

1. Johdanto

Gourmet-arkisto

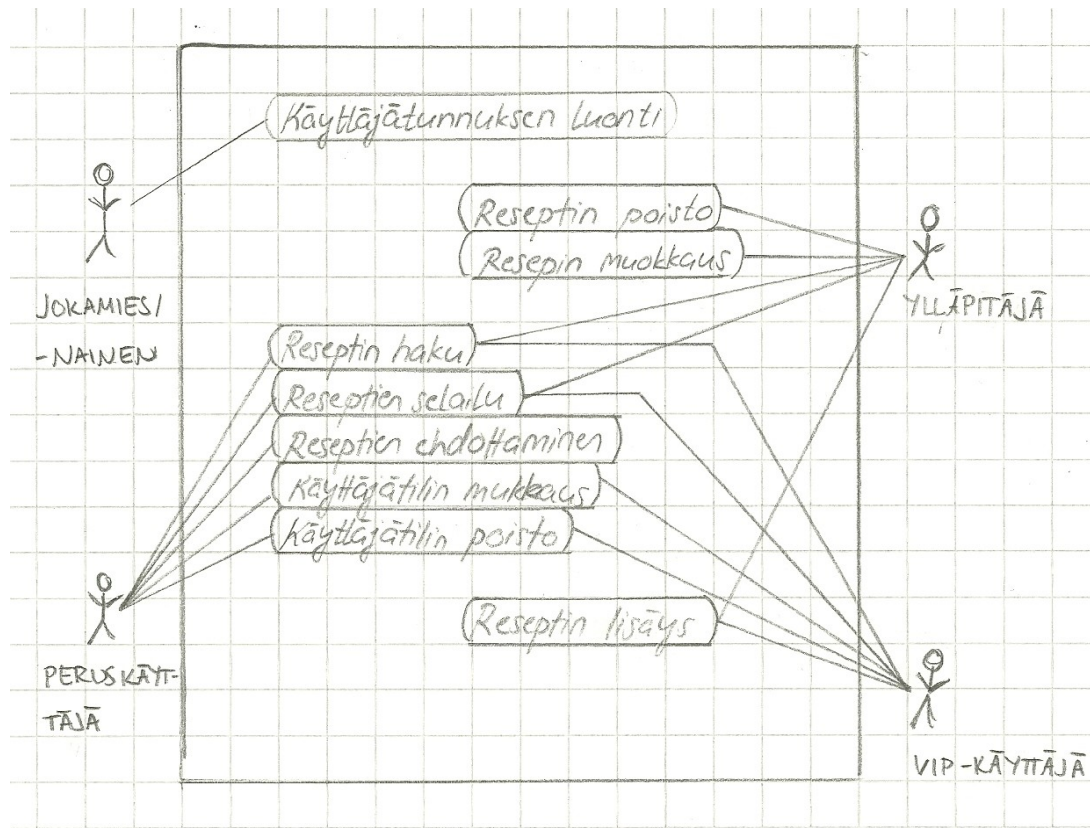
Gourmet-arkisto toimii Drinkkiarkiston tavoin, paitsi että sillä haetaan ruokareseptejä. Aluksi käyttäjän on kirjauduttava sivulle. Mikäli käyttäjällä ei vielä ole tunnusta, hän voi sellaisen luoda. Kirjautumisen jälkeen hän voi hakea reseptejä reseptin nimeen liittyvällä hakusanalla tai millä tahansa ohjeessa käytetyllä hakusanalla. Hän voi tietenkin saada useita osumia eli reseptejä samalla hakusanalla. Reseptien nimiin voi kuka tahansa käyttäjä lisätä vaihtoehtoisia nimiä, jolloin resepti löytyy useilla eri nimivaihtoehdoilla (esimerkiksi hernekeitto, hernesoppa, hernerokka). Sivulla annetaan myös lista kaikista resepteistä, defaultina lisäysjärjestyksessä (tuorein resepti ensin). Käyttäjä voi halutessaan hakea myös reseptit pääraaka-aineen tai ruokalajin mukaan (pääruoka, alkupala jne).

Käyttäjä voi koska tahansa muuttaa tietojaan tai poistaa tilinsä. Hän voi myös ehdottaa ylläpitäjälle reseptejä täyttämällä nettilomakkeen. Ylläpitäjän hyväksynnän jälkeen ehdotettu resepti lisätään muihin resepteihin ja se löytyy hauissa. Ylläpitäjä voi myös antaa suoran reseptien lisäysoikeuden valitsemilleen vip-käyttäjille.

