

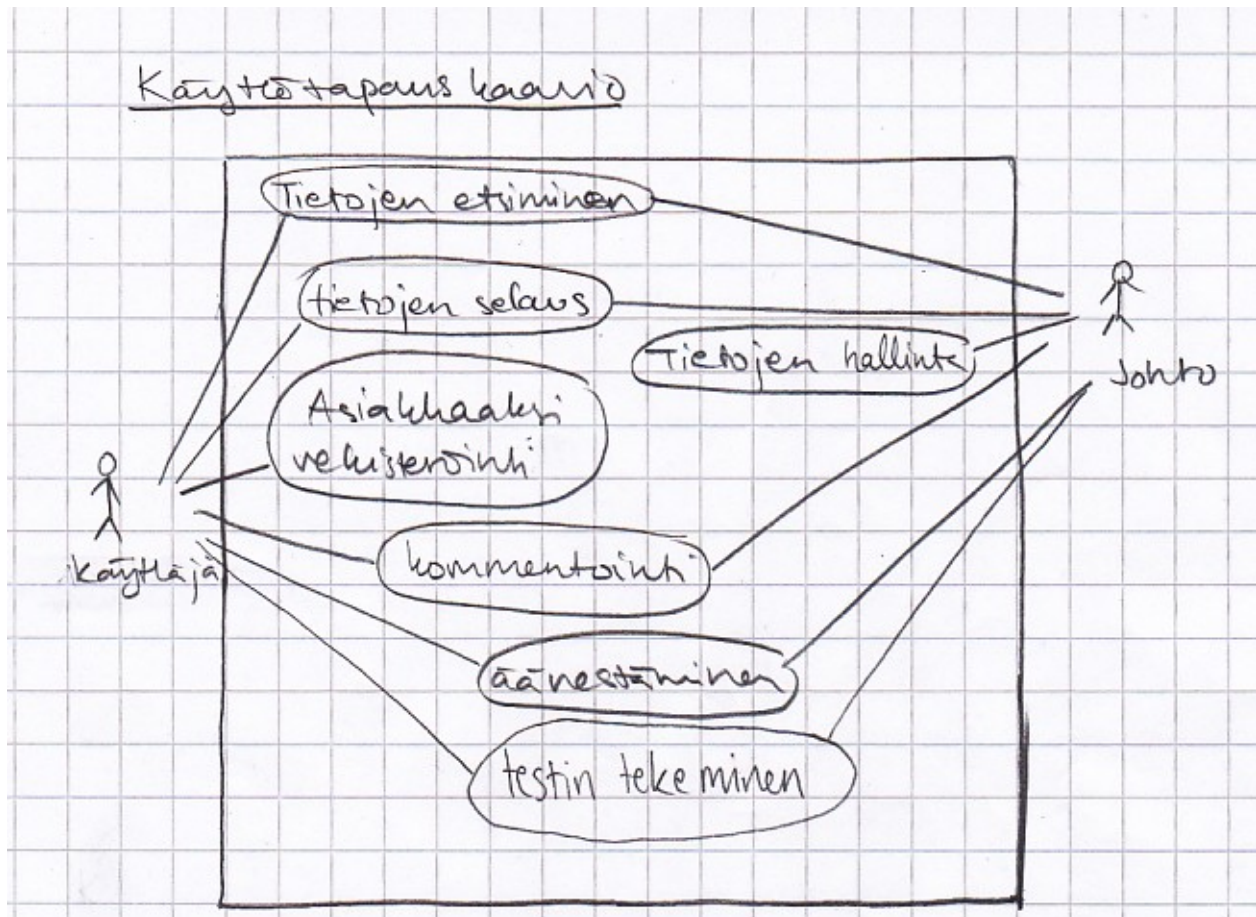
1. AIHEEN KUVAUS (Johdanto)

Kyseessä on George R. R. Martinin fantasiakirjasarjaan A Song of Ice and Fire perustuva tietokanta. Tietokantaan tallennetaan kirjassa muuten sekavina esiintyviä henkilöitä, ihmissuhteita, paikkoja yms ja kannan tarkoitus on olla apuna selventemässä asioita kirjoja lukeville henkilöille. Järjestelmän tavoitteena on siis tietokanta, jota pystyy helposti ja monipuolisesti käyttämään kirjasarjan lukemisen tukena. Tiedot tallennetaan mukaillen sarjan viidettä ja siis uusinta kirjaa, joten osa tietokannan tauluista tulee sisältämään juonipaljastuksia.

Tietojen selailujen lisäksi tietokannasta jo löytyviä objekteja voi päivittää ja kokonaan uusia objekteja voi lisätä kantaan admin-tunnuksilla. Sovelluksessa voi myös testata tietonsa kirjasarjasta, jättää kommentteja vieraskirjaan ja antaa äänensä neljässä eri äänestyksessä liittyen kirjan henkilöihin, sukuihin ja otuksiin.

Tietokanta toteutetaan PostgreSQL-palvelimella ja web-sovelluksen alustajärjestelmä tukee PHP-kieltä. Ohjelmisto edellyttää tietyn tietokannan käyttöä, eikä sitä voi vaihtaa helposti.

2.1 Käyttötapauskaavio



2.2 Käyttäjäryhmät

Käyttäjä

Käyttäjä on kuka tahansa, joka haluaa Internetin välityksellä tulla lukemaan tai etsimään tietoa 'A song of ice and fire' -kirjasarjasta rekisteröityttyään. Hän voi myös kommentoida tietokannan tietoja ja äänestää niistä.

