

# Tietokantasovellus (periodi I) – Vituttaako.com

Niki Ahlskog

# Sisällys

1. Johdanto

2. Yleiskuva järjestelmästä

# 1 Johdanto

Työn aihe on vituttaako.com verkkosovellus. Järjestelmän tarkoitus on auttaa ihmisiä avautumaan ja purkamaan ahdistustaan digitaalisesti. Toimintaidea on hieman samanlainen, kuin metrolehden tekstiviestipalsta. Tällä kertaa vain digitaalisessa muodossa. Järjestelmän tarkoituksena on auttaa ihmisiä kertomaan oma mielipiteensä anonyymisti.

Sovellukseen on mahdollista tallentaa tarinoita tekstimuodossa ja lisäksi kommentoida tarinoita. Kun käyttäjä julkaisee tarinan on sen jälkeen mahdollista osallistua keskusteluun tai jakaa kyseinen tarina sosiaalisessa mediassa. Järjestelmän tarkoitus on pääasiassa hauskuuttaa lukijoita, sekä pistää myös miettimään, miksi jotkut ihmiset ovat asioista sitä mieltä, että eivät uskalla kirjoittaa asiasta oikealla nimellään.

Työ toteutetaan laitoksen users-palvelimella ja kehitysyötä tullaan tekemään muutamalla eri tavalla. Vaihtoehto a) kehitys lokaalissa ympäristössä, josta julkaistaan versio "tuotantopalvelimelle" kerran viikossa, tai useammin. Vaihtoehto b) kehitys suoraan "tuotannossa". Vaihtoehto A on tällä hetkellä parempi vaihtoehto ja B:tä käytetään vain tarvittaessa.

Palvelimen, tai lokaalin ympäristön pitää pystyä kääntämään PHP:tä. Työ tullaan pääosin toteuttamaan PHP-ohjelmointikielellä. Palvelimella tulee olla asennettuna myös apache2. Alkuun sovellusta tullaan kokeilemaan Tsoha-Bootstrap frameworkilla, mikäli sen käyttö todetaan hyväksi kehitystä jatketaan. Framework hyödyntää myös Twig ja Slim kirjastoja, jotka asennetaan composerilla. Composer on pakettinhallintatyökalu PHP:lle. Käyttöjärjestelmänä kehitystyössä tullaan käyttämään OSX:ää ja Ubuntua, jota ajetaan virtualboxilla Windowsissa.

Sovelluksen tietokantana tullaan käyttämään PostgreSQL relaatiotietokantaa. Tietokanta tulee olla asennettuna ja tietokantayhteys muodostettuna sovelluksessa. PostgreSQL on avointa lähdekoodia. Tietokantana käytetään koulun users palvelimelle asennettua postgreä.

## 2 Yleiskuva järjestelmästä



### Käyttäjärühmät

#### Anonyymi käyttäjä

- Anonyymi käyttäjä on sivustolle satunnaisesti tai vaikka vakituisesti tuleva kävijä, joka ei ole rekisteröitynyt.
- Anonyymi käyttäjä voi selailla tarinoita tai jakaa niitä sosiaalisessa mediassa. Kaikki muut sidosryhmät kuuluvat myös tähän sidosryhmään.

#### Rekisteröitynyt käyttäjä

- Rekisteröitynyt käyttäjä voi editoida vanhaa tarinaa
- Poistaa tarinan
- Kommentoida tarinaa nimimerkillä
- Poistaa oman kommentin
- Listata omat tarinat profiilisivulla
- 

#### Admin

- Admin on sovelluksen ylläpitäjä. Admin vahtii että sivustolla ei julkaista “hirveää törkyä”
- Admin voi sensuroida tekstejä
- Poistaa kommentteja
- Selailla ja etsiä tarinoita

## Käyttötapaukset

### Anonyymi käyttäjä

1. Tarinan julkaiseminen  
Anonyymi käyttäjä voi julkaista tarinan heti etusivulla nimettömästi ilman sovellukseen kirjautumista
2. Tarinan jakaminen  
Anonyymi käyttäjä voi jakaa tarinan sosiaalisen median painikkeilla
3. Tarinoiden lukeminen  
Anonyymi käyttäjä voi lukea ja selailla tarinoita sivustolta
4. Tarinoiden etsiminen  
Anonyymi käyttäjä voi etsiä tarinoita hakusanoilla tai filttäreillä
5. Tarinoiden äänestäminen  
Anonyymi käyttäjä voi äänestää onko tarinan kanssa samaa vai eri mieltä.
6. Rekisteröityminen  
Anonyymi käyttäjä voi rekisteröityä sovellukseen

