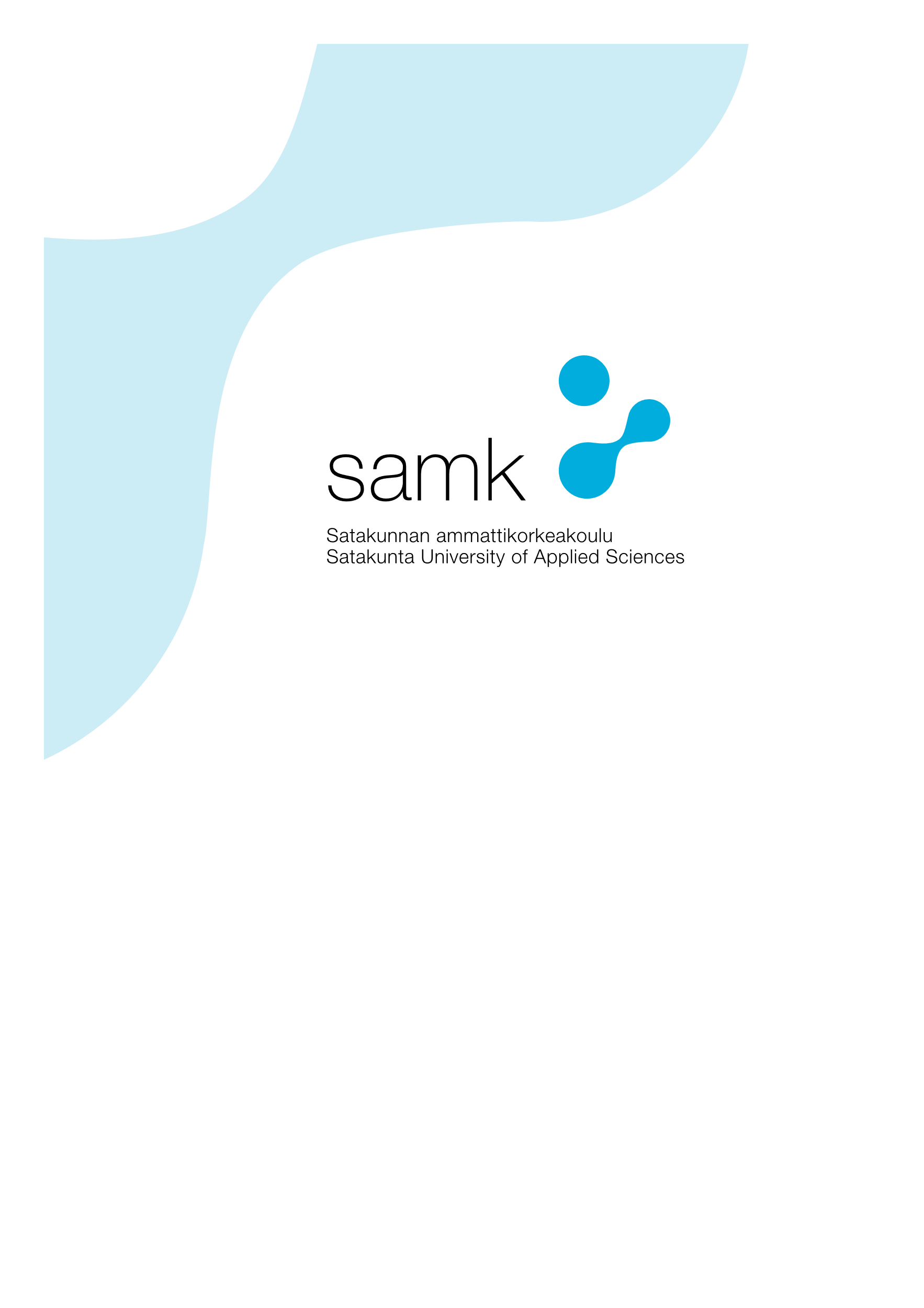
VILLE-MATTI LÖYTÖLÄ

**Interaktiivinen Virtuaalikierros**



|  |  |  |
| --- | --- | --- |
|  | |  |
| Tekijä(t)  Löytölä Ville-Matti | Julkaisun laji  Opinnäytetyö, AMK | Päivämäärä Kuukausi Vuosi |
| Sivumäärä | Julkaisun kieli |
| Julkaisun nimi  **Interaktiivinen virtuaalikierros** | | |
| Tutkinto-ohjelma  Sähkö- ja automaatiotekniikka | | |
| Halusin oppia tekemään Reactilla ja interaktiivinen virtuaalikierros mahdollista tämän. Myös aina kiinnostanut Vr-maailmat ja tähän sain opinnäytetyön kautta pikku kosketuksen.  Tehtävänä oli luoda nettisivut, joitten avulla pystyisi tutkimaan erillaisia interaktiivisia tiloja, nämä tilat muodostaisivat käyttäjän ympärille sylinterimäisen tilan, jonka näkymää pystyisi siirtämään hiiren avulla. Myös kyseistä tekniikka pystyisi hyväksi käyttämään seuraavissa projekteissa, koska teen itselleni opinnäytetyön.  Koodaus pohjana käytettiin Reactia ja React-nativea, joissa Reacti on JavaScript kirjasto. Tämä käyttä komponentteja ja tiloja hyväkseen. Näillä on nopea ja helppo luoda koodia. Myöhemmin löydettiin React-Vr framework, joka mahdollisti kaiken tarvittavan. Tätä käyttäessä huomattiin uusimman version menevän nimellä React-360 ja tämä oli viimeksi päivitetty 2018. Tämän johdosta tuli paljon virheitä vanhentuneista tekniikoista, mutta saatiin tehtyä loppuun asti.Vanhentumisen takia projekti vei yllättävän paljon aikaa.  Yksinkertaiset virtuaali-sivut saatiin tehtyä ja myöhemmin paransin toiminnallisuutta ja eniten sivuston ilmettä. Koska kyseistä kirjastoa ei ole päivitetty pitkään, uusien tekniikoitten käyttäminen oli hankalaa ja siitä johtuen sivuston pohja ensin näytti hyvin vanhalta. | | |
| [Asiasanat](http://www.finto.fi)  Haetaan asiasanaluettolosta mutta siihen ei tehdä linkitystä | | |

|  |  |  |
| --- | --- | --- |
| Author(s)  Löytölä Ville-Matti | Type of Publication  Bachelor’s thesis  ThesisAMK | Date Month Year |
| Number of pages | Language of publication: |
| Title of publication  **Interactive virtual tour** | | |
| Degree programme  Electrical and Automation Engineering | | |
| I wanted to learn what was possible with React, and the interactive Virtual Tour made this possible. I have also always been interested in VR worlds and got a little touch of this through my thesis.  The task was to create websites that could be used to explore various interactive spaces, these spaces would form a cylindrical space around the user, the view of which can be moved with the help of a mouse. I could also use this technique in the following projects, because I am doing a thesis for myself.  