

Kotlin-ohjelmoinnin verkkokurssi aloittelijoille

Teija Alasalmi

Tekijä(t) Teija Alasalmi	
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma	
Raportin/Opinnäytetyön nimi Kotlin-ohjelmoinnin verkkokurssi aloittelijoille	Sivu- ja liitesivumäärä 23 + 8
<p>Tämän toiminnallisen opinnäytetyön tarkoituksena oli luoda yhden opintopisteen laajuinen itseopiskeluun tarkoitettu, ajasta ja paikasta riippumaton suomenkielinen verkkokurssi ja sen aineisto Kotlin-ohjelmointikielestä GitHub-alustalle kenelle tahansa ohjelmoinnin perusteista kiinnostuneelle aloittelijalle. Toisena tavoitteena oli raportoida verkkokurssin suunnitteluun ja toteutukseen liittyvistä toimenpiteistä.</p> <p>Ajatus Kotlin-ohjelmoinnin verkkokurssista syntyi suomenkielisen Kotlinia käsittelevän sähköisen oppimateriaalin puutteesta. Siten opinnäytetyöllä ei ollut toimeksiantajaa. Opinnäytetyön tuloksena syntyneen verkkokurssin tavoitteena oli tutustuttaa kurssin opiskelija Kotlin-ohjelmointikielen perusteisiin. Verkkokurssia varten tietoa haettiin Kotlin-ohjelmointikielen dokumentaatioista, aloittelijoille suunnatusta Kotlinin opetussuunnitelmasta sekä Kotlinia käsittelevistä kirjoista. Kootun tiedon pohjalta tuotettiin viidestä teoriaosasta ja teoriaan liittyvistä tehtävistä, tehtävien malliratkaisuista, usein kysytyistä kysymyksistä ja Kotlin-suomi-sanastosta koostuva verkkokurssi, joka kirjoitettiin Markdown-merkintäkielellä ja julkaistiin GitHub-alustalla.</p> <p>Verkkokurssin suunnittelussa ja toteutuksessa on tärkeää huomioida etenkin kohderyhmä, jotta materiaalista saa helposti lähestyttävän ja tarkoituksenmukaisen. Lisäksi erityisesti ohjelmointia käsittelevää verkkokurssia suunniteltaessa ja toteuttaessa tulee kiinnittää huomioita kurssin sisältämiin aiheisiin, sillä aloittelijoille suunnatulle kurssille on hyvä valita ohjelmoinnin peruskäsitteitä käsitteleviä aihekokonaisuuksia. Myös verkkokurssin toteuttamiseen kuluu enemmän aikaa kuin itse kurssin opiskeluun.</p> <p>Kotlin-ohjelmoinnin verkkokurssin oppimateriaalin tuottaminen Markdown-merkintäkielellä osoittautui hyväksi valinnaksi, sillä oppimateriaalin luominen onnistui tavallisella tekstieditorilla. Lisäksi oppimateriaali on lähdekoodimuodossaan varsin luettavaa tekstiä, joka on helposti muunnettavissa esimerkiksi pdf-muotoon. Markdown-kielellä kirjoitettu kurssi on myös tarvittaessa siirrettävissä kokonaan toiselle kurssialustalle. GitHub kurssialustana toimi hyvin, sillä kurssin oppimateriaalin päivitys oli helppoa Git-versionhallintatyökalun avulla. Itse sivuston julkaisu oli vaivatonta GitHub Pages -työkalun avulla.</p> <p>Kurssin jatkokehitysideoita ovat muun muassa kurssin palaute- ja kommunikointikanavan luominen sekä kurssin laajentaminen Kotlin-hiekkalaatikon virheiden tulkitsemisvinkkien, ruutukaappausvideoiden ja ulkoisten materiaaliinlinkkien avulla.</p>	
Asiasanat Kotlin, verkkokurssi, ohjelmointi, sähköinen oppimateriaali	

Sisällys

1	Johdanto	1
2	Sähköiset oppimateriaalit ja laatukriteerit	2
2.1	Sähköinen oppimateriaali.....	2
2.2	eAMK verkkototeutusten laatukriteerit.....	3
3	Kotlin.....	5
3.1	Kotlinin ominaisuudet.....	5
3.2	Kotliiniin liittyvät avoimet verkko-oppimateriaalit.....	8
4	Verkkokurssin toteutus	9
4.1	Kohderyhmä ja oppimistavoitteet	9
4.2	Kurssialusta, kurssin ulkoasu ja navigointi	10
4.3	Verkkokurssin sisältö	12
4.4	Ajankäyttö	15
5	Pohdinta.....	17
	Lähteet	21
	Liitteet.....	24
	Liite 1. Kotlin-ohjelmoinnin verkkokurssin oppimistavoitteet	24
	Liite 2. eAMK verkkototeutuksen laatukriteerit.....	25
	Liite 3. Pikaohje laadukkaan verkkokurssin toteuttamiseksi.....	31

1 Johdanto

Verkko-opiskelu on nykypäivänä tuttua yhä useammalle opiskelijalle ja työssäkäyvälle. Etä- ja verkko-opiskelun suosio on kasvanut teknologian kehittyessä, ja erilaisia kurssisäilytöjä ja jopa kokonaisia tutkintoja voi nykyään opiskella verkossa ajasta ja paikasta riippumatta. Lisäksi maailmanlaajuinen pandemia vain kasvatti verkossa opiskelun tarvetta.

Tässä toiminnallisessa opinnäytetyössä käsitellään verkko-oppimista ja avoimen, itseopiskeluun sopivan verkkokurssin toteuttamista GitHub-alustalle. Verkkokurssin aiheena on Kotlin-ohjelmointikieli. Valitsin aiheeksi Kotlinin, koska kyseisen ohjelmointikielen taito on kysyttyä nykypäivän IT-alan työtehtävissä mobiiliohjelmoinnin saralla. Lisäksi olen kiinnostunut mobiiliohjelmoinnista, ja haluaisin työskennellä aiheen parissa tulevaisuudessa. Kotlinista ei myöskään löytynyt aiempaa suomenkielistä oppimateriaalia verkosta, ja lisäksi löytämäni englanninkieliset oppimateriaalit oli suunnattu pääasiassa ohjelmistokehittäjille, jotka osaavat jo entuudestaan esimerkiksi Java-ohjelmointikieltä.

Tässä opinnäytetyössä kuvataan verkkokurssin toteuttamisprosessin kulkua mallilla, jossa oppimateriaali tuotetaan vaiheittain. Verkkokurssille suunnitellaan aluksi tavoitteet ja oppimistavoitteet sekä valitaan kohderyhmä ja kurssialusta, minkä jälkeen siirrytään oppimateriaalin ja tehtävien laatimiseen. Lopuksi verkkokurssi itsearvioidaan eAMK verkkototeutusten laatukriteereitä apuna käyttäen. Työssä tuodaan esille myös verkkokurssin toteuttamisen aikana tehdyt havainnot ja ajatukset jatkokehitystä varten.

Tällä toiminnallisella opinnäytetyöllä oli kaksi tavoitetta. Ensimmäisenä tavoitteena oli luoda aloittelijoille sopiva, ajasta ja paikasta riippumaton, itseopiskeluun tarkoitettu suomenkielinen verkkokurssi Kotlin-ohjelmointikielen perusteista. Toisena tavoitteena oli raportoida verkkokurssin suunnitteluun ja toteutukseen liittyvistä toimenpiteistä, jotta kenen tahansa olisi tulevaisuudessa helpompaa luoda vastaavanlainen itseopiskeluverkkokurssi.

Toivon, että opinnäytetyöstäni olisi hyötyä etenkin sellaisille lukijoille, jotka ovat kiinnostuneita verkko-oppimisesta tai verkkokurssin suunnittelusta ja toteuttamisesta. Uskon, että työni tuloksena syntyneestä Kotlin-ohjelmoinnin verkkokurssista on hyötyä kaikille niille, jotka haluavat saavuttaa nykypäivän IT-alan työtehtävissä vaadittavia taitoja oppimalla ohjelmoinnin perusteita suomen kielellä. Tämän opinnäytetyön tuloksena tuotettu täysin suomenkielinen, Kotlinin perusteita käsittelevä kaikille avoin verkkokurssi on lisäksi toiseksi ainoa laatuaan. Kokonaisuutena tuotettu verkkokurssi toimii myös yhtenä osoituksena ohjelmointitaidostani ja erilaisten ohjelmointityökalujen käyttötaidosta, sekä henkilökohtaisena työnäytteenä ohjelmointiin liittyvän sähköisen oppimateriaalin tuottamisesta.

2 Sähköiset oppimateriaalit ja laatukriteerit

Tässä luvussa kuvataan sähköisen oppimateriaalin käsitettä sekä laadukkaiden oppimateriaalin ominaispiirteitä. Lisäksi kuvataan lyhyesti eAMK verkkototeutusten laatukriteereitä.

2.1 Sähköinen oppimateriaali

Sähköinen oppimateriaali voidaan käsitteenä määritellä monella eri tavalla. Laajassa merkityksessä sähköiseksi oppimateriaaliksi luetaan mikä tahansa verkossa saatavilla oleva, oppimateriaaliksi tarkoitettu sisältö (Ilomäki 2012). Sähköisiä oppimateriaaleja ovat siten esimerkiksi digitalisoidut kirjat tai kokonaiset verkko-oppimisympäristöt materiaaleineen (Ekonoja 2014) sekä muun muassa opetukseen tarkoitettut kuvapankit, itsenäiset verkko-kurssit ja oppikirjojen oheismateriaalit (Opetushallitus 2021).

Sähköisestä oppimateriaalista voidaan käyttää myös muita nimityksiä, kuten elektroninen oppimateriaali, e-oppimateriaali, digitaalinen oppimateriaali tai verkko-oppimateriaali. Näille erilaisille nimityksille yhteisenä piirteenä on, että oppimateriaali on käytettävissä tietokoneella tai tietoteknisellä laitteella. (Ekonoja 2014.) Mikään nimitys ei kuitenkaan ole vakiintunut (Opetushallitus 2021). Tässä opinnäytetyössä sähköisellä oppimateriaalilla tarkoitetaan loogisesti eri osista koostuvaa oppimateriaalia, joka on julkaistu verkossa ja joka on opiskeltavissa tietokoneella tai mobiililaitteella.

Mikä tahansa oppimateriaali on aina kytköksissä pedagogiikkaan, sillä oppimateriaalista löytyy yleensä jokin pedagoginen lähtökohta, joka omalta osaltaan vaikuttaa siihen, mitä oppimateriaalilla voi tehdä. Myös valittu oppimisalusta vaikuttaa siihen, miten hyvin oppimateriaali tukee tiettyjä toimintoja tai tietyntapaista oppimista. Oppimisalustat sisältävät erilaisia toimintoja yhteisöllisistä toiminnallisuuksista oppimisanalytiikkaan, mutta yksittäisestä oppimateriaalista ei välttämättä löydy kaikkia toimintoja. (Opetushallitus 2021.)

Kuten minkä tahansa oppimateriaalin, myös sähköisten oppimateriaalien laatu voi vaihdella suuresti. Kaikkia verkosta löytyviä materiaaleja ei välttämättä ole suunniteltu opetuskäyttöön. Lisäksi juuri tiettyyn aiheeseen liittyvään, opetuskäyttöön suunniteltujen sähköisten oppimateriaalien saatavuus voi ylipäätään olla haasteena. Erityisesti tietotekniikan oppimateriaaleja tuotetaan paljon alusta lähtien opettajien toimesta, ja lisäksi oppimateriaali on voitu koostaa useista eri lähteistä, joiden ajantasaisuus voi vaihdella. Tietotekniikassa opetettavat asiat myös vanhentuvat nopeammin kuin monilla muilla aloilla, mikä lisää oppimateriaalien päivitystarvetta. (Ekonoja 2014.)

Laadukkaan oppimateriaalin tunnistaa muun muassa siitä, että laadukkaassa oppimateriaalissa on huomioitu oppimistavoitteet ja kohderyhmä. Lisäksi oppimateriaalin sisältö on suunniteltu suhteessa oppimistavoitteisiin ja kohderyhmään. (Ellington & Race 1993.) Laadukkaassa oppimateriaalissa oppimistehtävät ovat haasteellisia, avoimia ja autenttisia (Opetushallitus 2021), tehtäviä on runsaasti ja niitä on eriytetty. Laadukas oppimateriaali motivoi oppilaita, kannustaa heitä monipuolisiin työtapoihin ja mahdollistaa myös oppilaiden itsenäisen opiskelun. (Heinonen 2005.) Sähköisissä oppimateriaaleissa opittavan asian esittämisessä tulee käyttää hyväksi verkon teknisiä mahdollisuuksia, kuten vuorovaikutteisuutta, jakamista ja linkityksiä (Opetushallitus 2021). Lisäksi oppimateriaalin rakenteen tulee olla havainnollinen (Heinonen 2005).

2.2 eAMK verkkototeutusten laatukriteerit

Sähköisen oppimateriaalin laatua voidaan arvioida etukäteen määritettyjen laatukriteerien avulla. Oppimateriaalin arvioimiseen on olemassa useita erilaisia laatukriteereitä, joita ovat Suomessa määritelleet ainakin Opetushallitus (2021) sekä eAMK-hanke (Varonen & Hohenthal 2017). Yleisesti laatukriteerien tarkoituksena on tarjota sähköisten oppimateriaalien kehittäjille eräänlainen tarkistuslista oppimateriaalin keskeisistä ja pedagogisesti järkevistä piirteistä (Ilomäki 2012).

Tässä opinnäytetyössä tuotettu verkkokurssi arvioidaan aAMK verkkototeutusten laatukriteerien avulla. Useisiin eurooppalaisiin laatukriteeristöihin pohjautuva eAMK verkkototeutusten laatukriteeristö on laadittu osana eAMK-hanketyötä syksyllä 2017. Alun perin laatukriteeristö on tarkoitettu ammattikorkeakoulujen ja CampusOnline.fi verkko-opintojen laadun kehittämiseksi, mutta sitä voidaan käyttää myös minkä tahansa verkkototeutuksen arviointiin sekä myös itsearviointiin arvioimalla toteutus käyttäen verkosta löytyvää arviointityökalua. (Varonen & Hohenthal 2017; eAMK 2021.)

Liitteessä 2 kuvatut eAMK verkkototeutusten laatukriteeristö koostuu yhdestätoista osa-alueesta, joita ovat kohderyhmä ja käyttäjät; osaamistavoitteet, oppimisprosessi ja pedagogiset ratkaisut; tehtävät; sisältö ja aineistot; työvälineet; vuorovaikutus; ohjaus ja palaute; arviointi; kehittäminen; käytettävyyden ulkoasu sekä tukipalvelut. Jokaisessa osa-alueessa huomioidaan lisäksi miten aihe – esimerkiksi kohderyhmä ja käyttäjät – tulisi ottaa huomioon suunnittelun ja tuotannon vaiheessa sekä varsinaisen toteutuksen aikana (Varonen & Hohenthal 2017.)

eAMK verkkototeutusten laatukriteeristön (Varonen & Hohenthal 2017) mukaan laadukas verkkokurssi huomioi käyttäjät ja heidän tarpeensa. Lisäksi osaamistavoitteiden, oppimisprosessin ja pedagogisten ratkaisuiden tulee olla työelämälähtöisiä ja osaamisperustaisia,

ja valittujen pedagogisten ratkaisuiden tulee olla linjassa opintojakson oppimiskäsityksen kanssa. Verkkokurssin tehtävien, sisällön, aineiston, työvälineiden ja vuorovaikutuksen tulee edistää osaamistavoitteiden saavuttamista työelämälähtöisesti, ja oppijoiden yksilölliset tarpeet tulee huomioida valitsemalla yhteisöllistä oppimista tukevia työtapoja. Lisäksi ohjauksen ja palautteen on hyvä olla oikea-aikaista, ja sitä tulee olla saatavissa koko opintojakson ajan. Kurssin arviointiperusteiden tulee olla läpinäkyviä, ja arvioinnin tulee olla jatkuvaa, monipuolista ja kannustaa itsearviointiin. Verkkokurssia tulisi myös kehittää jatkuvasti, kurssin ulkoasun ja käytettävyyden tulee olla selkeä. Kurssin tukipalveluiden tulee tarjota tukea pedagogisiin ja teknisiin haasteisiin. (Varonen & Hohenthal 2017.) Teknisiä haasteita voivat aiheuttaa esimerkiksi kurssialusta tai valittu merkintä- tai ohjelmointikieli.

3 Kotlin

Kotlin on avoimeen lähdekoodiin perustuva staattisesti tyyplitetty yleiskäyttöinen ohjelmointikieli, jonka kehitys aloitettiin vuonna 2010. Kotlin-ohjelmointikielen pääasiallisena kehittäjänä toimii tšekkiläinen JetBrains-yhtiö ja sen yli sadasta insinööristä koostuva ydintiimi sekä yli 350 ulkopuolista kehittäjää GitHubissa. Kotlinin ensimmäinen virallinen versio Kotlin 1.0 julkaistiin helmikuussa 2016, ja tämän opinnäytetyön kirjoitushetkellä Kotlinista on saatavilla jo versio 1.6.0. (Kotlin 2021a.)

Kotlinin kehittäjät suunnittelivat Kotlinista vaihtoehtoa Javalle, ja Kotlin onkin noussut suosituimmaksi ohjelmointikieleksi Android-kehityksessä Javan sijaan (Developer.android.com 2021a). Kielen suosiosta kertoo sekin, että vuonna 2017 Google ilmoitti tukevasa Kotlinia Android-kehitysympäristössään. Lisäksi lokakuussa 2018 JetBrains ja Google perustivat voittoa tavoittelemattoman Kotlin Foundation -säätiön, jonka tehtävänä on suojata ja edistää Kotlin-ohjelmointikielen kehitystä. Toukokuussa 2019 Google nosti Kotlinin Android-käyttöjärjestelmän ensisijaiseksi ohjelmointikieleksi Javan ja C++:n ohi. (Kotlin 2021b.)

Nykyään yli 4 miljoonaa ohjelmoijaa käyttää Kotlinia. Android- ja iOS-mobiiliohjelmoinnin lisäksi Kotlinia käytetään palvelinohjelmoinnissa, työpöytäsovelluksissa sekä web-ohjelmointiin esimerkiksi Spring Boot -kehityksen kanssa. Lisäksi Kotlinia voidaan käyttää data-analytiikassa. (Kotlin 2021b; Moskala & Wojda 2017.) Muun muassa Google, Uber, Atlasian, Pinterest, Square ja Corda käyttävät Kotlinia tuotantokoodissaan (Kotlin 2021b).

3.1 Kotlinin ominaisuudet

Kotlin-ohjelmointikielen ominaispiirteitä ovat Java-yhteensopivuus, ytimekkyys, turvallisuus, monipuolisuus ja helppous. Kotlin on 100-prosenttisesti yhteensopiva Java-ohjelmointikielen kanssa, sillä Kotlin perustuu Java-virtuaalikoneeseen (engl. *JVM, Java Virtual Machine*). Tämä tarkoittaa sitä, että Kotlin-ohjelmat käännetään Java-yhteensopivaksi tavukoodiksi, ja ohjelmia voidaan suorittaa missä tahansa JVM-ympäristössä. Lisäksi Kotlin-koodia voidaan kutsua Java-koodista ja päinvastoin, joten molempia kieliä voidaan käyttää sujuvasti rinnakkain samassa projektissa. (Kotlin 2021a.)

Kotlinilla voidaan kirjoittaa tiivistä ja ytimekästä koodia, jolloin koodirivejä tulee vähemmän esimerkiksi Javaan verrattuna (Griffiths & Griffiths 2019). Kuvassa 1 esitetään Task-luokka Java-kielellä kirjoitettuna, ja kuvassa 2 puolestaan esitetään sama dataluokka Kotlinilla kirjoitettuna, ja tästä huomataan, kuinka vastaava ohjelmakoodi Kotlinilla tehtynä on

hyvin lyhyt, vaikka se sisältää saman toiminnallisuuden. Kotlin luo automaattisesti dataluokan konstruktorin, luokan muuttujien asettamiseen ja hakemiseen tarvittavat metodit sekä itse luokan vertailuun, kopioimiseen, osiin purkamiseen ja merkkijonona tulostamiseen tarvittavat metodit (Kotlin 2021c). Lisäksi Kotlinista on karsittu kaikki ylimääräiset avainsanat ja toisteisuus esimerkiksi muuttujia luodessa (Jhonson 2021). Näin koodin kirjoittaminen, lukeminen ja ymmärtäminen nopeutuu. Kotlin on suunniteltu käyttämään juuri tarpeeksi toisteista yleiskoodia (engl. *boiler plate code*) luettavuuden varmistamiseksi pitäen asiat kuitenkin ytimekkäinä. (Griffiths & Griffiths 2019.) Javalla ohjelmoijan on kirjoitettava käsin paljon enemmän yleiskoodia (Samuel & Bocutiu 2017).

```
public class Task {  
    private String description;  
    private boolean isCompleted;  
    public Task(String description, boolean isCompleted) {  
        this.description = description;  
        this.isCompleted = isCompleted;  
    }  
    public String getDescription() {  
        return description;  
    }  
    public boolean getIsCompleted() {  
        return isCompleted;  
    }  
    public void setDescription(String description) {  
        this.description = description;  
    }  
    public void setIsCompleted(boolean isCompleted) {  
        this.isCompleted = isCompleted;  
    }  
}
```

Kuva 1. Task-luokka Java-koodilla.

```
data class Task(var description: String, var isCompleted: Boolean)
```

Kuva 2. Task-luokka Kotlin-koodilla.

Kuten kuvasta 2 nähdään, Kotlinilla kirjoitettu koodi on ytimekästä, minkä vuoksi projektissa ylläpidettävää koodia on usein määrällisesti vähemmän. Kun koodi on kokonaisuudessaan luettavampaa ja sitä on vähemmän, ohjelmointivirheitä tapahtuu myös todennäköisesti harvemmin (Jhonson 2021). Turvallisuutta lisää myös vaatimus muuttujan tyyppistä. Kotlin on staattisesti tyypitetty ohjelmointikieli, eli jokaisella muuttujalla on oltava määritelty tyyppi. Kotlinissa mikään muuttuja ei voi lähtökohtaisesti kuitenkaan olla arvotaan tyhjä eli null. Tyhjän arvon sijoittaminen muuttujaan aiheuttaa sen, ettei koodi käänny. Mikäli muuttujaan halutaan tallentaa tyhjä arvo, täytyy muuttuja merkitä ns. vaihtoehtoiseksi lisäämällä muuttujan tyyppin perään kysymysmerkki, kuten on tehty kuvassa 3. Näin muuttuja voi olla tyhjä ilman, että sovellus kaatuu ja antaa virheilmoituksen. Muuttujan arvoa voidaan tarkastella esimerkiksi funktiokutsun yhteydessä turvallisesti käyttämällä "?"-merkkiä, jolloin tyhjä arvo huomioidaan automaattisesti. Tyhjien arvojen automaattinen käsittely lisää siten Kotlinin turvallisuutta. (Heller 2020.)

```
var description: String? = "Kirjoita opinnäytetyö"
println(description?. toString())
```

Kuva 3. Muuttujan merkitseminen vaihtoehtoiseksi ja tyhjän arvon tarkastus.

Ohjelmointikielenä Kotlin on varsin monipuolinen, sillä Kotlinissa on tuki muun muassa korkeamman tason funktioille, lambda-funktioille, inline-funktioille, tail-rekursiolle sekä muille yleisille toiminnoille. Näin ollen Kotlinissa on kaikki funktionaalisen ohjelmointikielen ominaisuudet ja edut. (Heller 2020.) Vaikka Kotlin on täysimittainen funktionaalinen ohjelmointikieli, se säilyttää myös suurimman osan Javan olioluonteisuudesta vaihtoehtoisena ohjelmointityylinä. Lisäksi funktionaalisen ohjelmoinnin ja olio-ohjelmoinnin yhdistäminen on mahdollista. (Kotlin 2021a.) Kotlinin monipuolisuutta lisää myös se, että JetBrains on Kotlinin kehittämisen ohella julkaissut kielen Kotlin Native -version, joka mahdollistaa Kotlinin käytön sellaisillakin alustoilla, joilla JVM:n ajaminen ei ole mahdollista (Kotlin 2021d).

Kotlinin oppiminen on varsin helppoa etenkin kenelle tahansa Javaa entuudestaan osavalle kehittäjälle, sillä Kotlinissa on paljon yhtenäisyyksiä Javan kanssa. Esimerkiksi Kotlinin tyyppijärjestelmä muistuttaa paljon Javaa, joten kehittäjä voi helposti muuntaa olemassa olevan Java-koodinsa Kotlin-koodiksi (Jhonson 2021.) Ensisilmäyksellä Kotlinilla kirjoitettu ohjelmakoodi myös näyttää tiiviimmältä ja virtaviivaisemmalta Java-kielen versiolta. Tästä on esitetty esimerkki kuvassa 4, jossa neljällä koodirivillä luodaan myTask-olionmuuttuja, asetetaan tehtävä suoritetuksi ja tulostetaan tehtävän kuvaus ja tieto tehtävän suorittamisesta näytölle.

```

val myTask = Task("Kirjoita opinnäytetyö", false)
myTask.completed = true
val text = if (myTask.completed) "Kyllä" else "Ei"
println("Tehtävä: ${myTask.description}, suoritettu: $text")

```

Kuva 4. Kotlin-koodia.

Suhteellisen uutena ohjelmointikielenä Kotlin sisältää paljon ohjelmointia helpottavia ominaisuuksia, ja monet Java-editorit kuten IntelliJ IDEA, Android Studio ja Eclipse tukevat myös Kotlinia (Kotlin 2021a). Näin kehittäjä voi hyödyntää olemassa olevia ohjelmointitaitojaan Kotlin-koodia kirjoittaessaan. Kehittäjä voi luoda tehokkaita sovelluksia välittömästi, sillä Kotlinin perusteet ovat suhteellisen nopea oppia. Lisäksi Kotlinin maailmanlaajuinen kehittäjäyhteisö kasvaa nopeasti, joten apua on helposti saatavilla. Myös Kotlinin verkosta löytyvä dokumentaatio on varsin kattava ja ajantasainen. (Kotlin 2021e; Kotlin 2021f.)

3.2 Kotliniin liittyvät avoimet verkko-oppimateriaalit

Kotlinista löytyy paljon englanninkielistä ja ajantasaista itseopiskelumateriaalia verkosta. JetBrains ylläpitää omaa sivustoaan Kotlinin opetukseen, sekä päivittää maailmanlaajuista listaa Kotlinia opettavista yliopistoista, joita tällä hetkellä on 185 yliopistoa 44 maassa. Lisäksi JetBrains on toteuttanut EduTools-pluginin, joka on yhteensopiva yrityksen kehittämien ohjelmointiympäristöjen kanssa ja jonka avulla voidaan oppia ja opettaa muun muassa Kotlinia tai Javaa (JetBrains 2021, Kotlin 2021g.) Google puolestaan ylläpitää Android-kehittäjille suunnattua developer.android.com -sivustoa, ja sivuston Kotlin-osasta löytyy myös paljon materiaalia itseopiskeluun. Lisäksi sivusto sisältää neljä ilmaista, eritasoista Kotliniin ja Android-kehitykseen liittyvää englanninkielistä kurssia, joiden avulla voi itseopiskella Kotlinin perusteita. (Developer.android.com 2021a.)

Siinä missä Kotlinia käsittelevää englanninkielistä oppimateriaalia löytyy runsaasti verkosta sekä kirjojen ja YouTube-videoiden muodossa, suomenkielistä oppimateriaalia ei pääosin ole saatavilla erilaisia lehti- ja blogikirjoituksia tai Wikipedia-artikkeleita lukuun ottamatta. Suomenkielisen materiaalipuutteen yhtenä syynä voi olla se, että Kotlin on suhteellisen uusi ohjelmointikieli, eikä sitä opeteta ensisijaisena ohjelmointikielenä suomalaisissa korkeakouluissa. Tämän opinnäytteen päätavoitteena oli suunnitella ja toteuttaa aloittelijoille sopiva itseopiskeluun tarkoitettu suomenkielinen verkkokurssi Kotlin-ohjelmointikielen perusteista. Seuraavassa luvussa kuvataan luomani verkkokurssin toteutusprosessia.

4 Verkkokurssin toteutus

Sain idean aloittelijoille suunnatusta suomenkielisestä Kotlinin-ohjelmoinnin verkkokurssista maaliskuussa 2020. Maailmanlaajuisen koronapandemian saavuttua Suomeen jouduin itsekin karanteeniin etätöihin, ja työstin IT-tradenomin opintojani töiden ohella kotona. Päätin opiskella myös jotain kokonaan uutta, joten aloitin Kotlin-ohjelmoinnin opiskelun alkeista. Kun olin saanut ensikosketuksen Kotliniin etsiessäni itseopiskelumateriaalia, huomasin ettei aiheesta ollut kirjoitettu mitään suomen kielellä. Lisäksi suurin osa olemassa olevista englanninkielisistä verkkokursseista oli suunnattu Javaa entuudestaan osaaville kehittäjille. Vaikka itse osaan Javaa, mietin mitä haasteita ohjelmointia aloittava opiskelija saattaisi kohdata lukiessaan englanninkielistä materiaalia aiheesta, joka on täynnä vieraita käsitteitä, kuten *null safety* tai *single-expression functions*. Silloin päätin, että haluan tehdä toiminnallisen opinnäytetyön ja sen tuotoksena verkkokurssin Kotlinista.

Opinnäytetyön tuotoksena toteutin yhden opintopisteen laajuisen aloittelijoille suunnatun itseopiskeluun tarkoitetun aikatauluttoman verkkokurssin Kotlin-ohjelmoinnista Markdown-kielellä GitHub-alustalle. Kurssimateriaaliin perehtymistä varten etsin Haaga-Helia ammattikorkeakoulun verkkokirjaston kokoelmahakua apuna käyttäen kymmenen Kotlinin perusteita käsittelevää englanninkielistä elektronista kirjaa, jotka oli kirjoitettu korkeintaan viimeisen neljän vuoden sisällä (Griffiths & Griffiths 2019; Horton 2019; Saumont 2019; Späth 2019; Subramaniam 2019; Arias & Chakraborty 2018; Devcic 2018; Jemerov & Isakova 2017; Moskala & Wojda 2017; Samuel & Bocutiu 2017) Perehdyin kirjojen sisältöön selailemalla ja lukemalla kirjoissa esiteltyjä aihepiirejä. Lisäksi kävin läpi Kotlinin verkosta löytyvän dokumentaation (Kotlin 2021f).

Ennen kurssimateriaalin suunnittelua opiskelin itse Kotlinia Googlen Kotlin Bootcamp for Developers -verkkokurssilla (Developer.android.com 2021b) sekä Metropolia ammattikorkeakoulun järjestämällä Introduction to Kotlin -verkkokurssilla vuoden 2021 alussa (Campusonline.fi 2021). Lisäksi perehdyin Kotlin Curriculum for Beginners -sisältöön (Kotlin 2021h). Kirjoihin ja dokumentaatioon perehtyminen, Kotlinin itseopiskeleminen sekä Kotlin Curriculum for Beginners -dokumenttiin perehtyminen auttoivat oman verkkokurssin kohderyhmän, oppimistavoitteiden ja aiheiden valinnassa sekä kurssisisällön rajaamisessa.