### Rekisteröitynyt käyttäjä

Rekisteröitynyt käyttäjä voi tehdä samat kuin anonyymi käyttäjä, lisäksi seuraavat:

1. Tarinan julkaiseminen  
Rekisteröitynyt käyttäjä voi julkaista tarinoita nimimerkillään.
2. Tarinan editoiminen  
Käyttäjä voi editoida myöhemmin tarinaa
3. Tarinan poistaminen  
Käyttäjä voi poistaa tarinan tullessaan katumapäälle
4. Tarinan kommentointi  
Käyttäjä voi kommentoida omaa tai muiden tarinaa omalla nimimerkillään
5. Tarinoiden listaaminen  
Käyttäjä voi listata kaikki omat tarinat kätevästi omalta profiilisivultaan. Profiilisivulta näkee lisäksi statistiikat omista tarinoista
6. Kommentin muokkaaminen  
Käyttäjä voi muokata omaa kommenttiaan
7. Kommentin poistaminen  
Käyttäjä voi poistaa julkaisemansa kommentin

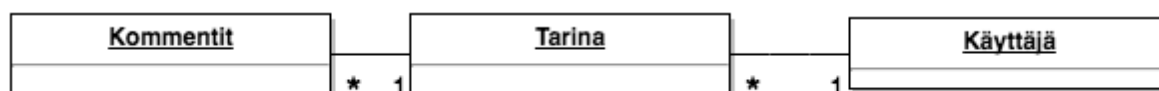
8. Sisäänkirjautuminen  
Rekisteröitynyt käyttäjä voi kirjautua sisään sovellukseen, jolloin käyttöön tulee lisää toimintoja
9. Uloskirjautuminen  
Rekisteröitynyt käyttäjä voi kirjautua ulos sovelluksesta

## Admin käyttäjä

Admin voi tehdä samat asiat kuin muut käyttäjät, mutta lisäksi:

1. Käyttäjän poistaminen  
Admin voi poistaa sivustolta epäasiallisen käyttäjän.
2. Kommenttien poistaminen  
Admin voi poistaa kenen tahansa käyttäjän epäasiallisia kommentteja sivustolta.
3. Tarinan poistaminen  
Admin voi poistaa minkä tahansa epäasiallisen tarinan sivustolta

## Järjestelmän tietosisältö



Kommentit = comments		
Attribuutti	Arvojoukko	Kuvailu
id	Int, autoindex, Primary Key	Kommenttien yksilöivä tunniste, joka kasvaa automaattisesti. Primary Key
comment	varchar	Tarinaan liittyvä kommentti. Merkkijono, tekstiä.
createdby	varchar, Foreign key	Henkilö tai nimimerkki