The coding was based on React and React native, where React is a JavaScript library. Components and the client are used here. These are quick and easy to generate code. Later, the React-Vr framework was discovered that made everything possible. At this point, we noticed that the latest version of React-360 was last updated in 2018. As a result, there were a lot of bugs due to outdated techniques, but we got it done to the end. Due to aging, the project took a surprisingly long time.  Simple virtual pages were made and later I improved the site's functionality and, above all, its appearance. Since the library has not been updated, it was difficult to use the new technologies and therefore I was very reluctant to use the site at first. | | |
| [Key words](http://finto.fi/en/)  Search from key word list but do not link | | |

ALKUSANAT

Lyhyt, tyyliltään ja sävyltään asiallinen ja myönteinen. Kiitossanat.

SISÄLLYS

Otsikon SISÄLLYS fonttikoko on 14, sisällysluettelon 12.

Sisällysluettelo päivitetään hiiren oikeanpuoleisesta painikkeesta sisällysluettelon päällä 🡪 Päivitä kenttä /Update field. Tekstiosaan otsikkotyyleillä kirjoittamasi otsikot tulevat tällöin näkyviin sisällysluetteloon. Jos sisällysluettelossa on kirjoitus- tai asiavirhe, korjaa se tekstiosaan, ei sisällysluetteloon. Jos lyhyen otsikon jälkeen sivunumero ei sisenny oikeaan reunaan, sisennä se tabulaattorilla.

Poista tämä ohje. Älä poista sisällysluetteloa tai sen alapuolella olevia tyhjiä rivejä.

1 Johdanto 6

2 Interaktiivinen Virtuaalikierros 7

2.1 Aiheen sisällön valinta 7

2.2 Frameworkin valinta 7

2.3 Pohjan rakenne 8

2.3.1 Dynamiikan luonti 10

2.4 Versiohallinta 11

2.5 360-Panoramakuvat 12

2.5.1 Solmupiste 13

2.5.2 Ricoh Theta SC2 15

3 React-360 16

3.1 Node.js 16

3.2 React-VR ja React-360 -framework 16

3.3 Surge 17

4 verkkosivun projektin luonti 17

5 Lähteet 20

LIITTEET

# Johdanto

Virtuaalitodellisuuden mahdollisuudet ovat viime aikoina saaneet huomattavasti julkisuutta johtuen teknisistä edistysaskelista edullisten virtuaalilasien saralla. Virtuaalitodellisuusteknologian käyttö ei ole kuitenkaan yleistynyt laajasti, ja vielä ei ole olemassa virtuaalitodellisuuden läpimurtosovellusta. Eräs syy tälle tilanteelle voi olla virtuaalitodellisuuden ohjelmistokehityksen vaikeus, ja siihen kuuluvat erityisongelmat. (Tuukka M. Takala, 2017)

Tämän opinnäytetyön tavoitteena on VR-tekniikalla tehtyjen verkkosivujen rakentaminen ja näitten rakenteiden haasteitten kuvaaminen. Verkkosivujen vähimmäistavoitteena on pystyä havainnollistamaan ja tutustuttamaan käyttäjät uusin tiloihin ja paikkoihin. Kyseistä tekniikka käytetään paljon museoiden, festivaalien ja muiden mielenkiintoisten kohteiden esittelyssä.

Virtuaalitodellisuuden käyttö on yleistynyt ja noussut suosioon lähiaikoina. Tähän ovat panostanut isot yhtiöt, kuten Facebook, Google, Microsoft, Samsung ja Sony, nämä kaikki ovat kehittäneet lisättyä todellisuutta ja virtuaalitodellisuutta, panostaen paljon rahaa kyseisiin tekniikoihin. Tämän myötä erilaisten virtuaalitodellisuus painotteisten pelien tekeminen on myös yleistynyt. Korona aikana rajoitettiin lähikontaktien ylläpitoa, jonka vuoksi kyseinen aihe on alkanut herättämään lisäkiinnostus. Opinnäytetyö esittelee React-kirjastoa, joka on kirjoitettu JavaScript kielelle. Kyseisen kirjasto valittiin helppouden ja komponentti rakenteen vuoksi, moni tekniikka on siirtynyt komponentti rakenteen puolelle. Kirjaston avulla tekijän ei tarvitse tehdä aivan kaikkia toiminnallisuuksia itse, vaan ne ovat valmiita kokonaisuuksia kirjastossa. Opinnäytetyö myös syventyy React-VR frameworkiin, joka on Reactin sisälle rakennettu framework. Tämä helpottaa VR-tekniikan tuontia alustalle, samaisella tavalla kuin kirjasto, mutta kyseessä on pienempi kokonaisuus. Projektin loppupäässä siirryttiin React-360:een, joka on uudempi versio React-VR:stä. Molemmat ovat Facebookin rahoittamia hankkeita, jotka ovat olleet jäädytettyinä muutaman vuoden. Koska kyseessä olevia paketteja ei ole päivitetty, tämä loi tekemiseen omia haasteita.

Lisäksi opinnäytetyössä käydään laajasti lävitse 360 kuvien ottamisesta, koska kyseinen asia loi isoja haasteita projektin alussa.

# Interaktiivinen Virtuaalikierros

## Aiheen sisällön valinta

Opinnäytetyön sisällön aiheena on luonto. Korona aikana on ihmisten aika sisätiloissa lisääntynyt ja tavoitteena on tuoda maailman luonto sisätiloihin VR-muodossa. Luonto kuvista ei tarvitse miettiä yksityisyyden suojaa, koska niissä ei näy muita ihmisiä tai ihmisten omistamia asioita. Kuvien korvaaminen ja vaihtaminen on helppoa, mikä mahdollistaa osaamisen ja tekniikan myynnin eteenpäin. Opinnäytetyön yhtenä päämääränä on verkkosivujen toimiminen kännykällä ja mahdollisimman monella selaimella optimoidusti.

## Frameworkin valinta

Verkkosivuston pakolliset toiminnallisuudet ovat, toimivuus selaimessa ja virtuaalitodellisuuden toimivuus selaimessa. Hyvänä toiminnallisuuden lisänä on toimivuus puhelimen selaimessa. Puhelin tuo lisä vaikeuksia, esimerkiksi kosketusnäytöllä tarvitsee toimia näkymän siirtäminen ja nappulat. Näitten vaatimuksien johdosta valitsin React-VR, joka käyttää React ja React-native kirjastoja. React on enemmän verkkoselainta varten tehty kirjasto, kun taas React-Native puhelin sovelluksia varten. Myös virtuaalitodellisuuden luominen on mahdollista kyseisessä frameworkissä. WebVR Polyfill tarvittiin myös asentaa, toimivuuden parantamiseksi. Ilman WebVR Polyfillia projekti ei toiminut uusimmissa verkkoselaimissa, aiheuttaen vakavia virheitä. Myöhemmin päivitettiin uusimaan versioon, joka oli vaihtanut nimeä. Tämän frameworkin nimi oli React-360 ja siirryttiin sen käyttämiseen. Uusin toteutus kyseisestä frameworkista on julkaistu 2018, mikä tietotekniikka maailmassa pitkä aika. Tästä johtuen uusimpien tekniikoiden käyttäminen ei ollut mahdollista, kuin myös pakotti asentamaan WebVR Polyfillin projektin toimivuuden parantamiseksi.

