

# Ohjelmistokehitys Angular-ja Symfony-kehyksillä

Vesa Kivistö

Opinnäytetyösuunnitelma

Lokakuu 2017

Tekniikan ja liikenteen ala

Insinööri (AMK), ohjelmistotekniikan tutkinto-ohjelma

## Sisältö

<b>1</b>	<b>Opinnäytetyön johdanto.....</b>	<b>1</b>
<b>2</b>	<b>Tavoite .....</b>	<b>1</b>
<b>3</b>	<b>Tietoperusta .....</b>	<b>1</b>
<b>4</b>	<b>Menetelmät ja aineisto.....</b>	<b>2</b>
<b>5</b>	<b>Aikataulu.....</b>	<b>2</b>
<b>6</b>	<b>Sisällysluettelo .....</b>	<b>2</b>

## 1 Opinnäytetyön johdanto

Opinnäytetyön aiheena ja alustavana työnimenä on ohjelmistokehitys Angular– ja Symfony-kehyksillä. Aiheen valintaan johti oma mielenkiinto verkkopalveluiden kehittämiseen sekä harjoittelun sallimat mahdollisuudet työskennellä verkkopalveluiden kehittämiseen tarkoitettujen ajankohtaisten teknologioiden kanssa. Opinnäytetyötä tehtäessä minulla on mahdollisuus laajentaa oppimistani edellä mainittujen kehysten parissa, perehtyä laajemmin ohjelmistokehityksen historiaan sekä ohjelmistokehityksen eri käytäntöihin, ja pohtia omaa kokemusta ja toimintatapoja ohjelmistokehityksen parissa. Opinnäytetyön toimeksiantajana toimii Protacon Solutions Oy.

## 2 Tavoite

Opinnäytetyön tavoitteena on toteuttaa Protacon Solutions Oy:n asiakkaan tilaamat verkkopalvelut. Verkkopalveluihin kuuluu kutsujärjestelmä, jolla voidaan lähettää kutsuja eri tapahtumiin sekä verkkokauppa, jonka kautta voi ostaa lippuja kyseisiin tapahtumiin. Opinnäytetyön aikana tutkin ohjelmistokehityksen sujuvuutta opinnäytetyöhön liittyvillä kehyksillä sekä toteuttamiseen käytettyjen ohjelmistokehitysmenetelmien sujuvuutta. Työssä käyn läpi hieman ohjelmistokehityksen historiaa ja menetelmien kehitystä, mutta en lähde tutkimaan kaikkia mahdollisia ohjelmistokehitysmenetelmiä vaan tutustun muutamaaan käytetyimpään menetelmään ja vertaan näitä työn toteutukseen. Työn lopputuloksena on tavoitteena saada asiakkaalle toimivat järjestelmät.

## 3 Tietoperusta

Opinnäytetyössä alustavana tietoperustana on ohjelmistotekniikan tutkinto-ohjelmassa oppimani tieto ohjelmistokehityksestä. Tutkittaviin ohjelmistokehyksiin tietolähteinä toimivat pääasiassa kehysten omat verkkosivut ja dokumentaatio ja tarvittaessa kehyksistä kirjoitetut kirjat tai muu painettu materiaali. Ohjelmistokehitykseen liittyvässä tietolähteinä uskon olevan sekä verkkojulkaisut että painettu materiaali yhtä suuressa osassa.

Opinnäytetyön hyötynä on konkreettisia esimerkkejä siitä, kuinka Angular- ja Symphony-kehysä hyödynnetään työelämässä. Pyrin myös analysoimaan työssä käytetyn ohjelmistokehitysmenetelmän vahvuuksia ja heikkouksia sekä vertaamaan näitä toimiviksi todettujen menetelmien vahvuuksiin ja heikkouksiin. Näin pyrin täsmentämään eri menetelmien eroja.

