Gacha Game: Duck Spinning

Project Report

Advanced Software Engineering 2024/2025
University of Pisa

Students:

Alexandra Pavlova, Othman Alhammali Shoaib Alhadiri,
Ermias Mulugeta Teklehaimanot

November 21, 2024

Overview

This project is a simple Gacha game where users can spin for ducks of varying rarities. The game implements user registration, login, profile management, and a feature that allows users to spin for a random duck. The ducks have different rarities, and once acquired, they are stored in the user's profile.

Key Features

1. User Authentication

- **Registration**: Users can register by providing a username and password. Passwords are securely hashed before storing in the database.
- **Login**: After successful registration, users can log in using their credentials.
- **Logout**: Users can log out and return to the login screen.
- **Profile**: After logging in, users are directed to their profile, where they can see all the ducks they've received.

2. Duck Spinning System

- Users can click the "Spin for a Duck" button to get a randomly selected duck.
- Ducks are assigned a rarity, which influences the chance of receiving them during a spin. The rarities are: Common, Rare, Super Rare, Ultra Rare, and Super Ultra Rare.
- The result of the spin is a duck object (with attributes like name, rarity, and image) which is then saved in the user's profile.

3. Duck Information

- Each duck has a name, rarity, and profession. An image of the duck is also displayed on the profile page once received.

4. Database Structure

- **Duck Model**: Stores information about the ducks, including their rarity and image.
- **UserDuck Model**: Links users to the ducks they've received and records the time they acquired them.

Technology Used

- **Django**: For web development, with Django's built-in user authentication system for managing user accounts.
- **SQLite**: For the database.
- **Django Templates**: For rendering HTML pages like login, registration, profile, and spin result.
- **HTML/CSS**: For front-end layout and design.
- **Random Selection Algorithm**: Weighted random selection based on the rarity of ducks.
- **Docker**: For containerizing the application to ensure consistent deployment and ease of setup.

Current Status

- User registration, login, and logout functionality are working as expected.
- The spinning feature is implemented and working, where a duck is randomly selected based on its rarity.
- Ducks acquired through spinning are stored in the user's profile.
- All ducks are listed on the profile page with details such as their name, rarity, and image.

Next Steps

- **Testing**: Implement unit and integration tests to ensure the application functions as expected.
- **Security**: Add methods to ensure the security of user data, including protection against common web vulnerabilities.