

Rollenbeschreibungen

Projektleiter:

Name: Levi Wanisch

Beschreibung: Zuständig für Ausarbeitung von Projektvision (was soll erreicht werden). Auch Anpassungen an der Projektvision werden in Zusammenarbeit mit Planung/Management vorgenommen.

Planung/Management:

Name: Manuel Ketz

Beschreibung: Erstellen eines konkreten Zeitplans für Programmierer und Designer. Festlegung welche Technologien und Werkzeuge zum Einsatz kommen. Entwickeln von Lösungsansätzen bei Komplikationen welche nach Absprache mit dem Projektleiter (ggf.) zum Einsatz kommen.

Programmierer:

Name: Saad Elhafiane

Beschreibung: Erstellung des Codes für Spiellogik(Server) sowie Code Gerüst für die Benutzeroberfläche(Client). Verwendete Technologien sowie Zeitplan werden vorgegeben. Bei Problemen Rückmeldung an Planung/Management.

Designer:

Name: Safouane Knani

Beschreibung: Design und Erstellung der Benutzeroberfläche mithilfe von den Programmierern vorgegebenen(erstellten) Werkzeugen. Bei Problemen Rückmeldung an Planung/Management.

Spielidee

Akteure:

Clients: Rechner mit verfügbarer Internet-Anbindung. Auf diesen wird die Clientseitige Doppelkopf-Client-Anwendung ausgeführt welche ihrerseits Daten von der Doppelkopf-Server-Anwendung anfordert. Es können mehrere Clients gleichzeitig Daten vom Server anfordern.

Server: Rechner auf welchem die Serverseitige Anwendung des Spiels sowie eine Datenbank zur Benutzerverwaltung ausgeführt wird.

Aktionen:

Client-Anwendung:

- Registrieren: Nutzer können sich nach Angabe ihrer e-Mail Adresse eines Benutzernamens und Passworts registrieren sollte die angegebene e-Mail nicht bereits registriert sein.
- Anmelden: Registrierten Nutzern ist es möglich sich mittels Angabe ihrer e-Mail Adresse oder ihres Benutzernamens und ihres Passworts anzumelden.
- Anmelden(Gast): Allen Nutzern ist es möglich sich ohne Angabe von Daten als Gast-Nutzer anzumelden.
- Rangliste einsehen: Alle Nutzer können die Rangliste der besten Spieler einsehen.
- Spiel suchen: Alle Nutzer können ein Spiel auf dem Server suchen

Server-Anwendung:

- Registrieren: Registriert Nutzer in der Datenbank
- Ausführen der Spiellogik: Verwendet werden die Standard Doppelkopf-Turnier-Regeln.
- Spielinstanz starten: Startet eigene Spielinstanz (in eigenem Thread). Eine Spielinstanz kann bis zu 4 Spieler beherbergen. Ist die Kapazität der Instanz erreicht wird ein Spiel gestartet.
- Nutzer-Information abfragen: Anfragen nach Nutzerdaten können an die Datenbank gesendet und die Antworten ausgewertet werden.
- Saisons verwalten: Verwaltet Zähler für Saison-Timer und setzt diesen sowie die Rangliste beim Start einer neuen Saison zurück.

Besonderheiten:

- Saison basiertes Ranglistensystem
- angemeldete Nutzer erhalten eine bestimmte Anzahl Punkte für gespielte Spiele, mehr durch Siege als durch Niederlagen (es sollen auch minus Punkte möglich sein, z.B bei mehreren Niederlagen in Folge, somit kann man in der Rangliste auch absteigen)
- abhängig von ihrer Punktzahl werden Nutzer Plätze in der Rangliste zugeordnet
- am Ende jeder Saison
 - > wird die Rangliste zurückgesetzt (ggf. werden die Punkte der Spieler nicht komplett zurückgesetzt, die Funktion hierzu muss noch konkret formuliert werden)
 - > werden die 3 besten Spieler geehrt indem ihre Namen und ihr Rang während der Gesamten Laufzeit der nächsten Saison sichtbar für alle in der Client-Anwendung angezeigt werden

Zusatzfunktionen:

- andere Darstellung der Karten zur besseren Unterscheidung zwischen 'Trumpf' und 'Fehl' sowie ihrer Wertmäßigkeiten
- Doppelkopf Tutorial

Projektbeschreibung

Komponenten und Technologien:

Client:

- *Darstellung:*
 - > Die Darstellung der Benutzeroberfläche auf dem Client wird mit JavaFX realisiert
 - > Thin-Client
- *Kommunikation:*
 - > Die Kommunikation des Clients mit dem Server wird mit JavaEE (ClientBuilder) umgesetzt

Server:

- *Spiellogik:*
 - > Die Spiellogik wird mit Java implementiert
- *Kommunikation:*
 - > die Kommunikation des Servers mit Clients wird mit JavaEE umgesetzt
- *Datenbankanbindung:*
 - > die Datenbankanbindung wird mit JDBC realisiert (ggf. MariaDB Connector/J)

Datenbank:

- *Datenbank-Management-System:*
 - > als RDBMS wird OracleDB eingesetzt
 - > alternativ MariaDB