

Beer Pong -rankkausjärjestelmä

Tietokantasovellus

Periodi 3, 2015

Juuso Strömberg

Johdanto

Beer Pong on viime aikoina noussut suosituimmaksi seurapeliksi Kumpulan kampuksen opiskelijoiden yhteisissä illanvietoissa. Pelille järjestetään joka vuosi turnauksia opiskelijavetoisesti ja peliä pelataan myös epävirallisissa tapahtumissa varsin paljon. On huomattu, että tulee tilanteita, joissa kokemattomampi pelaaja saa vastaanansa toista sataa ottelua pelannut veteraani, joka kuvainnollisesti pieksee kokemattomamman vastustajansa ottelussa. Tämä ei aina ole toivottua, sillä peli on suunniteltu rennoksi ajanvietoksi.

Tämän sovelluksen tavoitteena on korjata tämä tilanne. Sovellus on ranking-järjestelmä Beer Pong -pelaajille, jossa pelaajat saavat kukin oman ranking-arvonsa. Arvo perustuu pelaajan turnaus-sijoituksiin ja haastepelien tuloksiin. Jokainen pelaaja voi rekisteröityä palveluun. Rekisteröitymisen jälkeen pelaaja saa ranking-arvon 1000,0, joka on kaikille sama lähtökohta. Arvo nousee voittamalla turnaus- tai haasteotteluita ja laskee häviämällä niitä.

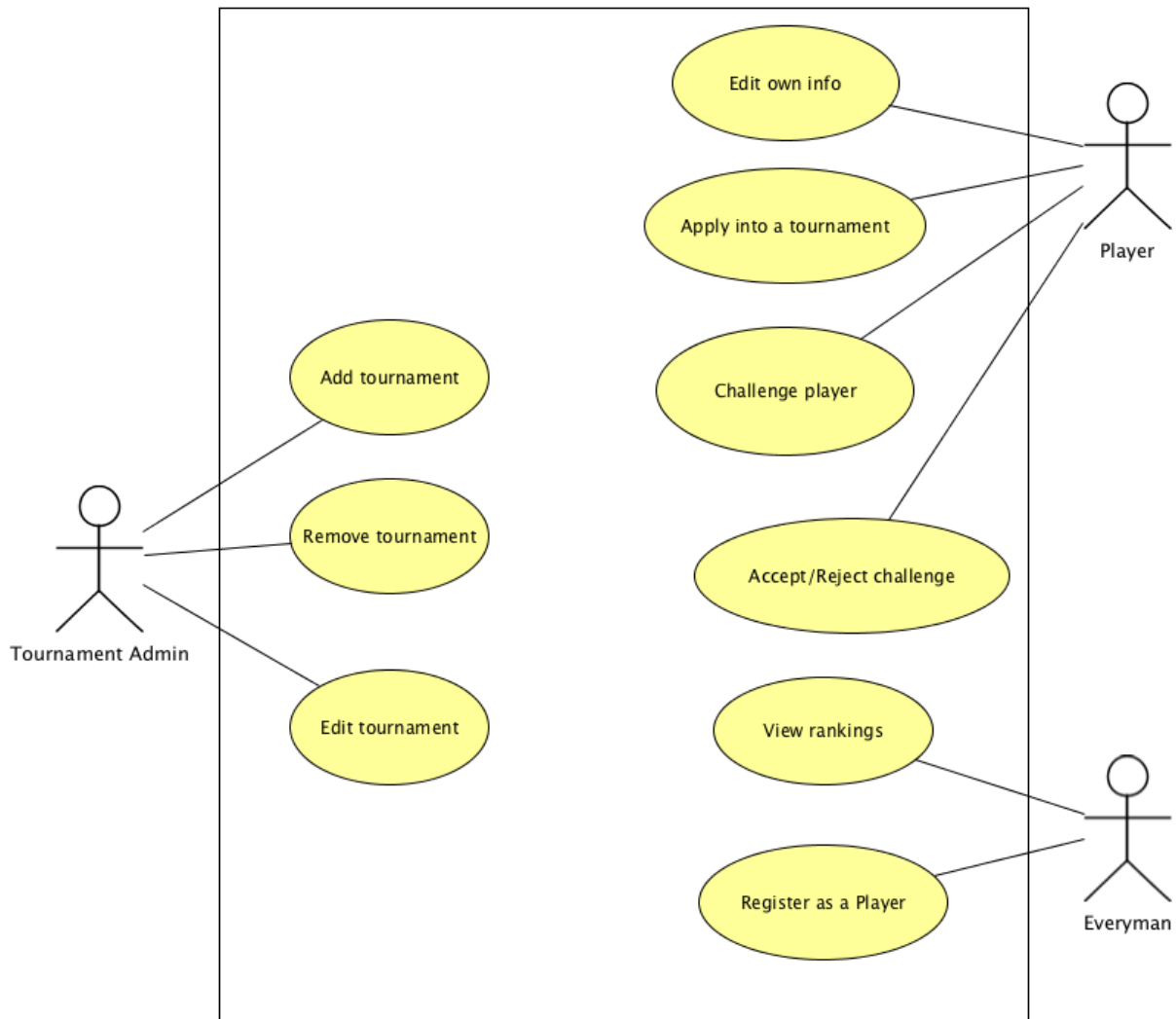
Järjestelmässä pelaajilla on mahdollisuus katsoa omia tilastojaan ja haastaa muita pelaajia haasteotteluun. Lisäksi heillä on mahdollisuus osallistua järjestelmään merkittyihin turnauksiin. Turnausotteluissa on oma painotus ranking-arvoille, joten niistä saa enemmän pisteitä kuin haasteotteluista.

