

Väliraportti

Ryhmä 10: Jarno Sundström, Lauri Järvisalo, Joni Kokko, Mikael Engström

Tehtävä: League of legends tilastointi ohjelma

1. Asiakasvaatimukset

Asiakas asetti ryhmälle tehtäväksi tuottaa League of legends-pelin pelaajia avustavan ohjelman. Pelin päätyttyä on pelaajan pystyttävä syöttämään käyttöliittymän kautta tietokantaan tiedot kyseisestä pelistä, jonka jälkeen ohjelma käsittelee ne seuraavalla tavalla:

- Ohjelma kerää pelaajan pelistä syöttämän datan tietokantaan
- Osa datasta menee tietokantaan laskurin kautta, jossa ohjelma laskee esimerkiksi paljonko kultaa, pelaaja on tienannut minuutissa
- Ohjelma osaa antaa pelaajalle rakentavaa palautetta jokaisesta pelatusta pelistä asianmukaisella tavalla
- Ohjelma tarjoaa pelaajalle mahdollisuuden tutkia pelihistoriaansa ja verrata omia tilastojaan esimerkiksi parempia pelaajia vastaan
- Ohjelman on pystyttävä noutamaan asiakkaan haluamia tietoja tietokannasta

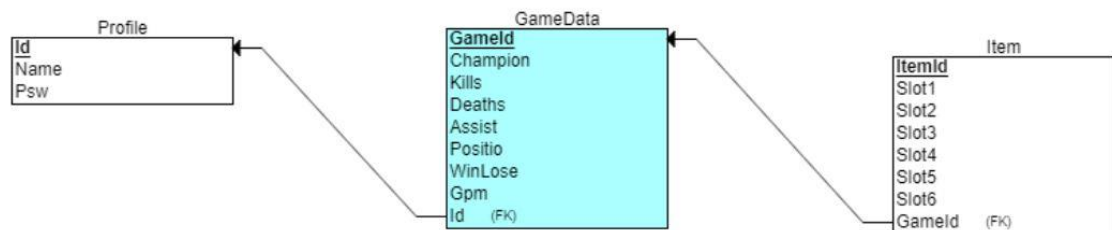
Asiakas haluaa testattavissa olevan helppokäyttöisen käyttöliittymän, jonka kautta hän pystyy syöttämään tietoja pelattavista peleistä tietokantaan. Jatkossa asiakasvaatimuksena on Riotin tarjoaman avoimen rajapinnan hyödyntäminen datan keräämisessä ohjelmaan. Aluksi asiakkaalle kuitenkin riittää, että ohjelma pystyy toteuttamaan yllä mainitut vaatimukset pelaajan itse syöttämillä tiedoilla.

2 Suunnittelu

2.1 Tietokanta

Aluksi ryhmä halusi suunnitella tietokantarakenteen, joka olisi mahdollisimman looginen ohjelman toiminnan ja asiakasvaatimusten näkökulmasta. Suunniteltu pohja koostui 3 erillisestä taulusta, jotka kaikki sisälsivät liitoksen. Ensimmäinen tauluista on Profile, johon liitetään tehdyt profiilit ja niille määritetään pääavaimeksi id, jonka tietokanta luo liukuvasti. Pelillinen informaatio hajautettiin kahteen tauluun, nämä taulut ovat Gamedata ja Item. Gamedata-tilu liitettiin suunnittelu vaiheessa viiteavaimella Profile-tilun id:hen. Gamedata tulee pitämään sisällään pääasiallisen pelaajan suoritusta kuvaavan tiedon, kuten pelin kierroksen aikana saadut tapot, kuoleman, kultaa, yms. Toinen pelillinen informaatiotaulu on Item, tämä taulu pitää sisällään kierroksen aikana pelaajan hallussa olleet tavarat pelin lopussa. Item-tilu liitettiin suunnittelu vaiheessa Gamedatan id:hen viiteavaimella. Molemmille informaatio tauluille luotiin myös oma pääavain id muodossa, syystä että se

voidaan automatisoida tietokannan avulla, kuten Profile taulun osalta on suunniteltu. suunnitelma oli, että Profile-taulusuhde Gamedata-tauluun on OnetoMany, koska yksi profiili voi sisältää useita peli-informaatioita, sen sijaan Gamedata-taulun suhde Item-tauluun on OnetoOne, koska ovat liitännäisinformaatioita, joista molemmat viittaavat vain yhteen pelaajan käymään peliin. Kuvassa 1 ryhmä taulurakenne.