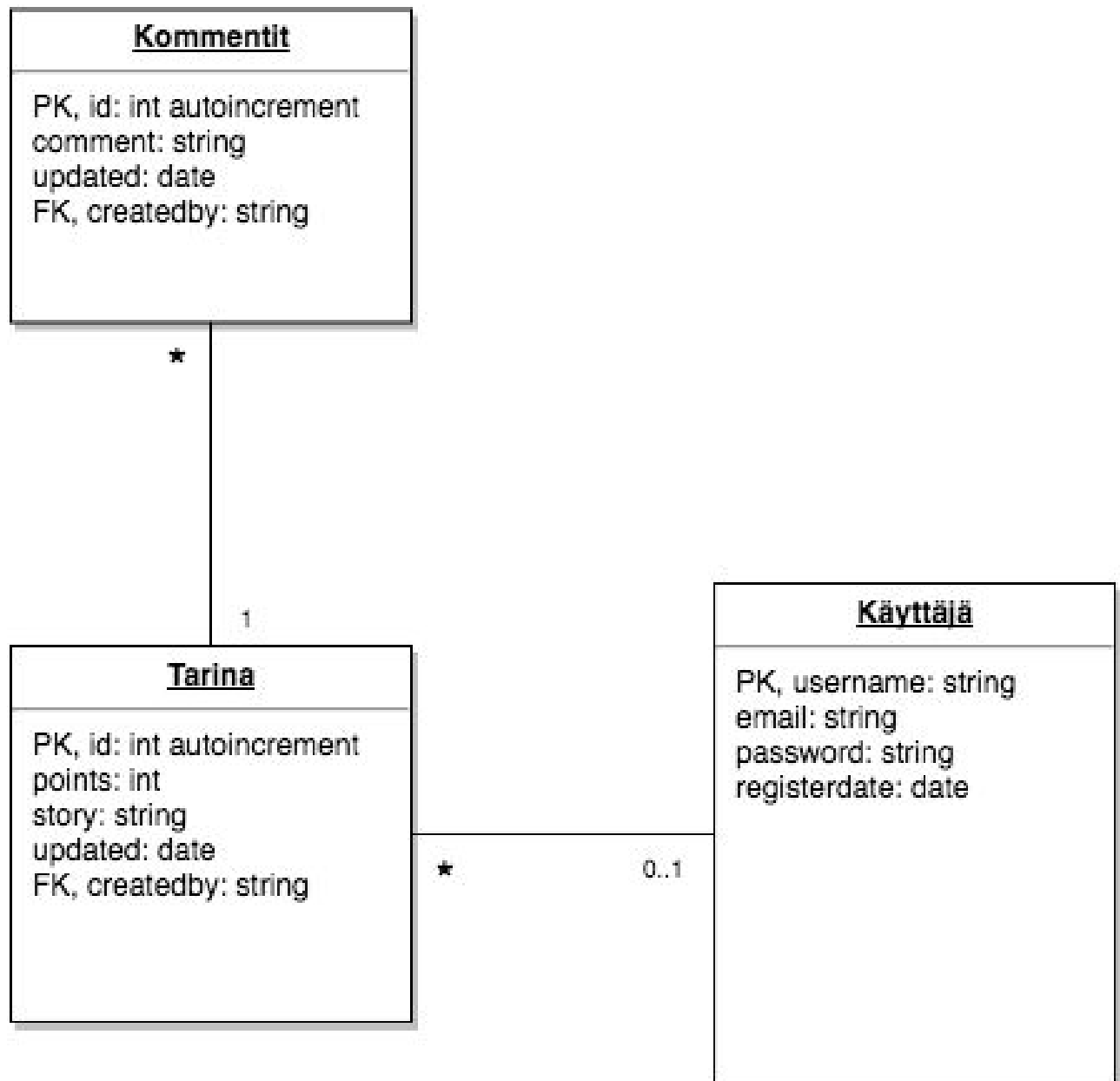
		joka tarinaa on kommentoinut
updated	timestamp	Päivämäärä milloin kommentti on lähetetty ja milloin sitä on päivitetty
story_id	INTEGER	ID joka viittaa tarinaan. Kommentteja voidaan hakea tietylle tarinalle.

Tarina = stories		
Attribuutti	Arvojoukko	Kuvailu
id	Int, autoindex, Primary Key	Tarinan yksilöivä tunniste, joka kasvaa automaattisesti. Primary Key
points	int	Tarinan pistemäärä joka tulee käyttäjien klikkauksista. Pistemäärä voi myös olla negatiivinen.
story	varchar	Varsinainen tarina, merkkijono.
updated	timestamp	Päivämäärä milloin tarina on lähetetty ja milloin sitä on päivitetty
createdby	varchar, Foreign key	Henkilö tai nimimerkki joka tarinan on lähettänyt

Käyttäjä = appuser		
Attribuutti	Arvojoukko	Kuvailu
username	varchar, Primary Key	Nickname tai käyttäjänimi sovelluksessa kommentointiin ja julkaisuun
email	varchar	Käyttäjän sähköpostiosoite
password	varchar	Käyttäjän salasana
registerdate	timestamp	Päivämäärä milloin käyttäjä on rekisteröitynyt sovellukseen
isadmin	boolean	Kertoo onko käyttäjällä sovellukseen admin oikeudet.

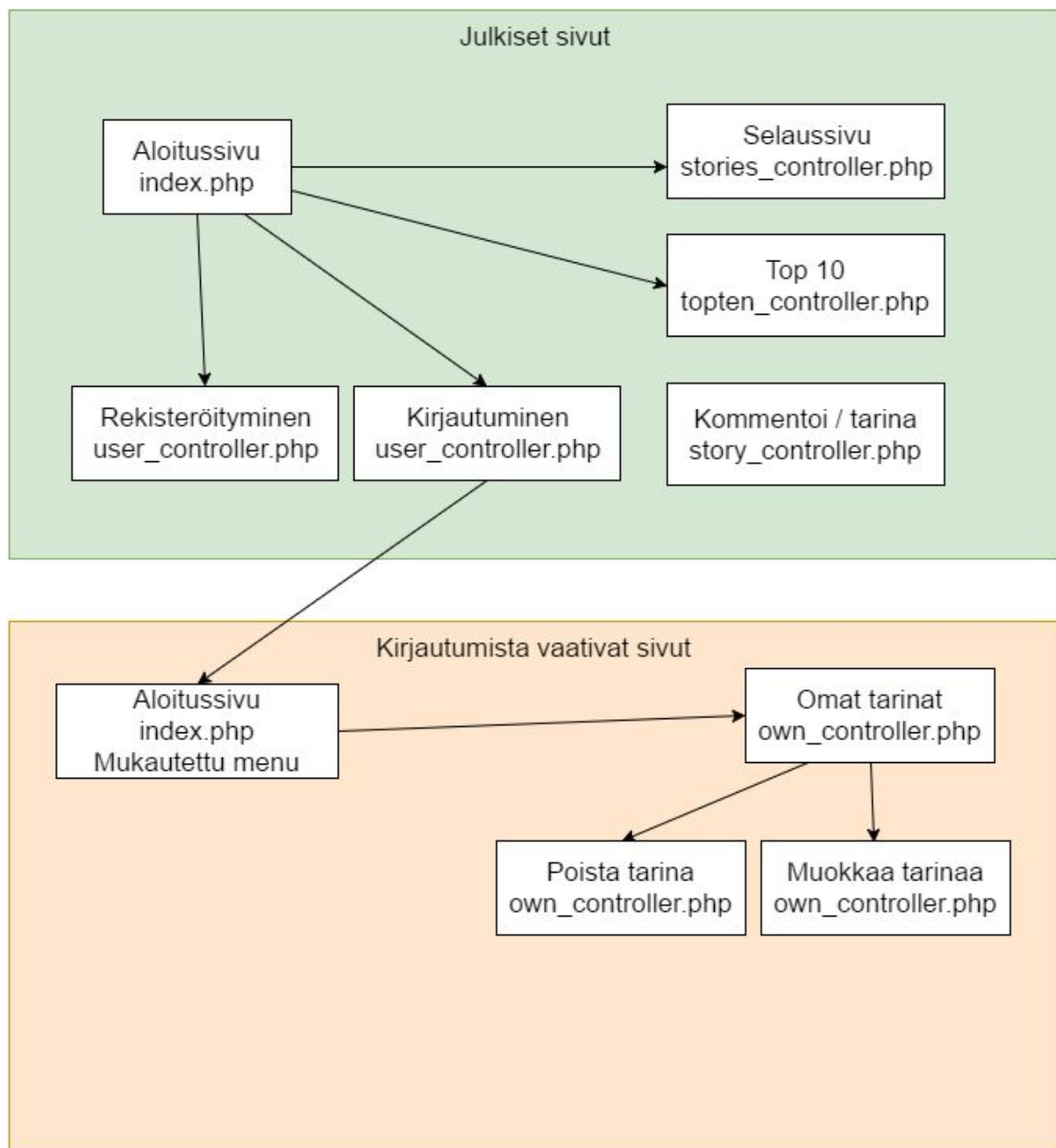


# Relaatiotietokantakaavio



# Järjestelmän yleisrakenne

Tietokantasovellus noudattaa MVC mallia ja sen pohjana on käytetty Tsoha-Bootstrap github forkkia. Kontrollerit, näkymät ja mallit on sijoitettu niitä kuvaaviin hakemistoihin app kansion alle, controllers, models ja views kansioihin. Sovelluksen asetukset löytyvät config hakemiston alta ja assets kansion sisällä on css, sekä javascript tiedostot. Kaikki tiedostot on kirjoitettu pienellä ja jos nimi on kaksiosainen, ne on yhdistetty alaviivalla \_. Sovelluksen julkaisu palvelimelle tapahtuu shell skriptien avulla, jotka löytyvät projektin juuresta.



# Asennustiedostot

Sovellus on asennettu koulun users palvelimelle, jossa on valmiiksi asennettuna web-palvelin ohjelmisto apache, sekä postgresql tietokanta. Sovelluksen siirtäminen uudelle palvelimelle vaatii siis php:tä tukevan web-palvelimen, postgresql tietokannan, sekä mahdollisuuden suorittaa shell komentoja. Tietokantayhteyden asetukset ovat kansiossa config/database.php

## Testaus

Sovellusta testattiin kehityksen aikana, vain niiltä osin kuin nähtiin tarpeelliseksi. Esimerkiksi xss tai mysql haavoittuvuuksia ei testattu erikseen. Käyttäjän syötteistä tehtiin vain minimaalinen validointi.

## Omat kokemukset

Sovelluksen teossa oli helppoa PHP:n syntaksi. Vaikeaa oli ymmärtää composerin luokkien lataaminen. Opin luokkien ja olioiden käyttämistä palvelinpuolella. Kurssi oli mielestäni melko työläs ja vaativa.