

**Verkkotyökalu Amazon Web Servicen palveluiden kuvausten
tiivistämiseksi**



Ammattikorkeakoulututkinnon opinnäytetyö

Tieto- ja viestintäteknikka, insinööri (AMK)

Kevät 2023

Tomi Markkinen

Tieto- ja viestintäteknikka

Tekijä Tomi Markkinen

Työn nimi Verkkotyökalu Amazon Web Servicesin palveluiden tiivistämiseksi

Ohjaaja Teemu Järvenpää

Tiivistelmä

Vuosi 2023

Tämä opinnäytetyö on tehty Amazon Web Service (AWS) yritykselle, ja sen Suomen toimipisteelle. Työn tarkoitus on toteuttaa AWS:n työntekijöiden käyttöön työkalu, josta näkee nopeasti mitä eri AWS:n palvelut tekevät. Tällä hetkellä AWS tarjoaa yli 200 erilaista palvelua asiakkailleen, joten työntekijöiden on haastavaa muistaa jokaisen palvelun sisältö ulkoa. Tämän takia työssä halutaan luoda työkalu, jonka käyttö on mahdollisimman nopeaa ja helppoa, ja josta näkee yhdellä silmäyksellä mitä eri palvelut tekevät. Työn lähtökohtana oli luoda työkalu, joka tarjoaa neljällä sanalla kuvauksen kaikista eri AWS:n palveluista suomeksi. Sanamääräksi valikoitui neljä, sillä se on vielä nopea lukea monellekin palvelulle, mutta neljään sanaan saa myös tarpeeksi informaatiota, jotta palveluiden erottaminen toisistaan on mahdollista.

Työssä halutaan tehdä verkkosovellus, joka toteuttaa siltä vaaditut tavoitteet, mutta helposti saavutettavissa olevalla tavalla. Työn toteutustapaa ei ollut ennalta määritetty, mutta työssä haluttiin pyrkiä kunnianhimoisempaan toteutukseen kuin vain muistilapun tekemiseen, joten lähdettiin tavoittelemaan työkalua, jota voi myös laajentaa tulevaisuudessa. Lopullinen työ toteutettiin web sovelluksena, käyttäen vain AWS:n omia palveluita ja Jekyll-ohjelmistoa, joka generoi dynaamista sisältöä staattisesti. Työ toteutettiin myös käyttämällä Amazon Amplify-palvelua, jonka ansiosta työkalun käyttö onnistuu myös verkossa.

Opinnäytetyö saavutti kaikki sille alun perin asetetut tavoitteet ja lisäksi myös muita matkan varrella tulleita lisäideoita. Valmiista tuotteesta saatiin yksinkertainen ja silti informatiivinen, sekä toteutus onnistuttiin luomaan myös verkossa käytettäväksi. Työllä olisi paljonkin jatkokehitysmahdollisuuksia, joita ei työn asettamissa aikamääreissä pystytty lisäämään projektiin.

Avainsanat Amazon Web Services, Amplify, pilvipalvelut

Sivut 21 sivua

Information and communication technology

Author Tomi Markkinen

Subject Support tool for Amazon Web Services

Supervisor Teemu Järvenpää

Abstract

Year 2023

This thesis was made for Amazon Web Services (AWS) Finnish branch. The purpose of the work was to implement a tool for internal AWS employees allowing a fast review of the AWS services. Currently, AWS offers over 200 different services, so it can be challenging to recall what each service is meant for. This was the motivation for creating a tool, which allows quick and easy access to see each service's description. The starting point of the work was to create a tool, which offers a four-word description for all the AWS services. The word limit is four due to this being amount that can provide enough information to allow the reader to distinguish each service while being fast to read.

The method of execution of the work was not predetermined, so the aim was to create a tool which could be extended in the future. The work was done as a web application which utilizes AWS's own services and Jekyll framework. Amazon Amplify was used to host the application so it can be accessed via internet.

The thesis reached all its goals as well as some additional ideas that encountered during the thesis project. The finished product was informational, and it was managed to be hosted online so using it through a web browser was plausible. There are opportunities for further development for the application, which could not be implemented within the time limit of the thesis project.

Keywords Amazon Web Services, Amplify, cloud computing

Pages 21 pages

Sisällys

1	Johdanto	1
2	Amazon Web Services, AWS.....	2
3	Käytetyt palvelut	3
3.1	Amazon EC2	3
3.2	AWS Cloud9.....	3
3.3	AWS CodeCommit.....	4
3.4	Amazon Amplify	4
3.5	Amazon Translate	5
3.6	Jekyll.....	5
4	Tavoitteet	6
5	Työkalun toteutus.....	7
5.1	Alkuperäinen suunnitelma	7
5.2	Muutos Jekylliin	7
5.3	Valmis työkalu	8
5.3.1	Toteutus	9
5.3.2	Hosting	13
5.3.3	Lopputulos.....	15
6	Jatkokehitysmahdollisuudet.....	16
	Lähteet.....	17

1 Johdanto

Pilvipalvelut ovat nopeasti kasvava osa tietotekniikan maailmaa. Kasvu näkyy suoraan myös pilvipalveluita tarjoavien yritysten suurista palveluiden määrässä, ja tästä syystä opinnäytetyön aihe on ajankohtainen. Tämä työ tehtiin yritykselle Amazon Web Services (AWS), jossa työn tarkoituksena oli toteuttaa sisäisten työntekijöiden työtä helpottava keino saada nopea muistutus siitä, mitä eri AWS:n palvelut tekevät. AWS:n palveluiden kirjo kattaa yli 200 eri palvelua, mikä tekee kaikkien palveluiden muistamisesta haastavaa. Tälle työlle määritetty lähtökohta oli luoda jokin keino tarjota nopea neljän sanan kuvaus kaikista eri palveluista, joita AWS tarjoaa.

Tavoitteena oli luoda keino tarjota tieto eri palveluista mahdollisimman yksinkertaisesti ja nopeasti. Työntekijä, joka etsii tarpeeseensa sopivinta työkalua Amazonin laajasta palveluvalikoimasta, haluaa löytää vastauksen tarpeeseensa mahdollisimman nopeasti. Etenenkin tarpeessa, jossa kaipaa tietoa useammasta eri palvelusta, kaikkien löytäminen yhdestä paikasta on kriittistä.

Työn lähtökohtana oli toteuttaa verkkosovellus, joka mahdollistaisi helpon saatavuuden verkossa. Työssä käytettiin staattisten sivujen kehittämiseen luotuja teknologioita, jotta sivu saatiin verkkoon helposti ja turvallisesti.

2 Amazon Web Services, AWS

Amazon Web Services (AWS) on markkinoiden suurin pilvipalveluiden tarjoaja. Vuoden 2022 viimeisellä neljänneksellä AWS:n markkinaosuus pilvi-infrastruktuuri markkinasta oli 32 %, eli kolmannes kaikista pilvipalveluiden käyttäjistä on Amazonin palveluiden käyttäjiä. (Statica, 2022.) Pilvipalvelut ovat käytön mukaan saatavilla olevia IT resursseja. Yhdysvaltain standardisointi- ja teknologiainstituutin kuvaus pilvipalveluista vuodelta 2011: ”Pilvipalvelu on malli, joka mahdollistaa kaikkialta, kätevän ja tarpeen mukaan olevan verkkoyhteyden jaetussa verkossa oleviin tietoteknisiin resursseihin, jotka voidaan nopeasti valmistella ja julkaista” (Yhdysvaltain standardisointi- ja teknologiainstituutti, 2011). Käytännössä tämä tarkoittaa sitä, että sen sijaan, että jokainen yritys tai muu organisaatio rakentaisi valtavan määrän omia konesaleja, voidaan pilven tuottajilta ostaa kapasiteettia ja resursseja käytön ja tarpeen mukaan. (Amazon, 2022d)

Etukäteismaksujen muuttaminen alhaisiin muuttuviin maksuihin, mahdollistaa yritysten keskittävän aikaa ja rahaa yrityksen toiminnalle kriittisiin toimintoihin. Pilvi-infrastruktuuri mahdollistaa erittäin nopean ja joustavan kehityksen, sillä resurssien käyttöönotto, kuten palvelimen provisiointi, tapahtuu minuuteissa. (Amazon, 2022d)

3 Käytetyt palvelut

Työn rakentamiseen vaadittiin monia erilaisia palveluita ja työkaluja. Tässä luvussa käydään läpi näitä tarvittavia työkaluja ja niiden perusteita. Kaikki paitsi yksi luetelluista palveluista ovat Amazonin palveluita ja käytössä heidän asiakkailleen.

3.1 Amazon EC2

Amazon Elastic Compute Cloud, EC2, on yksi AWS:n perusrakennuspalikoista. EC2 tarjoaa erittäin skaalautuvaa ja ketterää laskentatehoa pilvessä. EC2 on virtuaalipalvelin, jonka etuna kiinteisiin koneisiin ja servereihin verrattuna on nopea skaalautuvuus, niin ylös kuin alas, tarpeen mukaan. Erilaisia versioita EC2:sta kutsutaan instanssityypeiksi. Instansseja on yli 600 erilaista mallia, joiden prosessorien, päämuistin ja tarkoitusten määrä vaihtelee. Keskusmuisti RAM, Random Access Memory, on tietokoneen muistia, jossa pidetään käyttöjärjestelmää ja käytössä olevia prosesseja, muisti ei säily koneen sammuesssa (TechTarget, 2021). Omassa opinnäytetyössäni käytin T3.large instanssia, kyseisessä instanssissa on 2 virtuaalistaprosessoria ja 8 GiB RAMia. (Amazon, 2022c)

3.2 AWS Cloud9

Cloud9 on Amazonin IDE, integrated development environment, eli ohjelmointiympäristö. Cloud9:ssa on valmiiksi yleisimpien ohjelmointikielien vitaalit kirjastot, jolloin omalle koneelle asennettavien kirjaston ja pakettien määrä vähenee. Palvelu on täysin pilvipohjainen, joten ohjelmien rakentaminen ja editointi onnistuu mistä tahansa ja millä tahansa laitteella, jolla on pääsy internettiin. Etuna on myös työn vaivaton jatkaminen siitä, mihin on aiemmin jäänyt. Cloud9 mahdollistaa kehitysympäristön käytön hallitussa EC2 instanssissa ja tämä oli myös tapa. Sovelluksessa on myös suora terminaalilyhteys AWS:n command line interface:n, eli komentokehotteeseen, joten Amazonin muitakin palveluita on helppo hallita terminaali-ikkunasta. (Amazon, 2022a)

3.3 AWS CodeCommit

AWS CodeCommit on AWS:n versionhallintapalvelu. Palveluun voi ladata omat lähdekoodit, työskennellä projektin parissa tiimeissä, käyttää eri haaroja projektille, sekä hallita nykyistä versiota ja menneitä versioita. CodeCommitissa kaikki säilöt ovat salattuja, sekä levossa, että kuljetuksen aikana. Levossa oleva data, data at rest, on dataa, joka ei aktiivisesti liiku laitteiden välillä. Kuljetuksessa oleva data, data at transit, on dataa, joka liikkuu laitteiden välillä, joko lähiverkossa tai internetissä. (Amazon, 2022b)

3.4 Amazon Amplify

Amazon Amplify on AWS:n tarjoama ratkaisu full stack applikaatioiden luomiseen ja jakeluun. Full stack kehitys pitää sisällään applikaation front endin, eli gaafisen käyttöliittymän, sekä back endin joka vastaa applikaation logiikasta. Amplifyllä voi luoda applikaation taustapalvelu (backend) logiikan, eli miten applikaatio toimii ja kommunikoi tietokannan kanssa vain muutamalla klikkauksella. Applikaation käyttöliittymän voi koota klikkaus ja raahaus tyyppisesti AWS Amplify Studiolla. AWS Amplify Hosting, on palvelu, jota käytin työssäni lopullisen sovelluksen julkaisemiseen. (Amazon, Amplify, 2022.)

AWS Amplify Hosting on täysin hallittu CI/CD ja hosting palvelu. CI/CD, tulee sanoista continuous integration ja continuous delivery, eli jatkuva integraatio ja jatkuva toimitus.

Jatkuva integraatio tarkoittaa sitä, että koodi muutosten integrointi on automatisoitu keskeiseen koodisäilöön, jossa testit ja rakennukset suoritetaan. Jatkuvassa toimituksessa kehitetään ohjelmistoja lyhyissä sykleissä, jolloin päivityksiä voidaan tehdä melkein koska tahansa. Jatkuva toimitus on myös automatisoitu, jolloin päivitykset julkaisuun tapahtuvat välittömästi. Amplify Hosting tukee lukuisia yhdensivun-sovelluksen frameworkkeja.

Yksisivuinen sovellus on web applikaatio, joka lataa vain yhden web dokumentin ja päivittää sen rakennetta. (Amazon, 2022)

3.5 Amazon Translate

Amazon Translate tarjoaa käyttäjilleen erittäin laadukasta, nopeaa, edullista, sekä kustomoitua kielen käännöspalvelua. Käännöstyö tapahtuu hyödyntäen koneoppimista, ja tukee yli 5500 kielikombinaatiota. (Amazon, 2023f)

3.6 Jekyll

Jekyll ei ole Amazonin oma palvelu, vaan se on projektissani käyttämä Ruby-pohjainen kirjasto staattisen sisällön dynaamiseen generoimiseen. Jekyll kuvailee itseään ”yksinkertaiseksi” ja ”staattiseksi”. Käytännössä tämä tarkoittaa, että luodakseen dynaamista sisältöä Jekyll ei vaadi tietokantaa. Tietokannan sijaan Jekyll käyttää liquid templatea, joka generoi sisällön YAML, JSON, CSV tai TSV tiedostosta. Template kielen avulla voidaan sisällyttää dynaamista sisältöä staattisiin verkkosivuihin. Liquid on Shopify:n luoma, avoimenlähdekoodin template, joka on kirjoitettu Ruby ohjelmointikielellä. Toimiakseen Jekyll tarvitsee Rubyn asennuksen, ja tarkemmin Ruby 2.5.0 version tai sitä uudemman. (Jekyll, 2022.)

4 Tavoitteet

Työn tavoitteena on luoda toimiva työkalu, joka on mahdollista julkaista Amazonin työntekijöiden sisäiseen käyttöön ja helpottamaan työntekijöiden työskentelyä. Tavoitteena on toteuttaa AWS:n työntekijöiden käyttöön työkalu, josta näkee nopeasti mitä eri AWS:n palvelut tekevät.

Annetut vaatimukset projektille olivat kehittää ja toteuttaa AWS:n sisäiseen käyttöön tarkoitettu työkalu helpottamaan AWS:n palveluiden valinnassa. Työkalun tarkoituksena on lyhentää sovellusten kehitysaikoja, sekä helpottaa markkinointia vähentämällä aikaa, joka työntekijöillä menee oikeiden palvelujen valinnassa asiakkaalle. Oma vaatimukseni projektille on, että ratkaisun tulee pysyä hyvin yksinkertaisena, ja sitä tulee olla nopea käyttää. "Cheat sheet" tyyppisen ratkaisun täytyy olla sellainen, että siihen pääsee käsiksi nopeasti ja sitä ymmärtää todella helposti.

Tavoitteita lähdettiin muodostamaan keskusteluilla työtovereiden kanssa siitä, mitä mahdollisia ominaisuuksia sovelluksella voisi olla. Hakiessa tukea myös Amazonin omista ohjeista, leadership principleista, jotka ovat Amazonin ohjenuora työntekijöille. Ilmoille nousi myös ajatuksia, jotka olivat hyvin lähellä mahdottomia toteuttaa, mutta kaikista näistä ajatuksista ja keskusteluista nousi pinnalle päämäärä, sekä suuntaviivat, joita pitkin projektia lähdettiin toteuttamaan.

5 Työkalun toteutus

Valmis tuote on kehitetty Cloud9 ympäristössä. Cloud9 on rakennettu EC2:n päälle, ja käyttöjärjestelmänä on Amazon Linux 2. Amazon Linux2 on AWS:n Linux käyttöjärjestelmä. Se tarjoaa turvallisuuden keskittyvän ympäristön pilvisovellusten kehittämiseen sekä suorittamiseen. (Amazon, 2023e)

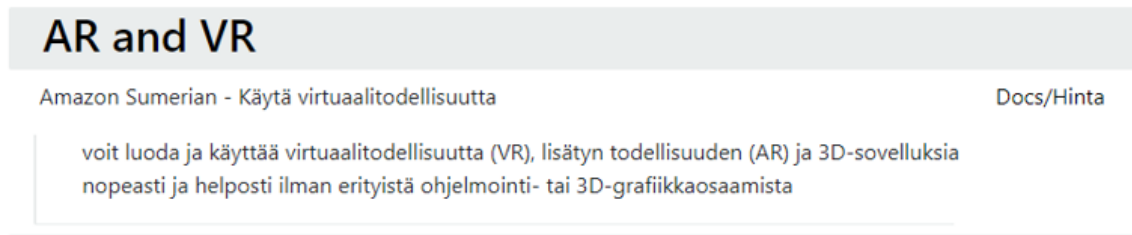
5.1 Alkuperäinen suunnitelma

Alkuperäinen toteutussuunnitelma oli luoda sivusto käyttäen ohjelmointikieli-pythonia ja sen web kehitykseen tarkoitettua kehystä, Djangoa. Kehitysvaiheessa hyödynnettiin Djangon tarjoamaa testitietokantaa, joka on integroituna Djangoon. Tämä mahdollisti sen, että kehitys vaiheessa ei ollut tarvetta yhdistää ohjelmaa ulkoiseen tietokantaan. Foreign key on relaatiotietokannoissa käytetty tieto, jolla yhdistetään kahdessa eri taulussa olevaa tietoa. Tietokannan rakenne luotiin sellaiseksi, että jokainen palvelu on linkitetty Foreign keyn avulla omiin kategorioihinsa, mahdollistaakseen palveluiden haun kategorioittain.

5.2 Muutos Jekylliin

Alkuperäisen toteutuksen ollessa jo melkein valmis, jouduttiin kuitenkin tekemään muutos käyttämään Jekylliä projektin toteutuksessa. Keskustelut työkavereiden kanssa projektin toteutuksesta ja tekniikoista, johtivat tulokseen, että on järkevämpää vaihtaa pois tietokantaa käyttävästä versiosta. Ratkaisuun päädyttiin, koska tietokanta lisää huomattavasti hyökkäyspinta-alaa pahantekijöille, ja lisäksi tietokanta vaatii säännöllisiä päivityksiä ja enemmän ylläpitoa. Hyökkäyspinta-alalla tarkoitetaan kaikkia internetiin näkyviä tietojärjestelmiä sekä niiden avoinna olevia portteja. Staattinen alusta varmistaa, ettei tiedot ole millään tavoin ulkoisesti muokattavissa.

Kuva 2. Miten lisätiedot palveluista tulee näkyviin pitämällä hiirtä palvelun päällä.



Suomennotokset palveluiden kuvauksiin, joita käytettiin koko virkkeen lisätiedossa, löytyi Amazonin omilta dokumentaationsivuilta ja ne käännettiin suomeksi käyttäen Amazonin Translate palvelua, eli Amazonin käännöspalvelua. Lyhyissä neljän sanan kuvauksissa päästiin käyttämään enemmän omaa luovuuttani, sillä mitään suoraa lähdemateriaalia, josta olisi kaikkien palveluiden suomennotokset mahdollista tuoda, ei ollut.

5.3.1 Toteutus

Projekti on kehitetty käyttäen Amazonin Cloud9 selain pohjaista IDE:tä. Käyttöjärjestelmänä on Amazon Linux 2. Kotisivu on dynaamisesti generoitua staattista sisältöä, joka on generoitu käyttäen Jekyllin liquid templatea, josta esimerkki (kuva 3.).

Kuva 3. Jekyllin liquid komentoja.

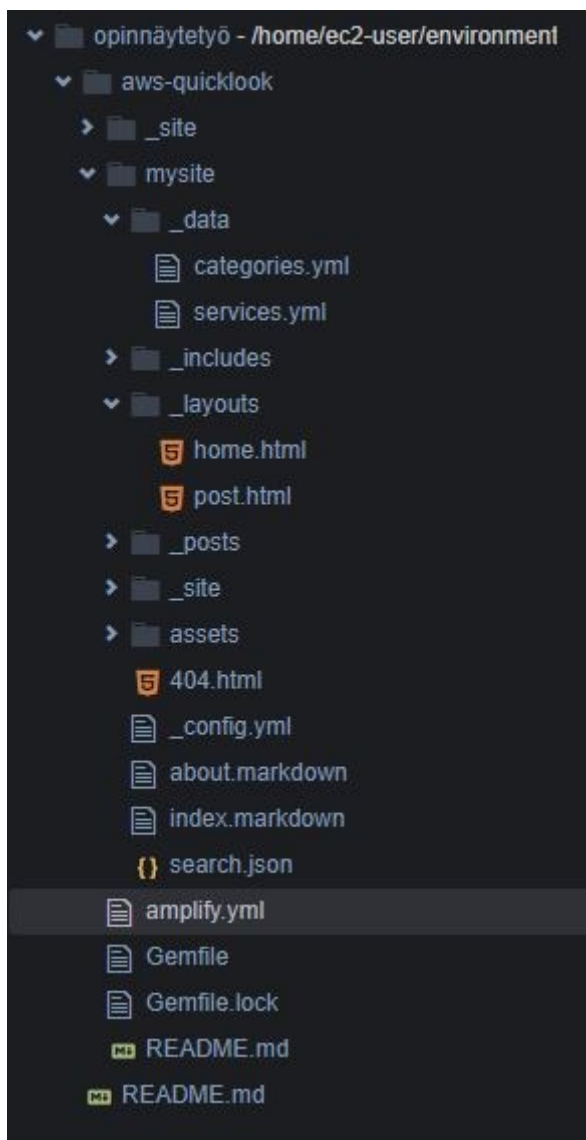
```

{% for post in sorted_posts %}
<div>
  <div class="otsikkoTausta">
    <h2 class="list-group-item otsikkoH2 h-font">
      <a href="{{ post.url }}.html">{{ post.title }}</a>
    </h2>
  </div>
  {% for service in site.data.services %}
  {% if service.category == post.title %}
  <!-- hover div-->
  <div class="hoverDiv">
    <p class="palvelut">{{service.name}} - {{service.four_worder}}
      <a target="_blank" class="linkki" href="{{service.url}}">Hinta</a>
      <a target="_blank" class="linkki" href="{{service.docs}}">Docs</a>
    </p>
    <!--hidden before hover-->
    <div class="piilosilla" id="piiloID">
      <p class="piilosillaP">{{service.suomeksi}}</p>
    </div>
  </div>
  {% endif %}
  {% endfor %}
</div>
{% endfor %}
</div>

```

Jekyll kykenee luomaan dynaamista sisältöä ilman tietokantaa, lataamalla sisältöä tiedostoista, jotka ovat `_data` tiedostopolkuun sijoitettuja selvennykseksi kuva tiedosto rakenteesta (kuva 4.)

Kuva 4. Tiedostorakenne



_layouts tiedostossa on kotisivun html tiedosto, sekä post.html tiedosto, joka toimii pohjana kaikille kategorioittain lajitelluille sivuille. Post.html pohjaa seuraavassa kuvassa (kuva5.).

Kuva 5. Post.html pohja

```

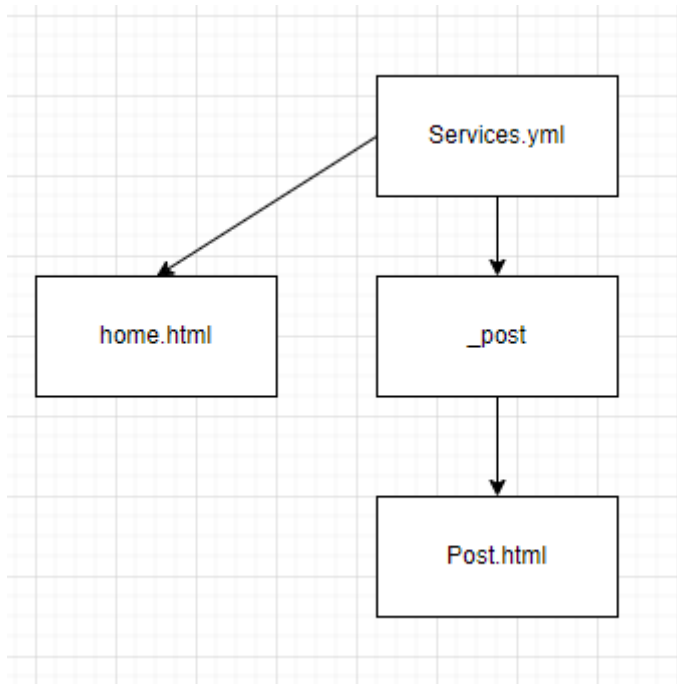
<div class="row">
  <div class="col-md-3 mt-2">
    <div class="card">
      <div class="card-header" style="background-color:#eaeaed;">
        Categories
      </div>
      <!--List of the categories we will have-->
      <ul class="list-group list-group-flush">
        <li class="list-group-item">
          <a href="index.html">All</a>
        </li>
        {% assign sorted_posts = site.posts | sort: 'title' %}
        {% for post in sorted_posts %}
        <li class="list-group-item">
          <a href="{{ post.url }}">
            {{post.title}}</a>
          </li>
        {% endfor %}
      </ul>
    </div>
  </div>
  <div class="col">
    {{ content }}
  </div>
</div>

```

Kategoria palkki kirjoitettiin uudestaan ja kuvan alareunassa näkyvässä div-elementissä, jonka luokka on col1, jekyll generoi sisällön, joka tulee _post kansioista.

Rakenne työssä toimii niin, että service.yml tietosto on kaiken avain. Etusivulla oleva data saadaan tuotua suoraan service tiedostosta. Kategorioittain lajitellessa, _post kansio sisältää html kielisen pohjan, johon sitten täytetään halutun kategorian palvelut. Tätä pohjaa hyödynnetään post.html tiedostossa kohdassa {{ content }}. Kaavio, jossa kuvattu miten tiedostot toimivat keskenään (kuva 6.)

Kuva 6. Tiedostojen toiminta keskenään.



5.3.2 Hosting

Sovelluksen hostaus verkossa tapahtuu käyttäen Amazonin Amplify-palvelua. Amplify käyttää projektin juuressa olevaa `amplify.yml` tiedostoa (kuva 7.), jossa on määritettynä rakenteen asetukset.

Kuva 7. amplify.yaml tiedosto, joka sisältää komennot, joiden avulla amplify rakentaa sovelluksen.

```
1 |version: 1
2 frontend:
3   phases:
4     preBuild:
5       commands:
6         - gem install bundler:1.17.3
7         - bundle install
8       # IMPORTANT - Please verify your build commands
9     build:
10      commands:
11        - cd mysite
12        - bundle exec jekyll build
13  artifacts:
14    # IMPORTANT - Please verify your build output directory
15    baseDirectory: /mysite/_site
16    files:
17      - '**/*'
18  cache:
19    paths: []
```

Version asetus määrittelee sen mitä YAML versiota Amplify käyttää. Frontend ajaa frontendin rakentamiskomennot, ja se koostuu kahdesta eri vaiheesta: preBuild, sekä build. PreBuild ajaa määritetyt komennot ennen kuin varsinaista koontiversiota. Build suorittaa tarvittavat komennot sovellukset saattamiseksi päälle. CD-komento siirtyy oikeaan kansioon ja sitten rakennetaan Jekyll-sovellus.

Lähdekoodin säilönä käytössä oli AWS CodeCommit. Amplify tukee CI/CD pohjaisia ratkaisuja, jolloin se seuraa koodisäilöä, jossa lähdekoodi on, ja aina kun seurattuun projektiin tehdään muutoksia Amplify rakentaa sovelluksen uudestaan. Vanha versio vaihtuu uuteen vasta, kun uusi on saatu valmiiksi, joten päivitykset tapahtuvat ilman alhaalla olo aikaa.