Kuva 1.

2.2 Käyttöliittymä

Käyttöliittymän suunnittelussa ryhmä halusi toteuttaa mahdollisimman helppokäyttöisen ja yksinkertaisen kokonaisuuden. Käyttäjän annetaan syöttää tietoa ohjelmaan vaihteittain ja monta eri ikkunaa hyödyntäen. Tällä tavalla uudet käyttäjät eivät kokisi uusille käyttäjille tyypillisiä ongelmia liian monimutkaisesta työympäristöstä johtuen.

Käyttöliittymän asiakasystävällisyyttä päätettiin viedä pidemmälle tarjoamalla käyttäjälle mahdollisuuden esimerkiksi varoitusteksteillä virheiden ennaltaehkäisemiseksi sekä tekstikenttien pyyhkiminen napin painalluksella vaivattomamman käytön mahdollistamiseksi.

Myös värimaailman ja tyylin suunnittelua aloitettiin ja päätettiin että käyttöliittymän tulisi olla mahdollisimman yksinkertainen. Yksinkertaisella värimaailmalla ja tyyllillä käyttöliittymä ei rasita käyttäjän silmiä ja tekee käyttäjäkokemuksesta miellyttävämmän.

3 Toteutus

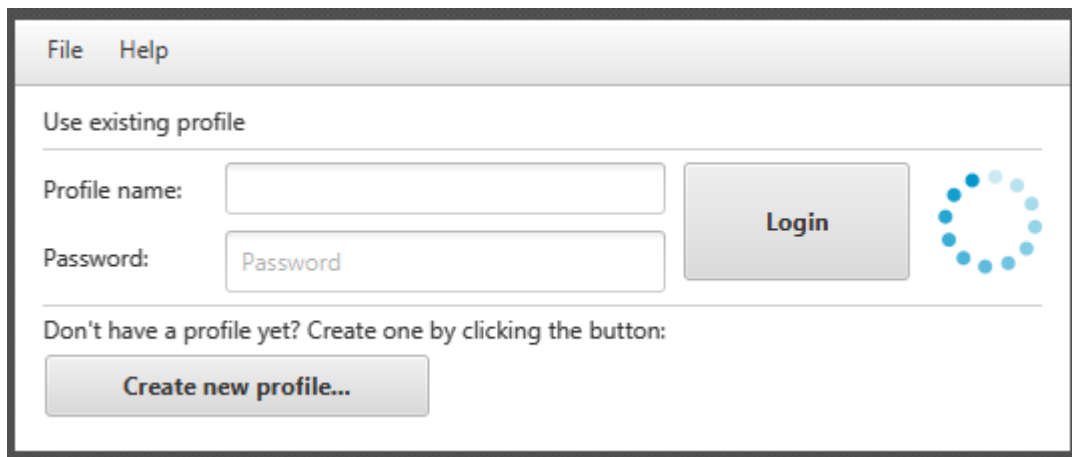
3.1 Tietokanta

Ryhmä lähti toteuttamaan tietokantaan rakentamalla ensin lokaalin tietokannan suunnitellun pohjan mukaisesti. Tämän rakentaminen ei aiheuttanut ryhmälle ongelmaa ja siitä päästiin tietokannan rakentamiseen eCloud koneelle, jonka ryhmä varasi käyttöönsä opintojakson alussa. Tietokanta saatiin myös eCloud koneelle ilman suurempia ongelmia ja lisäsimme tähän tietokantaan ryhmälle yhden yhteisen profiilin, jota käytetään hallinointiin. Tässä vaiheessa ryhmä lähti toteuttamaan tietokannan koodia. Ohjeistuksen mukaisesti käytettiin toteutuksessa käytettiin Hibernatea hyödyksi. Hibernaten cofiguroinnin yhteydessä ongelmia alkoi esiintyä. Ilmeisemmin configuroinnissa mapatut suhteen taulujen välillä eivät toimineet halutulla

tavalla tai kyselyt olivat viallisia tietokannan suuntaan. Lopputuloksena haluttujen tietojen haku ja asettaminen toimi vain puoliksi oikein. Tällä tarkoitetaan sitä että operaatioissa oli toiminallisuutta, mutta ne eivät toimineet halutulla tavalla. Kuitenkin demoaminen alkoi lähestymään ja ryhmämme käytti aivan tolkkuttomia tuntimääriä tämän asian selvittämiseen ja lopputulema ei korjaantunut. Tästä johtuen ryhmä päätti projektin demoa varten muuttaa tietokantarakenteen kaksi tauluiseksi. Täten annettiin profiilille oma taulu Profile ja kaikki peli.informaatio yhteen tauluun nimeltä Gamedata. Nämä taulut toimivat yksittäisinä ja korjausliikkeenä Gamedataan annetaan myös tieto profiilista omaan sarakkeeseen. Tämä suoraviivaisti toimintaa ja tietokantaoperaatiot tuottavat nyt demoa varten haluttuja tuloksia. Ryhmä katsoi tämän ratkaisun vain hetkelliseksi korjaukseksi. Demon jälkeen ryhmän tarkoituksena on yrittää saada tietokanta toimimaan liitoksilla ja muuttaa taulukot suunniteltuun kolmen taulukon malliin.

3.2 Käyttöliittymä

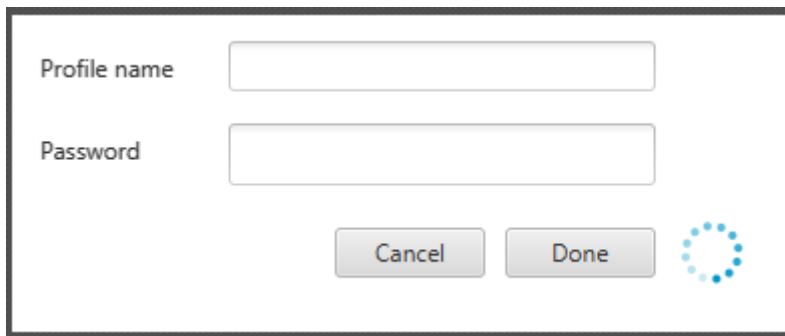
Ryhmä päätti toteuttaa käyttää käyttöliittymän rakentamiseen Scenebuilderiä. Ryhmä oli suunnitteluvaiheessa päättänyt tarvittavien ikkunoiden määrän (5) sekä niiden eri toiminnot. Ohjelman käynnistyessä ensimmäinen ikkuna on sisäänkirjautumista varten kuvassa 2.



Kuva 2.

Kuten kuva 2:stä ilmenee, käyttäjälle tarjotaan mahdollisuus kirjautua suoraan sisään tai luoda uusi käyttäjätili.

Käyttäjätilin luontia varten ryhmä toteutti kuvassa 3 näkyvän ikkunan.

