

Änderungen zum Pflichtenheft

July 2022

Änderung der Use Cases:

Spieler anmelden:

Es werden weitere Exceptions geworfen wenn der Benutzername nicht vorhanden ist oder der Benutzer bereits angemeldet.

Spieler registrieren:

Passwort hat keine Vorgaben.

Es wird eine Exception geworfen wenn die Passwörter nicht übereinstimmen oder wenn das Passwort oder der Benutzername ein unerlaubtes Teilwort enthält.

Spielraum erstellen:

Passwort hat keine Vorgaben.

Exception wird geworfen wenn Passwörter nicht übereinstimmen oder Spielraumname oder das Spielraumpasswort unerlaubte Teilwörter enthält.

Spieler bearbeiten:

Passwort hat keine Vorgaben.

Es wird eine Exception wenn das Passwort oder der Benutzername ein unerlaubtes Teilwort enthält.

Spieler löschen:

„keine Löschung erwünscht“ Exception ist nicht vorhanden.

Leaderboard:

Stattdessen ist eine Methode namens top_10_ausgeben vorhanden die die 10 Spieler mit den meisten Punkten zurück gibt und eine Methode namens bestenliste_anpassen die, die Bestenliste nach eines Spiels anpasst.

Chatten: Aufgeteilt in mehrere Methoden.

neue_nachricht ist für das Chatten in der Lobby verantwortlich, neue_spiel_nachricht ist für das Chatten in den Spielen verantwortlich und neue_spielraum_nachricht ist für das

Chatten in den Spielräumen verantwortlich.

Spieler hinzufügen:

Existiert nicht.

Stattdessen nutzen wir die Methode `spielraum_beiitreten`.

Es wird bei der Methode keine Exception geworfen.

Spielen:

Bei der Methode `spiel_starten` wird zusätzlich zu der Exception die geworfen wird wenn nicht genügend Spieler vorhanden sind eine Exception geworfen wenn der Spielraum nicht existiert.

Spiel Starten:

Bei der Methode `SpielStarten` wird eine Exception geworfen wenn das Spiel schon existiert oder wenn sich zu wenig Spieler im Spiel befinden.

Karten verteilen:

Wenn das Spiel gestartet wird werden die Karten verteilt.

Spieler-Zug:

Haben wir so nicht implementiert.

Stattdessen gibt es eine Methode namens `amZug` die zurückgibt ob ein Spieler am Zug ist.

Die Methoden für das Ziehen und Spielen einer Karte sowie für das Beenden eines Zuges existieren jedoch noch.

Spiel beenden:

Implementation der Methode ist so nicht vorhanden.

Stattdessen wird in einer Methode namens `neueRunde` die eine neue Runde initialisiert die das Spiel beendet und die Exceptions für Spiel beenden werden nicht geworfen.

Karte ziehen, Karte spielen:

Die Exceptions wurden nicht implementiert.

Stapel verschieben:

Die Exceptions wurden nicht implementiert.

Zug beenden:

Die Exceptions wurden nicht implementiert.

Hinzugefügte (Use Cases):

- Name: Spieler abmelden
Ziel: Den Spieler abzumelden.

Akteure: Spieler

Vorbedingungen: Spieler

Eingabedaten: Benutzername

Beschreibung: Der Spieler meldet sich ab.

Ausnahmen: Es wird keine Ausnahmen geworfen

Ergebnisse und Outputdaten: Der Spieler wird zu der anmelden Seite geführt und muss sich wieder anmelden.

Systemfunktionen: Zugriffsverwaltung

- Name: Lobby entfernen

Ziel: Ein Spieler wird aus der Lobby entfernt

Akteure: System

Vorbedingungen: Spieler befindet sich in der Lobby

Eingabedaten: Benutzername

Beschreibung: Spieler wird aus der Lobby entfernt

Ausnahmen: Wird geworfen wenn der Spieler nicht in der Lobby ist.

Ergebnisse und Outputdaten: Spieler wurde aus der Lobby entfernt.

Systemfunktionen: Zugriffsverwaltung

- Name: Lobby hinzufügen

Ziel: Spieler wird einer Lobby hinzugefügt.

Akteure: System

Vorbedingungen: Spieler meldet sich an.

Eingabedaten: Benutzername des Spielers.

Beschreibung: Spieler wird einer Lobby hinzugefügt.

Ausnahmen: Es wird keine Exception geworfen.

Ergebnisse und Outputdaten: Spieler wurde der Lobby hinzugefügt.

Systemfunktionen: Zugriffsverwaltung

- Name: Spieler entfernen

Ziel: Entfernt einen Spieler aus dem Spielraum

Akteure: System

Vorbedingungen: Spieler ist in einem Spielraum

Eingabedaten: Spieler name , Spielraum name

Beschreibung: Wenn der Spieler sich aus einem Spielraum entfernen will, wird der Spieler dann aus dem Spielraum genommen.