Johto

Johdolla tarkoitetaan henkilöitä, jotka ylläpitävät tietokantaa. Hekin voivat selata ja etsiä tietoa, kommentoida ja äänestää.

3.1 Asiakkaan käyttötapaukset

Tietojen etsiminen

Rekisteröityneet ihmiset voivat käydä etsimässä Internet-sivulla julkaistuja tietoja 'A song of ice

and fire' -kirjasarjasta.

Tietojen selaaminen

Rekisteröityneet voivat käydä lukemassa Internet-sivulla julkaistuja tietoja 'A song of ice and fire' -kirjasarjasta.

Rekisteröityminen

Rekisteröityessään käyttäjä pääsee selaamaan tietoja. Hän antaa käyttäjätunnuksen ja salasanan rekisteröityäkseen. Aktivointi tapahtuu heti.

Kommentoiminen

Rekisteröitynyt käyttäjä voi äänestää tietokannassa olevia tietoja.

Äänestäminen

Rekisteröitynyt käyttäjä voi äänestää tietokannassa olevia tietoja.

Testien tekeminen

Rekisteröitynyt käyttäjä voi tehdä testin 'A song of ice and fire' -kirjasarjasta ja testata omia tietojaan. Testin tulokset julkaistaan kommentilla vieraskirjassa.

3.2 Johdon käyttötapaukset

Tietojen etsiminen

Johto voi käydä etsimässä Internet-sivulla julkaistuja tietoja 'A song of ice and fire' -kirjasarjasta.

Tietojen selaaminen

Johto voi käydä lukemassa Internet-sivulla julkaistuja tietoja 'A song of ice and fire' -kirjasarjasta.

Tietojen hallinta

Johto voi päivittää vanhoja tietoja ja lisätä uusia tietoja.

Kommentoiminen

Johto voi äänestää tietokannassa olevia tietoja.

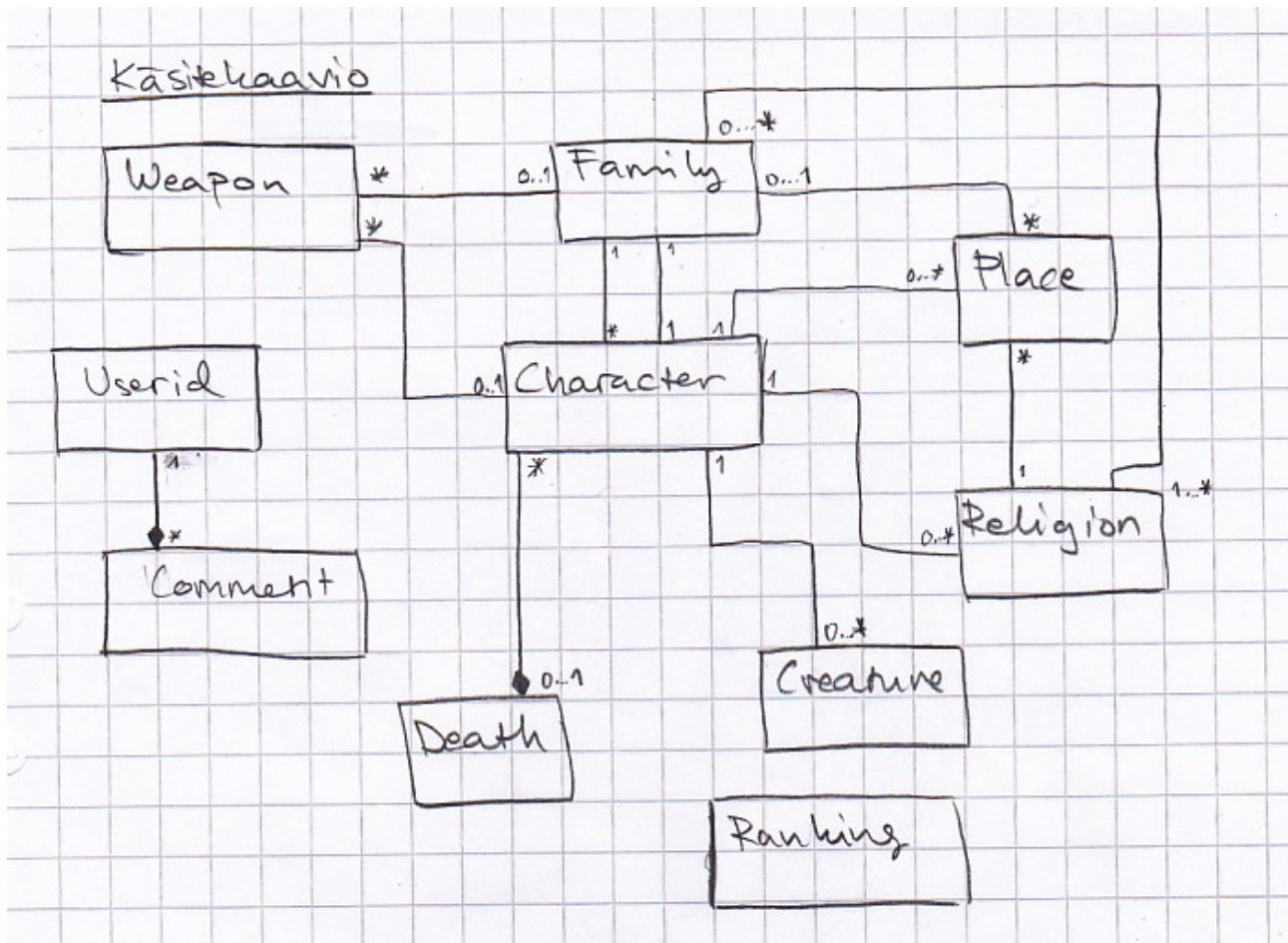
Äänestäminen

Johto voi äänestää tietokannassa olevia tietoja.