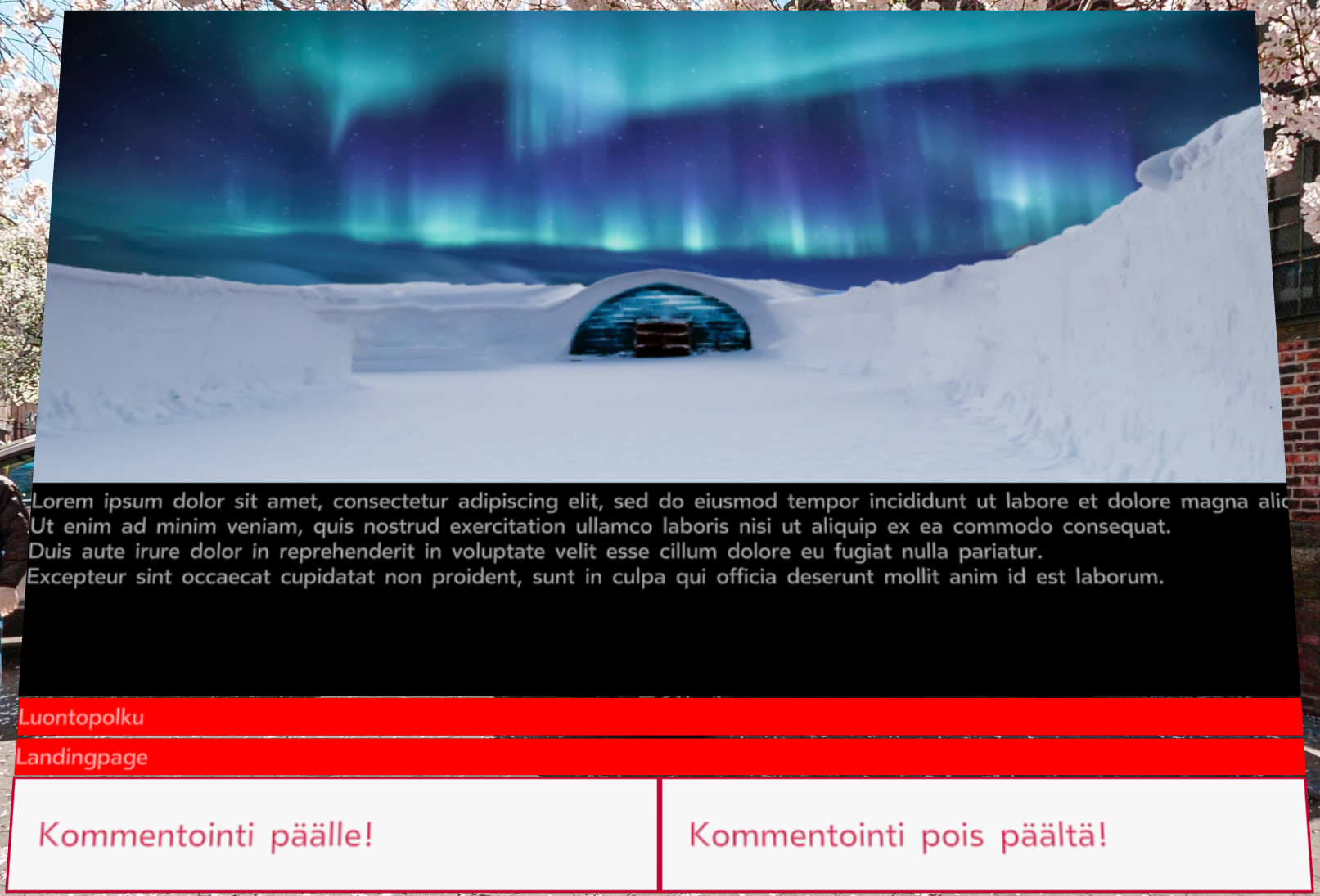
## Pohjan rakenne

Suunnitelma oli rakentaa interaktiivinen virtuaalikierros, kyseistä tekniikka on paljon käytetty esimerkiksi museoissa (Louvren virtuaalikierros hyvänä esimerkkinä), näistä on helppo ottaa mallia, mutta tehdä omalla tyylillä ja lisätoiminnallisuudella. Aloituksessa piti tehdä päätös koko pohjan rakenteesta. Projektin malliksi päädyttiin erotukseen jossa ensimmäisenä on aloitussivu, josta valitaan kohde ja navigoidaan kohteen sivuille. Aloitus-sivulla nähdään mahdollisten kohteiden nimet ja yleistietoa kohteista. Näistä nimistä voi valita mihin kohteeseen matkustetaan. Klikkaamalla kohteen nimeä näkymä vaihtuu, jolloin luodaan virtuaalitodellisuus uudesta valitusta paikasta. Tämä näkymä koostuu infopaneeleista, matkustamista ja yleisvalintoja varten oma paneeli. Matkustamis-paneelista pystytään palaamaan takaisin vanhaan paikkaan ja valitsemaan myös aloitussivu, jolloin kaikki alkaa alusta. Myöhemmin lisättiin kommenttiraita, joka yleisesti kertoo tietoa paikasta. Kommenttiraita alkaa pyörimään taustalla, kun valitaan ensimmäinen kohde. Kommenttiraidan myös saa mykistettyä matkustamis- ja yleisvalinta-paneelista.

Graphical user interface, website

Description automatically generated

(aloitussivu, jossa valintoina porin metsä ja luontopolku)



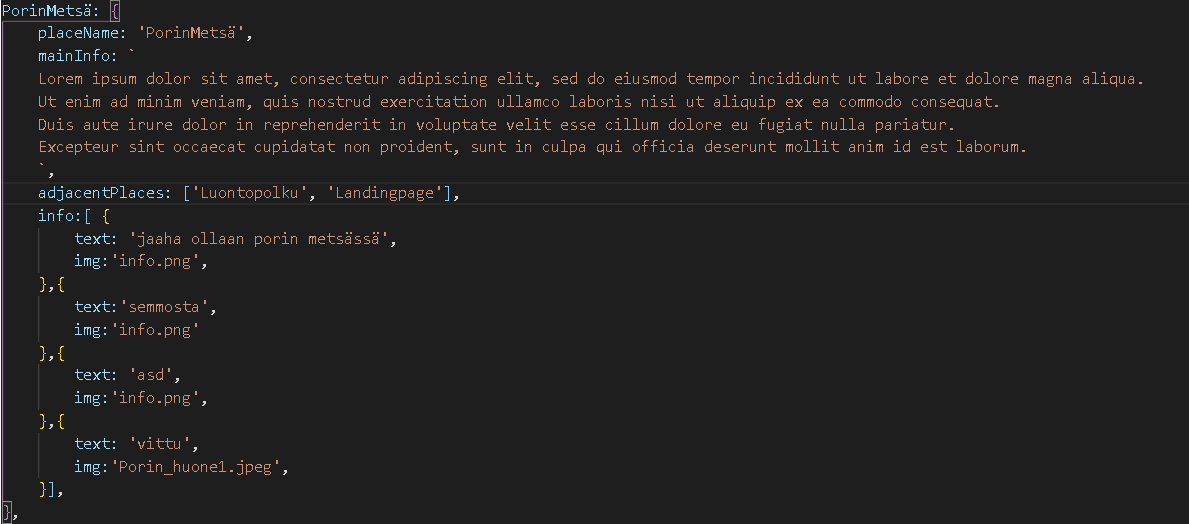
(kuvassa matkustamispaneeli, tässä kuva paikasta, yleistietoa paikasta, mahdollisuus matkustaa taaksepäin tai eteenpäin ja myös kommenttiraidan päälle kytkeminen tai pois.)



(kuvassa yleisnäkymä tilasta ja infopaneelin nappula.)

### Dynamiikan luonti

Hankalin asia projektissa oli tilojen dynaamisen rakenteen luominen. Tämä haluttiin toteuttaa niin ettei missään kohtaan tarvitse suoraan koodata tietoja eri tiloista ylätasolla olevaan paneeli määritykseen, jossa määritetään esimerkiksi infopaneelit, tiedot viereisistä paikoista mihin voi matkustaa, kommenttiraidat yms. Kyseiset tiedot tulisi yhdestä objektista, joka määritellään erikseen. Ylätasolla ei haluta määritellä tietoja, koska tämä ei olisi ollut hyvin dynaamista ja pitäisi aina luoda uusi pinta kaikkea varten. Dynamiikan avulla voidaan helposti määritellä uusia tiloja, joka helpottaa käyttämistä tilojen muuttuessa. Dynamiikan luonnissa tärkeintä on datatiedoston luominen ja sen muodon päättäminen. Koska koodauksessa käytetään JavaScriptiä, data-tiedoston muoto on ”.json”. Dynamiikan avulla pystytään määrittelemään uudet tilat ja sen viereiset tilat ja sovellus osaa ne rakentaa halutulla tavalla, näin ei tarvitse kuin datatiedostoa muuttaa kun halutaan lisää tiloja.



(Liitteenä kuva datatiedostosta ja yhdestä kohteen tiedoista. Paikan nimi on kohdassa ”placeName” ja myös ylimpänä, jonka mukaan pystytään etsimään kyseinen kohta. Kohta ”adjacentPlaces” kertoo mitä kohteita löytyy kyseisen paikan vierestä. Kohta ”info” kertoo info-panelien tekstit ja kuvan tiedot. Tämä annetaan objektina eteenpäin ja näin osaa piirtää näkymän tilasta.)

## Versiohallinta

Versiohallinta on hyvin käytännöllinen projekteissa, tällä turvataan tallentaminen ja jakaminen toisille tahoille. Versiohallinta auttaa, jos projektissa tulee toimivuutta vaarantavia virheitä, tällöin voidaan palata aiempaan toimivaan versioon ja poistaa vaarantavat tekijät projektista. Versiohallinta sovelluksena oli Git. Gitti on ilmainen ja avoinlähdekoodi sovellus, se on myös todella vanha ja hyvin toimiva, myös hyvin tunnettu ja virhetilanteissa helppo löytää vastaus virheeseen. Gitillä tehdään tiedostoista päähaara, josta pystyy tekemään uuden haaran ja kun tämä haara valmis voidaan se yhdistää eri haaroihin ja yleensä yhdistetään takaisin päähaaraan, kun on haluttua. Yleensä pidetään päähaaraa, kehityshaaraa ja tuotantohaaraa erikseen. Päähaaralle tulee halutut muutokset, jotka viedään tuotantoon hallitusti. Kehityshaaraan viedään pienellä kynnyksellä ja kokeillaan toimivuutta ja toiminnallisuutta. Nämä haarat päivitetään erilaisille alustoille esimerkiksi bitbucket tai github. Projekti perustettiin githubbiin helppouden ja yksinkertaisuuden vuoksi. Githubista on verkkoselain sivut ja sieltä pystyy päivittämään ja muokkaamaan haluttuja haaroja ja projekteja. Sieltä voidaan hakea projektin tiedostot eri tietokoneille. Tämä helpottaa hyvin paljon projektin kehittämistä, kun projekti ei ole tietokone kohtainen, myös helpottaa projektin jakamista eri ihmisille tai myöhemmin kyseisen projektin käyttöönottoa. Vaikka kyseisen projektissa oli vain yksi henkilö, oli githubista paljon hyötyä projektin aikana. Esimerksi projektissa oli muutaman kerran vaarantavia virheitä ja näin pystyttiin palaamaan haluttuun kohtaan. Projektia kehitettiin pöytäkoneella, kuin myös kannettavalla tietokoneella. Tämän myötä ei tarvinnut aina tarkistaa, että molemmat koneet olivat päivittyneet uusimpaan versioon projektista, vaan pystyttiin Gitin kautta päivittää uusimpaan versioon. Gitin peruskomentoihin kuuluu ”git pull, git push, git commit -m, git branch -b”. Näillä komennoilla pystyy hakemaan uusimman haaran, puskemaan haaran, tekemään uuden kommitin ja viimeisellä komennolla pystyy luomaan uuden haaran. Ainoa huono puoli Gitissä on, että se ei kysele ”haluatko oikeasti tehdä näin?”, vaan tekee. Minkä takia käyttäjä pohjaiset virheet ovat yleisiä ja tämän johdosta pitää olla hyvin varoivainen mitä tekee, ettei tule vaarantavia virheitä pelkästä versiohallinnan käytöstä.

Git on alkuperin hyväksynyt Linus Torvalds vuonna 2005 Linux kernelin kehittämiseen. Vuodesta 2005 lähtien Junio ​​Hamano on ollut ydinylläpitäjä. Kuten useimmat muut hajautetut versionhallintajärjestelmät, ja toisin kuin useimmat asiakas-palvelinjärjestelmät, jokainen Git-hakemisto jokaisessa tietokoneessa on täysimittainen arkisto, jossa on täydellinen historia ja täydet versionseurantaominaisuudet, riippumatta verkkoyhteydestä tai keskuspalvelimesta. Tämä mahdollistaa eri haarojen korjauksen ja virheiden etsimisen helpoksi, parantaen käyttöpitoa merkittävästi.

Git on ilmainen ja avoimenlähdekoodin ohjelmisto, jota jaetaan vain GPL-2.0 lisenssillä.

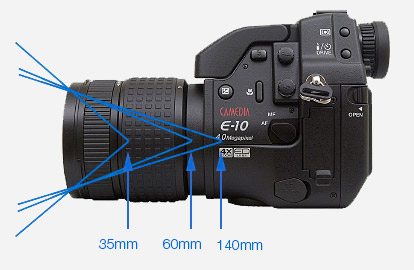
## 360-Panoramakuvat

Hankalin asia oli 360-panoramakuvien ottaminen. Opinnäytetyöllä ei ollut toimeksiantajaa ja tämän takia haluttiin välineitten kulut minimoida. Alussa otettiin kuvia järjestysvalvojakameralla ja kalansilmäobjektiivilla, mutta kyseinen tekniikka vaatii hyvin tarkkaa säätöä ja kalliita välineitä. Kyseinen vika tuli siitä, että nidonta ohjelmat eivät pystyneet tunnistamaan kuvien välistä yhteyttä ja näin sitoa kuvat toisiinsa luoden 360 kuvan. Tämä johtui siitä, että solmupisteen löytäminen halvoilla välineillä oli melkein mahdotonta. Tämän takia päädyin ostamaan Ricoh Theta SC2, 360-aktiivikamera. Kuvien laatu tulisi kärsimään tästä syystä, mutta helpottaen niitten ottamista huomattavasti. Panoraama pään ostaminen järjestelmäkameralle olisi ollut kalliimpaa kuin kyseisen kameran ostaminen ja kalansilmäobjektiivi oli lainassa, niin sekin olisi tarvinnut ostaa kuvien ottamista varten.

Kalansilmäobjektiivilla 360-kuvan ottamista varten tarvitsisi ottaa kuvat yhdessä vertikaalisessa linjassa 25-asteen erissä toisistaan ja myös kuvauskohdan maasta ja taivaasta. Ilman kalansilmäobjektiivia tarvitsisi vielä ottaa monesta erilaisesta vertikaalisesta linjasta kuvia. Tämä olisi nelikertaistanut kuvien tarpeen yhdestä pisteestä. Theta osaa tehdä tämän kaiken yhdellä kuvalla ja näin helpottaa merkittävästi kuvan ottamista, mutta kuvien laatu tuli kärsimään tällä tekniikalla ottaessa. Opinnäytetyön tavoitteena ei ollut kuvien parhaimman laadun vaatiminen, tästä johtuen kyseiset kuvat olivat enemmän kuin tarpeeksi hyviä. Konseptin mahdollistaminen oli tärkeämpää.

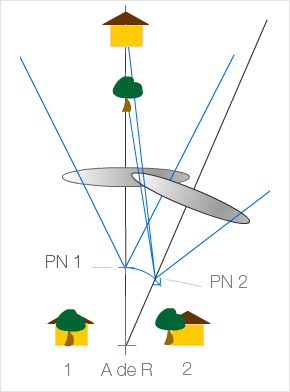
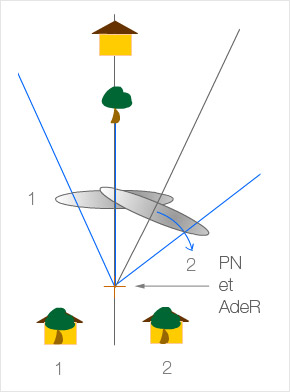
### Solmupiste

Solmupiste, toisella nimellä parallaksia piste. Solmupiste esittää linssin perspektiivin pistettä (sen silmä). Se on kohta linssin objektiivissa johon linssi kohdistaa kuvan. Zoomaus linssissä tämä piste siirtyy sisäänpäin verrattuna muissa linsseissä. Tämän johdosta solmupisteitä voi olla monta. Yleensä riittää tietää kolme erilaista solmupistettä. (panoramic-photo-guide)



(kolme erilaista solmupistettä erilaisissa linsseissä (panoramic-photo-guide))

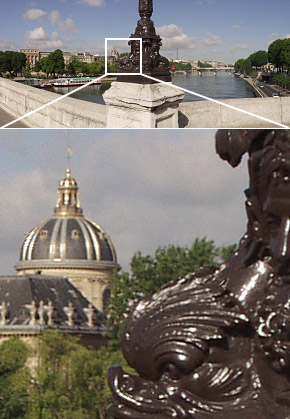
Solmupiste on hyvin tärkeä 360-kuvien ottamisessa, koska muuten kuvan perspektiivi muuttuu ja näin 360-kuvassa olevat esineet vaihtavat paikkaa luoden klooneja itsestään.

(kuvat.(panoramic-photo-guide))

Vasemmassa kuvassa solmupiste ei ole pyörivän akselin kohdalla. Kun kuvaaja siirtää kameraa oikealle, tuntuu kuin koko kuvaaja liikkuisi oikealle, josta syntyy kaksi samaa puskaa vähän eri kohdissa. Oikeassa kuvassa solmupiste ja pyörivä akseli ovat samassa kohdassa ja näin kuvien sitominen onnistuu. (panoramic-photo-guide).

Tämä aiheutti monta kertaa ongelmia opinnäytetyön aikana ja sidonta sovellukset loivat isoja eroavaisuuksia ja haamuja kuvissa.

(kuvat. (panoramic-photo-guide))

Näistä kuvista hyvin näkyy onko solmupiste kunnossa ja kummassa ei ole. Vasemmanpuolen kuva on otettu huomioimatta solmupistettä ja tästä johtuu kupolin ”varjo”, kun taas oikeassa on huomioitu solmupiste ja näin kuvien sitominen on onnistunut.

### Ricoh Theta SC2



(kuva. (ricoh))

Ricoh Theta SC2-actionkameralla kuvaa upeita 4K 30 FPS-resoluution videoita ja yksityiskohtaisia kuvia. Kamera on hyvin pienikokoinen ja helppokäyttöinen. Kameran aivan alapuolella on sokea piste, jota hyödyntäen otin kuvat. Seisoin kameran alapuolella, joka kuvassa.