Ausnahmen:

Spielraum existiert nicht

Spieler nicht in diesem Spielraum

Ergebnisse und Outputdaten: Der Spieler ist zurück in der Lobby.

Systemfunktionen: Verwaltung der Spielräume

- Name: Spielraum verlassen.
Ziel: Der Spieler kann den Spielraum verlassen.
Akteure: Spieler
Vorbedingungen: Spieler befindet sich im Spielraum.
Eingabedaten: Name des Spielraums und des Spielers.
Beschreibung: Sorgt dafür dass der Spieler den Spielraum verlassen kann.
Ausnahmen: Es wird keine Exceptions geworfen.
Ergebnisse und Outputdaten: Spieler hat den Spielraum verlassen.
Systemfunktionen: Verwaltung der Spielräume
- Name: Stapel verschieben
Ziel: Spieler können Stapel zu einem anderen passenden Stapel verschieben.
Akteure: Spieler
Vorbedingungen: Spieler sind im Spiel und am Zug.
Eingabedaten: Name des Spielraums, der Ausgangsstapel und der Zielstapel.
Beschreibung: Spieler kann ein Stapel auf einen passenden Stapel verschieben.
Ausnahmen: Es wird keine Exception geworfen.
Ergebnisse und Outputdaten: Der Stapel wurde wenn die Spielregeln erfüllt wurden auf einen anderen Stapel verschoben.
Systemfunktionen: Spielverwaltung
- Name: zahle Chip
Ziel: Der Spieler zahlt Chips.
Akteure: System
Vorbedingungen: Spieler ist im Spiel und beendet Zug.
Eingabedaten: Name des Spielraums und des Spielers und Anzahl der Chips die er bezahlen muss.
Beschreibung: Der Spieler bekommt so viele Chips abgezogen wie er bezahlen muss.
Ausnahmen: Es wird keine Exception geworfen.

Ergebnisse und Outputdaten: Der Spieler hat so viele Chips abgezogen bekommen wie er bezahlen musste.

Systemfunktionen: Spielverwaltung

- Name: handkartenSortieren
Ziel: Die Handkarten eines Spielers aufsteigend oder nach Farbe sortieren
Akteure: Der Spieler
Vorbedingungen: Der Spieler ist in einem Spiel
Eingabedaten: Spielraumname, Spieler name, filter
Beschreibung: Wenn der Filter auf true gesetzt ist, werden die Karten in aufsteigender Reihenfolge sortiert und wenn false , werden die Karten nach Farbe sortiert.
Ausnahmen: Es werden keine Ausnahmen geworfen.
Ergebnisse und Outputdaten: Die Karte des Spielers sind entweder aufsteigend oder nach Farbe sortiert.
Systemfunktionen: Spielverwaltung
- Name: rename Punktekonto
Ziel: Ändern des Namens des Spielers des Punktekontos
Akteure: Spieler
Vorbedingungen: Spieler bearbeitet Profil.
Eingabedaten: alter Name des Spielers und neuer Name des Spielers.
Beschreibung: Ändern des Namen des Spielers des Punktekontos.
Ausnahmen: Es wird keine Exception geworfen.
Ergebnisse und Outputdaten: Der Name wurde geändert.
Systemfunktionen: Bestenliste
- Name: neues Punktekonto
Ziel: Erstellen eines Punktekontos.
Akteure: System
Vorbedingungen: Spieler registriert sich.
Eingabedaten: Name des Spielers für den das Punktekonto angelegt werden soll.
Beschreibung: Erstellen und Speichern eines Punktekontos eines Spielers in einer Bestenliste.
Ausnahmen: Es wird keine Exception geworfen.
Ergebnisse und Outputdaten: Punktekonto wurde erstellt.

Systemfunktionen: Bestenliste

- Name: punktekonto_loeschen
Ziel: Die Punktekonto eines Spielers zu löschen.
Akteure: System
Vorbedingungen: Spieler hat eine Punktekonto
Eingabedaten: Benutzername
Beschreibung: Die Punktekonto eines Spielers wird gelöscht.
Ausnahmen: Es werden keine Ausnahmen hier geworfen
Ergebnisse und Outputdaten: Die Punktekonto des Spielers wird gelöscht.
Systemfunktionen: Bestenliste
- Name: Bot entfernen
Ziel: Ein Bot wird entfernt.
Akteure: Spieler
Vorbedingungen: Spieler befindet sich in einem Spielraum.
Eingabedaten: Name des Spielraums
Beschreibung: Ein Bot wird entfernt.
Ausnahmen: Es werden keine Ausnahmen hier geworfen
Ergebnisse und Outputdaten: Ein Bot wurde gelöscht
Systemfunktionen: Spielraumverwaltung

Hinzugefügte Produktdaten:

Name: Chips

Fachliche Beschreibung: Die Chips der Spieler.

Relevante Systemfunktionen: Spielverwaltung