5.3.3 Lopputulos

Alkuperäinen idea opinnäytetyölle tuli työharjoittelustani. Amazon Web Service:llä on yli 200 erilaista palvelua, joten myös teknisillä työntekijöillä riittää tehtävää muistaessa mitä minkäkin niminen palvelu tekee ja tarjoaa. Tämän idean ympärille sain vapaat kädet, miten lähteä toteuttamaan kyseinen työ. Tapaamisissa työtovereideni kanssa, opin myös sen, että varsinkin julkisen sektorin asiakkaat haluavat usein käyttää keskusteluissaan suomen kieltä, joten tämä toi myös lisää tarpeellisuutta luoda kyseinen ”cheat sheet” suomen kielellä.

Henkilökohtaisiin tavoitteisiini kuului myös oman osaamiseni ja ammattitaidon kehittäminen, erityisesti pilvipalveluiden ja verkkosovellusten kehittämisen osalta. Suoritin opinnäytetyön ohella harjoittelun samalle yritykselle, joten projektin lisätavoitteena oli saada suoritettua harjoittelu onnistuneesti, jotta minun olisi mahdollista myös palata työskentelemään Amazonille valmistuttuani.

6 Jatkokehitysmahdollisuudet

Sain sovelluksen pisteeseen, missä palvelu on toimiva, ja se täyttää kaikki projektin alkuperäiset tavoitteet. Projektia työstäessäni, sain kuitenkin lisää ajatuksia mitä työhän olisi mahdollista lisätä, mutta työn luomien aikaresurssien puitteissa kaikkea mitä ehdin suunnitella en projektiin ehtinyt sisällyttää. Sovellukseen olisi mahdollista lisätä automatiikkaa, joka mahdollistaisi tietojen lisäyksen, poistamisen, sekä päivittämisen. Sovellus voisi esimerkiksi hakea tiedot AWS:n dokumentaationsivuilta, sekä ajaa käännökset Translate API:n (Application programming interfacen) avulla. Toinen mahdollinen kehitysidea olisi mahdollisuus käyttää palvelua myös muilla kielillä, esimerkiksi englanniksi.

Lähteet

Amazon. (2022). *AWS Amplify*. Haettu 05. 12 2022 osoitteesta

<https://aws.amazon.com/amplify/hosting/?nc=sn&loc=2&dn=2>

Amazon. (2022a). *Cloud9*. Haettu 05. 12 2022 osoitteesta <https://aws.amazon.com/cloud9/>

Amazon. (2022b). *CodeCommit*. Haettu 05. 12 2022 osoitteesta

<https://aws.amazon.com/codecommit/>

Amazon. (2022c). *Instance types*. Haettu 05. 12 2022 osoitteesta

<https://aws.amazon.com/ec2/instance-types/t3/>

Amazon. (2022d). *What is AWS*. Haettu 05. 12 2022 osoitteesta

<https://aws.amazon.com/what-is-aws/>

Amazon. (2023e). *Amazon Linux 2*. Haettu 01. 03 2023 osoitteesta

<https://aws.amazon.com/amazon-linux-2/?amazon-linux-whats-new.sort-by=item.additionalFields.postDateTime&amazon-linux-whats-new.sort-order=desc>

Amazon. (2023f). *Translate*. Haettu 09. 03 2023 osoitteesta

<https://aws.amazon.com/translate/>

Jekyll. (2022). *Jekyllrb*. Haettu 05. 12 2022 osoitteesta <https://jekyllrb.com/>

Statista. (2022). *Statista*. Haettu 05. 12 2022 osoitteesta

<https://www.statista.com/statistics/967365/worldwide-cloud-infrastructure-services-market-share-vendor/>

TechTarget. (2021). *RAM*. Haettu 05. 12 2022 osoitteesta

<https://www.techtarget.com/searchstorage/definition/RAM-random-access-memory>

Yhdysvaltain standardisointi- ja teknologiainstituutti. (09 2011). *The NIST Definition of Cloud Computing*. Haettu 09. 03 2022 osoitteesta

<https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf>