

Verkkosivujen suunnittelu ja toteutus  
Case: Taiteilijaportfolio

Tommi Ilvonen

xx.xx.2018

|  |  |
| --- | --- |
| **Tekijä(t)**  Tommi Ilvonen. | |
| **Koulutusohjelma**  Tietojenkäsittely | |
| **Raportin/Opinnäytetyön nimi** Verkkosivujen tekeminen ja toteutus Case: Taitelijaportfolio | **Sivu- ja liitesivumäärä**  20 + 2 |
| Tiivistelmä edellytetään pääsääntöisesti vain opinnäytetöissä.  Opinnäytetyön tiivistelmässä esitetään työn keskeiset kohdat siten, että lukija ymmärtää tiivistelmän luettuaan työn sisältämät pääasiat. Tiivistelmässä esitetään selvitettävän asian tausta, työn tavoite ja rajaus, työn toteutustapa ja mahdolliset menetelmät, työn tekemisen ajankohta sekä tulokset ja päätelmät. Tiivistelmä etenee raportin mukaisessa järjestyksessä.  Tiivistelmä on enintään yhden sivun pituinen. Käytä tiivistelmässä asiatyyliä eli kirjoita tiiviisti, ymmärrettävästi ja kieliopillisesti oikein. Käytä kokonaisia lauseita ja virkkeitä. Jaa teksti muutaman virkkeen mittaisiin kappaleisiin ja erota kappaleet toisistaan ylimääräisellä rivinvaihdolla. Aloita uusi kappale siirtyessäsi uuteen asiaan. Tiivistelmä ei saa sisältää lähdeviitteitä.  Tiivistelmä on siis kuin itse opinnäytetyö pienoiskoossa. Tiivistelmän pitää olla itsenäinen kokonaisuus, joka on ymmärrettävissä itse opinnäytetyötä lukematta. | |
| **Asiasanat**  Tärkeysjärjestyksessä 3–6 asiasanaa, jotka kuvaavat työn sisältöä parhaiten. Käytä asiasanojen valinnassa Yleistä suomalaista asiasanastoa (YSA) osoitteessa <http://finto.fi/fi/> | |

Sisällys

[Käytetty sanasto 1](#_Toc508134970)

[1 Johdanto 2](#_Toc508134971)

[2 Tavoitteet ja suunnitelma 2](#_Toc508134972)

[3 Rajaus 3](#_Toc508134973)

[4 Suunnittelu 3](#_Toc508134974)

[4.1 JavaScript Frameworkin valitseminen 3](#_Toc508134975)

[4.2 JavaScript Frameworkien vertailu 6](#_Toc508134976)

[4.3 React.js **Virhe. Kirjanmerkkiä ei ole määritetty.**](#_Toc508134977)

[4.4 CMS 7](#_Toc508134978)

[5 Toteutus 7](#_Toc508134979)

[5.1 Verkkotunnus 7](#_Toc508134980)

[5.2 Verkkotunnuksen hankinta 8](#_Toc508134981)

[6 Yhteenveto 8](#_Toc508134982)

[Lähteet 9](#_Toc508134983)

[Liitteet 10](#_Toc508134984)

# Käytetty sanasto

JavaScript –

HTML –

CSS –

CMS –

MVC –

JavaScript Framework –

Front-end ­­­­­­­­–

# Johdanto

Opinnäytetyöni käsittelee verkkoportfolion toteutusta toimeksiantajalleni. Verkkoportfolio on perinteistä CV:tä nykyaikaisempi tapa esittää osaamista, saavutuksia ja työkokemusta. Verkkoportfolio on käytännössä sama asia, kuin yrityksen verkkosivut, mutta brändäyksen ja markkinoinnin kohteena on yrityksen sijasta yksityishenkilö.