Gourmet-arkisto toteutetaan laitoksen users-palvelimella Tomcat-palvelimen alla ja toteutuu siis sivustolle <http://t-vverkhov.users.cs.helsinki.fi/Gourmet-arkisto/> . Alustajärjestelmän on tuettava Java-ohjelmointikieltä. Tietokannaksi on valittu Postgresql, ja ohjelmisto vaatii sen käytön.

2. Järjestelmän kuvaus

Käyttötapauskaavio



3. Käyttötapaukset

Jokamiehen/-naisen käyttötapaukset

- *Käyttäjätunnuksen luonti*: Käytännössä rekisteröidytään peruskäyttäjäksi, minkä jälkeen saadaan oikeus nähdä, hakea ja ehdottaa reseptejä. Jokamies/-nainen luo itselleen tunnuksen ja salasanan.

Peruskäyttäjän käyttötapaukset

- *Reseptin haku*: Käyttäjä voi hakea reseptejä erilaisilla hakusanoilla. Hän voi löytää useamman reseptin yhdellä hakusanalla tai päätyä samaan reseptiin eri hakusanoilla.
- *Reseptien selailu*: Käyttäjä näkee listan kaikista resepteistä, joista hän voi valita mieleisensä. Hän voi myös järjestää listan ruoka-aineen tai ruokalajin mukaan.
- *Reseptin ehdottaminen*: Käyttäjä täyttää lomakkeen. Tiedot eivät kuitenkaan löydy reseptilistauksesta tai hauilla ennen kuin ylläpitäjä hyväksyy tämän.

- *Käyttäjätilin muokkaus*: Käyttäjä voi halutessaan muuttaa tietojaan.
- *Käyttäjätilin poisto*: Jos käyttäjä ei enää halua Gourmet-palvelua, hän voi poistaa tunnuksensa.

Vip-käyttäjän käyttötapaukset:

Samat käyttötapaukset kuin peruskäyttäjällä, paitsi Reseptin ehdottamisen sijasta reseptin lisäys.

- *Reseptin lisäys*: Vip-käyttäjä voi suoraan lisätä reseptejä ilman erillistä hyväksyntää.

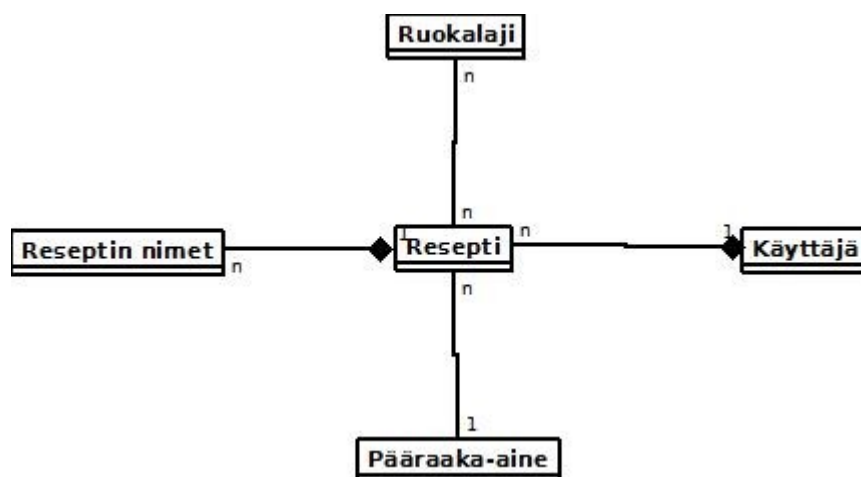
Ylläpitäjä:

Samat käyttötapaukset kuin vip-käyttäjällä, ja lisäksi:

- *Reseptin poisto*: Ylläpitäjä voi halutessaan poistaa reseptin tietokannasta.
- *Reseptin muokkaus*: Ylläpitäjä voi muokata reseptiä esimerkiksi löytäessään paremman variaation
- *Reseptiehdotusten hyväksyminen*: Ylläpitäjä hyväksyy peruskäyttäjän ehdottamia reseptejä ja liittää ne muiden käyttäjille näkyviksi.
- *Muiden käyttäjien oikeuksien hallinnointi*: Ylläpitäjä voi antaa muille käyttäjille vip-oikeuksia, jolloin nämä voivat lisätä reseptejä suoraan.
- *Käyttäjien poisto*: Ylläpitäjä voi myös poistaa käyttäjiä Gourmet-arkistosta.