4.1 Kohderyhmä ja oppimistavoitteet

Omat tietotaitoni Kotlinin suhteen huomioiden valitsin Kotlin-ohjelmoinnin verkkokurssin kohderyhmäksi aloittelijat, jotta kykenisin parhaiten suunnittelemaan kohderyhmän tarpeita vastaavan verkkokurssin. Lisäksi oli mielestäni loogista suunnitella verkkokurssi

aloittelijoille, koska minulla ei ole aiempaa kokemusta verkkokurssin suunnittelusta ja toteuttamisesta. Valitsin kurssiin aiheiksi Kotlin-ohjelman rakenteen kuvaamisen ja näytölle tulostamisen, muuttujat, ehto- ja valintalauseet, toistolauseet sekä funktiot, koska aiheet sopivat mielestäni hyvin aloittelijoille. Lisäksi suunnittelemani Kotlin-ohjelmoinnin verkkokurssisisältö kattaa käytännössä Kotlin Curriculum for Beginners -dokumentissa mainitun Basics-osuuden (Kotlin 2021h). Jätin kurssin ulkopuolelle luokat ja olio-ohjelmoinnin, taulukot, kokoelmat, poikkeukset sekä muut vaativammat aiheet, sillä muuten verkkokurssin toteuttamisen työmäärä tämän opinnäytteen puitteissa olisi kasvanut liian suureksi.

Aiheeseen perehtymisen, kurssisisällön rajaamisen ja kohderyhmän valinnan jälkeen suunnittelin Kotlin-ohjelmoinnin verkkokurssille aluksi oppimistavoitteet. Kurssin yleisenä oppimistavoitteena on tutustuttaa opiskelija Kotlin-ohjelmoinnin perusteisiin. Lisäksi suunnittelin jokaiselle viidelle osalle erilliset oppimistavoitteet, jotka on kuvattu liitteessä 1. Jokainen osa sisälsi neljä oppimistavoitetta, joista yksi liittyi teorian hallitsemiseen ja loput osan teorian käytännön osaamiseen. Poikkeuksen tekee osa 3, jossa kaksi oppimistavoitetta koski teoriaa ja kaksi oppimistavoitetta käsitti osan asioiden käytännön hallinnan.

4.2 Kurssialusta, kurssin ulkoasu ja navigointi

Toteutusta varten valitsin seuraavaksi Kotlin-ohjelmoinnin verkkokurssille alustan sekä versionhallinta- ja julkaisujärjestelmät. Ylläpidin ja hallinnoin Kotlin-ohjelmoinnin verkkokurssin materiaalia GitHubin kautta Git-versionhallintajärjestelmässä. GitHub on verkkotalusta Git-tietovarastojen isännöintiä ja ohjelmistokehitysyhteistyötä varten. GitHub pohjautuu Git-versionhallintatyökaluun -ja tekniikkaan. Git puolestaan on ilmainen, avoimen lähdekoodin hajautettu versionhallintajärjestelmä, jonka alkuperäinen kehittäjä on Linus Torvalds. (GitHub 2021a.)

Kirjoitin Kotlin-ohjelmoinnin verkkokurssin materiaalin käyttäen Markdown-kieltä, jota myös GitHub tukee. Markdown on tekstimuotoinen merkintäkieli strukturoitujen asiakirjojen kirjoittamiseen (GitHub 2021b), joten sitä voi lukea ja kirjoittaa millä tahansa tekstinkäsittelyohjelmalla. Kun materiaali oli valmis, julkaisin Kotlin-ohjelmoinnin verkkokurssin kokonaisuudessaan GitHub Pages -työkalun avulla. GitHub Pages on staattinen sivuston isännöintipalvelu, joka kokoaa HTML-, CSS- ja JavaScript-tiedostoja suoraan GitHubin tietovarastosta ja julkaisee verkkosivuston. Tämän jälkeen sivusto on saatavilla verkossa osoitteessa käyttäjätunnus.github.io (GitHub Docs 2021.) Toteuttamani Kotlin-ohjelmoinnin verkkokurssi löytyy verkosta osoitteesta <https://teijatestaaja.github.io/kotlin-peruskurssi/>

Valitsin GitHubin verkkokurssilla käytettäväksi alustaksi, koska GitHubin käyttäminen yksityishenkilönä on mahdollista ilmaiseksi. Näin ollen Kotlin-ohjelmoinnin verkkokurssin kehittämisestä, julkaisusta, ylläpidosta sekä mahdollisesta jatkokehittämisestä ei aiheudu mitään kustannuksia. Lisäksi GitHub toimi paitsi kurssin julkaisualustana, myös Git-versiohallintatyökalun kanssa käytettynä verkkokurssin versionhallintana, joten kurssimateriaali on varmassa tallessa pilvessä tietyn oman fyysisen koneen tai ulkoisen kovalevyn sijaan. Kolmas syy GitHubin valintaan oli se, ettei opinnäytetyölläni ja siten myös sen toiminnallisella osuudella ollut toimeksiantajaa, joten en ollut sidottu toimeksiantajan valitsemaan tai tarjoamaan kurssialustaan tai muuhun oppimisympäristöön. Neljäntenä syynä GitHubin valintaan oli sen helppokäyttöisyys, yhteensopivuus Git-versiohallintatyökalun kanssa sekä mahdollisuus julkaista staattisia verkkosivuja alustan kautta. Lisäksi Kotlin-ohjelmoinnin verkkokurssikokonaisuus toimii myös yhtenä työnäytteenä osana omaa GitHub-ansioluetteloani, koska kurssin GitHub-tietovarasto on täysin julkinen.

Valitsin Markdown-kielen kurssin oppimateriaalin toteutuskieleksi, koska merkkauškielen avulla voi tuottaa suoraan julkaisukelpoista tekstiä, jota on helppo editoida ja ylläpitää. Näin verkkokurssin toteuttaminen ei vaatinut esimerkiksi html-koodin kirjoittamista tai WordPress-julkaisujärjestelmän opettelua. Lisäksi Markdown-merkkauškielällä kirjoitetut dokumentit oli helppo muuntaa html-koodiksi GitHub Pages -työkalun avulla, ja dokumentit ovat lähdekoodimuodossaan varsin luettavaa tekstiä. Samaa lähdetekstiä voidaan myös käyttää esimerkiksi pdf-muotoisen kurssiversion tuottamiseen, joten muutokset tekstiin tarvitsee tehdä vain yhteen paikkaan. Markdown-kieltä käyttäen Kotlin-ohjelmoinnin verkkokurssin materiaalit oli myös helppo kirjoittaa useimmiten käyttämälläni Visual Studio Code -editorilla, joten kurssin toteuttaminen ei vaatinut uuden editorin käytön opettelua.

Kuvassa 5 esitetään Kotlin-ohjelmoinnin verkkokurssin etusivu avattuna vaakasuuntaisessa näkymässä iPhone 13 - mobiililaitteessa sekä navigointipalkki, joka löytyy jokaisen sivun yläosasta. Navigointipalkin avulla opiskelija voi navigoida Tietoa kurssista -etusivun ja eri osioiden välillä, sekä tarkastella kurssin yhteenvetoa, tehtävien malliratkaisuja, useimmiten kysyttyjä kysymyksiä sekä sanastoa.

kotlin-peruskurssi

Tietoa kurssista	Osa 1	Osa 2	Osa 3	Osa 4	Osa 5	Yhteenveto	Malliratkaisut	FAQ	Sanasto
------------------	-------	-------	-------	-------	-------	------------	----------------	-----	---------

Kotlin-ohjelmoinnin peruskurssi

Kotlin-ohjelmoinnin peruskurssi on kaikille Kotlin-ohjelmointikielestä kiinnostuneille suunnattu avoin ja ilmainen verkkokurssi. Kurssilla perehdytään Kotlin-ohjelmointikielen perusteisiin.

Mikä on Kotlin

Kuva 5. Kotlin-ohjelmoinnin verkkokurssin etusivu ja navigointipalkki.

Kotlin-ohjelmoinnin verkkokurssin jokainen osa koostuu yhtenevistä palasista. Jokaisen osan otsikon jälkeen esitetään osan oppimistavoitteet listaan koottuna. Tämän jälkeen esitetään osaan liittyvä teoria tekstinä ja kuvina. Koodiesimerkit esitetään erillisellä Kotlin-korostuksella, jonka Markdown-kieli mahdollistaa. Kunkin osan lopussa on yhteenveto, jossa kuvataan lyhyesti osassa käsitellyt asiat. Yhteenvedon jälkeen näytetään lista tehtävistä. Lopuksi opiskelija ohjataan siirtymään seuraavaan osaan klikkaamalla sivun lopussa näkyvää linkkiä. Jokaisen sivun alatunnisteessa näytetään copyright-tiedot sekä teksti, jossa kielletään kurssimateriaalin käyttö kaupallisiin tarkoituksiin.

4.3 Verkkokurssin sisältö

Toteuttamani Kotlin-ohjelmoinnin verkkokurssin kurssimateriaali koostuu Kotlin-ohjelmointikieleen liittyvästä teoriasta eli tekstistä ja tekstiä tukevista kuvista sekä erilaisista tehtävistä, tehtävien malliratkaisuista, yhteenvedosta, usein kysytyistä kysymyksistä ja Kotlin-suomi-sanastosta. Kotlin-ohjelmoinnin verkkokurssilla teoria ja tehtävät on jaettu viiteen osaan, joiden aiheet ovat: ohjelman rakenne, muuttujat, ehto- ja valintalauseet, toistolauseet sekä funktiot. Kuvassa 6 esitetään verkkokurssin etusivulla näkyvä yhteenveto kurssin osioista, osioihin liittyvien tehtävien määrä ja linkki malliratkaisuihin sekä aika, jonka arvioin kuluvan kunkin osion teorian opiskeluun ja tehtävien tekoon. Arvioin Kotlin-ohjelmoinnin verkkokurssin teorian opiskeluun ja tehtävien tekoon kuluvan aikaa noin 27 tuntia, mikä laajuudeltaan vastaa yhtä opintopistettä.

Osa	Aihe	Tehtäviä	Aika-arvio	Malliratkaisut
Osa 1	Ohjelman rakenne	5	2 h	Osa 1 ratkaisut
Osa 2	Muuttujat	15	10 h	Osa 2 ratkaisut
Osa 3	Ehto- ja valintalauseet	10	5 h	Osa 3 ratkaisut
Osa 4	Toistolauseet	10	5 h	Osa 4 ratkaisut
Osa 5	Funktiot	10	5 h	Osa 5 ratkaisut
	Yhteensä	50	27 h	

Kuva 6. Yhteenvedo Kotlin-ohjelmoinnin verkkokurssin aiheista, tehtävistä ja opiskeluun kuluvasta viitteellisestä ajasta.

Suunnittelin osiin sisältyvän teorian siten, että se etenee loogisesti aiheesta toiseen, alkaen helpoimmin omaksuttavimmasta aiheesta (ohjelman rakenne) kurssin monimutkaisimpaan aiheeseen (funktiot). Teoriaosassa kuvataan kuhunkin aihealueeseen liittyvät käsitteet, ja pyrin huomioimaan valitsemani kohderyhmän siten, että teksti olisi helposti luettavaa ja silmäiltävää. Tekstin seassa olevien kuvien on tarkoitus tukea ja selventää joitakin teoriassa esitettyjä asioita, sekä helpottaa erilaisten oppijoiden opiskelua. Kurssin Yhteenvedo-osassa listataan kaikki kurssin oppimistavoitteet, ja siten opiskelija voi helposti tarkastaa oppimansa asiat.

Teorian lisäksi Kotlin-ohjelmoinnin verkkokurssi sisältää yhteensä 50 tehtävää. Kurssin tehtävät on numeroitu juoksevasti siten, että ensimmäinen numero kuvaa osion numeroa ja toinen numero tehtävän järjestysnumeroa osion sisällä. Lisäksi jaoin kurssin tehtävät kahteen tasoon. Suurin osa tehtävistä on suhteellisen helppoja perustehtäviä, joiden avulla opiskelija harjoittelee Kotlinin perusteita. Kurssin perustehtävissä opiskelija lukee ja analysoi ohjelmakoodia, korjaa tai muuttaa valmista ohjelmakoodia sekä kirjoittaa itse ohjelmakoodia. Kuvassa 7 esitetään esimerkki kurssin perustehtävästä.

Perustehtävät

5.1 Mitä seuraava funktio tekee?

```
fun omaLaskuri() {
    var tulo = 1;
    for (luku in 1..5) {
        tulo = tulo * luku
    }
    print(tulo)
}
```

Kuva 7. Esimerkki Kotlin-ohjelmoinnin verkkokurssin perustehtävästä.

Jokainen kurssin osa sisältää lisäksi vähintään yhden Guru-tehtävän, joka vaatii hoksamista tai osiossa opitun asian syvempää osaamista. Guru-tehtävät on merkattu verkkokurssin materiaaliin kuvassa 8 näkyvällä sinisellä tähtikuvakkeella.

★ Guru-tehtävä

5.10 Kirjoita funktio, joka saa parametrinaan kolme kokonaislukua satunnaisessa järjestyksessä ja joka tulostaa luvuista pienimmän.

Kuva 8. Esimerkki Kotlin-ohjelmoinnin verkkokurssin Guru-tehtävästä.

Kotlin-ohjelmoinnin verkkokurssin yksi kantava käyttöidea on se, että kurssin tehtävät tehdään Kotlin-hiekkalaatikossa eikä varsinaisessa ohjelmointiympäristössä. Suunnittelin Kotlin-ohjelmoinnin verkkokurssin tehtävät suoritettavaksi Kotlin-hiekkalaatikossa, koska se on käytettävissä millä tahansa selaimella eikä Kotlin-hiekkalaatikon käyttö siten vaadi tietyn ohjelmointiympäristön asennusta ja hallintaa. Lisäksi Kotlin-hiekkalaatikko on käytettävissä mobiililaitteella. Tehtävien teon keskittäminen selaimeen madaltaa kynnystä lähteä opettelemaan uutta ohjelmointitaitoa, kun aloittaminen ei vaadi suurta työtä ja asentelua. Opiskelija voi halutessaan tehdä tehtävät myös jollakin Kotlinia tukevalla editorilla tai ohjelmointiympäristöllä, joihin annetaan suosituksia kurssin etusivulla. Editoreiden käyttöön ei kuitenkaan anneta kurssilla erillistä ohjeistusta. Näin ollen editorin valinta ja käytön opiskelu jää opiskelijan itsensä vastuulle.

Kurssin opiskelua tukevia komponentteja ovat tehtävien malliratkaisut, useimmin kysytyt kysymykset sekä Kotlin-suomi-sanasto. Tehtävien malliratkaisut noudattavat tehtävien numeerointia, jotta opiskelijan olisi helppo löytää tehtävää vastaava ratkaisu. Lisäksi kokosin useimmin kysytyihin kysymyksiin listan kysymyksistä, joita materiaali minussa herätti ja joita uskon tulevien opiskelijoiden kysyvän, sekä vastaukset kysymyksiin. Kysymysten ja niihin annettujen vastausten tarkoitus on ohjata opiskelijaa itseopiskelemaan aiheesta enemmän materiaalin ulkopuolella. Kysymysten listan päivittäminen jätettiin myös yhdeksi kehitysideaksi. Kokosin Kotlin-suomi-sanastoon kaikki teoriaosassa esitetyt Kotliniin liittyvät termit sekä niiden suomenkieliset selitykset. Sanaston tarkoitus on helpottaa kurssilla opiskelua, sillä termin selityksen voi halutessaan tarkastaa yhdeltä sivulta. Sanastoon voisi vielä lisätä sarakkeen, joka kertoisi missä kurssin osiossa kyseinen termi esiintyy ensimmäisen kerran. Tämä jätettiin yhdeksi kehitysideaksi.

Koska Kotlin-ohjelmoinnin verkkokurssi on suunniteltu opiskeltavaksi omaan tahtiin täysin itseopiskeluna, en suunnitellut kurssille kiinteää aikataulua tai arviointiperusteita. Tämä

mahdollistaa sen, että kurssin opiskelija voi suorittaa kurssin täysin omassa aikataulussaan ilman, että hänen tarvitsisi esimerkiksi palauttaa tehtäviä tai mahdollista harjoitustyötä etukäteen määrätyn aikataulun mukaan. Toinen kurssia rajoittava tekijä on yhteistoiminnallisuuden puute, sillä en suunnitellut kurssille minkäänlaista yhteydenottokanavaa tai yhteistoiminnallisen oppimisen mahdollistavaa kanavaa. Kurssia ei myöskään testattu oikeilla käyttäjillä esimerkiksi tiettynä ajankohtana toteutettavana verkkokurssina tai ohjatuna verkkokurssina, sillä ajankäytön takia se ei ollut mahdollista tämän opinnäytetyön puitteissa. Kiinteä aikataulu, arviointiperusteiden luominen esimerkiksi harjoitustyön muodossa, yhteydenpitokanavan mahdollistaminen esimerkiksi Slackin tai Telegramin avulla sekä kurssin varsinainen käyttöönotto esimerkiksi jossakin oppilaitoksessa jätettiin jatkokehitysideoiksi.

Harkitsin kurssin tehtävien tekoa varten sopivan editorin esittelyä ja käyttöönottoa kurssimateriaalin tehtävänannoissa, mutta koska aihetta oli rajattava, päätin jättää editorin käytön opiskelun opiskelijan omalle vastuulle, sekä editorin käyttöohjeen ja opastusvideon yhdeksi kurssin jatkokehitysideaksi. Toinen syy editorin poisjättämiseen kurssin oppisisällöstä oli se, että Kotlin-ohjelmien suorittamiseen pöytäkoneella tai kannettavalla tietokoneella tarvitaan JDK (Java Development Kit), ja sen asentamiseen olisi pitänyt luoda erillinen ohje aloittelijoita varten.

Valittuun kurssialustaan liittyy myös omia rajoituksia. GitHub ei ole varsinainen verkkokurssialusta, joten se ei toistaiseksi mahdollista esimerkiksi minkäänlaisen oppimisanalytiikan käyttöä tai tehtävien palautusta alustalle, verrattuna esimerkiksi Moodlen oppimisympäristöön. Lisäksi valitun Markdown-kielen tekstin ulkoasun muotoilun muokkausmahdollisuudet ovat rajalliset.

4.4 Ajankäyttö

Yhden opintopisteen laajuisen Kotlin-ohjelmoinnin verkkokurssin materiaalin tuottaminen vei yhteensä noin 80 tuntia (taulukko 1), josta eniten aikaa kului aiheeseen perehtymiseen, toisin sanoen Kotlinia käsittelevien kirjojen lukemiseen ja muistiinpanojen tekemiseen. Tämä vei melkein puolet käytetystä ajasta. Toiseksi eniten aikaa kului luonnollisesti myös osien tekstien ja tehtävien luomiseen. Tässä tehtävien malliratkaisujen luominen on esitetty omana kohtanaan, sillä monet tehtävät vaativat koodaamista, jolloin malliratkaisun toimivuus varmistettiin sekä Kotlin-hiekkalaatikolla että oikealla ohjelmistoympäristöllä.

Taulukko 1. Kurssin tuottamiseen käytetty aika.

Toiminta	Aika (h)
Aiheeseen perehtyminen	35
Verkkokurssin sisällön suunnittelu	9
Osien tekstin ja tehtävien luominen	28
Tehtävien malliratkaisujen luominen	5
FAQ:n ja sanaston kokoaminen	2
Kurssin testaus kannettavalla tietokoneella ja mobiililaitteella	1
Yhteensä	80

Vähiten aikaa kului kurssin testaamiseen. Suoritin Kotlin-ohjelmoinnin verkkokurssin testauksen Chrome-selaimella kannettavalla tietokoneella, jonka käyttöjärjestelmänä oli Windows 10 Pro. Lisäksi testasin verkkokurssin iPhone 13 mobiililaitteella ja siinä olevalla Safari-selaimella iOS 15.1.1 -käyttöjärjestelmäversiolla. Kotlin-ohjelmoinnin verkkokurssin testauksessa ei ilmennyt ongelmia.

On hyvä huomata, että kokonaisuutena kurssimateriaalin tuottamiseen meni noin 80 tuntia eli hieman yli kahden normaalin työviikon verran. Yhden opintopisteen laajuisen Kotlin-ohjelmoinnin verkkokurssin opiskeluun kuluu puolestaan arviolta vain noin 27 tuntia. Kurssimateriaalin tuottamiseen menee siis lähes kolminkertainen aika oppimateriaalin opiskelemiseen verrattuna.

5 Pohdinta

Seuraavissa luvuissa pohditaan opinnäytetyön tavoitteiden täyttymistä, arvioidaan toteutettu verkkokurssi eAMK verkkototeutusten laatukriteereiden avulla, pohditaan kurssin tuomia hyötyjä, arvioidaan valitun kurssialustan sopivuutta käyttötarkoitukseen sekä tarkastellaan opinnäytetyön tekijän ammatillista kehittymistä. Lopuksi annetaan ideoita jatkotutkimukseen ja kurssin kehittämiseen.

Tällä toiminnallisella opinnäytetyöllä oli kaksi tavoitetta: luoda aloittelijoille sopiva itseopiskeluun tarkoitettu suomenkielinen verkkokurssi Kotlin-ohjelmointikielen perusteista sekä raportoida verkkokurssin suunnitteluun ja toteutukseen liittyvistä toimenpiteistä. Nämä tavoitteet toteutuivat mielestäni hyvin, sillä sain opinnäytetyöni puitteissa luotua yhden opintopisteen laajuisen avoimen Kotlin-ohjelmoinnin verkkokurssin GitHub-alustalle. Lisäksi opinnäytetyöni avulla opin, mitä verkkokurssin suunnittelu ja toteuttaminen pitävät sisällään. Nyt minulla on lähtötasoa selkeämpi kuva siitä, mitä vaiheita verkkokurssin toteuttaminen vaatii, ja miten paljon aikaa tulee varata kurssin toteuttamiseen aivan nollasta.

Varsinaisen opinnäytetyöraportin kirjoittamisen tekisin hieman ripeämmällä aikataululla, sillä vuoden lähestyessä loppuaan opinnäytetyön kirjoittamisen aikataulu kävi melko tiukaksi päivätöiden ohella. Toiminnallinen osuus tuli tehtyä hyvissä ajoin, mutta verkkokurssille toteutettavan materiaalin määrää oli pakko rajata, jotta opinnäytetyön laajuus ei kasvaisi liian suureksi. Olisin kuitenkin halunnut toteuttaa muutamia ruutukaappausvideoita, jotta erilaiset oppijat saisivat itseopiskelukurssista enemmän irti.

Itsearvioin Kotlin-ohjelmoinnin verkkokurssin toteutuksen käyttämällä verkkototeutusten arviointityökalua (aAMK 2021). Arviointityökalu on käytettävissä verkossa selaimella, ja työkalu koostuu yhdestä lomakkeesta. Lomake sisältää eAMK verkkototeutusten osat (11 kappaletta), ja jokaisen osan alla on vaihteleva määrä kriteereitä, joiden toteutumista voi itsearvioida neliportaisella asteikolla, jossa vastausvaihtoehtoina ovat Ei, Osittain, Enimmäkseen ja Kyllä. Kotlin-ohjelmoinnin verkkokurssitoteutuksen itsearvioinnin tulokset löytyvät liitteestä 2.

Kuten liitteestä 2 käy ilmi, Kotlin-ohjelmoinnin verkkokurssilla laatukriteeristön *Osaamistavoitteet, oppimisprosessi ja pedagogiset ratkaisut* -osan *Opiskelijan kansainvälisyysosaaminen vahvistuu opintojakson myötä* -kriteeri täyttyy vain osittain. Kotlin-ohjelmoinnin verkkokurssi on kokonaan suomenkielinen, joten kielitaitoa tarvitaan ymmärtämään aino-

astaa Kotlin-ohjelmointikielen englanninkielisiä sanoja. Kansainvälisyysosaaminen toteutuisi paremmin, jos kurssille olisi toteutettu materiaalinkit vieraskieliseen Kotlin-verkkomateriaaliin. Linkkien lisääminen ulkopuolisiin materiaaleihin jätettiin yhdeksi kehitysideaksi.

Liitteessä 2 kuvatus laatuksriteristön *Työvälineet* -osan kriteereistä *Ohjeet verkkoalustan ja työvälineiden käyttöön, sovellusten lataamiseen ja käyttäjätilin luomiseen löytyvät verkkoalustalta* -kriteeri ei täyty, koska oppimateriaalissa ei anneta ohjeita Kotlin-hiekkalaatikon käyttöön. Tähän auttaisi ruutukaappausvideo, jonka tekeminen jätettiin jatkokehitysideaksi. Kuten liitteestä 2 käy ilmi, laatuksriteristön *Vuorovaikutus* -osan kaikki kriteerit toteutuvat kurssilla vain osittain, ja ohjaukseen ja palautteeseen liittyvät kriteerit eivät toteudu juuri ollenkaan. Vuorovaikutuksen osalta kurssin laatua parantaisi kehitysideaksi jätetyn vuorovaikutuskanavan lisääminen. Tällä hetkellä opiskelijan ja opettajan välisen kommunikoinnin mahdollistaa ainoastaan sähköposti. Laatuksriteristön *Arviointi* -osaan liittyviä kriteereitä ei arvioitu, koska kurssille ei toteutettu erillistä arviointia kurssin itseopiskeluluonteen takia.

Liitteessä 2 kuvatus laatuksriteristön *Kehittäminen* -osan palautteeseen liittyvä kriteeri toteutuu vain osittain, koska opintojaksoa kehitetään opinnäytetyön tekemisen jälkeen toiseksi vain sähköpostitse saadun palautteen perusteella. Varsinaisen palautelomakkeen tekeminen jätettiin kehitysideaksi. *Käytettävyyys ja ulkoasu* -osan *Videot ja äänitiedostot on tekstitetty tai sisältö on muutoin saatavilla tekstimuodossa* -kriteeri ei toteudu, koska kurssille ei ajanpuutteen vuoksi toteutettu materiaalia video- tai äänimuodossa.

Tämän opinnäytetyön toiminnallisena tuotoksena toteuttamani Kotlin-ohjelmoinnin verkkokurssi aloittelijoille on tietääkseni tällä hetkellä ainoa suomenkielinen, avoin verkkokurssi Kotlinin perusteista. Lisäksi kurssi on mielestäni erityisen sopiva valitulle kohderyhmälle, koska kurssin aiheet käsittelevät ohjelmoinnin peruskäsitteitä, kuten ohjelman rakennetta, muuttujia, ehto- ja toistolauseita, valintalauseita sekä funktioita. Kaikki nämä aihealueet ovat läsnä missä tahansa ohjelmointikielessä, joten kurssin avulla voi oppia paitsi Kotlinia, myös yleisesti ohjelmointia.

GitHub toimi kurssialustana yllättävän hyvin. Kurssin oppimateriaalin päivitys oli helppoa Git-versionhallintatyökalun avulla suoraan komentoriviltä. Lisäksi sivuston julkaisu tapahtui kätevästi, kun ei tarvinnut asentaa mitään lisäosia. Kurssimateriaalin päivitys onnistui jopa sivuston julkaisun jälkeen automaattisesti viemällä muutokset tietovarastoon normaalisti. Markdown-kieli toimi valittuna merkkauksikielenä suhteellisen hyvin. Kaipasin ainoastaan mahdollisuutta merkata tekstiä eri väreillä. Värien vaihtaminen olisi voinut onnistua muokkaamalla GitHub Pagesin generoimia html-sivuja käsin, mutta koska värikoodausta

tarvittiin vain muutamassa kohtaa 5. osassa, koin helpommaksi käyttää kuvaa, johon teksti on väritetty eri värillä. Joka tapauksessa Markdown-kielen käytön ansiosta sivustosta saa helposti vaikkapa pdf-muotoisen kurssiversion, ja lisäksi kurssi on helposti siirrettävissä kokonaan toiselle kurssialustalle. GitHubin ja Markdown-kielen tarjoamat edut helpottavat myös mahdollista kurssin jatkokehitystä. Opinnäytetyön palauttamisen jälkeen Kotlin-ohjelmoinnin verkkokurssi jää joka tapauksessa GitHub-alustalle kenen tahansa Kotlinin opiskelusta kiinnostuneen saataville.

Oma ammatillinen osaamiseni kehittyi hyvin paljon tämän opinnäytetyön myötä. Vaikka olen aiemmin kirjoittanut neljä erilaista opinnäytetyötä eri kielillä aiemmissa opinnoissani, tämänkin opinnäytetyöraportin tekeminen opetti minulle tiedonetsintää, systemaattisen raportin kirjoittamista ja lähdekriittisyyttä. En juuri kaivannut työlleni ohjausta, vaan työstin raportin lähes kokonaan itsenäisesti. Ainoastaan mietin sitä, mikä olisi paras tapa kuvata toiminnallisen osuuden tuotoksen – tässä tapauksessa verkkokurssin – toteuttamisprosessi. Lopulta päädyin käyttämään 1. persoonaa passiivin sijaan, koska koin näin saavani ääneni paremmin kuuluviin Kotlin-ohjelmoinnin verkkokurssin ainoana toteuttajana.

Opinnäytetyön toiminnallisen osuuden eli verkkokurssin toteuttaminen oli äärettömän opettavaista. Ensinnäkin opin paljon verkkokurssin suunnittelusta sekä siitä minkälaisia asioita tulee huomioida ennen kuin lähtee tekemään varsinaista toteutusta. Näiden pohjalta koostin laadukkaasti verkkokurssin toteuttamisen pikaohjeen, joka löytyy liitteestä 3. Toisena opittuna asiana sain työkalupakkiini uudeksi työkaluksi Intelli J Idea -editorin osaamisen, sillä käytin editoria verkkokurssin tehtävien malliratkaisuiden luomiseen. Lisäksi opin kirjoittamaan Markdown-kieltä, sekä käyttämään sujuvasti Git-versionhallintajärjestelmää yhdessä GitHub-alustan kanssa. Luonnollisesti opin myös itselleni kokonaan uuden ohjelmointikielen Kotlinin, sillä en ollut aiemmissa opinnoissani käynyt Kotlinia käsitteleviä kursseja.

Jatkotutkimuksen kannalta olisi kiinnostavaa selvittää millaisia avoimia, suomenkielisiä verkkokursseja on olemassa ohjelmoinnista erilaisilla moderneilla ohjelmointikielillä. Näiden kurssien toteutuksia voisi vertailla keskenään, jotta saataisiin selville yleisimmin modernien ohjelmointikielten kursseilla opetettuja käsitteitä. Lisäksi eräs jatkotutkimusidea voisi olla siinä, miten Kotlin- ja Java-ohjelmointikieltä voitaisiin parhaiten opettaa samalla kurssilla.

Toteuttamaani Kotlin-ohjelmoinnin verkkokurssia voisi myös kehittää monella eri tavalla. Verkkokurssille voisi toteuttaa arvioitavaksi suoritustavaksi pienen harjoitustyön palauttamisen. Harjoitustyö voisi olla esimerkiksi yksinkertainen tehtävälister-sovellus. Sitä varten

sisältöön olisi hyvä tehdä uusia osia seuraavista aiheista: merkkijonomallit, luokat ja tyhjiä muuttujien turvallinen luominen. Lisäksi kurssille voisi lisätä osion, jossa käsitellään ohjelmakoodin testausta esimerkiksi yksikkötesteillä. Kurssista voisi myös kerätä palautetta kurssin loppupalautelomakkeen tai muun verkkolomakkeen avulla. Tällaisen lomakkeen voisi toteuttaa esimerkiksi Google Formilla.

Verkkokurssin tekstiä voisi elävöittää lisäämällä enemmän aiheeseen liittyviä kuvia. Osaan 3 (ehto- ja valintalauseet) sekä osaan 4 (toistolauseet) voisi myös lisätä välitehtäviä, ja lisäksi koko kurssille voisi lisätä uudeksi tehtäväkategoriaksi bonustehtävät, jotka laajentaisivat aihetta. Pitempiin osiin voisi toteuttaa myös sivun sisäisen navigoinnin, ja lisäksi kurssille voisi lisätä Linkit-osan, joka sisältäisi linkkejä Kotlinia käsitteleviin sivustoihin ja englanninkielisiin avoimiin Kotlin-verkkokursseihin.

Kurssin olemassa olevia osia voisi myös laajentaa. Ensimmäiseen osaan voisi lisätä luvun siitä, millaisia virheitä Kotlin-hiekkalaatikon käytössä voi syntyä, ja miten virheitä tulkitaan. Jokaiseen osaan voisi myös lisätä linkin Kotlinin viralliseen dokumentaatioon siihen kohtaan, jossa osassa käsiteltyä asiaa esitellään. Lisäksi jokaiseen osaan voisi lisätä pienen vertailua Javaa -osan. Tämä helpottaisi etenkin Java-ohjelmointitaustan omaavien opiskelijoiden opiskelua. Erilaisia oppijoita voisi myös paremmin huomioida tekemällä nykyisten osioiden teoriasta YouTube-videoita, tai ruutukaappausvideoita esimerkiksi Kotlinia tukevan ohjelmointiympäristön asentamisesta. Kotlin-ohjelmoinnin verkkokurssin nykytoteutus ei sisällä videoita, koska niiden käsikirjoittamiseen ja tekemiseen olisi mennyt liikaa aikaa. Tästä huolimatta toivon, että lopputulos näyttäytyy kurssin tuleville itseopiskelijoille laadukkaana ja selkeänä itseopiskelupakettina.

Lähteet

Arias, M. & Chakraborty, R. 2018. Functional Kotlin. Packt Publishing. Birmingham.

Campusonline.fi 2021. Introduction to Kotlin. Luettavissa: <https://campusonline.fi/course/introduction-to-kotlin/>. Luettu: 11.12.2021.

Devicic, M. 2018. Kotlin Quick Start Guide. Packt Publishing. Birmingham.

Developer.android.com 2021a. Develop Android apps with Kotlin. Luettavissa: <https://developer.android.com/kotlin>. Luettu: 9.12.2021.

Developer.android.com 2021b. Kotlin Bootcamp for Programmers. Luettavissa: <https://developer.android.com/courses/kotlin-bootcamp/overview>. Luettu: 11.12.2021.

eAMK 2021. Verkkototeutusten arviointityökalu. Luettavissa: <https://www.eamk.fi/fi/campusonline/arviointilomake/>. Luettu: 19.12.2021.

Ekonoja, A. 2014. Oppimateriaalien kehittäminen, hyödyntäminen ja rooli tieto- ja viestintätekniikan opetuksessa. Jyväskylän yliopisto. Jyväskylä Studies in Computing 193. Luettavissa: <https://jyx.jyu.fi/handle/123456789/44175>. Luettu: 19.12.2021.

Ellington, H. & Race, P. 1993. Producing Teaching Materials: A Handbook for Teachers and Trainers. (Second edition). Kogan Page. Lontoo.

GitHub 2021a. Git Cheat Sheet. Luettavissa: <https://training.github.com/downloads/github-git-cheat-sheet.pdf>. Luettu: 12.12.2021.

GitHub 2021b. GitHub Flavored Markdown Spec. Luettavissa: <https://github.github.com/gfm/>. Luettu: 12.12.2021.

GitHub Docs 2021. About GitHub Pages. Luettavissa: <https://docs.github.com/en/pages/getting-started-with-github-pages/about-github-pages>. Luettu: 12.12.2021.

Griffiths, D. & Griffiths, D. 2019. Head First Kotlin. O'Reilly Media, Inc. Sebastopol.

Heinonen, J. P. 2005. Opetussuunnitelmat vai oppimateriaalit. Peruskoulun opettajien käsityksiä opetussuunnitelmien ja oppimateriaalien merkityksestä opetuksessa.

Helsingin yliopisto. Soveltavan kasvatustieteen laitos. Tutkimuksia 257. Luettavissa: <https://helda.helsinki.fi/handle/10138/20002>. Luettu: 19.12.2021.

Heller, M. 2020. What is Kotlin? The Java alternative explained. Luettavissa: <https://www.infoworld.com/article/3224868/what-is-kotlin-the-java-alternative-explained.html>. Luettu: 11.12.2021.

Horton, J. 2019. Android Programming with Kotlin for Beginners. Packt Publishing. Birmingham.

Ilomäki, L. 2012. Johdanto. Teoksessa L. Ilomäki (toim.) Laatus e-oppimateriaaleihin: E-oppimateriaalit opetuksessa ja oppimisessa. Helsinki: Opetushallitus. Oppaat ja käsikirjat 2012:5, 5–6. Luettavissa: https://www.oph.fi/sites/default/files/documents/144415_laatus_e-oppimateriaaleihin_2.pdf. Luettu: 19.12.2021.

Jemerov, D. & Isakova, S. 2017. Kotlin in Action. Manning Publications Co. New York.

JetBrains 2021. Educator Start Guide. Luettavissa: <https://plugins.jetbrains.com/plugin/10081-edutools/docs/educator-start-guide.html>. Luettu: 10.12.2021.

Jhonson, E. 2021. Kotlin vs Java: Which is the Best Choice for Android App Development? Luettavissa: <https://medium.com/javarevisited/kotlin-vs-java-which-is-the-best-choice-for-android-app-development-7c9fc782d2c9>. Luettu: 11.12.2021.

Kotlin 2021a. FAQ. Luettavissa: <https://kotlinlang.org/docs/faq.html>. Luettu: 9.12.2021.

Kotlin 2021b. Kotlin programming language. Luettavissa: <https://kotlinlang.org/assets/kotlin-media-kit.pdf>. Luettu: 9.12.2021.

Kotlin 2021c. Data classes. Luettavissa: <https://kotlinlang.org/docs/data-classes.html>. Luettu: 10.12.2021.

Kotlin 2021d. Kotlin Native. Luettavissa: <https://kotlinlang.org/docs/native-overview.html>. Luettu: 11.12.2021.

Kotlin 2021e. Kotlin for Android. Luettavissa: <https://kotlinlang.org/docs/android-overview.html>. Luettu: 11.12.2021.

Kotlin 2021f. Kotlin docs. Luettavissa: <https://kotlinlang.org/docs/home.html>. Luettu: 11.12.2021.

Kotlin 2021g. Teach. Luettavissa: <https://kotlinlang.org/education/>. Luettu: 10.12.2021.

Kotlin 2021h. Kotlin Curriculum for Beginners. Luettavissa: https://docs.google.com/document/d/1XIJJaV3zhn-tJhDc_6Kr00lmTo5zCBuES3Yt67wX752M/edit#heading=h.zc4tzol1fpqp. Luettu: 11.12.2021.

Moskala, M. & Wojda, I. 2017. Android Development with Kotlin. Packt Publishing. Birmingham.

Opetushallitus 2021. E-oppimateriaalin laatukriteerit. Luettavissa: <https://www.oph.fi/fi/julkaisut/e-oppimateriaalin-laatukriteerit>. Luettu: 19.12.2021.

Samuel, S. & Bocutiu, S. 2017. Programming Kotlin. Packt Publishing. Birmingham.

Saumont, P. Y. 2019. The Joy of Kotlin. Manning Publications. New York.

Späth, P. 2019. Learn Kotlin for Android Development. Apress. Leipzig.

Subramaniam, V. 2019. Programming Kotlin. Pragmatic Bookshelf.

Varonen, M. & Hohenthal, T. 2017. eAMK verkkototeutusten laatukriteerit. Luettavissa: <https://www.eamk.fi/fi/campusonline/laatukriteerit/>. Luettu: 19.12.2021.

Liitteet

Liite 1. Kotlin-ohjelmoinnin verkkokurssin oppimistavoitteet

Osa 1: Ohjelman rakenne

- Osaat kirjoittaa yksinkertaisen Kotlin-ohjelman, joka tulostaa tekstiä.
- Tiedät mikä on Kotlin-hiekkalaatikko.
- Osaat tulkita ja korjata Kotlin-ohjelman suorittamisessa aiheutuneita yksinkertaisia virheitä.
- Osaat kirjoittaa koodiin kommentteja.

Osa 2: Muuttujat

- Tunnet käsitteen muuttuja, ja tiedät mitä ovat muuttujan tyyppi, muuttujan nimi ja muuttujan arvo.
- Osaat luoda ja käsitellä kokonaisluku-, liukuluku-, merkki-, merkkijono- ja totuusarvomuuttujia.
- Osaat tehdä yksinkertaisia laskutoimituksia.
- Osaat käyttää sujuvasti Kotlin-hiekkalaatikkaa.

Osa 3: Ehto- ja valintalauseet

- Tunnet käsitteet ehtolause ja valintalause.
- Osaat luoda ehtolauseita, joiden avulla ohjelmaan saadaan vaihtoehtoista toimintaa.
- Tunnet ehtolauseissa tyypillisesti käytettävät vertailuoperaattorit ja loogiset operaatiot ja, tai sekä ei.
- Osaat luoda valintalauseita.

Osa 4: Toistolauseet

- Tunnet käsitteen toistolause ja osaat luoda Kotlin-ohjelman, joka sisältää toistolauseen.
- Osaat käyttää while-toistolauseita.
- Osaat käyttää do..while-toistolauseita.
- Osaat käyttää for-toistolauseita.


Osa 5: Funktiot

- Tunnet käsitteet funktio, funktion parametri ja funktion paluuarvo.
- Osaat luoda pala palalta parametrillisia ja parametrittomia funktioita.
- Osaat luoda funktioita, jotka palauttavat arvon.
- Osaat kutsua funktioita sekä pääohjelmasta että muiden funktioiden sisältä.

Liite 2. eAMK verkkototeutuksen laatukriteerit


Kohderyhmä ja käyttäjät

Käyttäjät ja heidän tarpeensa huomioidaan suunnittelussa ja tuotantovaiheessa sekä toteutuksen aikana.

Lähtötasovaatimukset löytyvät opintojaksokuvauksesta.	Kyllä
Lähtötaso voidaan tarvittaessa selvittää. Opintojakson osallistujilla on riittävät pohjatiedot ja -taidot opintojakson suorittamiseksi.	Kyllä
Toteutus ja pedagogiset ratkaisut onnistuvat valitulla osallistujamäärällä.	Kyllä
 100.00%	

Osaamistavoitteet, oppimisprosessi ja pedagogiset ratkaisut

Osaamistavoitteet on määritelty osaamisperustaisesti, työelämälähtöisesti ja geneeristen taitojen kehittyminen huomioiden. Opintojaksolla sovelletaan tarkoituksenmukaisia pedagogisia malleja, toimintatapoja ja menetelmiä, jotka ovat linjassa opintojakson oppimiskäsityksen kanssa.

Opintojakson sisältö ja menetelmät sekä tekniset ja pedagogiset ratkaisut tukevat osaamistavoitteiden saavuttamista.	Kyllä
Opiskelijan kansainvälisyysosaaminen vahvistuu opintojakson myötä.	Osittain
Toteutuksen työtavat tukevat geneeristen taitojen kehittymistä ja saavuttamista.	Kyllä
Opiskelijalla on mahdollisuus asettaa omat tavoitteensa suhteutettuna opintojakson tavoitteisiin.	Kyllä
 83.33%	

Tehtävät

Oppimistehtävät ovat osaamistavoitteiden saavuttamista edistäviä, työelämäläheisiä ja mahdollistavat opiskelijoiden yksilöllisyyden huomioimisen. Toteutukselle valitut työtavat tukevat yhteisöllistä tiedonrakentelua ja osaamisen jakamista.

Opintojakson sisältö ja menetelmät sekä tekniset ja pedagogiset ratkaisut tukevat osaamistavoitteiden saavuttamista.	Kyllä
Tehtävät kytkeytyvät osaamistavoitteisiin ja työelämän todellisiin tilanteisiin.	Kyllä
Tehtävänannot ovat ymmärrettäviä ja ne ohjaavat opiskelijan työskentelyä.	Kyllä
Tehtävät sopivat verkko-opiskeluun ja niitä on mahdollista suorittaa verkossa yksin tai yhdessä toisten opiskelijoiden kanssa.	Kyllä
Tehtävien tekemisessä opiskelija voi hyödyntää erilaisia teknologisia ratkaisuja, mm. ääni, video, kuva, erilaiset tekstit.	Kyllä



100.00%

Sisältö ja aineistot

Sisältö ja aineistot tukevat osaamistavoitteiden saavuttamista.


Sisältö auttaa oppijaa yhdistämään uutta tietoa aiemmin oppimaansa. Sisältö tukee tiedon soveltamista.	Kyllä
Opiskelija saa ohjausta osaamistavoitteita tukevan oppimateriaalin valintaan.	Kyllä
Opettajat ja opiskelijat tuottavat ja valitsevat aineistoja, jotka ovat ajantasaisia ja koottu luotettavista lähteistä.	Kyllä
Lähdeviitteet ja tieto käyttöoikeuksista ovat näkyvillä aineistossa.	Kyllä
Tieto opiskelijan tuottaman materiaalin säilyttämiseen ja hyödyntämiseen liittyvistä käytänteistä löytyy verkkoalustalta.	Kyllä



100.00%


Työvälineet

Verkkotyövälineet tukevat oppimista.

Toteutuksella käytetään verkkotyövälineitä, jotka tukevat osaamistavoitteiden saavuttamista, ovat pedagogisesti perusteltuja sekä soveltuvat ammattialan työprosesseihin.	Kyllä
Jos osaamistavoitteiden saavuttaminen vaatii sovellusten lataamisen tai uuden käyttäjätilin luomisen, asia on perusteltu verkkoalustalla. Ladattavat sovellukset ovat maksuttomia ja tietoturvallisia.	Kyllä
Ohjeet verkkoalustan ja työvälineiden käyttöön, sovellusten lataamiseen ja käyttäjätilin luomiseen löytyvät verkkoalustalta.	Ei
Opettaja ja opiskelija saavat opintojakson aikana verkkoalustalta metatietoa. Opettaja voi hyödyntää tietoa mm. ohjauksessa ja opiskelija mm. opintojensa edistymisen seuraamisessa.	Kyllä
Tieto opintojakson suorittamiseen vaadittavasta peruslaitteistosta ja muista tarvittavista sovelluksista löytyy opintojaksokuvauksesta.	Kyllä
Opintojakson suorittaminen onnistuu tavanomaisen nopealla verkkoyhteydellä.	Kyllä
 83.33%	


Vuorovaikutus

Vuorovaikutus tukee osaamistavoitteiden saavuttamista.

Opettajalla ja opiskelijoilla on mahdollisuus keskinäiseen vuorovaikutukseen, yhteisölliseen tekemiseen, toisilta oppimiseen ja kokemusten jakamiseen verkossa.	Osittain
Sidosryhmät, esim. työelämän edustajat, pystyvät osallistumaan vaivatta yhteistyöhön toteutuksen aikana.	Osittain
 33.33%	


Ohjaus ja palaute

Ohjaus ja palaute ovat oikea-aikaista ja niitä on saatavissa koko opintojakson ajan.

Tieto ohjauksesta ja sen toteuttamisen tavoista löytyy verkkoalustalta.	Kyllä
Opiskelijoilla on mahdollisuus saada ohjausta ja palautetta opintojakson aikana opettajilta, muilta opiskelijoilta ja sidosryhmien edustajilta.	Ei
Opiskelija pystyy osallistumaan aktiivisesti ohjaukselliseen keskusteluun eri työvälineitä hyödyntäen.	Ei
Verkkoalustan analyysityökaluja hyödynnetään opiskelijoiden edistymisen seuraamisen, etenemisen, kannustamisen ja ohjaamisen tukena.	Ei
Ohjauksen vastuuhenkilöt, kanavat ja aikataulut löytyvät helposti verkkoalustalta.	Kyllä
Opiskelija voi antaa palautetta ja esittää kysymyksiä koko opintojakson ajan.	Kyllä
 50.00%	


Arviointi

Arviointi on läpinäkyvää, jatkuvaa, monipuolista ja reflektio-osaamista kehittävää.

Arvioinnin toteutus, kohteet ja muodostuminen löytyvät opintojaksokuvauksesta. Arviointi suoritetaan arviointikriteerien mukaisesti.	Ei
Arviointia tapahtuu koko oppimisprosessin ajan ja sitä toteutetaan monipuolisia menetelmiä hyödyntäen. Opiskelija osallistuu itsearviointiin ja vertaisarviointiin verkkoalustan työvälineitä käyttäen.	Ei
 0.00%	


Kehittäminen

Verkkototeutusta kehitetään jatkuvasti.

Opintojakso on ajanmukainen ja päivitetty mm. osaamistavoitteiden, sisältöjen, menetelmien, arvioinnin ja verkkotyövälineiden osalta.	Kyllä
Opettajilta ja opiskelijoilta kerätään palaute vähintään toteutuksen jälkeen. Opintojaksoa kehitetään saadun palautteen perusteella.	Osittain
 66.67%	

Käytettävyys ja ulkoasu

Toteutus on selkeä, käytettävä ja tietoturvallinen.

Opintojakson rakenne ja etenemisen vaiheet on esitetty selkeästi. Suorittamisen kannalta oleelliset, opintojaksokuvausta vastaavat asiat löytyvät kootusti verkkoalustalta.	Kyllä
Sisällöt (mm. kansiot, sivut, tiedostot) ovat tunnistettavia ja toimivia.	Kyllä
Sisältökokonaisuus näyttäytyy yhtenäisenä.	Kyllä
Fontit ovat helposti luettavia. Teksti on riittävän suurta tai suurennettavissa.	Kyllä
Tekstit ovat luettavissa ruudunlukijalla.	Kyllä
Visuaaliset elementit tukevat sisältöä.	Kyllä
Videot ja äänitiedostot on tekstitetty tai sisältö on muutoin saatavilla tekstimuodossa.	Ei
Linkkien kuvaustekstit ovat ymmärrettäviä. Linkit aukeavat uuteen selainikkunaan.	Osittain
Materiaali, joka ei ole esteetöntä, erottuu selkeästi.	Kyllä
Opiskelu onnistuu päätelaitteesta riippumatta.	Kyllä
Opiskelu verkkoalustalla ja eri sovelluksia sekä työvälineitä käyttäen on tietoturvallista. Sisältö ja materiaalit ovat tietoturvallisia.	Kyllä
 84.85%	

Tukipalvelut

Pedagogisiin ja teknisiin haasteisiin on saatavilla tukea.

Opettaja on tietoinen pedagogisen ja teknisen tuen kanavista. Hän voi tehdä tukipyynnön eri työvälineitä käyttäen.	Kyllä
Opiskelija löytää verkkoalustalta yhteystiedot, mistä hän saa apua opintojen sisältöjä ja suorittamista tai teknisiä ongelmia koskeviin kysymyksiinsä. Hän voi tehdä tukipyynnön eri työvälineitä käyttäen.	Kyllä
Tukipyyntöihin saa vastauksen sovitun aikataulun mukaisesti.	Kyllä
<div><div></div><div>100.00%</div></div>	

Tuloksesi:**78.91%**

Verkkototeutuksesi täyttää osan arviointikriteereistä, mutta vielä löytyy parannettavaa. Jos vastauksistasi löytyy ei- ja osittain-vastauksia, panosta näiden kohtien kehittämiseen. Opiskelijoilta, kollegoilta ja verkkopedagogiikan tukihenkilöiltä voit saada hyviä ajatuksia kehitystyöhön.

Liite 3. Pikaohje laadukkaan verkkokurssin toteuttamiseksi

1. Todenna tarve verkkokurssille.
2. Mieti verkkokurssin kohderyhmä ja tavoitteet.
3. Valitse kurssialusta ja merkintäkieli materiaalin kirjoitusta varten. Mieti samalla kurssin ulkoasu.
4. Perehdy lähdemateriaaliin ja suunnittele verkkokurssin osat.
5. Toteuta verkkokurssin materiaali, tehtävät, malliratkaisut, arviointi ja muut oppimista tukevat komponentit.
7. Testaa valmis verkkokurssi opiskelemalla se itse eri laitteita käyttäen.
8. Itsearvioi verkkokurssin toteutus käyttämällä verkkokurssin arviointityökalua.
9. Päivitä kurssia itsearviointiin ja saadun palautteen perusteella.