A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



# HOBBES - Pokémon trading application

## Functionality specification

## Version: 1.0.0

## 29.03.2025.

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



# Version history

| Version | Date | Author | Comment |
|---|---|---|---|
| **1.0.0** | 29.03.2025. | Kai Šabijan | |
| | | | |
| | | | |

# Document certification

| Name | Role | Company | Date | Signature |
|---|---|---|---|---|
| **Noa Tisaj** | Project Manager | Pokempany | 29.03.2025 | Noa |
| | | | | |

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



# Content

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



# Introduction

As part of our academic project, our team of four is developing a Pokémon trading application that makes exchanging Pokémon easier, safer, and more enjoyable. Trading has always been an important part of the Pokémon experience, helping players complete their collections and connect with others. However, finding the right trade partner and making fair exchanges can sometimes be complicated. Our app is designed to solve these problems by providing a clear and user-friendly way to manage trades, search for specific Pokémon, and interact with other players.

This document lays out the functionality specification for our application. The goal is to clearly define what the app will do, how users will interact with it, and what features need to be implemented. Having this kind of document is important because it helps keep everyone on the same page during development. It prevents misunderstandings, makes planning and coding more efficient, and serves as a reference when testing or improving the app later on.

Beyond the technical side, this project is also about creating a fun and engaging experience for Pokémon fans. Our goal is to design an app that feels intuitive and enjoyable while making trading simpler and more accessible for everyone

# Scope of the project

The HOBBES Pokémon Trading Web Application is designed to provide a structured and interactive platform for users to trade Pokémon. The project will include a secure authentication system, a mechanism for selecting and managing Pokémon, and a trading system that allows users to exchange Pokémon seamlessly. Additionally, a reward system will incentivize user engagement based on trading activity. The application will be built using a .NET backend and a React frontend, with PokeAPI as the primary source for Pokémon data. While the core focus is on trading, additional features such as Pokémon evolution may be implemented if time permits. The project aims to ensure a smooth user experience by leveraging modern web technologies and maintaining clear separation between frontend and backend development.

# Concepts

In the development of the HOBBES Pokémon Trading Web Application, several fundamental principles shape the project's execution and ensure its successful completion. These include:

Development Stages – The project follows a structured process, from initial planning and design to implementation, testing, and deployment.

Feature Scope – The primary focus is on Pokémon trading functionalities, authentication, inventory management, and reward systems. Additional features will be evaluated for feasibility before inclusion.

Project Timeline – The schedule includes defined milestones for feature implementation, testing phases, and the final launch to ensure timely delivery.

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



Risk Mitigation – Potential challenges, such as API downtimes, security vulnerabilities, or low user engagement, are identified early, with proactive solutions in place.

# Role description

In our application, every user plays the same role: the registered user. There are no admin or moderator roles—everyone has equal access to all the core features of the platform. To start using the application, users must create an account and log in. This ensures that each user has a personal inventory, can track their trades, and receive rewards.

Once logged in, users can manage their Pokémon collection, search for available trades, and propose their own offers. They will also receive daily Pokémon rewards and can participate in a reward system that grants additional Pokémon after completing a certain number of trades. The application will ensure that all users have a fair trading experience by implementing safeguards against one-sided exchanges.

By centralizing all these features under a single user role, the platform remains simple and accessible. There are no complicated permission levels—everyone has the same ability to trade, collect, and engage with the system. This approach ensures that every user gets the full experience while keeping the development process straightforward.

# Fences

-Scope Fence: The application won't deal with Battles or catching Pokemon, it will completely focus on trading

-Budget Fence: During development we will only use free or open source tools such as Pokeapi and Figma's free design option

-Time Fence: The project must be completed inside of the first term, seeing as one of our team members is only staying for this term, if needed 3 members can continue but the last possible deadline is early 2026

-Quality Fence: The application must be fast, responsive and with minimal down time, it should be scaleable and easily maintainable

# Assumptions and dependencies

The development and functionality of the HOBBES Pokémon Trading Web Application rely on several key assumptions and dependencies. It is assumed that users will have stable internet access to interact with the platform, as real-time trading and data retrieval require continuous connectivity. The project depends on PokeAPI as the primary source for Pokémon data, including names, types, and sprites. Any changes or downtime in this external API could impact the application's functionality.

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



Additionally, the system assumes that users will engage in fair trading practices, as no direct enforcement mechanism will initially be in place to prevent unfair trades. Authentication and security measures will depend on established frameworks within the .NET ecosystem, ensuring secure user registration and login processes. The frontend relies on React, and its proper integration with the backend is critical for seamless functionality.

Furthermore, the project assumes that hosting and storage solutions will support the necessary data handling, particularly if Pokémon sprites need to be self-hosted. Any limitations in these resources could require adjustments in the implementation. The successful deployment of the application also depends on the team's ability to collaborate effectively, following best practices for development, testing, and documentation.

# List of requirements

## Requirements related to functionality

### User Authentication

A User will need to log in using their password and username to access the functionalities of the trading platform/website. If a User doesn't log in, they won't be able to receive their starting 6 Pokemon and subsequently, they won't have access to the randomly generated daily rewards in the form of new Pokemon. Furthermore, they aren't going to be able to post or interact with trades. Passwords have to be hashed and emails must be unique.

### Choosing the starter Pokemon

Once a User has logged in, they are presented with the option of choosing between 6 starter Pokemon that they can then use to trade. These Pokemon will be of common rarity to facilitate and ecourage progression and long term interest as well as User retention. This happens only once per account.

### Daily Pokemon Generation

After the initial 6 starting Pokemon that the User chooses between and receives upon login, they will still have the ability to receive new Pokemon outside of trades. One of these sources will be random daily Pokemon generation. These Pokemon will be randomly generated and their rarity will be dependent on a percentage system. The higher the rarity, the lower the percentage and chance to get them. This feature will have a timer of 24 hours behind it, meaning that the User will be able to receive only one new Pokemon within a 24 hour period.

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



## Pokemon Filtering

To make Pokemon trading and storafe more intuitive and simpler to manage, a Pokemon filtering system will be implemented. The filters will include Pokemon type and name.

## Trading System

The main allure of this platform will be the ability to trade Pokemon. Users will be able to trade any of their received Pokemon by interacting with trade offers that other Users posted on the platform and will be able to post trade offers of their own. There is not going to be a daily limit for trading and no limitations will be imposed on the rarity of the Pokemon within the trade offers. Trading will be possible only between two Users and will not include the possibility of live trading, only interacting with already existing offers.

## Reward Program

As previously mentioned, Users will have multiple avenues of Pokemon acquisition on the platform. A reward program tied to the trades will be implemented. After a User has successfully completed 5 trades, they will receive an egg from which they will be able to acquire a Pokemon. In addition, after 25 completed trades, the User will have the ability to manually choose a Pokemon. All Pokemon recevied will be immediately added to the Users' inventory.

## Pokemon Storage

Every User will have access to their own personal inventory system. Any and all Pokemon received through any means on the platform will be immediately added to the personal inventory. Through his inventory system, Users will be able to manage their Pokemon with the help of the aforemenetioned filtering system.

## Pokemon evolution (If time permits)

Once a User has received a Pokemon, they will have the ability to evolve it to their next evolution stage. For evolution to occur, a certain amount of time will need to have passed. There will be 3 separate stages of evolution and each one will require more time than the previous.

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr



# Requirements Related to Characteristics

To ensure our Pokémon trading platform is enjoyable, efficient, and sustainable, we define the following non-functional requirements:

## Usability

- The user interface must be intuitive and allow users to easily navigate their inventory, search for specific Pokémon, and initiate trades.
- The application should be fully responsive, providing a consistent experience on both desktop and mobile devices.
- Filtering options must be available for Pokémon type, name, and evolution stage.
- The trading process must be clear and guided, including at least two confirmation steps to prevent accidental trades.

## Performance

- The application must support up to 500 concurrent users without noticeable slowdowns.
- Search operations (e.g., finding a Pokémon) must return results within 150–200ms.
- Trade interactions (posting, accepting) should be executed and confirmed in less than 300ms.
- Users with up to 500 Pokémon in their inventory should experience no lag or excessive loading times.
- The system should be able to process at least 10 trades per second at peak usage.

## Security

- All users must log in with secure passwords (minimum 8 characters, including at least one number and one special symbol).
- Passwords must be securely hashed and never stored in plain text.
- Account details and trade histories must be protected using strong encryption.
- The platform must include verification steps during trades to prevent accidental or unfair exchanges.
- Real-time monitoring of trading activity, with suspicious trades flagged for review within 30 seconds.

## Scalability

- The platform must remain stable and responsive as the number of users grows.
- The backend infrastructure must support at least 10,000 active users per month and handle over 50,000 trades per day without degradation.
- The architecture should allow for future updates (e.g., adding new Pokémon features) to be deployed within 48 hours, without disrupting existing functionalities.

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr