

Halpa korttipakka rakentaja

Mikko Leppänen
Tuomas Kyttä

IIO13200
Joulukuu 2015



Contents

1	Asennus	2
1.1	Labranetissä	2
1.2	Labranetin ulkopuolelta	2
1.3	Ulkopuoliset kirjastot	2
2	Tietoa ohjelmasta	2
2.1	Funktionaalisten vaatimusten toteutus	2
2.2	Ei-funktionaaliset vaatimukset	3
2.3	Ominaisuudet jotka toteutettiin ohi vaatimusten	3
3	Lyhyt käyttöohje	4
3.1	Perusnäkyä	4
3.2	Create deck	4
4	Riippuvuudet	5
5	Tiedossa olevat bugit	5
6	Mitä opittiin	6
6.1	Suurimmat haasteet	6
6.2	Mitä opiskella lisää	6
7	Tekijät ja vastuunjako	7
7.1	Tuomas Kyttä	7
7.2	Mikko Leppänen	7
7.3	Molemmat	7

1 Asennus

1.1 Labranetissä

Kloonaa repo. Käynnistä visual studio. Avaa projekti. Valitse repo. Valitse Harkka kansio ja käynnistä Harkka.sln. Käynnistä default.aspx. Jos heittää virheilmoituksen väärästä config tiedoston polusta sulje visual studio ja käynnistä uudelleen. Uudelleen käynnistys ja sen välissä tallentaminen muuttaa paikallisen config tiedoston polun.

1.2 Labranetin ulkopuolelta

Asenna MySQL palvelin. Asenna samalla MySQL workbench ja aja sillä repossa oleva luonti.sql tiedosto. Luo kannalle käyttäjä ja anna SELECT ja INSERT oikeuden ainakin. Avaa solution visual studio 2015:sta. Aseta web configgiin polku ja luodun käyttäjän tunnistustiedot luodulle kannalle. Aja test.aspx. Tämä repii apin ja unirest kirjaston avulla dataa netistä aikaisemmin luodulle kannalle. Tämän jälkeen sivua voi ajaa defaultin kautta.

1.3 Ulkopuoliset kirjastot

Datan saamiseksi kantaan projektissa käytettiin yhtä ulkopuolista kirjastoa: unirest.net. Kirjaston asentaminen toteutettiin repimällä tarvittavat luokat osaksi projektia.

2 Tietoa ohjelmasta

2.1 Funktionaalisten vaatimusten toteutus

- Selaa kortteja: Toimii
- Luo pakka: Toimii
- Lisaa pakkaan kortteja: Toimii

- Tallenna pakka: Toimii
- Selaa valmiita pakkoja: Toimii
- Tietoturva: Ohjelman ainoa kirjoitettava input tarkistetaan RegularExpressionValidatorilla. Ohjelma on täysin tietoturvallinen.
- Muokkaa valmista pakkaa: Ei toimi
- Kirjautuminen ja käyttäjän luonti: Ei ole
- Admin käyttäjä joka voisi admin sivulla käydä uusimassa kannan kortteja, hallinnoimassa käyttäjiä jne: Ei ole

2.2 Ei-funktionaaliset vaatimukset

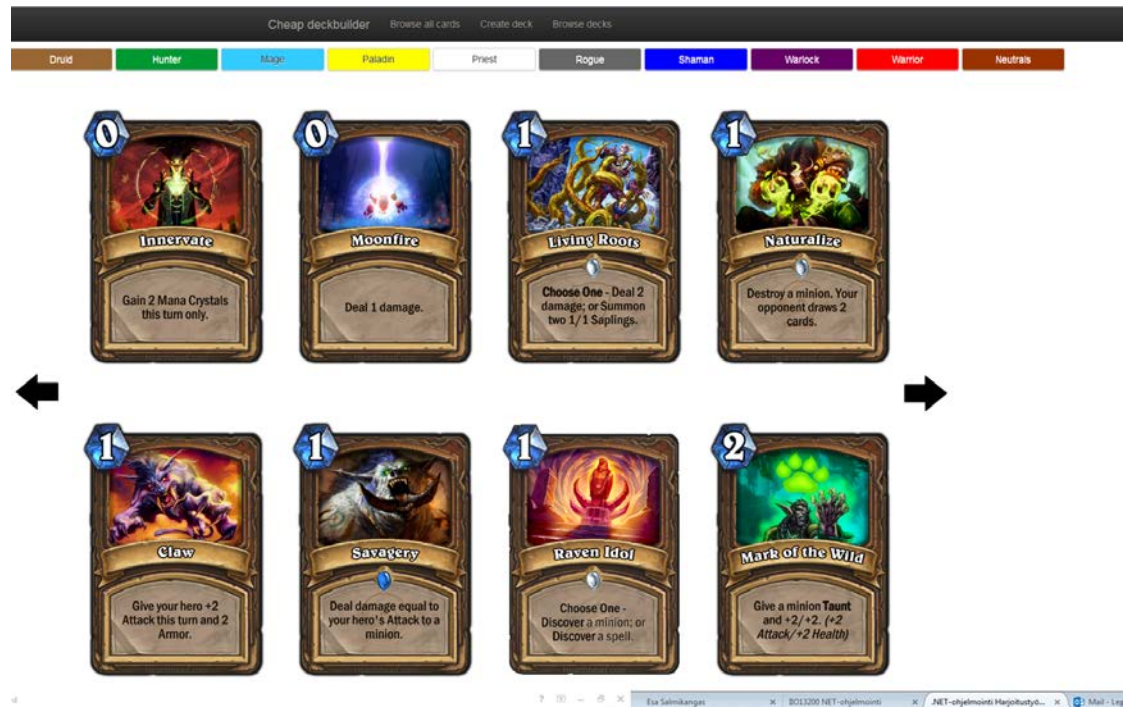
- Korttien selaus sulavaa: .NET:n updatepaneelilla saatiin korttien selaus ikkuna käyttämään ajaxia ja välttyttiin koko sivun uudelleenluomisesta selatessa uudelleen. Seuraavien kuvien esilatausta ei saatu toimimaan annetussa ajassa.
- Pakan tulisi päivittyä kortteja lisätessä: Kortteja lisätessä paneeli päivittyy, mutta poistaessa se päivittyy vain kun suoritetaan toinen operaatio. Dynaamisten triggerian asettaminen update paneelille ei siis onnistunut.

2.3 Ominaisuudet jotka toteutettiin ohi vaatimusten

- Pakkaa luodessa voi valita luokan ja muiden luokkien kortit piilotetaan
- Niin edistyksellinen ulkoasu että se määrittää huonon ulkoasun uudelleen

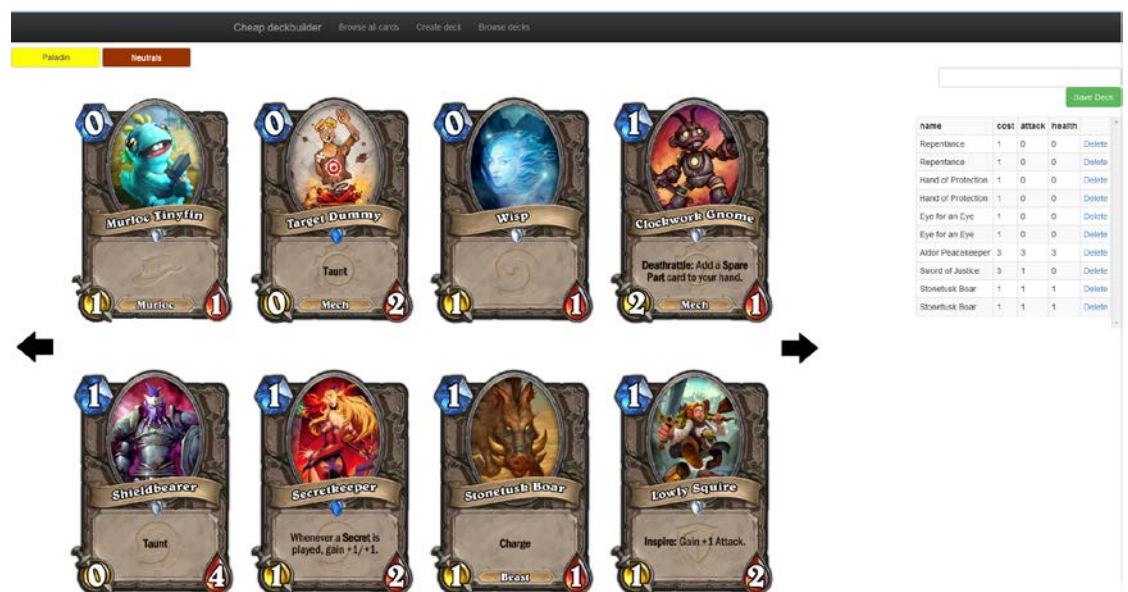
3 Lyhyt käyttöohje

3.1 Perusnäkö



Ylimmästä palkista voi navigoida eri sivuilla ja niissä on toiminnollisuutta mitä odottaisi nimen perusteella. Värikoodatut nappulat joissa on eri luokkien nimiä rajaavat näytettäviä kortteja. Nuolinäppäimillä voi navigoida näytettäviä kortteja eteenpäin.

3.2 Create deck

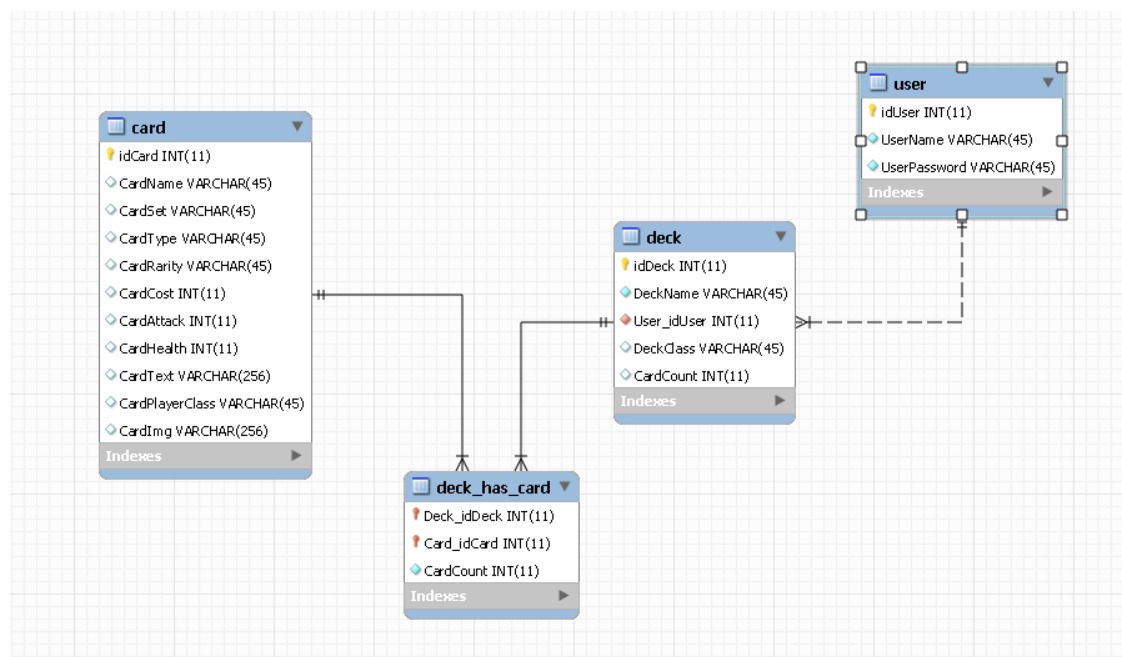


Create deck sivulla valitaan ensin luokka värikoodatuilla nappuloilla jonka jälkeen voi selata kortteja tutuilla menetelmillä ja kortteja klikkaamalla ne lisätään pakkaan joka näkyy sivussa. Save deckista luotu pakka tallennetaan kantaan

Browse decks on hyvin ilmiselvä kontrolleiltaan.

4 Riippuvuudet

MySQL tietokanta. Labranetistä ajettuna webconfig tiedostoon on annettu polku ja hyvin rajoitetun käyttäjän tunnistus tiedot.



Kannanluonti scripti on innodb engineä käyttävä ja on repon juuressa. Testidatan asettaminen tapahtuu test.aspx:n kautta. Test.aspx:n toiminnallisuus oli alkujaan tarkoitus olla adminille tarkoitettulla sivulla.

Kannassa olevien korttien päivittäminen on riippuvainen test.apsx:ssa sijaitsevasta mashape.com api avaimesta. Ohjelman koodissa sijaitseva avain tapetaan Tuomas Kytän toimesta joulukuun 31 päivä, jonka jälkeen kannan korttien päivitys ei onnistu.

5 Tiedossa olevat bugit

- Delete nappi korttien luonnissa ei päivitä pakka paneelia. Delete nappi toimii, mutta sen vaikutus ei näy käyttäjälle välittömästi.

- Selauksessa jos seuraavia kortteja on alle 8, niitä ei näytetä.

6 Mitä opittiin

- Kuinka käytetään update paneelia
- Kuinka käytetään nugettipaketti hallinta systeemiä ja revitään uusia liitännäisiä projektiin
- ASP on http pohjainen, raskas ja asiakaspäähän toiminnollisuuden saaminen on masentavaa
- IComparerin käyttö
- ASP komponenttien rakentuminen / sijoittuminen ja etenkin löytäminen
- Validatorin käyttö
- Dynaamisten nappien luonti datagridiin
- Masterpagen käyttö
- ASP ja MySQL yhdessä käyttöä.
- Entity modelien teko ja labranetissä MySQL entityjen käytön vaikeus
- Istunnon hallinta, sessioon voi tunkea kaikenlaista

6.1 Suurimmat haasteet

- Unirest. Ohjeet käyttöön olivat olemattomat
- Toiminta clientti päähän
- Sivuston dynaamisuus asetti mielenkiintoisia haasteita

6.2 Mitä opiskella lisää

Suurin osa ASP.NET:n toiminnasta tapahtuu palvelin päässä, mutta nykyään on normaalia, että sivusto on dynaaminen. Olisi kiva oppia miten oikeasti ASP sivuja kannattaa puukottaa javascriptilla tai miten ASP.NET:n scriptmanageria voisi käyttää tähän tarkoitukseen.

7 Tekijät ja vastuunjako

7.1 Tuomas Kyttä

- Korttien alustus ja järjestäminen
- Default sivu
- Update paneelien asettaminen
- Bootstrapin asettaminen
- Tyylit
- Korttien selaaminen eteen ja taaksepäin
- Korttien tallennus kantaan
- Tietokanta ja tietokannan käyttäjän luonti

7.2 Mikko Leppänen

- Create deck sivu
- Browse deck sivu
- Pakkojen lisäys kantaan ja luku kannasta
- Pakan korttien täyttö
- Deck model
- Turhien nappien piilotus
- Datagridien täyttö
- 90-80% CardLogic.cs tiedostosta, joka toimii kontrollerina näytösivujen ja modelien välillä.

7.3 Molemmat

- Unirest laajennukset käyttö
- Kortit Apin avulla netistä

- Card model
- Istunnon hallinta