

# Project Protocol MTGC Fink

## 1. Project Overview

The MTGC (Magic: The Gathering Card) Server application is designed to handle user registration, login, and game functionalities, including managing user cards and tokens. It utilizes a client-server architecture where the server listens for incoming requests and processes them accordingly (HTTP Request, JSON).

## 2. Design Decisions

I decided to choose a layered architecture style where I can separate different parts of the server (Controller, Services, Repository, Models). I did that to have a better overview of the project and to make working easier.

At the moment the project works with HashMap's as storage system. This will be replaced with a database.

Unfortunately, I haven't reached all my goals until this deadline, but I plan to catch up until the next hand in.

## 3. Structure

The project's architecture is organized into several packages, each with a specific responsibility (based on my layered architecture):

- **org.example:** Contains the main server class and server management, as well as the TokenManager.
- **org.example.Controller:** Contains controllers responsible for handling user requests.
- **org.example.Service:** Contains business logic for user registration and login.
- **org.example.Model:** Contains data models for User, Card, MonsterCard, and SpellCard.
- **org.example.Repository:** (Not shown in the provided code) Expected to manage data persistence for User entities.

Later the rest of the classes (Trading, Deck etc.) will be added and I also want to rename the package to fit the project.

## 4. GITHUB Link

[https://github.com/SilverPheonix/MTCG\\_Fink.git](https://github.com/SilverPheonix/MTCG_Fink.git)