Koska turnausvoitoista saa paremmat pisteet, turnausten lisääminen järjestelmään tapahtuu vain turnausjärjestäjiksi valittujen käyttäjien toimesta. Heillä on mahdollisuus luoda uusia turnauksia, poistaa ja muokata niitä. He hyväksyvät pelaajien hakemukset turnauksiin ja voivat samalla myös kutsua niitä.

Järjestelmä toteutetaan Ruby-ohjelmointikielellä käyttäen Sinatra-ohjelmistokehystä. Demonstraatioksi ohjelma tullaan julkaisemaan Heroku-palvelussa. Tietokantana käytetään PostgreSQL-relaatiotietokantajärjestelmää. Sovelluksen käyttöönottamiseksi omalle palvelimelle palvelin tarvitsee Ruby- ja PostgreSQL-tuen lisäksi jonkin Rubylle soveltuvan julkaisualustan kuten Phusion Passengerin. Mikäli halutaan saada Heroku-tyylistä toiminnallisuutta sovelluksen julkaisemiseen, niin voidaan käyttää Docker-alustaan perustuvaa Dokku-julkaisujärjestelmää tai suuremmille infrastruktuureille tarkoitettua Deis-julkaisujärjestelmää. Käyttäjän selain täytyy tukea javascriptiä.

Käyttötapaukset

Kaavio



Käyttäjärühmät

Jokamies (Everyman)

Jokamiehellä tarkoitetaan jokaista rekisteröitymättömää käyttäjää, joka tulee sivustolle. Kaikki muut ryhmät kuuluvat tähän ryhmään.

Pelaaja (Player)

Pelaajalla tarkoitetaan pelaajaksi rekisteröitynyttä käyttäjää.

Turnauksen ylläpitäjä (Tournament admin)

Turnauksen ylläpitäjä on pelaajaksi rekisteröitynyt henkilö, jolle on annettu oikeudet luoda ja ylläpitää turnauksia. Ylläpitäjä kuuluu siis myös pelaaja-ryhmään.

Käyttötapauskuvaukset

Jokamies

- Tilastojen katselu
 - Jokainen sivulle tulija pääsee näkemään pelaajien ja turnausten tilastoja
- Rekisteröityminen pelaajaksi
- Kirjautuminen

