



# Angularin ja Reactin erot front-end kehityksessä

Antti Vainikka

TkK-tutkielma  
Huhtikuu 2017

TULEVAISUUDEN TEKNOLOGIOIDEN LAITOS  
TURUN YLIOPISTO

TURUN YLIOPISTO  
Tulevaisuuden teknologioiden laitos

VAINIKKA, ANTTI: Angularin ja Reactin erot front-end kehityksessä  
TkK-tutkielma, x-määrä s., x liites.  
Tietotekniikka  
Huhtikuu 2017

---

Angular ja React ovat molemmat laajalti käytössä olevia front-end työkaluja, jotka ovat luotuja laajentamaan verkkosovellusten toiminnallisuuksia sekä helpottamaan niiden kehitystä. Nykyään harvemmin enää kirjoitetaan verkkosovelluksia tai edes yksinkertaisempia verkkosivuja ilman mitään kehikkoa tai kirjastoja, joten niistä on tullut teollisuus standardeja.

Angular ja React ovat nykyisistä front-end työkaluista kehittyneimpien joukossa. Angular on front-end kehikko, jolle voidaan ohjelmoida käyttämällä Javascript tai Typescript -ohjelmointikieliä. Typescriptillä kirjoitetut Angular sovellukset käännetään Javascriptiksi ennen suoritusta. React on Javascript kirjasto, joten sitä kirjoitetaan enimmäkseen Javascriptiä käyttäen. DOM (document object model) manipulaatio Reactilla tapahtuu käyttäen JSX Javascript laajennussyntaksia, jolla voidaan kuvailla HTML-tageja käyttäen Javascriptiä.

Asiasanat: tiivistelmä sivu, Kandidaatti -tutkielma, Javascript, React, Angular, Front-end.

# Sisältö

1	Johdanto	1
2	Kehitys	2
3	Backendin kanssa kommunikointi	3
4	Datan sitominen ja DOM manipulaatio	4
5	Suorituskyky ja kehittämisen vaikeudet	5

# 1 Johdanto

Nykyaikaisessa verkkosovelluskehityksessä on tavallisesti käytössä yksi tai useampi erilainen kehys tai kirjasto, joka auttaa kehittäjää tekemään sovelluksesta laajemman ja ominaisuuksiltaan monipuolisemman. Kehykset ja kirjastot tekevät usein vain tietyn osan sovelluksesta. Angular ja React ovat

tunnetuimpien joukossa front-end puolen työkaluista. Ne ovat laajalti käytössä monissa verkkopalveluissa ympäri maailman.

Angularia on käytetty muun muassa Wolfram Alpha laskentasovelluksessa[1], suoritinvalmistaja Intelin verkkosivuissa sekä uutisyhtiö NBC:n (National Broadcasting Company) sekä uutisyhtiö ABC Newsin (American Broadcast Company News) verkkosivuissa.[2] Reactilla toteutettuja tunnettuja verkkosovelluksia ja sivuja on esimerkiksi median suoratoistopalvelu Netflix, joka on erikoistunut elokuviin ja televisiosarjoihin, rss-syötteiden lukupalvelu Feedly sekä vähittäismyyntiyritys Walmartin kotisivut. [3]

Angular on laajalti käytetyn MEAN-fullstack kehyspinon front-end kehys. MongoDB, ExpressJS, Angular, NodeJS ovat MEAN-pinon kehykset kokonaisuudessaan. Fullstack tarkoittaa kehyspinossa sitä, että se kattaa sovelluksen kaikki osa-alueet eli back-endin, front-endin sekä tietokannan.

## 2 Kehitys

Angularin ja Reactin kehittyneisyyttä ja laajaa levinneisyyttä selittää suuret yritykset niiden kehityksen takana. Angularin kehittäjänä toimii selainjätti Google ja Reactin pääkehittäjä on Facebook. Reactin kehityksessä on mukana myös Instagram sekä yhteisö.

Angularia kehittää sekä ylläpitää hakukonejätti Google. Angular on jaettu versioihin, joista tällä hetkellä vakaa käytössä oleva versio on 1.6.1, mutta versio 2 on ollut jo pitkän aikaa kehityksessä. Versio 2 tai vain Angular 2 on toteuttanut monia asioita hyvin erilailla aikaisempiin versioihin verrattuna. Käynkin asioita läpi pääasiallisesti Angular 2 kannalta, vaikkakin monet asiat pätevät vanhempiinkin versioihin.

Reactia puolestaan kehittää yhteistyössä yhteisön kanssa Facebook sekä tämän tytäryhtiö Instagram. Reactissa ei ole suurempia muutoksia tuovia versioita vaan sitä kehitetään kasaamalla edellisen päälle uusia ominaisuuksia ja korjauksia.

Google vaihtoi Angularin kielen Javascriptistä Typescriptiin versionumeron vaihtuessa 1 alkuisista numeroista 2 alkuisiin numeroihin.

### 3 Backendin kanssa kommunikointi

Angular käyttää sovelluksen backendin kanssa kommunikointiin palveluita, jotka lähettävät kutsuja backendille ja jakavat tiedot haluttuun paikkaan. Palvelut ovat erillisessä tiedostossa määriteltyjä koodinpätkiä, jotka tekevät kutsuja backendille.