Projektin toimeksiantajana toimii taitelija Kalle Pitkänen. Hän on öljymaalauksiin suuntautunut taitelija Sipoosta. Kalle Pitkänen on tuottelias ja hän laittanut töitään esille kuvien jakopalvelu Instagramiin. Myyntiä edistääkseen Kalle on ajatellut tarvitsevansa itselleen verkkoportfolion, jonne voisi laittaa teoksiaan esille, sekä kertoa itsestään.

Verkkoportfolion tekeminen taitelijalle tarkoittaa, että verkkoportfolion pitää olla persoonallinen ja tuoda taitelijan teokset mahdollisimman hyvin esille. Tämä vaatii perehtymistä taitelijan persoonaan ja yhteisiä suunnittelupalavereja hänen kanssaan, jotta persoonallisuus saadaan tuotua esille verkkoportfolioon. Kuvien selaamisen pitää olla mahdollisimman vaivatonta ja kuvista täytyy saada myös yksityiskohdat esille.

Verkkoportfoliosta toimeksiantaja hyötyy saadessaan teoksilleen, sekä itselleen enemmän näkyvyyttä. Parempi näkyvyys helpottaa näyttelyihin pääsemistä, sekä töiden myyntiä. Verkkoportfolio antaa henkilöstä ammattimaisen kuvan ja sillä voi erottua edukseen.

# Tavoitteet ja suunnitelma

Tavoitteenani on tehdä verkkoportfolio itselleni uusilla teknologioilla, joten projektilla kasvatan omaa osaamistani. Hyvin onnistunut verkkoportfolio olisi oiva näyte omasta osaamisestani. Koska uudella teknologialla tehdessä oppimiskäyrä on alkuun hidas, pitää suunnitella hyvin teknologioiden käyttö ja tavoitteiden pitää olla realistiset.

Toimeksiantaja antoi minulle joustavan aikataulun ja toteutuksen suhteen vapaat kädet. Sovimme, että opinnäytetyöprojektin vähimmäisvaatimus on tuottaa verkkoportfolio, jossa on perusasiat kunnossa. Eli verkkosivujen pitää olla julkisesti saavutettavissa, responsiivinen, pitää olla näyttävä etusivu, kuvagalleria, henkilöesittely, yhteydenottolomake ja yhteystiedot.

Vähimmäisvaatimuksilla verkkoportfolion ulkonäkö on valmis ja esittelykelpoinen. Kun vähimmäisvaatimukset on toteutettu, seuraavat askeleet olisivat toteuttaa hakukoneoptimointi, ottaa käyttöön sisällönhallintajärjestelmä ja toteuttaa jatkuvan integroinnin automatisointi. Näitä ei kuitenkaan tarvitse saada tehtyä opinnäytetyöhön, vaan nämä pyrin toteuttamaan sitä mukaa, kun ehdin.

Varsinkin vähimmäisvaatimuksista seuraavat tavoitteet ovat itselleni oppimisen ja ammattitaidon kannalta mielenkiintoisia. Nämä tavoitteet ovat myös toimeksiantajan kannalta tärkeitä verkkosivujen ylläpidon ja tavoitettavuuden osalta.

# Rajaus

Opinnäytetyön ei ole tarkoitus kertoa, kuinka tehdä verkkosivut mahdollisimman tehokkaasti ja helposti. Opinnäytetyössä ei keskitytä kertomaan verkkosivujen historiasta tai asioista, jotka pitää ottaa huomioon verkkosivuja tehdessä. Edellä mainittuja ohjeitä on Internet täynnä, joten en koe niiden tuovan lisäarvoa tälle opinnäytetyölle.

Projektin tarkoitus on yksinomaan tuottaa toimeksiantajalle ratkaisu, perustella valitut teknologiat ja kuvata toimeksiannon työvaiheiden kulkua. Olen enemmän ohjelmistokehittäjä, kuin graafikko tai Web Designer, joten lähestymistapani on suunnitella toteutus, toteuttaa ja oppia. En kuitenkaan tingi verkkosivujen ulkonäöstä tai käytettävyydestä, sillä front-end-kehitys on lähellä sydäntäni. Front-end tarkoittaa selainpuolen ohjelmointia eli koodia, joka ajetaan selaimessa verkkosivua selatessa. Tähän kuuluu esimerkiksi sivun rakenne (HTML), ulkoasu (CSS) ja selaimessa tapahtuvat toiminnallisuudet (JavaScript). (Laine, 2015.)

# Suunnittelu

Projektin alkuun minun pitää suunnitella käyttämäni teknologiat ja kuinka hyödynnän niitä. Projektin etenemisen kannalta oleellisin valinta on sopivan JavaScript ohjelmointiympäristön valitseminen. En ole juurikaan käyttänyt uusimpia JavaScript ohjelmointiympäristöjä, joten edessä on paljon opettelua. Työssäni käytän päivittäin Knockout.js ohjelmointiympäristöä, mutta se on hieman vanhentunut ja monimutkainen, joten ei näin ollen sovellu tähän projektiin.

## JavaScript Frameworkin valitseminen

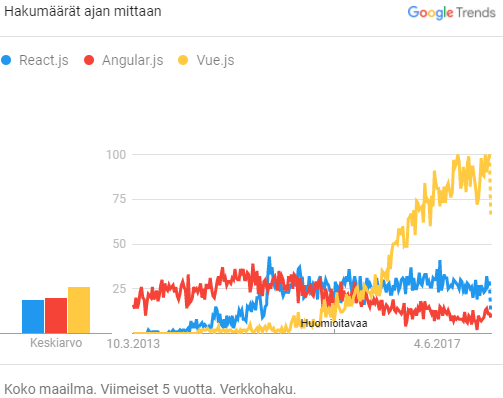
JavaScript Framework tarkoittaa ohjelmointiympäristöä, jossa ohjelmoidaan JavaScript-ohjelmointikielellä. Ohjelmointiympäristö on ohjelma tai joukko ohjelmia, joilla ohjelmoija suunnittelee ja toteuttaa ohjelmistoa. (Wikipedia 2018.) JavaScript Frameworkit ovat tarkoitettu web-ohjelmointiin eli verkkosivujen interaktiivisuuden tuottamiseen. (Wikipedia 2018.)

JavaScript Frameworkien ikä on tyypillisesti melko lyhyt ja ne kehittyvät ominaisuuksiltaan nopeasti. Jokin mikä on ollut viisi vuotta sitten suosituin, saattaa olla nyt unohdettu ja korvattu uudemmilla teknologioilla. Frameworkin ikä on hyvä pitää mielessä, sillä tämän hetken kuumimmat Frameworkit ovat myös parhaiten työllistäviä. (lähde?)

Tämän hetken kolme suosituinta JavaScript Frameworkia ovat React.js, Vue.js ja Angular.js. Näillä on toteutettu valtaosa isoista nykyisistä verkkopalveluista ja ovat ohjelmistokehityksessä eniten työllistäviä ohjelmointiympäristöjä. (lähde?)

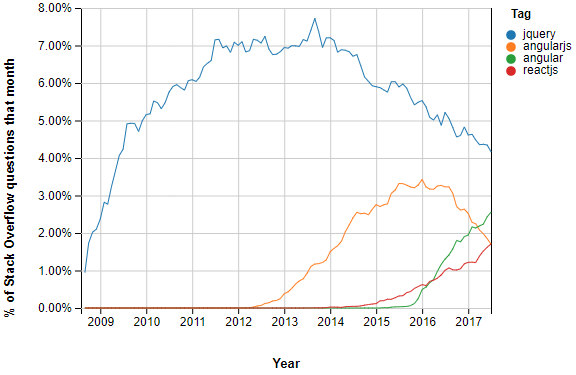
## Suosituimmat JavaScript Frameworkit

Selvitin Google Trends -työkalulla mikä kolmesta näistä kolmesta JavaScript Frameworkista on haetuin. Google Trends tulokset ovat vertailukelpoisia, sillä jokainen tietopiste on jaettu maantieteellisen sijainnin hakujen kokonaismäärällä ja sen edustamalla ajanjaksolla, jotta suhteellista suosiota voidaan verrata. (Google Trends 2018.)

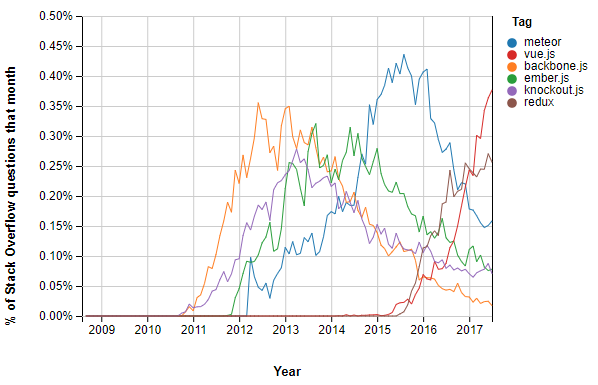


Google Trends kuvaajasta voi päätellä, että viimeisen viiden vuoden aikana Vue.js:n hakumäärät ovat kasvaneet Angular.js:n ja React.js:n ohi. Hakuja on tapahtunut todennäköisesti eniten, kun etsitään apua kyseisellä Frameworkilla kehittäessä tai kun haetaan tietoa hakijalle tuntemattomasta Frameworkista. Tästä voi päätellä, että Vue.js:n suosio on kasvamassa.

Tein vertailuja myös Stack Overflow verkkosivun Stack Overflow Trends työkalulla. Stack Overflow on maailman isoin ohjelmistokehittäjäyhteisö yli 50 miljoonalla kuukausittaisella käyttäjällä. (Stack Overflow Insights, 2018.) Stack Overflow Trends työkalulla voi verrata Stack Overflow:ssa tehtyjä hakuja vuodesta 2008 lähtien. Työkalu ehdotti suosituimmiksi hauiksi JavaScript Frameworks ja Smaller JavaScript Frameworks.



JavaScript Frameworks kuvaajasta voi päätellä jQueryn olleen pitkään Stack Overflow:n haetuin Framework. jQueryn suosio alkoi laskea Angularin ensimmäisen version myötä ja Angularin suosio alkoi laskea Angularin uusimman version ja Reactin myötä. Kuvaajan perusteella vuoden 2017 puolessa välissä Angularin uusin versio on ollut maailmanlaajuisesti Reactia suositumpi, mutta uskoisin maailmanlaajuisen tilanteen olevan tällä hetkellä varsin tasan. Tässä kuvaajassa ei ole Vue.js:ää, sillä se on vasta kasvattamassa suosiotaan ja yhteisö on vielä pieni verrattuna kuvaajan Frameworkeihin.



Smaller JavaScript Frameworks kuvaajasta voi päätellä Vue.js:n olleen Stack Overflow:n haetuin pienempi Framework vuoden 2017 puolivälissä. Nykyisellä kasvuvauhdilla Vue.js tulee todennäköisesti löytymään käytetyimpien Frameworkien joukosta.

## JavaScript Frameworkien vertailu

Angular.js on Googlen kehittämä ja ylläpitämä avoimen lähdekoodin JavaScript Framework. Angularin arkkitehtuuri on MVC eli model-view-controller, joka tarkoittaa mallia, näkymää ja käsittelijää. Malliin varastoidaan tieto, käsittelijä käsittelee tietoa ja näkymässä näytetään tieto. (Lähde?)

Tutkimuksieni perusteella Angular.js on jäykkä ratkaisu ja oppimiskäyrä on melko pitkä. Jäykällä tarkoitan, että tekijällä ei ole vapaita käsiä päättää miten jokin ratkaisu toteutetaan, vaan Framework pakottaa johonkin tiettyyn malliin. Tämä on hyvä ominaisuus, jos on isompi tiimi kehittämässä, jolloin koodi pysyisi mahdollisimman samanlaisena tekijästä riippumatta. Näin ollen Angular.js ei oikein palvele minun tarpeitani.

React.js on Facebookin kehittämä ja ylläpitämä avoimen lähdekoodin JavaScript Framework. Kyseessä ei ole oikeastaan Framework, vaan Library eli kirjasto, sillä React.js koostuu komponenteista, eikä pakota käyttäjäänsä tietynlaiseen malliin. (Lähde?) Toisin kuin Angular on MVC arkkitehtuurin Framework, React.js käsittää vain MVC:n view-tason eli näkymän. React.js:llä on tehty Facebookin lisäksi lukuisia suosittuja verkkosivuja kuten Airbnb, Dropbox, Imgur, Instagram, Netflix, Paypal, Periscope, Reddit, Uber, Tesla ja WhatsApp. (Medium 2018.) Mediumin artikkelin mukaan voi todeta, että valtaosa sosiaalisen median verkkosivuista on rakennettu Reactilla.

Vaikka React.js on luultavasti tämän hetken trendikkäin ja suosituin JavaScript Framework, siinäkin on ongelmansa. React.js antaa todella vapaat kädet toteutukseen ja oppimiskäyrä on Angularia pidempi. React.js on valtavan yrityksen tekemä Framework palvelemaan omaa tarkoitustaan eli monimutkaisia ja laajoja käyttöliittymiä. Reactia osatessaan varmasti saisi hyvin web-ohjelmointi töitä, mutta päädyin siihen, ettei React.js palvele parhaiten tämän projektin tarkoitusta.

Päädyin siis itsellenikin yllätykseksi käyttämään Vue.js Frameworkia tähän projektiin. Vue.js on avoimen lähdekoodin JavaScript Framework, jonka kehitti aluilleen Evan You, entinen Googlen työntekijä ja Angularin kehittäjä. Evan You päätyi kehittämään Vue.js:n, sillä hän halusi Angularia kevyemmän Frameworkin ilman ylimääräisiä konsepteja. (Cromwell, 2016.) Vue.js:n on siis tarkoitus olla helposti omaksuttava ja ketterä Framework. Se on kehitetty pienempään tarpeeseen, kuin suuren kompleksisen verkkosivun tekemiseen. Toisaalta Vue.js:llä onnistuu myös monimutkaisemman kokonaisuuden tekeminen (Lähde?).

Vue.js siis erottui edukseen opittavuudellaan ja ketteryydellään. Vue.js palvelee selkeästi parhaiten tämänkaltaista yksinkertaisempaa projektia. Ehkäpä joskus tulevaisuudessa refaktoroin eli rakennan uudestaan omat verkkosivuni Reactilla.

## CMS

<https://hackernoon.com/how-i-built-a-content-management-system-for-a-react-app-in-one-day-269df17f5509>

<https://hackernoon.com/how-i-built-a-cms-and-why-you-shouldnt-daff6042413a>

# Toteutus

## Verkkotunnus

Projektin alkuun varaan toimeksiantajalleni domainin (engl. domain name) eli verkkotunnuksen. Domain on verkkosivun nimi ja osoite, jolla Internetin käyttäjät pääsevät verkkosivuille. Domain Name System (DNS) eli nimipalvelujärjestelmä kääntää verkkotunnukset IP-osoitteiksi. Domain piilottaa verkkosivun IP-osoitteen ja näyttää käyttäjälle helpommin muistettavan verkkotunnuksen, kuten esimerkki.com. (Gil, 2018.) Domain Name System eli DNS on Internetin nimipalvelujärjestelmä, joka muuntaa verkkotunnuksia IP-osoitteiksi.

Toimeksiantaja haluaa käyttää omaa nimeään verkkotunnuksena, sekä haluaa verkkotunnuksen päätteeksi kaupalliseen tarkoitukseen käytetyn .com -päätteen. Verkkotunnukseksi valikoitui kallepitkanen.com.