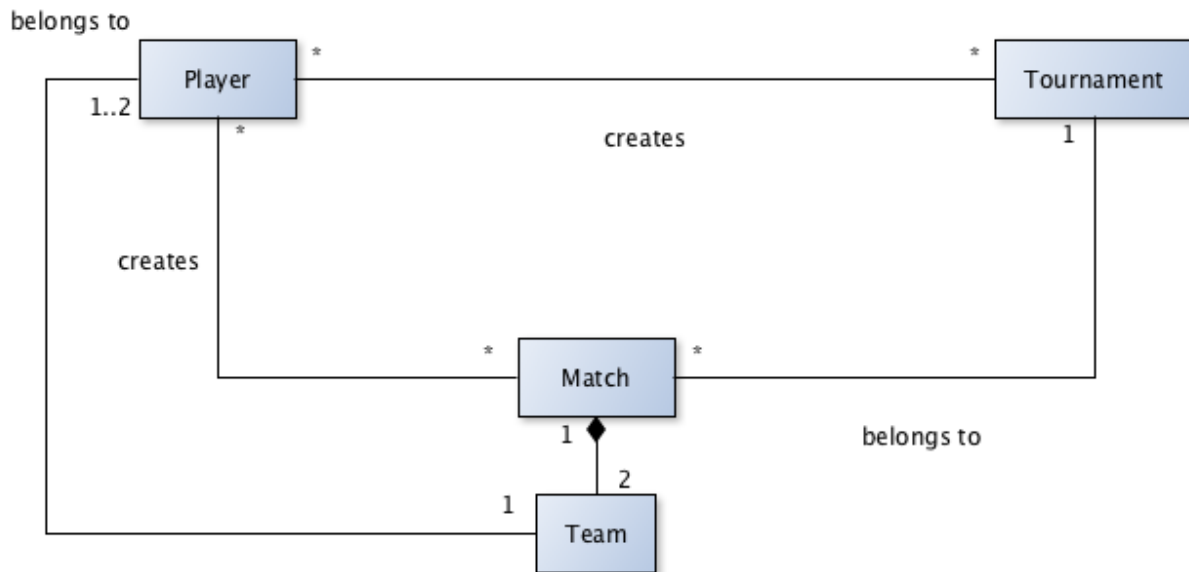
Pelaaja

- Turnaukseen liittyminen
 - Jokainen rekisteröitynyt pelaaja voi ilmoittautua turnaukseen.
- Pelaajan haastaminen
 - Jokainen pelaaja voi haastaa toisen pelaajan
- Haasteen hyväksyminen / hylkääminen
 - Jokainen pelaaja voi joko hyväksyä tai hyljätä toisten pelaajien lähettämät haasteet
- Omien tietojen muokkaaminen

Turnauksen ylläpitäjä

- Turnauksen luominen / editoiminen / poistaminen
- Turnaushakemusten hyväksyminen / hylkääminen

Tietosisältö



Pelaaja (player)

Attribuutti	Arvojoukko	Kuvaus
username	merkkijono, pakollinen	käyttäjätunnus, jota käytetään kirjautumiseen ja sivustolla toimimiseen
password	merkkijono, pakollinen	käyttäjän salasanan bcrypt-tiiviste
player_name	merkkijono	käyttäjän nimi
ranking_score	desimaali, vakio 1000.0	käyttäjän ELO-ranking-pistemäärä
games_played	kokonaisluku	pelattujen ottelujen määrä
wins	kokonaisluku	voittojen määrä
losses	kokonaisluku	tappioiden määrä
ties	kokonaisluku	tasapelien määrä
is_admin	boolean-arvo	onko käyttäjä järjestelmänvalvoja

Joukkue (team)

Attribuutti	Arvojoukko	Kuvaus
team_name	merkkijono, pakollinen	joukkueen nimi
player_one_id	avain, pakollinen	joukkueen pelaajan avain
player_two_id	avain	joukkueen toisen pelaajan avain, mikäli on

Turnaus (tournament)

Attribuutti	Arvojoukko	Kuvaus
tournament_name	merkkijono, pakollinen	turnauksen nimi
start_date	päivämäärä, pakollinen	aloituspäivämäärä
start_time	aika, pakollinen	aloitusaika
end_date	päivämäärä, pakollinen	päättymispäivämäärä
end_time	aika, pakollinen	päättymisaika
location	merkkijono, pakollinen	turnauspaikan osoite
admin_id	avain, pakollinen	turnauksen järjestäjän avain
status	merkkijono	turnauksen tilanne

Turnaustiimi (tournament_team - turnauksen osallistujien kirjaamiseen tarvittava liitostaulu)