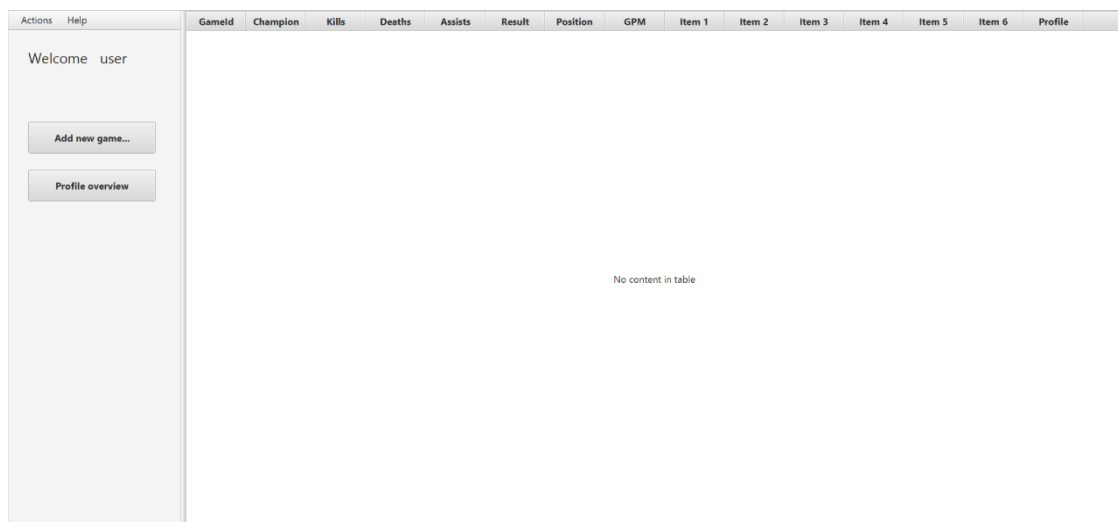


A user creation dialog box with a dark border. It contains two text input fields: 'Profile name' and 'Password'. Below the fields are two buttons: 'Cancel' and 'Done'. To the right of the 'Done' button is a circular loading spinner icon.

Kuva 3.

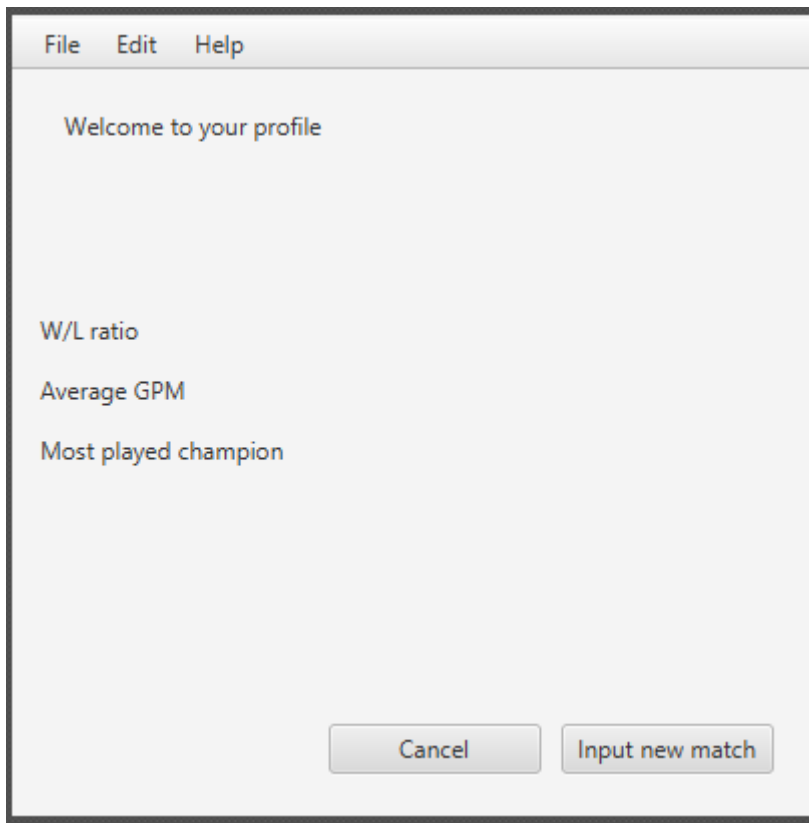
Kuvassa 2 käyttäjän luonti-ikkuna.

Sisäänkirjautumisen jälkeen käyttäjälle tulee näkyviin keskusikkuna, josta käyttäjä voi nähdä pelihistoriansa ja halutessaan lisätä uusia pelaamiansa pelejä tietokantaan tai vain tarkastella profiilinsa yksityiskohtia. Keskusikkuna näkyvissä kuvassa 4.



Kuva 4.