Oma nimi verkkotunnuksena on hyvä tapa markkinoida itseään, sillä se on helppo muistaa ja parantaa hakukoneessa löytymistä. Verkkotunnuksessa kannattaa käyttää ainoastaan kansallisia symboleita, jotta verkkotunnus olisi mahdollisimman helppo kirjoittaa Suomenkin ulkopuolella. Tämän takia nimen ääkkönen on vaihdettu aakkoseksi.

Verkkotunnuksen ylätasoksi .com on hyvä ratkaisu, sillä se viittaa kaupalliseen tarkoitukseen (commercial). Taitelijaportfolion on tarkoitus markkinoida henkilöä, joten tarkoitus on kaupallinen.

## Verkkotunnuksen hankinta

Vertailin erilaisia verkkotunnusvälittäjiä Internet -artikkeleilla, jotta sain hyvän kuvan mistä kannattaisi varata verkkotunnus. Ensimmäisenä minulle tuli mieleeni aiemmin käyttämäni Gandi.net, sillä sieltä olen hankkinut itselleni verkkotunnuksen. Halusin kuitenkin saada laajemman yleiskuvan, mikä olisi tällä hetkellä paras verkkotunnusvälittäjä, sillä Gandin hallintanäkymä oli omasta kokemuksestani hieman vanhanaikainen.

Luin verkkoartikkelit <https://makeawebsitehub.com/reviews/domain-registrars/> ja <https://bloggingthing.com/best-domain-name-registrars-2017-2018>, joista molemmat suosittelivat parhaana vaihtoehtona Namecheap:ia. Tärkeimmät perustelut olivat käyttäjäystävällinen käyttöliittymä, järkevät hinnat ja nopea sekä luotettava tukipalvelu.

Verkkotunnus kallepitkanen.com maksoi 8,95 € + 0,15 € ICANN maksu (rekisteröintimaksu) = 9,09€

Valitsin käyttööni ilmaisen WhoIsGuard palvelun, joka suojaa yksityistiedot whois pyynnöiltä.

# Yhteenveto

# Lähteet

Paul Gil 2018. What Is a Domain Name? Luettavissa:  
<https://www.lifewire.com/what-is-a-domain-name-2483189>. Luettu: 12.2.2018.

Tuulikki Laine 2015. Mitä markkinoijan tulee ymmärtää web-ohjelmoinnista. Luettavissa:  
<https://www.dagmar.fi/verkkopalvelukehitys/mita-markkinoijan-tulee-ymmartaa-web-ohjelmoinnista/>. Luettu 22.2.2018

Google Trends 2018. Luettavissa:  
<https://support.google.com/trends/answer/4365533?hl=fi&ref_topic=6248052>. Luettu 26.2.2018

Google Trends 2018. Luettavissa:  
https://trends.google.com/trends/explore?date=today%205-y&q=react.js,angular.js,vue.js.  
Luettu 26.2.2018

Stack Overflow Insights 2018. Luettavissa:  
https://insights.stackoverflow.com/. Luettu: 26.2.2018.

Stack Overflow Trends 2018. Luettavissa:   
<https://insights.stackoverflow.com/trends?tags=jquery%2Cangularjs%2Cangular%2Creactjs>. Luettu 26.2.2018.

Stack Overflow Trends 2018. Luettavissa:  
<https://insights.stackoverflow.com/trends?tags=angularjs%2Cvue.js%2Creactjs>. Luettu: 26.2.2018.

Medium 2018. Luettavissa:  
<https://medium.com/@coderacademy/32-sites-built-with-reactjs-172e3a4bed81>. Luettu: 6.3.2018.

Cromwell 2016. Luettavissa:  
<https://betweenthewires.org/2016/11/03/evan-you/>. Luettu: 6.3.2018.

# Liitteet