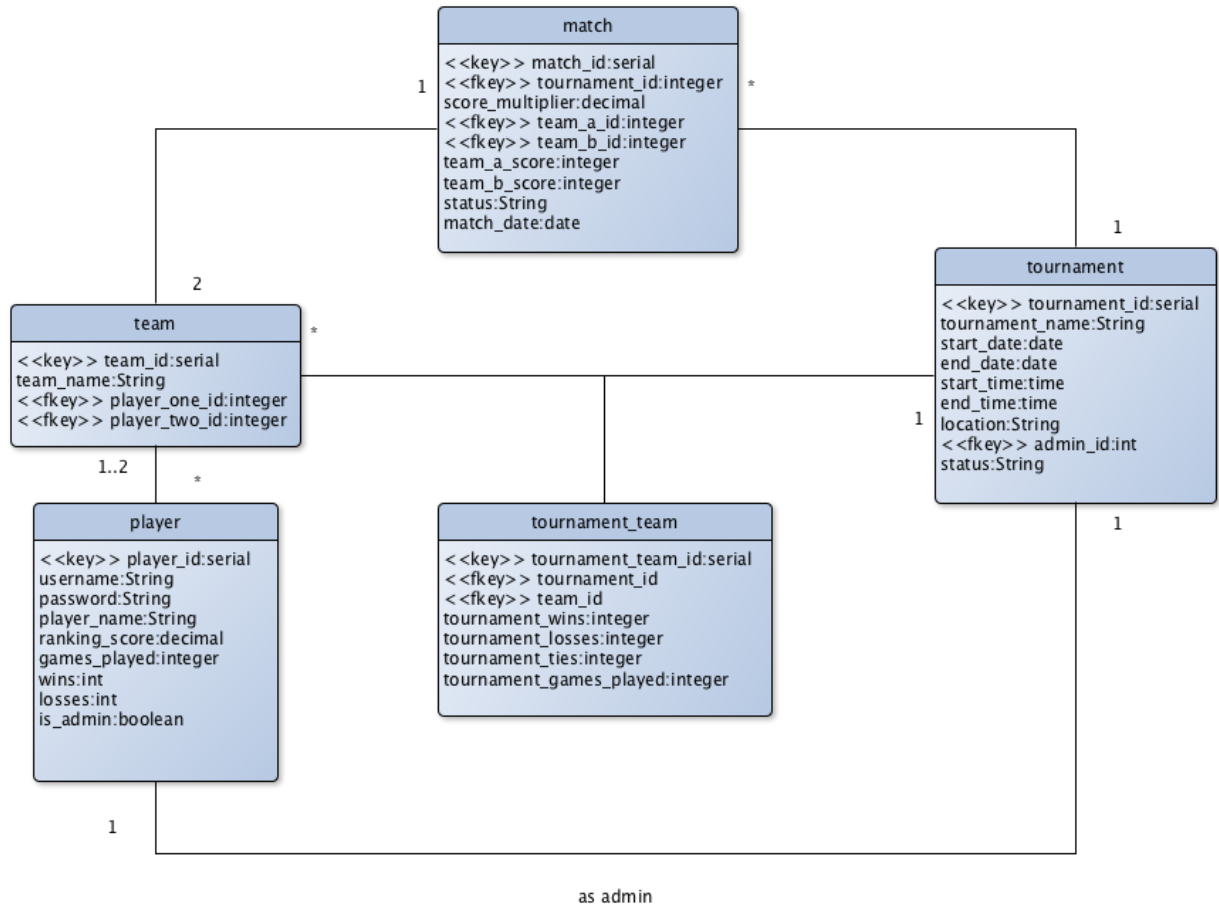
Attribuutti	Arvojoukko	Kuvaus
tournament_id	avain, pakollinen	turnauksen avain
team_id	avain, pakollinen	joukkueen avain
tournament_wins	kokonaisluku	turnauksessa tulleet voitot

tournament_losses	kokonaisluku	turnauksessa tulleet häviöt
tournament_ties	kokonaisluku	turnauksessa tulleet tasapelit
tournament_games_played	kokonaisluku	turnauksessa pelatut ottelut

Ottelu (match)

Attribuutti	Arvojoukko	Kuvaus
tournament_id	avain	turnauksen avain, mikäli turnausottelu
score_multiplier	desimaali	ranking-pistelaskuun käytettävä muuttuja. 1.0 Mikäli haasteottelu, 1.5 jos turnausottelu.
team_a_id	avain,pakollinen	kotijoukkueen avain
team_b_id	avain, pakollinen	vierasjoukkueen avain
team_a_score	kokonaisluku	kotijoukkueen pisteet
team_b_score	kokonaisluku	vierasjoukkueen pisteet
status	merkkijono	ottelun tilanne
match_date	päivämäärä	ottelun päivämäärä

Relaatiotietokantakaavio



Järjestelmän yleisrakenne

Sovellus on tehty MVC-mallia noudattaen. Kontrollerit, mallit ja näkymät löytyvät kansioista controllers, models ja views. Kontrollerit on toteutettu mallien mukaan, mutta lisäyksenä on etusivuun ja kirjautumiseen liittyviä toimintoja kontrolloiva main_controller.rb.

