**Kalle Vesanterä**

**Joona Nuorala**

**Frisbeegolf-tulosalustan ohjelmistotuotantosuunnitelma**

**Ohjelmistotuotanto**

**CENTRIA-AMMATTIKORKEAKOULU**

**Tieto- ja viestintätekniikan insinööri**

**Joulukuu 2024**



**KÄSITTEIDEN MÄÄRITTELY**

Mikäli opinnäytetyössä esiintyy vaikeasti ymmärrettäviä ammattialan käsitteitä tai lyhenteitä, jotka vaativat määrittelyä, tehdään erillinen käsitteitä määrittelevä luettelo, johon lisätään myös mahdolliset lyhenteet. Luettelossa käytetään riviväliä 1,5. Käsitteiden määrittely tulee abstract:in jälkeen. Käsitteet kirjoitetaan aakkosjärjestykseen.

Esimerkki luettelosta

**ABS**

(Anti-lock Braking System) on ajoneuvon jarrujärjestelmä, jonka vaikutuksesta ajoneuvon pyörät eivät lukkiudu jarrutuksessa.

**XYZ123**

Tätä ohjetta varten keksitty esimerkkikäsite.

**KÄSITTEIDEN MÄÄRITTELY**

**SISÄLLYS**

[1 Projektisuunnitelma ja roolitus 1](#__RefHeading___Toc2793_3343872354)

[1.1 Tavoitteet 1](#__RefHeading___Toc2795_3343872354)

[1.2 Roolitus ja vastuut 1](#__RefHeading___Toc2797_3343872354)

[1.3 Yhteydenpito 2](#__RefHeading___Toc2799_3343872354)

[1.4 Prosessimallin valinta 2](#__RefHeading___Toc2801_3343872354)

[1.5 Tehtävät ja aikataulu 3](#__RefHeading___Toc2803_3343872354)

[1.6 Riskianalyysi 3](#__RefHeading___Toc2805_3343872354)

[2 Vaatimusmäärittely 6](#__RefHeading___Toc2807_3343872354)

[2.1 Toiminnalliset ja ei-toiminnalliset vaatimukset, priorisointi 6](#__RefHeading___Toc2809_3343872354)

[2.2 Vaatimusten validointi 6](#__RefHeading___Toc3504_3343872354)

[2.3 Webropol-kysely 6](#__RefHeading___Toc3222_3343872354)

[2.4 User story -esimerkit 8](#__RefHeading___Toc3224_3343872354)

[2.5 Projektin sidosryhmät 10](#__RefHeading___Toc3226_3343872354)

[3 Arkkitehtuuri- ja moduulisuunnittelu, saavutettavuus 12](#__RefHeading___Toc2817_3343872354)

[3.1 Koko järjestelmän kuvaus 12](#__RefHeading___Toc3617_3343872354)

[3.2 Ositus 15](#__RefHeading___Toc3619_3343872354)

[3.3 Moduulisuunnittelu 15](#__RefHeading___Toc3621_3343872354)

[3.4 Rajapinnat 16](#__RefHeading___Toc3623_3343872354)

[3.5 Riippuvuudet 16](#__RefHeading___Toc3625_3343872354)

[3.6 Saavutettavuus 16](#__RefHeading___Toc3627_3343872354)

[4 lähdeviittauksista esimerkkejä 17](#__RefHeading___Toc2819_3343872354)

**LÄHTEET 7**

**LIITTEET**

**KUVIOT**

KUVIO 1. Sään huomioiminen 5

KUVIO 2. Otsikko 5

**KUVAT**

KUVA 1. Moottoripyöräilyä Suomen talvessa 6

**TAULUKOT**

TAULUKKO 1. Vastaajien ikäjakauma 5

TAULUKKO 2. Otsikko 5

Päivitä sisällysluettelo napsauttamalla sitä hiiren oikealla ja valitsemalla:

**Päivitä kenttä, Päivitä koko luettelo, OK**

Kirjoita itse LÄHTEIDEN, KUVIOIDEN, KUVIEN ja TAULUKOIDEN sivunumerot sisällysluetteloon.

# Projektisuunnitelma ja roolitus

## Tavoitteet

Projektin tavoitteena on luoda uudenlainen alusta, jota käytetään frisbeegolf-turnauksissa ja peleissä. Vastaavanlaisia palveluita on jo olemassa, mutta suunnittelemaamme täysimittaista alustaa, joka toimii sekä täysimittaisena web-sovelluksena että natiivina mobiilisovelluksena ei ole olemassa.

Suurimpana tavoitteena onkin tavoittaa entistä suurempi yleisö, kuten pelaajat, kisajärjestäjät ja ratojen ja kilpailujen suunnittelijat ja järjestäjät.