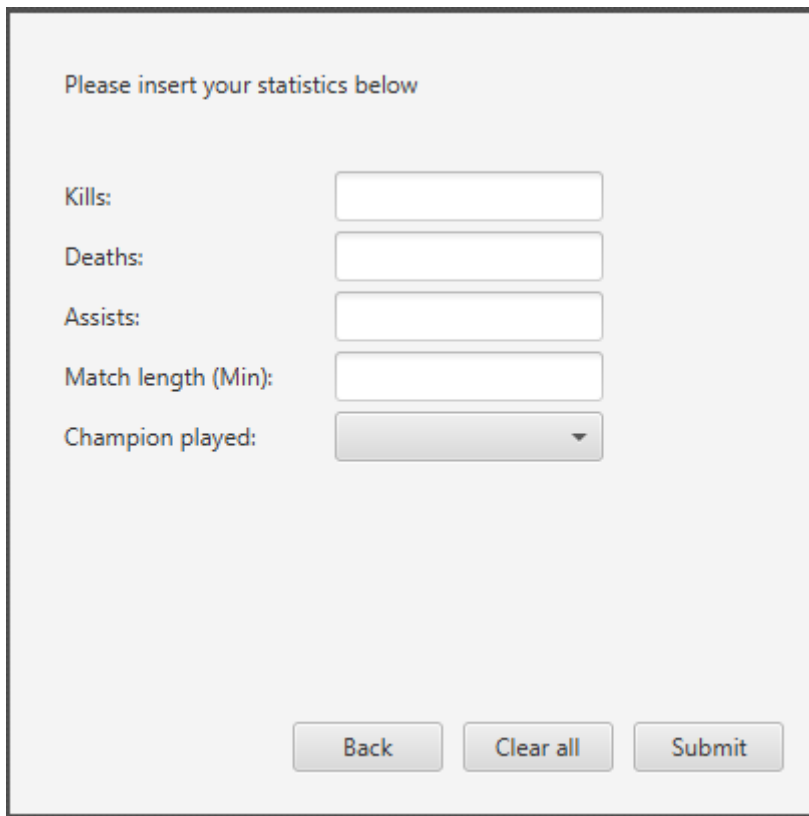
Painamalla 'Profile overview' -painiketta, pystyy käyttäjä näkemään oman profiilinsa, jossa tulee näkyä käyttäjälle tärkeää dataa peleihin liittyen. Kuvassa 5 profiili-ikkuna.



Kuva 5.

Kuvassa 5 käyttäjän oma profiili josta selviää W/L ratio (voitot / tappiot), average GPS (Paljonko kultaa tienattu per minuutti) sekä käyttäjän eniten pelaama hahmo.

Seuraavaksi tarvittiin ikkuna jonka kautta käyttäjä voi syöttää olennaista dataa tietokantaan, kuvassa 6 ryhmän ratkaisu.



Please insert your statistics below

Kills:

Deaths:

Assists:

Match length (Min):

