events: Details of sports events, schedules, and participation.

These data sets are managed within Cloud Firestore, ensuring real-time synchronization and secure access control.

Data synchronization between collections and documents is handled by triggered cloud functions. Cloud functions are also used as the main intermediary between the frontend and the backend with most api calls going via cloud functions. They are also used to invoke public soroban RPC's to perform actions on the blockchain.

Google Cloud Services

In the proof of concept Gladius uses Key Management Service for secure handling of wallet private keys belonging to the clubs and users. BigQuery is used to store event data associated with various Gladius smart contracts, transactions and web2 interaction on the frontend.