Alustan on tarkoitus mahdollistaa sekä suurempien turnausten järjestäminen, että pienempien kaveriporukoiden väliset hupipelit. Mobiilisovellusta käyttäessä pelaaja voi helposti syöttää palveluun pelinaikaista dataa, kuten heitossa käytetyn kiekon ja kiekon sijainnin. Muut pelaajat voivat samalla seurata muiden pelaajien edistymistä esimerkiksi sovelluksen sisäisen karttapalvelun avulla.

Pelin aikainen data on saatavilla reaaliaikaisesti myös täysimittaisessa web-sovelluksessa, josta esimerkiksi kisajärjestäjät voivat seurata turnauksen edistymistä.

Kaikki pelien aikainen data myös tallennetaan, jotta esimerkiksi pelaajat voivat käydä tarkastelemassa heidän pelattuja turnauksiaan.

## Roolitus ja vastuut

Alustavasti roolitusta on mietitty seuraavan kaltaiseksi:

Kalle Vesanterän vastuulla on olla teknisen puolen ”asiantuntija” joka selvittää mitä teknisiä vaatimuksia ohjelmiston eri osilla tulee olemaan. Käytännössä tämä myös tarkoittaa sitä, että Kallen vastuulla on ohjelmiston backend ja sen suunnittelu.

Joona Nuoralan vastuulla on toimia eräänlaisen ”loppukäyttäjä-asiantuntijana”, eli tarkoitus on ideoida ominaisuuksia sekä sitä miten UI/UX puoli tulee toimimaan sovelluksessa. Tämä siis vastaa ohjelmiston frontend-puolta ja suunnittelua.

Molemmat tekevät oman osa-alueensa alustavaa dokumentaatiota. Yhteisissä tapaamisissa myös käydään tarkemmin läpi mitä suunnitellut ohjelmiston palat sisältävät, samalla myös sovittaen dokumentaatiota yhteen.

Koska kysymys on pienestä tiimistä, tullaan työtä tekemään myös paljon yhdessä. Asioita sekä frontend- sekä backend-asioissa voidaan päättää yhdessä, mutta epäselvyyksien välttämiseksi pääasiallinen vastuu on näiden osa-alueiden omistajilla.

## Yhteydenpito

Projektia tehdään pienessä ryhmässä ja työ tulee olemaan pääasiassa etätyöskentelyä. Yhteydenpito on suunniteltu tapahtuvan niin, että yhteisiä tapaamisia on tiiviisti ja näiden aikana katsotaan tarkasti mitä projektissa on tapahtunut, samalla sopien tulevasta.

Jatkuvaa yhteydenpitoa varten käytämme parhaaksi katsomaamme sähköistä viestintää, joka mahdollistaa niin teksti-, kuin ääni- ja videomuotoisen yhteydenpidon. Lisäksi yhteydenpitoon liittyy oleellisesti työkalut kuten Trello, Github ja muut työkalut joita projektin aikana käytetään.

## Prosessimallin valinta

Prosessimalliksi valittiin inkrementaalisen- ja prototyyppimallin hybridi. Tämä päätös tehtiin sen takia, että pienellä tiimillä toimiessa, sekä täysin uudenlaisen projektin kanssa työskennellessä, on tämän kaltainen malli helpompi hallita. Lisäksi tämän mallin myötä ei kulu aikaa turhaan suunnitteluun ja erilaisiin kokouksiin, vaan työn pääpainona voidaan pitää itse ohjelmiston parissa työskentely.

Koska kyseessä on myös tiimin ensimmäinen projekti, prototyyppimainen komponenttien kehittely ja kokeilu on huomattavasti hedelmällisempää. Täten saamme itse kehitettyä projektin sellaiseksi kuin haluamme, varaten kuitenkin mahdollisuuden muuttaa asioita ikään kuin lennosta.

Olemme myös itse projektimme täysimittaiset omistajat, joten myös sen takia tämä malli sopii meille parhaiten.

## Tehtävät ja aikataulu

Alustavasti tehtävät on määritelty seuraavalla tavalla:

• Projektin ideointi, alustavan projektisuunnitelman tekeminen ja roolien valinta

• Ohjelmiston vaatimusmäärittely, joka sisältää mm. toiminnalliset ja ei-toiminnalliset vaatimukset, ominaisuuksien priorisoinnin ja käyttäjiin ja heidän tarpeisiinsa liittyvät vaatimukset

• Ohjelmiston arkkitehtuurien ja moduulien suunnittelu, sisältäen frontend- ja backend-toiminnallisuuden, rajapinnat, riippuvuudet ja muut tekniset yksityiskohdat. Näistä koostuu myös koko järjestelmän kuvaus.

• CI/CD-suunnittelu, joka sisältää tuotteen- ja versionhallinnan sekä testauksen ja laadunvarmistuksen suunnitelmat

Aikataulu:

Projektin suunnitelluksi aikajanaksi on arvioitu seuraavaa:

• Projektin suunnittelu on parhaillaan käynnissä ja suunnitelman esityskelpoiseksi valmistumisen määräaika on 16.12.2024

• Ohjelmistokokonaisuuden rakentaminen alkaa 6.1.2025

• Ohjelmistokokonaisuuden odotetaan olevan valmis ensimmäiseen tuotantoversioon (alpha) 22.12.2025

• Lisäksi projektisuunnitelman kehittyessä, tulemme arvioimaan tarkempia aikajanoja ohjelmiston eri osille ja niiden määräajoille. Nämä tullaan päivittämään suunnitelmaraporttiin niiden selkeytyessä.

## Riskianalyysi

Projektille on kirjoitushetkellä tehty alustavaa riskiarviota. Tämä on aihe joka tulee myös elämään projektin kasvaessa ja sitä päivitetään tapauskohtaisesti reaaliajassa.

Tällä hetkellä riskiarvio sisältää seuraavat kohdat ja niiden todennäköisyydet ja vaikutukset:

• Taloudellinen riski (kustannuksia ei saada katettua) – Riski on pieni, sillä projektiin ei nykytiedon mukaan tulla käyttämään suuria taloudellisia panostuksia. Vaikutus on myös minimoitu, sillä se ulottuu ainoastaan projektin sisäisiin omistajiin. Vaikutuksia saadaan myös hallittua tekemällä harkittuja päätöksiä hankinnoissa ja pitämällä kustannukset mahdollisimman pieninä.

• Käyttäjäkunta ei koskaan kasva halutuksi – Riski on kohtalainen. Tähän vaikuttaa suhteellisen pieni kohderyhmä ja se että olemassa olevia palveluita on markkinoilla. Toisaalta alustan onnistuessa halutusti, voi tulos ottaa tuulta alleen suhteellisen nopeasti. Vaikutus on vakava, sillä käytännössä tässä kohtaa projekti on epäonnistunut. Riskin toteutuessa on mahdollista katsoa, voiko mahdollisesti epäonnistuneen projektin osia käyttää jossakin tulevassa projektissa, jotta työ ei olisi ollut täysin turhaa. Lisäksi tämän kaltainen projektin kaatuminen ei välttämättä ole epäonnistuminen, sillä kyseessä on kuitenkin arvokasta tietoa jota voidaan hyödyntää myöhemmin uralla, varsinkin kun suurta taloudellista panostusta ei ole tehty.

• Ryhmän jäsenen sairastuminen tai äkillinen poisjäänti – Riski on pieni, mutta tämän vaikutukset voivat olla katastrofaalisia ja projekti voi viivästyä tai jopa kokonaan keskeytyä tämän johdosta. Tämä on myös asia, johon ei käytännössä voi aina vaikuttaa. Riskin toteutuessa vaikutuksia voidaan minimoida sillä, että kaikki tiimin jäsenet ovat perillä projektin eri osa-alueista ja toimintatavoista. Tällä voidaan mahdollistaa se, että projekti ei täysin pysähdy, vaikkakin aikataulutus saattaa muuttua radikaalisti. Lisäksi tehokas viestintä on suuri osa tämän kaltaisen riskin minimointia.

• Kehityslaitteiston tai infra-laitteiston hajoaminen – Riski on pieni ja vaikutukset ovat pienet, jos ne on huomioitu projektin kulussa. Tässä tulee kuitenkin huomioida mahdollinen taloudellinen riski. Riskin vaikutuksia voidaan hallita esimerkiksi pitämällä tarpeelliset varmuuskopiot kaikesta projektiin liittyvästä. Lisäksi päätös ostaa laitteistopuolen palvelut ulkoiselta toimijalta takaa pitkälti sen, että laitteiston pettäminen ei vaikuta projektin kulkuun.

• Aikataulutuksen pettäminen – Riski on kohtalainen. Kyseessä on tiimin ensimmäinen vastaavanlainen projekti, jolloin riski virheelliseen aikataulutukseen kasvaa. Lisäksi yllättäviä muutoksia saattaa tapahtua, joka vaikuttaa projektin kulkuun suunnitellussa aikataulussa. Riskiin voi vaikuttaa valmistautumalla etukäteen mahdollisiin tilanteisiin, jotka tähän vaikuttaa, sekä myös laskemalla aikataulun eri osiin tarpeeksi ylimääräistä aikaa. Lisäksi projektin aikainen työajanseuranta helpottaa aikataulutuksen arviointia projektin edetessä, jolloin riski tämän toteutumiseen pienenee.

• Jos projekti ei ole päätoimista toimintaa, miten ajanhallinta saadaan toimimaan – Tähän sisältyy monia riskejä ja niiden vaikutukset voivat olla erilaisia. Parhaimmassa tapauksessa vaikutus on minimaalinen mutta pahimmillaan projekti voidaan joutua keskeyttämään.

• Kyberturvariskit – Kyberturva on osa-alue, joka tulee ottaa tosissaan, sillä riskit voivat olla hyvinkin suuret. Esimerkiksi ransomware-hyökkäyksen kohteeksi joutuminen voi olla kohtalokasta projektille. Tähän voidaan kuitenkin varautua noudattamalla hyviä toimintatapoja projektin datan hallinnassa, laitteiston ja ohjelmistojen käsittelyssä sekä yleisessä toiminnassa, jota projekti sisältää.

# Vaatimusmäärittely

## Toiminnalliset ja ei-toiminnalliset vaatimukset, priorisointi

Toiminnalliset ja ei-toiminnalliset vaatimukset määriteltiin pääosin projektin omistajien, eli kehittäjien, toimesta. Vaatimuksia käytiin läpi suunnittelupalavereissa. Niissä hyödynnettiin omia ideoita ja visiota alustaa kohtaan, alustavia käyttäjäkyselyjä sekä vastaavanlaisten alustojen käyttäjäkokemuksesta syntyneitä ideoita ja tarpeita.

Lisäksi vaatimusten priorisointia mietittiin vaatimuksia suunnitellessa. Priorisoinnissa pidettiin tärkeimpänä niitä ominaisuuksia, jotka koemme välttämättömäksi alustan suunnitellun toiminnallisuuden kannalta. Vaatimuksissa on myös ominaisuuksia, pienemmällä prioriteetilla, joiden koemme tuovan alustalle lisää syvyyttä. Nämä eivät kuitenkaan ole kriittisen tärkeitä alustalle kokonaisuus huomioiden.

Kokonaisuudessaan luettelo vaatimuksista ja ei-toiminnallisista vaatimuksista priorisointeineen löytyy taulukkona suunnitelman liitteistä.

## Vaatimusten validointi

Tekstiä tähän...

## Webropol-kysely

Vaatimusmäärittelyä varten teimme alan harrastajille suunnatun ytimekkään kyselyn, jossa kartoitimme tärkeimpiä ominaisuuksia, joita ohjelmiston tulisi sisältää. Kirjoitushetkellä kysely on ollut auki n. 4 päivää. Vastauksia tähän mennessä on tullut 16. Kyselyn alkuperäinen Webropol-raportti tämän suunnitelman liitteenä.

Kyselyssä kartoitettiin seuraavia asioita:

• Mitä frisbeegolfiin liittyviä palveluja olet käyttänyt?

100% vastaajista on käyttänyt Disc Golf Metrix -palvelua. Lisäksi PDGA Live sekä UDisc ovat olleet suosittuja alan harrastajien keskuudessa. Näihin ohjelmistoihin kannattaa siis kiinnittää erityisesti huomiota uutta suunnitellessa.

• Mistä seuraavista ominaisuuksista olet eniten kiinnostunut frisbeegolf-tulosalustoissa?

Tässä kysymyksessä pyydettiin valitsemaan 1-4 ominaisuutta, jotka ovat tärkeimmät alustalle. Yli 50% kannatukselle ylsi 3 vaihtoehtoa:

1. Mahdollisuus selata omaa peli- ja turnaushistoriaa

2. Mahdollisuus luoda omia kilpailuja, esimerkiksi muutaman hengen kaveriporukalle

3. Kilpailun tulosten reaaliaikainen seuranta

Lisäksi noin 40% kannatuksesta oli seuraaville ominaisuuksille:

4. Karttanäkymä lähialueen radoista

5. Erilaisia valmiita pelimuotoja (kuten paripeli ja skins)

Yllättäen, karttaominaisuus, jossa voi seurata kilpailun etenemistä, ei tässä osassa kyselyä saavuttanut suurta suosiota. Kyse on kuitenkin yhdestä ydinominaisuudesta, jota alustalle on suunniteltu, joten sille saattaa olla kannattavaa antaa joka tapauksessa suurempi prioriteetti.

• Millaisista lisäominaisuuksista olet kiinnostunut?

Lisäominaisuuksista kyseltäessä annettiin vastaajien valita vapaasti niin monta vaihtoehtoa kuin he halusivat. 3 vastausta ylsi 50% kannatusrajaan:

1. Heittopituuden mittaaminen

2. Ratakohtainen Highscore-taulukko

3. Tuloskortin jakaminen esim. sosiaalisessa mediassa

Nämä ovat ominaisuuksia, jotka kannattaa todennäköisesti nostaa ohjelmiston kokonaisuus huomioon ottaen melko korkealle prioriteeteissa.

Noin 40% kannatuksen keräsivät myös 2 lisäominaisuutta:

4. Puttiharjoitus

5. Radalla käytetyn reitin tallentaminen karttapalveluun

Nämä ominaisuudet yhdistettynä alustaan suunniteltuun karttapalveluun toimivat todennäköisesti melko hyvin yhteen. Tämän takia nämä kannattaa myös huomioida ohjelmiston kokonaisuutta toteutettaessa, vaikkakaan niiden prioriteettia ei tarvitse alkuun pitää kovin korkealla.

• Millä seuraavista alustoista olisit kiinnostunut käyttämään sovellusta?

100% vastaajista on halukkaita käyttämään palvelua mobiililaitteella, kuten älypuhelimella. Lisäksi 38% kertoi olevansa kiinnostuneita käyttämään palvelua tietokoneella. Tämä viestii siitä, että alustan mobiiliversion kehittäminen kannattaa nostaa prioriteettijärjestyksessä mahdollisimman korkealle.

• Mitä ominaisuuksia olet jäänyt kaipaamaan olemassa olevissa palveluissa?

Tähän kysymykseen saimme seuraavat vastaukset:

1. *” Esim metrixissä voisi olla tulostenkirjausnäkymässä sijoitusseuranta, esim. niin, että näkyy omaa sijaa 4 lähintä pelaajaa ja heidän kokonaistulos.”*

2. *” Selkeä ja yksinkertaistettu kisaluettelo jotka saa järjestettyä aika järjestykseen milloin ilmoittautuminen aukeaa. Tai karttapalvelu jossa näät kisat jotka on avoinna*

*ilmottautumiselle.”*

3. *” Karttoja”*

Alustavasti suunniteltu karttapalvelu vaikuttaa siis olevan ominaisuus, jolle olisi kysyntää, muodossa tai toisessa.

• Vapaa ideoiden antaminen.

Tähän kysymykseen saimme seuraavat vastaukset:

1. *” Selkeä ja yksinkertaistettu kisaluettelo jotka saa järjestettyä aika järjestykseen milloin ilmoittautuminen aukeaa. Tai karttapalvelu jossa näät kisat jotka on avoinna*

*ilmottautumiselle”*

2. *”Tarkempaa dataa radoista.”*

## User story -esimerkit

Yksi osa User story -esimerkkien keräämisessä on tekemämme Webropol-kysely ohjelmiston kohderyhmän käyttäjille. Lisäksi User storyjä kehitettiin itsenäisesti ohjelmistosuunnittelun aikana, jotta saatiin käsitys siitä, miten alustan suunnitellut ominaisuudet tukevat käyttäjäkokemusta.

Alla on listattuna User storyjä, liittyen ohjelmiston tiettyihin ominaisuuksiin:

1. Rekisteröityminen - Käyttäjänä haluan mahdollisuuden rekisteröityä sivustolle, jotta voin tallentaa tietojani palveluun ja katsella niitä jälkikäteen.

2. Salasanan nollaus - Käyttäjänä haluan mahdollisuuden nollata käyttäjätunnuksen salasanan, jos vaikkapa unohdan salasanani.

3. PDGA-numero käyttäjätilissä - Käyttäjänä haluan mahdollisuuden tallentaa PDGA-numeron käyttäjälleni, jotta PDGA-rating saadaan näkyviin palvelussa.

4. Tilin poistaminen - Käyttäjänä haluan mahdollisuuden poistaa käyttäjätili koska tahansa palvelusta, ilman mitään syytä, sillä haluan säilyttää päätösvaltani omien tietojeni käytöstä.

5. Profiilin tietojen julkisuus - Käyttäjänä haluan päättää itse mitkä tiedot profiilissani ovat julkisia, jotta voin päättää itse mahdollisesti yksityisyyteen vaikuttavien tietojen julkisuudesta.

6. Karttanäkymä lähialueen radoista - Käyttäjänä haluan nähdä itseäni lähellä olevat radat helpossa karttanäkymässä, jotta löydän reitin radoille jopa vieraalla paikkakunnalla ollessani.

7. Kilpailuihin liittyminen - Käyttäjänä haluan nähdä helposti listattuna kilpailut, joiden rekisteröityminen on auki.

8. Kilpailuihin liittyminen - Käyttäjänä haluan nähdä kartalla tulevat kilpailut, jotta voin helposti osallistua minua lähellä järjestettäviin kilpailuihin.

9. GPS-sijainnin käyttö - Käyttäjänä haluan mahdollisuuden käyttää puhelimen sijaintitietoja, jotta voin tallentaa esimerkiksi heittojen pituuksia ja radalla käytetyn reitin.

10. Tulosten kirjaaminen - Käyttäjänä haluan, että tuloksien kirjaaminen on mahdollisimman helppoa ja nopeaa, jolloin voin keskittyä paremmin pelin muihin osa-alueisiin.

11. Pelihistoria - Käyttäjänä haluan nähdä oman pelihistoriani, jotta voin seurata kehittymistäni ajan mittaan.

12. Heittotekniikan harjoittelu - Käyttäjänä haluan mahdollisuuden mitata heittojeni pituutta, tallentaen sen historiaan, jotta voin seurata kehittymistäni.

13. Sosiaaliseen mediaan jakaminen - Käyttäjänä haluan jakaa tuloskortin sosiaaliseen mediaan, esimerkiksi hyvän kierroksen pelattuani, jotta ystäväni pääsevät näkemään hyvän tuloksen.

14. Turnausten hallinta - Turnausjärjestäjänä haluan mahdollisuuden lisätä ja poistaa pelaajia turnauksesta, jotta hallinta olisi mahdollisimman helppoa.

15. Turnausten hallinta - Turnausjärjestäjänä haluan mahdollisuuden lisätä käyttäjiä turnauksen hallintaryhmään, jotta vaikkapa tuomarit saadaan osaksi turnauksenhallintaa.

16. Turnausten hallinta – Turnauksen järjestäjänä haluan mahdollisuuden lähettää osallistuville pelaajille viestejä, jotta kommunikaatio pelaajien kanssa olisi mahdollisimman sujuvaa.

17. Turnausten hallinta – Turnauksen järjestäjänä haluan mahdollisuuden lisätä turnaukseen eri luokkia, jotta esimerkiksi eri ikäluokat voidaan jakaa keskenään omiin peleihinsä.

18. Turnausten hallinta – Turnauksen järjestäjänä haluan mahdollisuuden jakaa pelaajat eri ryhmiin, jotta esimerkiksi mahdollisen sponsorin vaatimukset saadaan täytettyä.

19. Alustan käyttö - Käyttäjänä haluan mahdollisuuden selata alustan ohje-sivua, tai vastaavaa, jotta löydän tietoa ja apua siitä miten ominaisuuksia tulee käyttää.

20. Säännöt - Käyttäjänä haluan mahdollisuuden nähdä frisbeegolfin säännöt ja turnauksen sisäiset säännöt, jotta mahdollisilta väärinkäsityksiltä vältyttäisiin.

## Projektin sidosryhmät

Ohjelmistokokonaisuuden sidosryhmiksi on määritelty seuraavat:

1. Ohjelmiston omistaja:

Ohjelmiston omistajaksi on määritelty ohjelmiston tuottava ryhmä. Tämä perustuu siihen, että tarkoituksena on tuoda markkinoille uusi ohjelmisto, joka on täysin itse kehitetty ja tullaan tuomaan omasta toimesta markkinoille.

2. Toimittaja:

Ohjelmiston toimittajaksi on määritelty ohjelmiston tuottava ryhmä. Tuomme itse ohjelmiston markkinoille ja vastaamme sen julkisaattamisesta.

3. Tukiryhmä:

Tämä sidosryhmä kattaa kaikki ohjelmiston kehitykseen mahdollisesti osallistuvat ulkopuoliset asiantuntijat. Esimerkiksi ulkopuolelta palkattu konsultti tai tietoturva-asiantuntija kuuluu tukiryhmään.

4. Ohjelmistotestaajat:

Esimerkiksi ohjelmiston tuotannon alpha- ja betavaiheessa voidaan ohjelmistoa testata kutsumalla rajattu määrä käyttäjiä käyttämään ohjelmistoa. Heitä käsitellään omana sidosryhmänään, joka tuottaa palautetta ohjelmiston toiminnasta jatkokehittämistä varten.

5. Käyttäjät:

Ohjelmiston loppukäyttäjät ovat todennäköisesti kaikista laajin sidosryhmä. Tämä kategoria kattaa kaikki lopullisen ohjelmiston käyttäjät. Siihen sisältyy niin yksittäiset pelaajat, kuin myös mahdollisten kilpailuiden järjestäjät.

# Arkkitehtuuri- ja moduulisuunnittelu, saavutettavuus

## Koko järjestelmän kuvaus

Ohjelmistokokonaisuus on jaettu kahteen pääosaan, frontendiin sekä backendiin. Nämä molemmat koostuvat useammista pienistä komponenteista, jotka voivat oman pääosansa sisällä toimia yhteistyössä keskenään. Lisäksi frontend ja backend voivat keskustella toisiensa kanssa ohjelmistorajapintojen kautta.

Frontend-kokonaisuus koostuu kahdesta isosta osasta; web-pohjaisesta ja mobiililaitteille tarkoitetusta frontendistä. Nämä tulevat pitämään sisällään pitkälti saman toiminnallisuuden, mutta ne tulee kuitenkin rakentaa erikseen. Tämän työn helpottamiseksi ja nopeuttamiseksi, on näiden toteutusteknologioiksi valittu React ja React Native, joista ensimmäinen on tarkoitettu web-pohjaiseen frontendiin ja jälkimmäinen mobiililaitteiden frontendiin. Frontend koostuu monesta eri pääkomponenteista, jotka lähestulkoon kaikki voivat olla yhteydessä toisiinsa. Erilaisia komponentteja (tai näkymiä) yhdistellessä saadaan aikaan kokonaisuus, joka toimii yhteen saumattomasti ja yhteneväisesti.

Backend käsittää kaiken sen, mitä sovellus tekee taustatyönä. Käyttäjien-, turnausten- ja karttojenhallinta frontendissä saa tarvitsemansa tiedot backendin tietokannoista, rajapintojen kautta. Nämä tiedot ovat tallennettuna omiin tietokantoihinsa. Lisäksi tietoja muokattaessa frontendin puolella, välitetään data backendin rajapintojen kautta tietokantoihin. Lisäksi ulkoisista palveluista haetaan data backendissä. Esimerkiksi pelaajan tietoihin haettava tieto PDGA:n rajapinnoista, käsitellään backendissä, jonka jälkeen tämä tallennetaan vastaavaan tietokantaan. Myös erilaisten notifikaatioiden, kuten sähköpostien ja push-notifikaatioiden, lähettäminen tapahtuu backendin kautta.

Ohjelmisto on kokonaisuutena suunniteltu toimimaan pilvipohjaisessa Platform-as-a-Service (PaaS) -palvelussa. Tällä tavalla kehittäjien ei tarvitse itse huolehtia esimerkiksi laitteistopuolen hankinnoista ja ylläpidosta. Lisäksi pilvipohjaisissa järjestelmissä on pitkälti sisäänrakennettuna tietoturvapuolen ominaisuuksia, kuten myös nopeaa skaalautumista tukevia ratkaisuja. Lisäksi ohjelmistovaatimus ohjelmiston saatavuudesta saadaan varmemmin täytettyä pilvipohjaisella ratkaisulla. Web-pohjainen frontend ja backend tullaan toteuttamaan PaaS-palveluun omina instansseinaan, jotka toimivat yhteistyössä keskenään.

Seuraavaksi liitetyt kaaviot kuvaavat järjestelmän toimintaa ja sitä, miten kokonaisuuden eri komponentit viestivät toistensa kanssa:



*Frontend ja backend muodostavat keskenään kokonaisuuden.*



*Frontend koostuu erinäisistä komponenteista, jotka keskustelevat toistensa kanssa ja ovat myös yhteydessä backendiin.*



*Backend koostuu komponenteista, joilla hallitaan datan kulkua. Data voi kulkea sisäisesti rajapintojen kautta tietokantoihin. Lisäksi dataa voidaan hakea tai lähettää ulkopuoliselle palvelulle.*

## Ositus

Tähän tekstiä...

## Moduulisuunnittelu

Tähän tekstiä...

## Rajapinnat

Tähän tekstiä...

## Riippuvuudet

Tähän tekstiä...

## Saavutettavuus

Tähän tekstiä...

# lähdeviittauksista esimerkkejä

Opinnäytetyössä käytetyt lähteet ilmoitetaan tekstiviitteinä yleisesti käytössä olevan nimi-vuosi-viitejärjestelmän mukaan. Tekstiviite on sulkeissa oleva lyhyt viittaus lähdeluettelossa olevaan tarkempaan lähdetietoon.

Aho (2015, 106) tarkastelee sitä, kuinka lähdeluettelo ja lähteisiin viittaaminen tekstissä tehdään oikein. Jokaiselle viittaukselle tulee löytyä tiedot lähdeluettelossa. Jokaiseen lähdeluettelossa olevaan lähteeseen tulee viitata. (Aho 2015, 35).

Jos viittaus kohdistuu **useampaan virkkeeseen**, lähdeviite merkitään sulkeisiin asiakokonaisuuden loppuun pisteen jälkeen, ikään kuin itsenäiseksi virkkeeksi. ﻿Myös **sulkeiden sisään merkitään tuolloin piste﻿.**

**﻿Esimerkki viittauksesta useampaan virkkeeseen﻿:**

Opiskelija osaa lähdeviittaustekniikan. Hän on huolellinen ja noudattaa ohjeita. Silloin lukija tietää, mikä asia on mistäkin lähteestä ja mikä on opiskelijan omaa mielipidettä. (Aho 2015, 21–23.)

Jos viittaus koskee vain **yhtä virkettä**, lähdeviite sisällytetään kyseisen virkkeen sisäpuolelle. **Piste merkitään tuolloin siis sulkumerkin ulkopuolelle:**

**Esimerkki viittauksesta vain yhteen virkkeeseen:** Tämä on helppoa (Aho 2015, 18).

Jos tekijöitä on kolme tai useampia, tekstiviitteeseen merkitään ensimmäistä kertaa viitattaessa kaikki nimet (Hirsjärvi, Remes & Sajavaara 2014, 21–22), mutta myöhemmin merkitään ensimmäinen tekijä ja lyhenne ym. (Hirsjärvi ym. 2014, 23–24).

Lisää esimerkkejä lähdeviittauksista on oppaassa OPINNÄYTETYÖ- JA KIRJOITUSOHJEET >

6 Lähteisiin viittaaminen.

**LÄHTEET**

Aho, H. 2015. *Ajoneuvoja Suomen talvessa*. Kokkola: Kirjakustannus Oy.

Hirsjärvi, S., Remes, P. & Sajavaara, P. 2009. *Tutki ja kirjoita*. 15., uudistettu painos. Helsinki: Tammi.

Sieppinen, A. 2011*. Resurssit osana asiakkaan arvonluontia palveluliiketoiminnassa. Pankkien lainapalvelut kuluttajille*. Maisterin tutkinnon tutkielma. Helsinki: Aalto-yliopisto. Saatavissa: http://urn.fi/URN:NBN:fi:aalto-201301151018. Viitattu 24.4.2023.

Lisää esimerkkejä lähteiden merkitsemisestä lähdeluetteloon on oppaassa [OPINNÄYTETYÖ- JA KIRJOITUSOHJEET](https://libguides.centria.fi/yhteinenoppariohje) > 7 Lähdeluettelon laatiminen.

LIITE 1

**Ohjeita liitteiden käyttöön**

Liitteitä ovat työn sisältöön kuuluvat ohjekirjat, haastattelulomakkeet, taulukot ja kuviot sekä muu materiaali, jonka näkeminen varsinaisen työn yhteydessä ei ole välttämätöntä. Liitteet sijoitetaan opinnäytetyössä viimeiseksi.

Liitteissä ei käytetä sivunumeroita. Sen sijaan liitteet numeroidaan erikseen juoksevasti: LIITE 1, LIITE 2 jne. oikeaan yläreunaan. Jos esimerkiksi liitteessä kaksi on kolme sivua, merkitään seuraavasti: LIITE 2/1, LIITE 2/2, LIITE 2/3. (Liitteitä on kuitenkin vain se määrä, minkä liitteen päänumero osoittaa eli tässä esimerkissä kaksi, siis liitteet 1 ja 2.)

Tekstissä viitattaessa liitteeseen ei liitteen sivunumeroita mainita, vaan viitataan vain liitteen numeroon (esim. LIITE 1). Liitteessä olevaan kuvioon ei myöskään erikseen viitata, vaan viittaus kohdistuu suoraan liitteeseen. Esimerkiksi: Kokkolan seutukunnan sekä Keski-Pohjanmaan koulutusyhtymän ja Centria-ammattikorkeakoulun runkoverkko koostuu Hewlett-Packardin ProCurve-tuotesarjan kytkimistä (LIITE 1).

LIITE 2/1

Esimerkki monisivuisen liitteen 2 sivusta 1.

LIITE 2/2

Esimerkki monisivuisen liitteen 2 sivusta 2. Tähänkin liitesivuun viitattaessa käytetään viittausta liitteeseen 2 (LIITE 2).

LIITE 2/3

Esimerkki monisivuisen liitteen 2 sivusta 3. Tähänkin liitesivuun viitattaessa käytetään viittausta liitteeseen 2 (LIITE 2).