Testien tekeminen

Johto voi tehdä testin 'A song of ice and fire' -kirjasarjasta ja testata omia tietojaan. Testin tulokset julkaistaan kommentilla vieraskirjassa.

4.1 Tietosisältö käsitekaaviolla



4.2 Tietokohteet

Character

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono max 30 merkkiä, not null	Henkilön etunimi
famillyname	Merkkijono max 30 merkkiä, not null	Henkilön sukunimi
alias	Merkkijono max 150 merkkiä	Henkilön muut kutsumanimet
title	Merkkijono max 150 merkkiä	Henkilön arvonimet
specialfeatures	Merkkijono max 150 merkkiä	Henkilön erityisominaisuudet

Henkilö voi olla kuulumatta yhteenkään sukuun tai kuulua yhteen sukuun. Yhteen sukuun voi kuulua monta henkilöä. Monta henkilöä voi olla kuollut. Yksi henkilö voi olla monen otuksen omistaja tai ei yhdenkään. Olennoilla on kaikilla vain yksi omistaja. Henkilö voi omistaa monta asetta tai ei yhtäkään ja aseella voi olla vain yksi omistaja. Henkilö voi olla sukunsa tämän hetkinen päämies ja suvun päämiehenä voi olla vain yksi henkilö kerrallaan.

Death

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono max 30 merkkiä	Kuolleen henkilön etunimi
familyname	Merkkijono max 30 merkkiä	Kuolleen henkilön sukunimi
book	Kokonaisluku	Sen kirjan järjestysnumero, jossa henkilö kuolee
causeofdeath	Merkkijono max 50 merkkiä	Kuolinsyy
nameofslayer	Merkkijono max 30 merkkiä, not null	Tappajan tai pääsyyllisen etunimi
familynameofslayer	Merkkijono max 30 merkkiä, not null	Tappajan tai pääsyyllisen sukunimi

Moni henkilö voi olla kuollut, mutta kaikkien ei tarvitse olla kuolleita. Jos henkilö on kuollut ennen kirjansarjan alkua, attribuutin book arvoksi on laitettu 0. Jos henkilö uskotaan kuolleen tai henkilön kuoleman ajankohdasta ei ole selvyyttä, on attribuutin book arvoksi laitettu 100. Jos tappajan tai pääsyylliseen henkilöllisyys on epäselvä, on etu- ja sukunimiksi merkitty unknown. Jos taas henkilö on kuollut johonkin tautiin tai kuolinsyy ei muuten ole kenenkään toisen henkilön tai olennon aiheuttama, on tappajan nimien kohdalle kirjoitettu none.

Death-taulu sisältää juonipaljastuksia!

Religion

Attribuutti	Arvojoukko	Kuvailu
gods	Merkkijono max 30 merkkiä, not null	Jumalhahmojen kutsumanimi
numberofgods	Kokonaisluku, not null	Jumalhahmojen lukumäärä
symbol	Merkkijono, max 30 merkkiä	Uskonnon symboli
servant	Merkkijono, max 30 merkkiä	Uskonnon parissa työskentelevien nimike
mainarea	Merkkijono, max 50 merkkiä, not null	Alue, jossa uskontoa pääasiallisesti harjoitetaan

Uskontoa voi esiintyä monessa eri paikassa, mutta pääasialliseksi palvontapaikaksi voi laittaa vain yhden paikan. Jos jumalten lukumäärä on tuntematon, on attribuutin numberofgods arvoksi laitettu 100.

Creature

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 30 merkkiä, not null	Olennon kutsumanimi
species	Merkkijono, max 30 merkkiä	Laji, jota olento edustaa
nameofowner	Merkkijono, max 30 merkkiä, not null	Olennon omistajan etunimi
familynameofowner	Merkkijono, max 30 merkkiä, not null	Olennon omistajan sukunimi
favoritefood	Merkkijono, max 30 merkkiä	Olennin lempiruoka
specialfeatures	Merkkijono, max 100 merkkiä	Olennon erityisominaisuudet

Olennolla on vain yksi omistaja, mutta henkilö voi omistaa monta olentoa.

Place

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 50 merkkiä, not null	Paikan nimi
location	Merkkijono, max 50 merkkiä	Paikan sijainti
familynameoforiginallord	Merkkijono, max 30 merkkiä, not null	Alkuperäisen päämiehen sukunimi
familynameofcurrentlord	Merkkijono, max 30 merkkiä, not null	Nykyisen päämiehen sukunimi
mainreligion	Merkkijono, max 30 merkkiä, not null	Yleisin uskonto alueella
notableplaces	Merkkijono, max 200 merkkiä	Tunnetuimpia paikkoja sijainnissa
type	Merkkijono, max 30 merkkiä	Millaisesta paikasta on kyse

Paikalla voi olla sekä yksi alkuperäinen päämies että yksi nykyinen. Jos päämiehien nimistä ei ole tietoa, on attribuutin kohdalle kirjoitettu unknown, ja jos päämiestä ei yksinkertaisesti ole, attribuutin arvo on none. Kaikissa paikoissa on yksi yleisin uskonto.