4. Järjestelmän tietosisältö

Käsitekaavio



Tietokohde: Ruokalaji

Attribuutti	Arvojoukko	Kuvaus
Id	Int, autoincrement	Ruokalajin tunnus
Nimi	Merkkijono	Ruokalajin nimi

Tietokohde: Pääraaka-aine

Attribuutti	Arvojoukko	Kuvaus
Id	Int, autoincrement	Pääraaka-aineen tunnus
Nimi	Merkkijono	Pääraaka-aineluokan nimi

Tietokohde: Käyttäjä

Attribuutti	Arvojoukko	Kuvaus
Id	Int, autoincrement	Käyttäjän tunnus
Tunnus	Merkkijono	Käyttäjän itsevalitsema tunnus, jota käytetään kirjautuessa
Salasana	Merkkijono	Käyttäjän valitsema salasana
Admin-oikeudet	Boolean	Määrittelee käyttäjälle maksimaaliset eli ylläpito-oikeudet
Lisäysoikeus	Boolean	Määrittää käyttäjälle oikeuden lisätä reseptejä

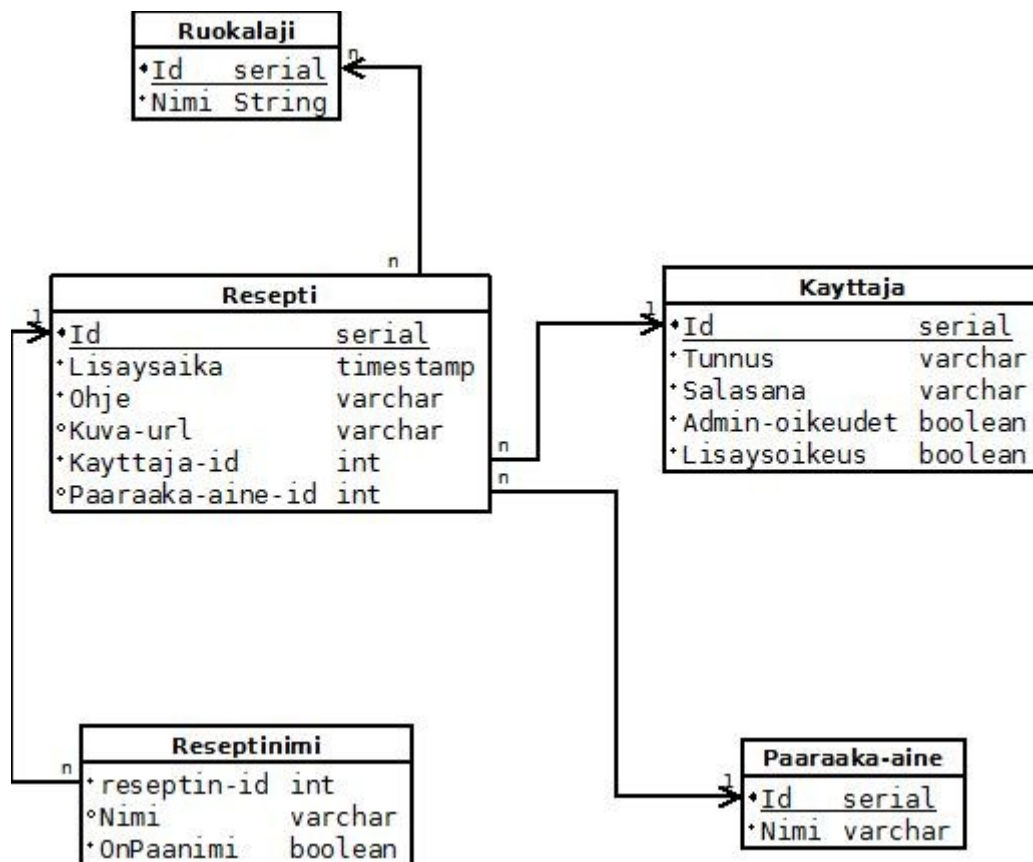
Tietokohde: Resepti

Attribuutti	Arvojoukko	Kuvaus
Id	Int, autoincrement	Reseptin tunnus
Lisäysaika	Timestamp	Reseptin lisäyshetki
Ohje	Merkkijono	Reseptin ohjeet
Kuva-url	Merkkijono, vapaaehtoinen	Url ruokakuvaan
Käyttäjä-id	Int	Viittaus reseptin tehneeseen käyttäjään
Pääraaka-aine-id	Int, vapaaehtoinen	Viittaus reseptin pääraaka-aineluokkaan