## 4 Menetelmät ja aineisto

Opinnäytetyön toteutetaan Protacon Solutions Oy:n tiloissa asiakasprojektina. Kehitysmenelmissä pyritään mahdollisimman ketteriin menetelmiin, aktiiviseen kommunikointiin asiakkaan kanssa ja tiiviiseen tiimityöhön. Asiakasprojektin alussa aineistonkeruuta on runsaasti Angular- ja Symphony-kehysiin perehtymisen merkeissä ja perehtymisen jälkeen aineistonkeruu jatkuu koko asiakasprojektin ajan. Opinnäytetyön alussa tiedonkeruu myös keskittyy paljon ohjelmistokehityksen historiaan ja eri menetelmiin.

## 5 Aikataulu

- Opinnäytetyön aihe-ehdotus: lähetetty ja hyväksytty viikolla 40.
- Aineistohaku: aloitettu viikolla 41, kestää opinnäytetyön loppuun kirjoittamiseen asti.
- Raportoinnin kirjoittaminen: tavoite aloittaa viikolla 42 taustatietojen ja teorian osuuksilta.
- Varsinainen toteutus: opinnäytetyössä käsiteltävää ohjelmistokokonaisuutta toteutettu alkukeväästä lähtien, toteutus jatkuu vielä
- Työn toteutus ja raportoinnin ”release candidate” valmis marraskuun puoliväliin, jotta ehtii joulukuun käsittelykokoukseen. Tähän tavoitteeseen pääseminen tiukkaa, joten realistisempi tavoite on saada opinnäytetyö valmiiksi tammikuun 2018 käsittelykokoukseen.

## 6 Sisällysluettelo

Alla alustava suunnitelma työn sisällysluettelosta. Otsikot ja otsikkonumeroinnit alustavan suunnitelman mukaiset.

## **Termit/Sanasto**

### **1 Työn lähtökohdat**

#### **1.1 Taustaa**

Jotain taustaa ratkaistavasta ongelmasta. Toteutettavana tuotoksena kutsujärjestelmä, joten mahdollisesti jotain liittyen vastaavanlaisiin ohjelmistoihin ja niiden taroituksiin. Opinnäytetyö myös käsittelee ohjelmistokehitystä, joten mahdollisesti myös taustatietoja ohjelmistokehitykseen liittyen.

#### **1.2 Toimeksiantaja**

Toimeksiantajan esittely.

### **2 Tehtävän kuvaus**

#### **2.1 Ongelma ja tavoitteet**

Ratkaistavan ongelman ja toteutettavan tuotoksen esittely

#### **2.2 Vaatimusmäärittely**

##### **2.2.1 Yleistä**

Yleistä teoriaa vaatimusmäärittelystä, miksi tehdään ja mitä varten tehdään. Vaatimusmäärittelyn tavoitteet ja merkitys osana ohjelmistoprojektia.

##### **2.2.2 Toiminnalliset vaatimukset**

Tuotoksen toiminnalliset vaatimukset, mitä ohjelmistolla tulee voida tehdä.

##### **2.2.3 Ei-toiminnalliset vaatimukset**

Ei-toiminnalliset vaatimukset, saatavuus ja skaalautuvuus yms.

## 3 Teoriaa ja teknologiavalinnat

### 3.1 Ohjelmistokehitys

Ohjelmistokehityksen teoriaa, pääasiassa hieman historiaa (miten kehittynyt nyky-muotoonsa?) ja eri menetelmien vahvuuksia ja heikkouksia.

### 3.2 Ohjelmointikielet

#### 3.2.1 Yleistä

Mahdollisesti yleistä taustatietoa ohjelmointikielistä.

#### 3.2.2 TypeScript

TypeScriptin esittely.

#### 3.2.3 PHP

PHP:n esittely.

#### 3.2.4 C#

C#:n läpikäynti. En itse tehnyt tällä varsinaisesti kovin paljon, mutta authorisointi-, raportointi- ja PDF-järjestelmä käyttävät tätä. Lyhyt läpikäynti.

### 3.3 Ohjelmointikehykset

#### 3.3.1 Yleistä

Tähän yleistä juttua ohjelmointikehyksistä. Mitä ne ovat ja miksi niitä käytetään.

#### 3.3.2 Angular

Angularin esittely. Kehyksen rakenne ja tärkeimmät komponentit ja ominaisuudet. Taustatietoja ja historiaa, Angular ei ole sama kuin AngularJS.

### 3.3.3 Symfony

Symfonyn esittely. Mikä Symfony on ja mihin sitä käytetään. Tärkeimpien ominaisuuksien läpikäynti. Symfonyn alla käsitellään myös Doctrine, joka ei varsinaisesti ole osa Symfony-kehystä, mutta se on laajasti käytetty Symfonyn parissa.

## 3.4 Ohjelmointiympäristöt

Käytettävät ohjelmointiympäristöt. Ainakin JetBrainsin PhpStormin läpikäynti, mahdollisesti myös Microsoftin Visual Studio tai Visual Studio Code.

## 3.5 Versionhallinta

### 3.5.1 Yleistä

Yleistä teoriaa versionhallinnasta. Merkitys ja tärkeys ohjelmistokehityksessä.

### 3.5.2 Git

Gitin teoriaa.

### 3.5.3 Phabricator

Phabricatorin esittely. Mikä on ja mihin käytetään. Vertailut muihin versionhallintajärjestelmiin (GitHub ja GitLab).

## 3.6 Muut työkalut

### 3.6.1 JIRA

JIRA:n esittely. Mikä merkitys JIRA:lla oli opinnäytetyön toteutuksessa.

### 3.6.2 Slack

Slackin esittely. Mikä merkitys Slackilla oli opinnäytetyön toteutuksessa.

## 4 Toteutus

### 4.1 Kutsujärjestelmän front end

Tässä käydään yksityiskohtaisesti läpi kutsujärjestelmän front endin toteutus. Käyttöliittymä ja tärkeimmät toiminnot tulevat omien alalukujen alle. Kuvia toteutetun järjestelmän tärkeimmistä osista, mahdollisesti koodisnippettejä tärkeimmistä osista.

### 4.2 Kutsujärjestelmän back end

Kutsujärjestelmän back endin toteutus. Esitellään käytettyjä luokkia ja tärkeimpiä kontrollereita ja toimintoja, kuten käyttäjien hallintaa ja kutsujen luontia. Tärkeimmät ja yksityiskohtaisimmat vaiheet omien alalukujen alle. Keskustelu muiden järjestelmien kanssa.

### 4.3 Tietokanta

Tietokannan esittely. Käydään läpi tietokannan rakenne (taulut ja taulujen sidokset).

### 4.4 Verkkokauppa

Kutsujärjestelmän rinnalle toteutetun verkkokaupan läpikäynti suhteellisen lyhyesti, mikäli koetaan tarpeelliseksi. Pääpaino on kutsujärjestelmässä, mutta verkkokaupan läpikäynti auttaa lukijaa hahmottamaan toteutetun ohjelmiston kokonaisuutena.

### 4.5 Authorisointijärjestelmä

Authorisointijärjestelmän läpikäynti. Tämä aika keskeisessä osassa toteutetussa ohjelmistossa, mutta koska en itse työskennellyt tämän parissa paljoa, niin käydään vain nopeasti läpi. Tarkoitus saada lukijalle perus ymmärrys järjestelmän toiminnasta ja merkityksestä.

### 4.6 Raportointijärjestelmä

Raportointijärjestelmän läpikäynti. Sama kuin authorisointijärjestelmässä.



## 4.7 PDF-järjestelmä

PDF-järjestelmä, PDF-tiedostojen luonti, säilytys ja käsittely. Suhteellisen lyhyt läpikäynti kuten authorisointi- ja raportointijärjestelmän kanssa.

## 5 Tulokset

Mitä opinnäytetyötä tehdessä saatiin varsinaisesti tehtyä. Saatiinko ratkaistua opinnäytetyön alussa esitetyt vaatimukset järjestelmälle.

## 6 Pohdinta

Ohjelmistokehitysmenetelmien näkökulmasta: mikä onnistui ja missä olisi ollut parantamisen varaa. Järjestelmässä ilmenneet ongelmat ja niiden ratkaisu sekä jatkokehitysideat.

## Lähteet

## Liitteet