Reactissa puolestaan ei ole mitään sisäänrakennettua standardia tapaa kommunikoida backendin kanssa sillä se on vain kirjasto, eikä täysi front-end kehys. Sensijaan Reactissa ei ole standardia tapaa toteuttaa backend kommunikaatiota vaan se toteutetaan usein tavallisilla Javascriptin ajax kutsuilla tai käyttäen muita kehyksiä apuna.

Angularin palvelut injektoidaan riippuvuuksina tarvittaviin elementteihin ja niistä luodaan ilmentymä kun komponentti tarvitsee niitä.

## 4 Datan sitominen ja DOM manipulaatio

Front-end työkalujen tärkein tehtävä on datan näyttäminen näkymissä. Data jota näytetään voi muuttua ja usein käyttäjälle halutaan näyttää muutokset niin pian kuin mahdollista. Tämä on mahdollistettu datan sitomisella molemmissa tapauksissa. Toteutus työkalujen välillä tosin eroaa huomattavasti.

Reactin toteutuksessa muuttujia voidaan lisätä JSX koodin sekaan kaarisulkeilla. Kaarisulkeiden sisällä olevat muuttujat ottavat arvonsa koodista. React komponenteilla on erilaisia tiloja, joihin kuuluu muunmuassa komponentin luonti ja tuhoutuminen. Lisäksi voidaan määritellä intervalli, jonka välein tietty funktio suoriutuu. Näkymän päivittäminen vaatii joko intervalin tai komponentin tilan muutoksen.

Angularin toteutus eroaa Reactin toteutuksesta monella tavalla. Angularissa voidaan sitoa data samalla lailla yksisuuntaisesti kaarisulkeilla kuten Reactissa, mutta sen lisäksi on myös mahdollista tehdä kaksisuuntainen sidonta käyttämällä kaksia sulkeita. Yksisuuntaisessa sidonnassa data päivittyy muuttujien muuttuessa. Mikäli muuttuja on sidottu kaksisuuntaisesti toimii tämä myös toisin päin, eli muuttujan sisältö muuttuu näkymää muutettaessa. Esimerkiksi kun käyttäjä syöttää tekstikenttään jotain, muuttujan arvo muuttuu tekstikentän sisällöksi. Näkymien päivitystä ei tarvitse sitomisen jälkeen tehdä itse vaan se tapahtuu automaattisesti muuttujan muuttuessa.

Angular on siis kehittäjän kannalta helpompi, sillä siinä ei tarvitse miettiä päivitysintervalleja tai komponentin tilaa, toisin kuin Reactissa. React sovelukset ovat yleensä kevyempiä

## 5 Suorituskyky ja kehittämisen vaikeudet

Angular on käyttänyt versiosta 2 alkaen kielenään Typescriptiä, joka on Javascriptin syntaksin ja semantiikan perivä kieli, joka kääntyy Javascriptiksi kääntövaiheessa. Sekä Typescript, että Reactin JSX vaativat molemmat käännöksen ennen kuin selain voi niitä suorittaa.[4]

Toisaalta typescript tuo uusia työkaluja tavallisen Javascriptin päälle. Typescriptissä voi tyypitellä muuttujia eikä tarvitse siten tyytyä Javascriptin automaattisesti antamaan muuttujatyyppeihin. Tyypittelyllä saadaan ehkäistyä epätoivottuja lopputuloksia.

Monet asiat, jotka Angularissa ovat parin rivin koodinpätkiä vaativat Reactissa pidemmän pätkän koodia. Esimerkiksi sidotun muuttujan päivitys, jota ei Angularissa tarvitse miettiä, vaatii React-ohjelmoijalta toimenpiteitä. Monet Angularin tarjoamat helppokäyttöisyyttä edistävät seikat saattavat tehdä Angularista Reactia raskaamman.

Vaikka Typescript on Javascriptin "paranneltu versio"joidenkin mielestä Reactin JSX on mukavampaa kirjoittaa. Reactin lähestymistapa on siitä erikoinen, että kaikki komponenttien HTML määrittelyt tehdään JSX:llä, joka on periaatteessa Javascriptiä, joka on tehty näyttämään HTML kieleltä. Esimerkiksi komponentin luominen JSX:llä tapahtuu seuraavalla tavalla.

```
<MyButton color="blue"shadowSize=2> Click Me </MyButton>
```

Mutta se muuttuu käännösvaiheessa muotoon:

```
React.createElement( MyButton, color: 'blue', shadowSize: 2, 'Click Me' )
```

,joka onkin jo tyypillistä Javascript koodia, jossa ollaan käytetty vain React kirjaston metodia.



## Viitteet

- [1] Made With Angular  
<https://www.madewithangular.com/>
- [2] AngularJS - Wikipedia  
<https://en.wikipedia.org/wiki/AngularJS>
- [3] Sites Using React  
<https://github.com/facebook/react/wiki/sites-using-react>
- [4] Typescript - JavaScript that scales  
<https://www.typescriptlang.org/>