Tietokohde: Reseptin nimet

Attribuutti	Arvojoukko	Kuvaus
Reseptin id	Int	Reseptin tunnus
Nimi	Merkkijono	Reseptin nimi
OnPäänimi	boolean	Määrittelee esiintyykö nimi listauksessa

5. Relaatietietokantakaavio



6. Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Näkymät löytyvät Web-hakemiston WEB-INF-hakemistosta. Source-packagesta löytyvät taas mallit ja kontrollerit: tietokantaan liittyvät mallit database-hakemistosta, muut mallit models-hakemistosta, filterit filters-hakemistosta, utilityt utilities-hakemistosta sekä kontrollerit servlets-hakemistosta. Käytetyt apukirjastot on sijoitettu Libraries-hakemistoon.

7. Järjestelmän komponentit

Servletit (kontrollerit)

KayttajanlistausServlet.java

Käyttäjien listaamiseen tarkoitettu sivu. Sivun on tarkoitus näkyä vain ylläpitäjälle, joka voi selata käyttäjiä ja hakea niitä tunnuksen tai sähköpostin mukaan. Sivulta voi myös lähettää käyttäjille sähköpostia, antaa heille vip-oikeudet tai poistaa ne.

KayttajanLisaysServlet.java

Käyttäjän lisäämiseen tarkoitettu sivu. Sisältää lomakkeen, johon uuden käyttäjän on syötettävä uniikki tunnus, vapaaehtoinen sähköposti ja salasana.

KayttajanTiedotServlet.java

Käyttäjälle itselleen tarkoitettu sivu, josta näkyvät omat tiedot, eli tunnus ja sähköposti. Sivulta käyttäjä voi siirtyä käyttäjätilin muokkaussivulle tai poistaa tilinsä.

KayttajanTietojenMuokkausServlet.java

Sisältää lomakkeen, jonka avulla käyttäjä voi vaihtaa tunnuksensa, sähköpostinsa ja salasana.

KirjautumisServlet.java

Sovelluksen etusivu. Tarjoaa tarvittavat kentät sisäänkirjautumiseen ja linkin uuden käyttäjätilin luomiseen.

ReseptinListausServlet.java

Tällä sivulla tarjotaan hakulomake, jossa on sanahaku ja ruokalaji- sekä pääraaka-aine-filtteröinti. Aluksi sivulla listataan kaikki reseptit, haun jälkeen hakutulokset. Admin näkee samalta sivulta myös reseptiehdotukset.

ReseptinLisaysServlet.java

Reseptin ehdotukseen, lisäykseen ja muokkaukseen tarkoitettu sivu. Sivulla tarjotaan kentät reseptin nimelle, kuva-urlille, ruokalajille, pääraaka-aineelle ja varsinaiselle ohjeelle.

ReseptinTiedotServlet.java

Yksittäisen reseptin esittelysivu, jossa näkyvät kaikki reseptin tiedot. Admin-oikeudet omaavalle käyttäjälle sivulla on myös linkit reseptin muokkaukseen ja poistoon.

VikaTilaServlet.java

Kyseinen kontrolleri pitää huolta siitä, että virheen sattuessa käyttäjä ohjataan kirjautumissivulle ja ilmoitetaan virheestä virheboksissa.

YleisServlet.java

Yleistä kontrolleri-logiikkaa, kuten sivujen avaaminen ja sivuille siirtäminen sekä käyttäjän oikeuksien tarkistaminen ja virheviestien näyttäminen.

Mallit

Kayttaja.java

Käyttäjäolio, joka pitää sisällään kaikki käyttäjälle oleelliset tiedot.

PaaraakaAine.java

Pääraaka-aineolio, johon sisältyy id ja pääraaka-aineen nimi.

Resepti.java

Resepti-olio, sisältää reseptin kannalta oleelliset tiedot.

