

Semesterprojekt SWEN 1: MTCG

Projektprotokoll

Daniel Rohrweckh

Decisions

Für den http-server habe ich drei Klassen erstellt. Die Klasse httpserver startet den Server. Die Klasse httprequest behandelt die http-Requests. Die Klasse httpresponse betätigt die http-Responses. Das Login und die Registrierung von User wird durch die Klasse userendpoint geregelt. Beim Login wird überprüft, ob ein User schon in der Datenbank existiert und ob das eingegebene Passwort, mit dem in der Datenbank übereinstimmt. Bei der Registrierung der Datenbank das Passwort des Users gehasht zu übergeben, da Passwörter nie im Klartext in einer Datenbank gespeichert werden sollen. Außerdem wird geprüft, ob ein User mit demselben Namen schon in der Datenbank existiert.

Die card Klasse habe ich als abstrakte Klasse definiert, weil es nur Spell- oder Monsterkarten geben kann und keine normalen Karten.

Ich hatte mich entschieden drei Tabellen für die Datenbank zu erstellen. Einmal eine für die Userdaten, eine für die Packages wo nur ihre ID und ob Sie gekauft wurden, vermerkt wurde und eine für die Karten, wo alle Informationen von der Karte so, wie wenn Sie gehört und ob Sie sich in einem Deck Protokoll MTCG Intermediate Hand-In befindet. Im Programm habe ich deck und stack als eigene Klassen entfernt und sie gegen zwei Listen in der Userklasse ersetzt, weil ich nur die Karten in einer Liste in ihnen gespeichert hatte und mich entschied das ich Sie nicht als eigene Klasse brauche.

Structure

Das Programm ist in einen Teil der sich mit den http/REST-Elementen befasst und in einem Teil, der sich mit den Modelklassen befasst, sowie einen Battleteil welcher sich mit der Kampflogik befasst und einen Datenbankteil, welcher die Verbindung zur Postgres Datenbank übernimmt.

Unit testing

Ich habe mich fokussiert die Kampflogik mittels Unittests zu testen da sie keine Datenbankeinbindung gebraucht haben, um Tests für Sie zu erstellen. Ich habe mich besonders auf die Sonderregeln fokussiert wo ich jede einzelne abgeprüft hab. Ich habe auch die ELO-Kalkulation getestet.

Lessons Learned

Ich habe gelernt, wie man einen Rest Server mit C# aufsetzt und diesen mit einer Postgres Datenbank verbindet. Außerdem habe ich schmerzlich lernen müssen, wie einfach es ist kleine Fehler zu machen die Stunden brauchen, um gefunden zu werden.

Time

Für das Projekt habe ich um die 80 bis 100 Stunden gebraucht

Git link

git link: <https://github.com/Danijoh1/MTCG.git>