Family

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 30 merkkiä, not null	Suvun nimi
sigil	Merkkijono, max 100 merkkiä	Suvun vaakunan ominaispiirteet
words	Merkkijono, max 150 merkkiä	Suvun motto
placeoforigin	Merkkijono, max 30 merkkiä, not null	Paikka, josta suku on lähtöisin
nameofcurrentlord	Merkkijono, max 30 merkkiä	Nykyisen päämiehen etunimi

Sukuun voi kuulua monta henkilöä, mutta henkilö voi kuulua vain yhteen sukuun. Suku on voinut olla monen paikan alkuperäisenä ja myös monen paikan nykyisenä päämiehenä, mutta paikoilla on vain yhdet päämiessuvut kerrallaan. Jos suvun lähtöpaikasta ei ole tietoa, on attribuutille placeoforigin asetettu arvo unknown.

Weapon

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 30 merkkiä, not null	Aseelle annettu nimi
kind	Merkkijono, max 30 merkkiä	Aseen tyyppi
nameoforiginalowner	Merkkijono, max 30 merkkiä, not null	Aseen alkuperäisen omistajan etunimi
familynametofooriginalowner	Merkkijono, max 30 merkkiä, not null	Aseen alkuperäisen omistajan sukunimi
material	Merkkijono, max 150 merkkiä	Aseen valmistusmateriaali

Aseella on omistajana joko tarkka henkilö tai pelkkä suku, mutta vain yksi omistaja. Henkilöllä tai suvulla voi olla omistuksessa monta asetta tai ei yhtään.

Comment

Attribuutti	Arvojoukko	Kuvailu
username	Merkkijono, max 10 merkkiä, not null	Kommentin jättäjän käyttäjänimi
comment	Merkkijono, max 500 merkkiä, not null	Käyttäjän kommentti

Ranking

Attribuutti	Arvojoukko	Kuvailu
title	Merkkijono, max 30 merkkiä, not null	Äänestyksen nimi
name	Merkkijono, max 30 merkkiä	Äänestettävän etunimi
familyname	Merkkijono, max 30 merkkiä	Äänestettävän sukunimi
ranking	integer, not null	Äänestettävän tulos

userid

Attribuutti	Arvoluokka	Kuvailu
username	Merkkijono, max 10 merkkiä, not null	Käyttäjänimi
password	Merkkijo, max 15 merkkiä, not null	Salasana
admin	boolean	Tieto siitä, onko käyttäjä admin vai ei

Käyttäjä voi jättää monta kommenttia.

5. Relaatiotietokantakaavio

RELAATIOKAAVIO

Weapon	
PK	name : String
	kind : String
FK	name of original owner : String
FK	familyname of original owner : String
	material : String

Family	
PK, FK	name : String
	sigil : String
	words : String
FK	place of origin : String
	name of current lord : String

Character	
PK	name : String
PK	familyname : String
	alias : String
	Title : String
	special features : String

Place	
PK	name : String
	location : String
FK	family name of original lord : String
FK	familyname of current lord : String
FK	main religion : String
	notable places : String
	type : String

Ranking	
	Title : String
	Name : String
	familyname : String
	Ranking : int

Death	
PK, FK	name : String
PK, FK	familyname : String
	book : int
	Cause of Death : String
FK	name of Slayer : String
FK	familyname of Slayer : String

Comment	
FK	User id : String
	Comment : String

User id	
PK	Username : String
	password : String
	admin : boolean
	points : int

Creature	
PK	Name : String
	Species : String
FK	Name of owner : String
FK	familyname of owner : String
	favorite food : String
	Special features : String

Religion	
PK	Gods : String
	number of Gods : int
	symbol : String
	Servant : String
FK	main area : String

6. Järjestelmän yleisrakenne

Järjestelmän aloitussivun toiminnot löytyvät tiedostosta got.php.

Istunto (sessio) alustetaan tiedostossa yla.php. Yhteys tietokantaan otetaan tiedostolla yhteys.php. Kyselyt tietokannasta toteutetaan tiedostolla kysely.php.

Kirjautumisen onnistumisen tarkastuksen toiminto on tiedostossa kirjaudu.php, jossa myös istunto otetaan käyttöön, jos kirjautuminen onnistuu. Jos käyttäjätunnusta ei ole luotu, käsittelee tilanteen tiedosto failusername.php. faillogin.php käsittelee kirjautumisen, joka epäonnistuu jostain muusta syystä.

Kirjautumisen jälkeisen valikon ohjelmakoodit sisältyvät tiedostoon etusivukirjau.php.

Rekisteröitymisen ulkoasun hoitaa tiedosto rekisterointi.php ja itse rekisteröitymisen eli käyttäjänimen ja salasanan lisäämisen tietokantaan tiedosto lisays.php.

Eri kategorioiden pariin pääsee valitsemalla etusivukirjau:n kohdalla haluamansa vaihtoehdon, ja kategoriasta riippuen järjestelmä ohjaa sivulle, jonka toiminnallisuus löytyy tiedostosta henkilot.php, suku.php, paikka.php, ase.php, religion.php, otus.php tai kuolema.php. Haluamiaan tietokannasta löytyviä tauluja voi valita katseltavaksi haluamillaan ehdoilla ja tämä toiminnallisuus löytyy kategoriasta riippuen tiedostosta [kategoriannimi]tarkka.php (eli esim sukutarkka.php).

Tiedostot, joiden nimessä esiintyy sana "aanestys" sisältävät ohjelmakoodit äänestyksille. "aanestys"-alkuiset tiedostot hoitavat äänestyksen aloituksen ja "lisaaaanestys"-alkuiset tiedostot hoitavat äänestyksen tarkemman toiminnallisuuden eli tarkastavat, löytyykö äänestetty objekti tietokannasta. Tiedostot, joiden nimessä esiintyy sana "tulokset", hoitaa äänestyksen tulosten esittämisen katsojalle.

Kommenttien jätöstä huolehtii tiedosto kommentit.php ja tiedostot kaikkikommentit.php ja minunkommentit.php tulostavat jätetyt kommentit luettavaksi.

paivitus.php huolehtii sivusta, jolta voi valita, mitä kategoriatahtoo päivittää. Tiedostot, joiden nimi on muotoa lisaa[kategoriannimi].php (esim.lisaasuku.php) antaa adminille mahdollisuuden lisätä kantaan täysin uusi objekti. Tiedostot lisays[kategoriannimi].php tarkistavat, että lisattava objekti on ok. Sama pätee tiedostoille paivita[kategoriannimi].php ja paivitus[kategoriannimi].php, jotka toteuttavat kannasta jo löytyvien objektien päivityksen ja päivityksen tarkastuksen. Jos käyttäjä ei ole admin ja yrittää päivittää sivustoa, järjestelmä herjaa samalla tiedostolla paivitus.php. nitwit.php on järjestelmän herjaussivu, joka herjaa, jos käyttäjä yrittää lisätä kantaan jotain epäpätevää.

Quiz-toiminnallisuuden tarjoaminen käyttäjälle tapahtuu tiedostolla testi.php ja tiedosto testitarkastus.php puolestaan tarkistaa testin tuloksen ja tallentaa tulokset käyttäjän tietoihin.

7. Järjestelmän komponentit

[aanestyshenkilo.php](#)

Henkilöäänestysten etusivu. Voi äänestää Fairest Maiden tai Bravest Warrior -äänestyksessä tai katsoa niiden tämän hetkiset tulokset.

[aanestysotus.php](#)

Otusäänestyksen etusivu. Voi äänestää Cutest Creature -äänestyksessä tai katsoa sen tämän hetkiset tulokset.

[aanestyssuku.php](#)

Suvun äänestyksen etusivu. Voi äänestää Most Magnificent Family -äänestyksessä tai katsoa sen tämän hetkiset tulokset.

[addfrontpage.php](#)

Jos valitsee paivitys.php:ssä uuden asian lisäämisen ja valitsee jonkun vaihtoehtoista niin tämä tiedosto ohjaa valintojen mukaan oikealle sivulle.

[ala.php](#)

Sisältää html-koodia, joka pitäisi aina lisätä tiedoston loppuun erikseen.

[ase.php](#)

Ase-tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa aseista tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

[asetarkka.php](#)

Tiedoston ase.php valintojen mukaan tulostaa sivulle halutut tiedot aseista.

[bwtulokset.php](#)

Bravest Warrior-äänestyksen sen hetkiset tulokset.

[cctulokset.php](#)

Cutest Creature -äänestyksen sen hetkiset tulokset.

[etusivukirjau.php](#)

Kirjautumisen jälkeen tuleva etusivu. Mahdollisuudet valita haluaako selata tietoja vai päivittää tietoja. Tietoja voi päivittää vain jos on admin.

[faillogin.php](#)

Jos kirjautu.php:ssä todetaan, että salasana oli väärä niin ohjataan tälle sivulle ja siitä pääsee takaisin etusivulle.

[failusername.php](#)

Jos kirjautu.php:ssä todetaan, että käyttäjää ei ole olemassa niin ohjataan tälle sivulle ja siitä pääsee takaisin etusivulle.

[fntulokset.php](#)

Fairest Maiden-äänestyksen sen hetkiset tulokset.

frontpage.php

Jos valitsee etusivukirjau.php:ssä tietojenselauksen ja jonkun vaihtoehtoista niin tämä tiedosto ohjaa valintojen mukaan oikealle sivulle.

got.php

Sovelluksen etusivu. Sisältää kentät sisäänkirjautumista varten, linkin rekisteröitymissivulle.

henkilot.php

Henkilö-tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa henkilöistä tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

henkilotarkka.php

Tiedoston henkilot.php valintojen mukaan tulostaa sivulle halutut tiedot henkilöistä.

kaikkikommentit.php

Tulostaa kaikki kirjatut kommentit ja keiden ne ovat.

kirjaudu.php

Tarkistaa onko käyttäjä olemassa ja onko salasana oikein kirjoitettu. Ohjaa virheilmoitussivulle, jos käyttäjä ei ole olemassa tai salasana on väärä. Jos kaikki täytetyt tiedot ovat oikein niin ohjaa etusivukirjau.php:seen.

kommentti.php

Pystyy halutessaan jättämään kommentin sivusta. Sisältää linkit listaukseen kaikista kommenteista ja omista kommenteista.

kuolema.php

Kuolleiden henkilöiden tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa kuolleista henkilöistä tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

kuolematarkka.php

Tiedoston kuolema.php valintojen mukaan tulostaa sivulle halutut tiedot kuolleista henkilöistä.

kysely.php

Sisältää funktion, joka hakee tietokannasta tarvittavat rivit taulusta.

lisaaaanestyshenkilo.php

Lisää henkilö-äänestyksien tulokset Ranking-tiluun.

lisaaaanestysotus.php

Lisää otuksen-äänestyksen tulokset Ranking-tiluun.

lisaaaanestyssuku.php

Lisää suku-äänestyksen tulokset Ranking-tiluun.

lisaaase.php

Uuden aseisen lisäämisen tietokantaan etusivu. Sisältää kentät aseisen tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisaahenkilo.php](#)

Uuden henkilön lisäämisen tietokantaan etusivu. Sisältää kentät henkilön tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisaakommentti.php](#)

Lisaa uuden kommentin tietokantaan.

[lisaakuolema.php](#)

Uuden kuolleen henkilön lisäämisen tietokantaan etusivu. Sisältää kentät kuolleen tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisaootus.php](#)

Uuden olennon lisäämisen tietokantaan etusivu. Sisältää kentät olennon tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisaapaikka.php](#)

Uuden paikan lisäämisen tietokantaan etusivu. Sisältää kentät paikan tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisaasuku.php](#)

Uuden suvun lisäämisen tietokantaan etusivu. Sisältää kentät suvun tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisaausko.php](#)

Uuden uskonnon lisäämisen tietokantaan etusivu. Sisältää kentät uskonnon tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[lisays.php](#)

Lisää rekisteröityneen asiakkaan tietokantaan ja siirtää etusivukirjau.php:hen.

[lisaysase.php](#)

Tarkistaa ensin onko lisaase.php:ssä pyydytyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden asean tietokantaan, jos se ei ollut valmiiksi jo siellä.

[lisayshenkilo.php](#)

Tarkistaa ensin onko lisaahenkilo.php:ssä pyydytyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden henkilön tietokantaan, jos se ei ollut valmiiksi jo siellä.

[lisayskuolema.php](#)

Tarkistaa ensin onko lisaakuolema.php:ssä pyydytyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden kuolleen henkilön tietokantaan, jos se ei ollut valmiiksi jo siellä.

[lisaysotus.php](#)

Tarkistaa ensin onko lisaootus.php:ssä pyydytyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden otuksen tietokantaan, jos se ei ollut valmiiksi jo siellä.

[lisayspaikka.php](#)

Tarkistaa ensin onko lisaapaikka.php:ssä pyydettyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden paikan tietokantaan, jos se ei ollut valmiiksi jo siellä.

[lisayssuku.php](#)

Tarkistaa ensin onko lisaasuku.php:ssä pyydettyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden suvun tietokantaan, jos se ei ollut valmiiksi jo siellä.

[lisaysusko.php](#)

Tarkistaa ensin onko lisaausko.php:ssä pyydettyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Lisää uuden uskonnon tietokantaan, jos se ei ollut valmiiksi jo siellä.

[logout.php](#)

Sisältää sovelluksesta uloskirjutumiseen tarvittavat asiat. Ohjaa takaisin etusivulle.

[minunkommentit.php](#)

Listaa käyttäjän omat kommentit;

[mmtulokset.php](#)

Most Magnificent Family-äänestyksen sen hetkiset tulokset.

[nitwit.php](#)

Kertoo, että käyttäjä ei ole täyttänyt tarvittavia kohtia.

[otus.php](#)

Olento-tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa olennoista tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

[otustarkka.php](#)

Tiedoston otus.php valintojen mukaan tulostaa sivulle halutut tiedot olennoista.

[paikka.php](#)

Paikka-tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa paikoista tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

[paikkatarkka.php](#)

Tiedoston paikka.php valintojen mukaan tulostaa sivulle halutut tiedot paikoista.

[paivitaase.php](#)

Jo tietokannassa olevan aseiden päivittäminen tietokantaan etusivu. Sisältää kentät aseiden tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitahenkilo.php](#)

Jo tietokannassa olevan henkilön päivittäminen tietokantaan etusivu. Sisältää kentät henkilön tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitakuolema.php](#)

Jo tietokannassa olevan henkilön kuoleman päivittäminen tietokantaan etusivu. Sisältää kentät kuolleen henkilön tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitaotus.php](#)

Jo tietokannassa olevan olennon päivittäminen tietokantaan etusivu. Sisältää kentät olennon tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitapaikka.php](#)

Jo tietokannassa olevan paikan päivittäminen tietokantaan etusivu. Sisältää kentät paikan tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitasuku.php](#)

Jo tietokannassa olevan suvun päivittäminen tietokantaan etusivu. Sisältää kentät suvun tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitausko.php](#)

Jo tietokannassa olevan uskonnon päivittäminen tietokantaan etusivu. Sisältää kentät uskonnon tietojen täyttämistä varten, tarkennukset että mitkä kohdat on pakko täyttää, tietoa kuinka pitkiä jotkut kohdat voivat olla ja esimerkin.

[paivitys.php](#)

Tarkistaa, onko käyttäjä admin. Jos ei ole admin niin kertoo, että ei pysty päivittämään. Jos on admin niin tervehtii ja saa valita haluaako päivittää vanhoja tietoja vai lisätä uusia ja antaa vaihtoehdot näille.

[paivitysase.php](#)

Tarkistaa ensin onko paivitaase.php:ssä pyydetty pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää asean tietokantaan.

[paivityshenkilo.php](#)

Tarkistaa ensin onko paivitahenkilo.php:ssä pyydetty pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää henkilön tietokantaan.

[paivityskuolema.php](#)

Tarkistaa ensin onko paivitakuolema.php:ssä pyydetty pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää kuolleen henkilön tietokantaan.

[paivitysotus.php](#)

Tarkistaa ensin onko paivitaotus.php:ssä pyydetty pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää olennon tietokantaan.

[paivityspaikka.php](#)

Tarkistaa ensin onko paivitapaikka.php:ssä pyydetty pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää paikan tietokantaan.

[paivityssuku.php](#)

Tarkistaa ensin onko paivitasuku.php:ssä pyydettyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää suvun tietokantaan.

[paivitysusko.php](#)

Tarkistaa ensin onko paivitauskko.php:ssä pyydettyt pakolliset tiedot täytetty, jos ei niin ohjaa nitwit.php:hen. Päivittää uskonnon tietokantaan.

[rekisterointi.php](#)

Rekisteröinnin etusivu. Sisältää kentät rekisteröintiä varten.

[suku.php](#)

Suku-tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa suvuista tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

[sukutarkka.php](#)

Tiedoston suku.php valintojen mukaan tulostaa sivulle halutut tiedot suvuista.

[testi.php](#)

Sisältää 'A song os ice and fire'-monivalintatestin.

[testitarkistus.php](#)

Tarkistaa 'A song of ice and fire' -monivalintatestin pisteet. Päivittää uudet pisteet tietokantaan ja lisää kommentin tietokantaan, jossa kerrotaan paljon henkilö sai pisteitä.

[updatefrontpage.php](#)

Jos valitsee paivitys.php:ssä vanhojen tietojen päivittämisen ja valitsee jonkun vaihtoehtoista niin tämä tiedosto ohjaa valintojen mukaan oikealle sivulle.

[uskonto.php](#)

Uskonto-tietojen selauksen etusivu. Sisältää eri vaihtoehtoja, joista käyttäjä voi valita mitä haluaa uskonnoista tietää. Jotkin kohdat sisältävät kentän, johon käyttäjä joutuu itse tarkentamaan, että mistä haluaa tietää.

[uskontotarkka.php](#)

Tiedoston uskonto.php valintojen mukaan tulostaa sivulle halutut tiedot uskonnoista.

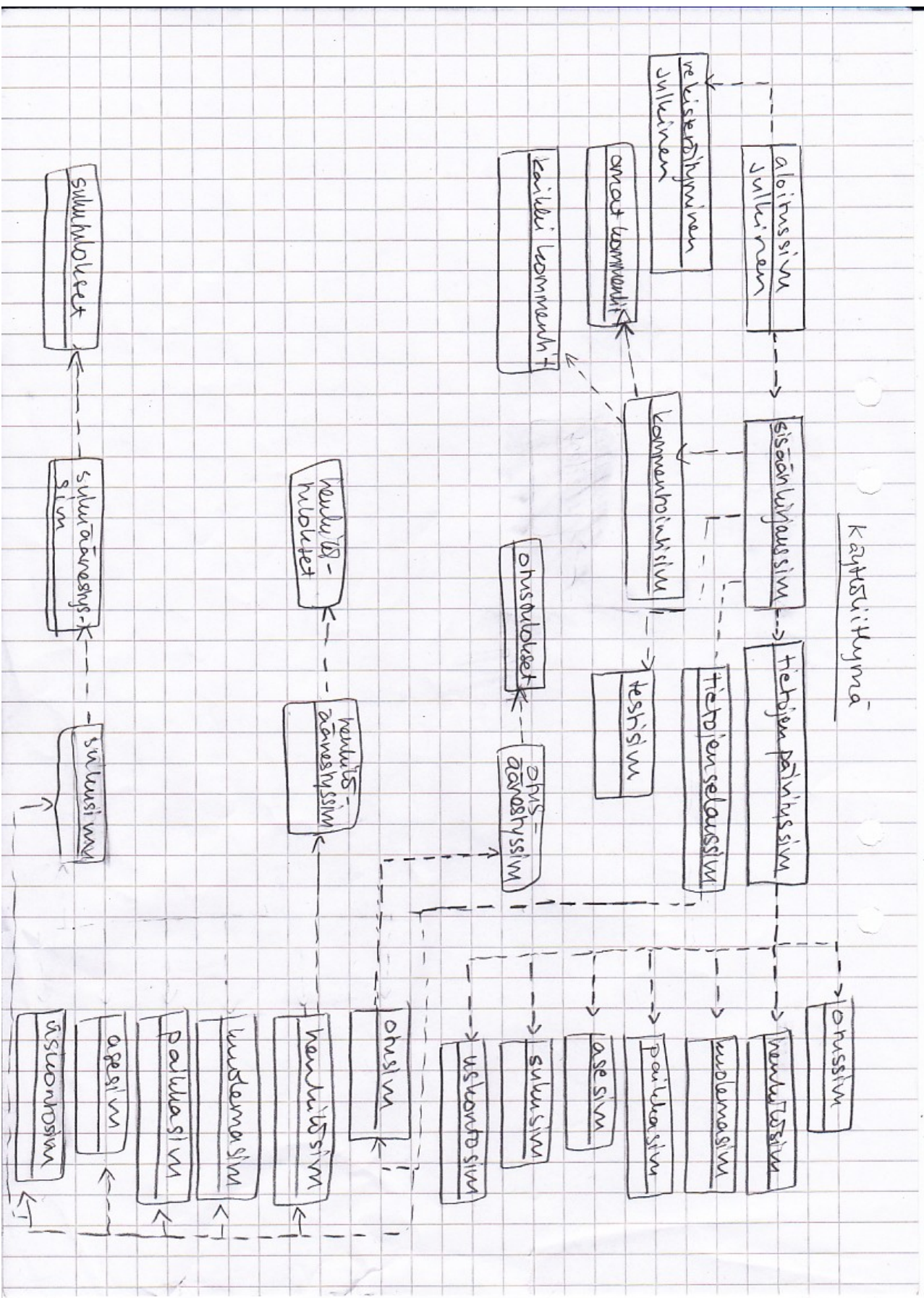
[yhteys.php](#)

Muodostaa yhteyden tietokantaan.

[yla.php](#)

Sisältää html-koodia, joka pitäisi aina lisätä tiedoston alkuun erikseen. Alustaa myös istunnon (sessio).

8. Käyttöliittymä



9. Asennustiedot

Sovelluksen tiedostot sijaitsevat paikassa nimeltään: hpalva@users.cs.helsinki.fi: htdocs/tsoha/
Kaikki tarvittavat tiedostot ovat samassa hakemistossa. Tiedostot on mahdollista saada omaan käyttöön kopioimalla ne sieltä. Mikäli tietokantaa vaihtaisi niin tulisi yhteysasetukset eli käyttäjätunnus ja salasana muuttaa yhteys.php -nimiseen tiedostoon.

10. Käynnistys- ja käyttöohje

Harjoitustyö sijaitsee osoitteessa <http://hpalva.users.cs.helsinki.fi/tsoha/got.php>. Kirjautuakseen sisään voi käyttää admin -tunnuksia username: hpalva ja password: hemuli tai voi itse rekisteröityä.
Vain admin -tunnuksilla pääsee päivittämään ja lisäämään tietoa tietokantaan.

11. LIITE 1: CREATE TABLE -lauseet

```
CREATE TABLE Userid(  
username VARCHAR(10) NOT NULL,  
password VARCHAR(15) NOT NULL,  
admin BOOLEAN,  
Points INTEGER,  
CONSTRAINT pk_Userid UNIQUE KEY (username)  
CONSTRAINT fk_Comment_userid FOREIGN KEY (userid)  
REFERENCES Userid.username  
);
```

```
CREATE TABLE Comment(  
userid VARCHAR(10),  
comment VARCHAR(500),  
CONSTRAINT fk_Userid_username FOREIGN KEY (username)  
REFERENCES userid_username  
);
```

```
CREATE TABLE Character(  
name VARCHAR(30) NOT NULL,  
familyname VARCHAR(30) NOT NULL,  
alias VARCHAR(150),  
title VARCHAR(150),  
specialfeatures VARCHAR(150),  
  
CONSTRAINT pk_Character PRIMARY KEY (name, familyname),  
);
```

```
CREATE TABLE Family(  
name VARCHAR(30) NOT NULL,  
sigil VARCHAR(100),  
words VARCHAR(150),  
placeoforigin VARCHAR(30),  
nameofcurrentlord VARCHAR(30),  
CONSTRAINT pk_Family PRIMARY KEY (name),  
CONSTRAINT fk_Family_placeoforigin FOREIGN KEY (placeoforigin)  
REFERENCES Place ON DELETE SET NULL,  
);
```

```
CREATE TABLE Weapon(  
name VARCHAR(30) NOT NULL,  
kind VARCHAR(30),  
nameoforiginalowner VARCHAR(30),  
familynameoforiginalowner VARCHAR(30),  
material VARCHAR(30)  
CONSTRAINT pk_Weapon PRIMARY KEY (name),  
CONSTRAINT fk_Weapon_familynameoforiginalowner FOREIGN KEY  
(familynameoforiginalowner)  
REFERENCES Family  
);
```

```
CREATE TABLE Religion(  
  gods VARCHAR(30) NOT NULL,  
  numberofgods INTEGER NOT NULL,  
  symbol VARCHAR(30),  
  servant VARCHAR(30),  
  mainarea VARCHAR(50),  
  CONSTRAINT pk_Religion PRIMARY KEY (gods),  
  CONSTRAINT fk_Religion_mainarea FOREIGN KEY (mainarea)  
  REFERENCES Place,  
);
```

```
CREATE TABLE Creature(  
  name VARCHAR(30) NOT NULL,  
  species VARCHAR(30),  
  nameofowner VARCHAR(30),  
  familynamEOFowner VARCHAR(30),  
  favoritefood VARCHAR(30),  
  specialfeatures VARCHAR(100),  
  CONSTRAINT pk_Creature PRIMARY KEY (name),  
  CONSTRAINT fk_Creature_nameofowner FOREIGN KEY (nameofowner)  
  REFERENCES Character,  
  CONSTRAINT fk_Creature_familynamEOFowner FOREIGN KEY (familynamEOFowner)  
  REFERENCES Character  
);
```

```
CREATE TABLE Death(  
  name VARCHAR(30) NOT NULL,  
  familyname VARCHAR(30) NOT NULL,  
  book INTEGER,  
  causeofdeath VARCHAR(50) NOT NULL,  
  nameoslayer VARCHAR(30),  
  familynamEOFslayer VARCHAR(30),  
  CONSTRAINT pk_Death PRIMARY KEY (name, familyname),  
  CONSTRAINT fk_Death_name FOREIGN KEY (name)  
  REFERENCES Character,  
  CONSTRAINT fk_Death_familyname FOREIGN KEY (familyname)  
  REFERENCES Character  
);
```

```
CREATE TABLE Ranking(  
  title VARCHAR(100) NOT NULL,  
  name VARCHAR(20) NOT NULL,  
  familyname VARCHAR(20) NOT NULL,  
  ranking INTEGER,  
);
```

```
CREATE TABLE Place(  
  name VARCHAR(50) NOT NULL,  
  location VARCHAR(50),  
  familynamEOForiginalowner VARCHAR(30),
```

```
familynameofcurrentowner VARCHAR(30),
mainreligion VARCHAR(30),
notableplaces VARCHAR(200),
type VARCHAR(30),
CONSTRAINT pk_Place PRIMARY KEY (name),
CONSTRAINT fk_Place_familynameoforiginallord FOREIGN KEY (familynameoforiginallord)
REFERENCES Family,
CONSTRAINT fk_Place_familynameofcurrentlord FOREIGN KEY (familynameofcurrentlord)
REFERENCES Family
CONSTRAINT fk_Place_mainreligion FOREIGN KEY (mainreligion)
REFERENCES Religion
);
```