ReseptinNimi.java

Reseptin nimi-olio, joka sisältää reseptin id:n, nimen ja tiedon siitä, onko nimi päänimi vai ei.

Ruokalaji.java

Ruokalajiolio, sisältää ruokalajin id:n ja nimen.

Tietokantamallit

Tietokanta.java

Yleiset tietokantatoimintaan liittyvät asiat, kuten tietokannan avaus ja sulkeminen

TkKayttaja.java

Tässä luokassa käsitellään kaikki käyttäjään liittyvät tietokantaoperaatiot, kuten lisääminen, poisto, muokkaus ja erilaiset haut.

TkResepti.java

Tässä luokassa käsitellään kaikki reseptin ja siihen liittyvien taulujen tietokantaoperaatiot, kuten reseptin lisääminen, poisto, muokkaus ja haut, sekä pääraaka-aineen, ruokalajin ja nimien haut.

Virheet

GourmetException.java

Ekstendoi RuntimeExceptionin, antaa käyttäjälle GourmetExceptioneita.

Filtterit

KirjautumisFilter.java

Filtterin on tarkoitus tarkistaa aina, onko käyttäjä kirjautunut ja onko tällä siis oikeudet /arkisto-etuliitteillä merkittyihin urleihin.

Utilityt

SalausUtility.java

Tämän tarkoitus on salata käyttäjän salasana, ettei sitä voisi nähdä tietokannasta suoraan.

Näkymät(jsp)

Käyttäjänäkymät

kayttajalistaus.jsp

KayttajaListaServlet.javan jsp-puoli.

Kayttajanlisays.jsp

KayttajanLisaysServlet.javan jsp.

kayttajantiedot.jsp

KayttajanTiedotServlet.javan jsp.

kayttajantietojenmuokkaus.jsp

KayttajanTietojenMuokkaus.javan jsp.

kirjautuminen.jsp

KirjautumisServlet.javan jsp.

Reseptinäkymät

reseptilistaus.jsp

ReseptiListaServlet.javan jsp.

reseptinlisays.jsp

ReseptinLisaysServlet.javan jsp.

reseptintiedot.jsp

ReseptinTiedotServlet.javan jsp.

Tags

pohja.tag

Sivujen yleinen header, näkyy kaikilla /arkisto-etuliitteisillä urleilla.

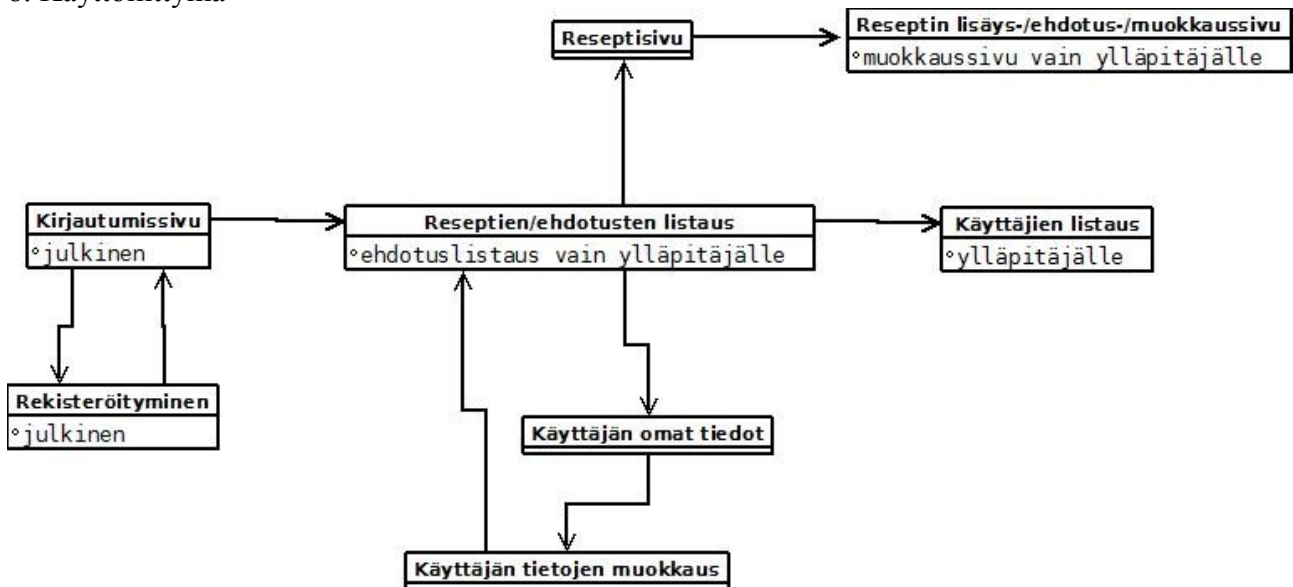
Kuvat

Pohja.tagin kuvat löytyvät omasta kansiostaan webin alta.

Kirjastot

Libraries-kansiosta löytyy jstl-kirjasto, joista käytin ”t” ja ”c”-komponentteja.

8. Käyttöliittymä



Sovelluksessa on ylhäällä navigaatiopalkki, josta pääsee reseptilistaukseen, reseptin lisäykseen/ehdotukseen, käyttäjän omiin tietoihin sekä admin käyttäjien listaukseen. Siellä on myös uloskirjauspainike ja reseptien pikahaku.

9. Asennustiedot

Sovellusta varten on asennettava PostgreSQL-tietokanta, jonka saa ladattua web-osoitteesta <http://www.postgresql.org/download/>. Tietokannan asetukset on määriteltävä ohjelmiston web/META-INF-hakemistosta löytyvässä context.xml-tiedostossa. Resource name -kohdassa määritellään käytettävä tietokanta, user-kohdassa tunnus ja password-kohdassa salasana.

Sovellus vaatii myös Tomcat-palvelimen, joka löytyy web-osoitteesta

<http://tomcat.apache.org/download-60.cgi>.

10. Käynnistys-/käyttöohjeet

Harjoitustyö on asennettuna osoitteessa <http://t-vverkhov.users.cs.helsinki.fi/Gourmet-arkisto/> ja tällä hetkellä sisään pääsee tunnuksella admin ja salasanalla admin. Sivuilla voi luoda myös oman tunnuksen.

11. Testaus, tunnetut bugit ja puutteet & jatkokehitysideat

Koska sovelluksessa ei ole oikeastaan testattavaa logiikkaa, on sovellusta testattu lähinnä kokeilemalla. Olen kokeillut itse ja pyytänyt muita ihmisiä kokeilemaan. Tällä tavalla on löytynyt useita bugeja ja näitä on yritetty parhaan mukaan korjailla.

Jatkokehitysideoita on kuitenkin paljon. Esimerkiksi kuvien lataamisen mahdollisuus reseptiin suoraan olisi hyvä lisäominaisuus. Samoin mahdollisuus tarkastella tietyn käyttäjän tekemiä reseptejä listana. Käyttäjä voisi myös nähdä omat reseptinsä omalla sivulla ja mahdollisesti muokata omia reseptejään.

Käyttöliittymää voisi tietysti hioa loputtomiin, valitettavasti toistaiseksi tiedot käyttöliittymäkoodauksesta ovat vain aika vähäiset.

12. Omat kokemukset

Tietokantasovelluksessa helppoa oli ehkä se, ettei tarvinnut kehittää monimutkaisia algoritmeja, vaan kyse oli enemmän ohjelmahierarkian rakentamisesta ja logiikan jakamisesta järkevästi eri blokkien kesken.

Vaikeaa taas oli se, että sovelluksessa tuli niin paljon uutta, että ajankäytöstä tuli ongelma. Itse en ole html:ään tai css:ään koskaan koskenut, joten web-sovelluspuoli toi paljon lisätyötä tietokantaohjelmoinnin rinnalle.

Toisaalta tämä teki kurssista todella opettavaisen. Oppi todella paljon paitsi tietokantaohjelmoinnista, myös web-sovellusten rakenteesta, Javan Servleteistä, html:ää, css:ää ja jstl:n käyttöä. Ehkä kurssin nimi kannattaisikin vaihtaa, ”Tietokantapohjainen web-sovellusohjelmointi” tai vastaava voisi olla kuvaavampi.

13. Liitteet

Liitteet löytyvät muista projektin kansioista GitHubissa. Sieltä löytyvät ohjelmakoodit src-hakemistosta ja create table -lauseet sql-hakemistosta. Tehtäväkuvaus löytyy osoitteesta <http://advancedkittenry.github.io/aiheet/Drinkkiarkisto.html> .