Sovellukseen käsin kirjoitettu kevyt validaattori ja ranking-pisteiden laskuun käytettävä apuluokka löytyvät kansioista helpers.

Sovelluksen alustus tapahtuu tiedostossa app.rb, joka löytyy juurikansiosta. Koska kaikki osat ovat jakautettu omiin kansioihinsa, on jokaiseen kansioon luotu erikseen alustusta varten tiedosto init.rb, jossa kansion luokat ladataan sovelluksen käyttöön. Kaikki konfiguraatio tapahtuu myös tiedostossa app.rb.

Sovellukseen käytettävät apukirjastot on lueteltu tiedostossa Gemfile. Sovelluksen ajamiseen tarvittavat ympäristömuuttujat ovat lueteltu tiedostossa .env.

CSS-tiedostot löytyvät kansioista public/css.

The diagram illustrates the application's page structure, organized into two main sections: 'Julkiset sivut' (Public pages) and 'Kirjautumista vaativat' (Pages requiring login).

Julkiset sivut (Public pages):

- Aloitussivu index.slim** is the central hub, linking to:
 - Rekisteröintisivu register.slim**
 - turnauksen sivu show_tournament.slim**
 - pelaajan profilisivu show_player.slim**
 - Kaikki ottelut -sivu matches.slim**
 - Kaikki pelaajat -sivu players.slim**
 - Kaikki turnaukset -sivu tournaments.slim**
- Rekisteröintisivu register.slim** and **turnauksen sivu show_tournament.slim** are also linked to each other.

Kirjautumista vaativat (Pages requiring login):

- Kaikki ottelut -sivu matches.slim** links to **Ottelun editointi edit_match.slim** and **Ottelun lopetus end_match.slim**.
- Kaikki pelaajat -sivu players.slim** links to **Pelaajan haastosivu challenge_player.slim**.
- Kaikki turnaukset -sivu tournaments.slim** links to **Turnauksen lisäys add_tournament.slim**.

```
graph TD; index[Aloitussivu index.slim] --> register[Rekisteröintisivu register.slim]; index --> show_tournament[turnauksen sivu show_tournament.slim]; index --> show_player[pelaajan profilisivu show_player.slim]; index --> matches[Kaikki ottelut -sivu matches.slim]; index --> players[Kaikki pelaajat -sivu players.slim]; index --> tournaments[Kaikki turnaukset -sivu tournaments.slim]; register --> show_tournament; matches --> edit_match[Ottelun editointi edit_match.slim]; matches --> end_match[Ottelun lopetus end_match.slim]; players --> challenge_player[Pelaajan haastosivu challenge_player.slim]; tournaments --> add_tournament[Turnauksen lisäys add_tournament.slim];
```

Asennusohjeet

Sovellus on tällä hetkellä asennettu Heroku-palveluun. Mikäli sovellus halutaan asentaa omalle palvelimelle, niin pitää PostgreSQL:n lisäksi olla seuraavat asennettu:

- Ruby, vähintään versio 2.2.0
- Bundler (gem install bundler), jolla apukirjastot saa asennettua

Tietokantayhteyteen tarvittavat tiedot menevät .env-nimiseen tiedostoon. Mikäli sovellus halutaan ajaa toisessa Heroku-instanssissa, sovelluksen kansioista löytyvät siihen tarvittavat config.ru- ja Procfile-tiedostot.

Kehitysympäristössä voi käyttää foreman-työkalua, joka käynnistää lokaalin testausversion, joka päivittää näkymät automaattisesti, kun lähdekoodia muutetaan (HUOM! koska Sinatra:ssa kontrollerit ja mallit latautuvat ennen tämän live-reload-toiminnon käynnistymistä, pitää niiden muutokset nähdäkseen käynnistää foreman uudestaan). Tämä tapahtuu komennolla

```
foreman start
```

Käynnistysohjeet

Sovelluksen testiversio löytyy osoitteesta <http://beerpongranking.herokuapp.com/> . Koska sovellus on Herokussa, saattaa sovelluksen käynnistyminen kestää pidempään, jos siellä ei ole ollut vähään aikaan aktiviteettia. Syynä on se, että Heroku käynnistää sovellukset vain tarvittaessa.