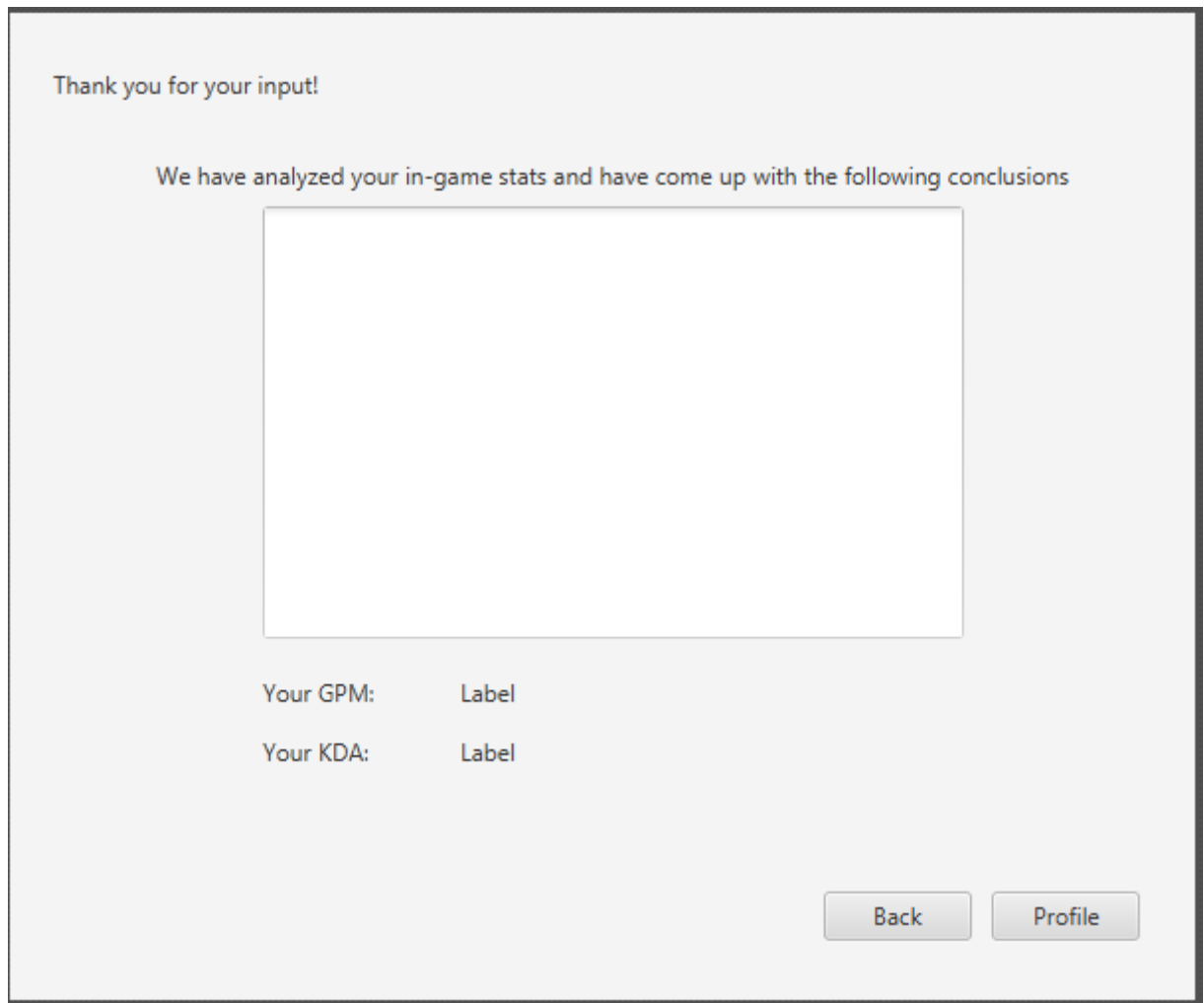
Champion played:

Back Clear all Submit

Kuva 6.

Kuten kuvasta 6 ilmenee, käyttäjälle tarjotaan yksinkertainen näkymä, johon on helppo syöttää kaikki olennainen tieto pelattuun peliin liittyen. Pudotusvalikosta käyttäjä saa valita pelaamansa hahmon, tällä hetkellä valikosta löytyy kolme pelattavaa hahmoa.

Viimeisenä tarvittiin enää palaute ikkuna, josta käyttäjä saa heti palautetta pelatusta ottelusta datan analysoinnin jälkeen, esimerkiksi jos käyttäjä on kuollut liian monta kertaa pelissä sen pituuteen nähden, niin käyttöliittymä mainitsee asiasta palautteessa ja kertoo myös muun olennaisen datan laskutoimituksen jälkeen. Kuvassa 7 ryhmän toteutus palaute ikkunasta.



Kuva 7.

Kuten kuvasta 7 ilmenee, käyttöliittymä antaa palautteen teksti-ikkunassa ja sen alla olennaiset tiedot numerosyötteinä (KDA on tappojen ja avustusten summa jaettuna kuolemilla). Tästä ikkunasta käyttäjälle tarjotaan mahdollisuus navigoida joko profiiliinsa tai takaisin tiedonsyöttö ikkunaan.

4 Tulos

Lopputulos on jakson pituuteen ja tavoitteisiin nähden hyvä. Tavoitteista suurin osa toteutui ja osaa joudutaan jatkamaan ensi jaksossa, ryhmän olisi tarkoitus opetella käyttämään avointa rajapintaa tiedonkeruussa sekä edistää käyttöliittymän toimivuutta ja lisätä toiminnallisuuksia.