# React-360

## Node.js

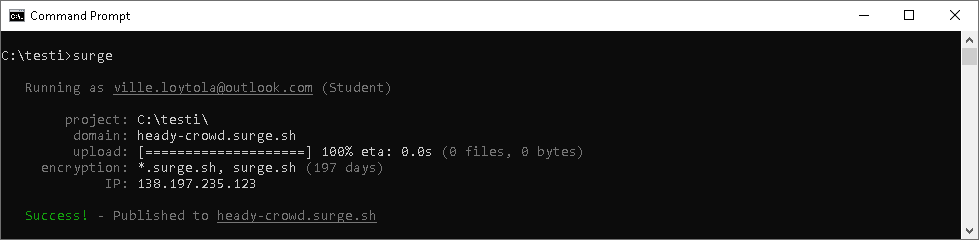
Node.js on avoinlähdekoodi, järjestelmäriippumaton JavaScript käyttöympäristö, projektin pyörittämiseen käytetään Node.js. Node.js asentaessa asentuu myös npm pakettien hallinta sovellus JavaScript paketeille. Tätä käytetään paljon projektin pakettien, kirjastojen, frameworkkien asentamiseen ja on pakollinen olla ennen kuin aloitetaan. Npm sovelluksen komento on ”npm”, jolla asennetaan paketit. Node.js yleisin komento on ”node”. Kahta kyseistä komentoa tulee näkymään paljon tulevissa kappaleissa ja käytetään vain käynnistämään kyseinen tarvittava sovellus, joka aloittaa toiminnallisuuden pyörittämisen.

## React-VR ja React-360 -framework

React-VR nimettiin uudelleen 2017 React-360:seksi. React-360 kirjastoa on viimeksi 2018 päivitetty, jonka johdosta kyseessä on vanhentunut frameworkki. React-360 frameworkin tavoitteena oli helpottaa kehittäjien työtä luodessa mukaansatempaavia 360- sisältöä selaimelle. React-360 mahdollistaa interaktiivisen virtuaalitodellisuuden luomisen selaimelle, jota pystytään käyttämään kännykällä ja tietokoneella. On myös mahdollista käyttää VR-laseja tietokoneen ja kännykän kautta. Tämän saavuttaakseen frameworkiin kuuluu React ja React-Native. React on selainpuolen luontia varten oleva avoinlähdekoodi kirjasto. Tällä luodaan selain pohjaisia sovelluksia. React-Native kirjasto perustuu enemmän puhelin selainpuolen sovelluksien luontiin. Reactin isoin hyöty on tilojen hallitseminen, mikä mahdollistaa sivuston interaktiivisuuden ja sivuston tilojen päivittämisen lennossa. Frameworkki käyttää Three.js kirjastoa ohjelman esittämisenä näytöllä. Käytin Surge.sh sivustoa ja kirjastoa sovelluksen levittämiseen.

## Surge

Surgen käyttäminen on tehty helppoksi. Tarvitsee ainoastaan asennetaan koneelle komennolla ”npm install –global surge”, mikä asentaa surgen kyseiselle koneelle. Kirjaston voidaan asentaa projektin sisään ilman ”-global” komentoa, silloin sitä voidaan käyttää projektin sisäisesti. Tällöin kun projektista tehdään koontiversio, surge käynnistetään projektin sisäisesti, eikä globaalisti. Vain projektia varten käytettävä vain komentoa ”npm install surge”, kehittämisympäristöön ainoastaan ”npm install -dev surge” ja sitten yleisesti, jotta voidaan käyttää tietokoneelta muissakin projekteissa ”npm install -global surge”. Sitten tarvitsee kyseinen projektin koontiversio rakentaa ja sinne laittaa projektin yleiset käytössä olevat tiedostot (kuvat yms). Tämän jälkeä siirtyä komentorivissä kyseiseen kansion ja syöttää komento ”surge”. Tämän jälkeen surge tarkistaa kansion sijainnin, kuten kuvassa (liite1) näet. Tämän jälkeen surge tarkistaa haluamasi domain nimen saatavuuden, jonka on loputtava surge.sh, mutta alun saa itse päättää. Sitten surge siirtää palvelimelle kyseiset tiedostot ja näet domain nimesi, jolla pääset katsomaan sivustoja. Domain nimi on webbiosoite sivustollesi.



# verkkosivun projektin luonti

Projektin luonti alkaa pakettien asentamisella. ”npm install -g react-360-cli”. Tämä asentaa react-360 frameworkin globaalisti kyseiselle koneelle, kyseinen paketti on hyvin isossa keskiössä nettisivujen luonnissa. Kyseisen projekti luodaan komennolla ”react-360 init PROJECT\_NAME” project\_name kohtaan lisätään halutun projektin nimi. Tämä luo yksinkertaisen VR-sivun pohjan. Tämän jälkeen aloitetaan haluttujen toimintojen luontia kyseiselle nettisivulle. Projektin pyörittämiseen käytetään kirjaston omaa paketointi tiedostoa (packaget.js), tämä käynnistetään node.js:än avulla komennolla ”npm start”, joka käynnistää projektin package.json tiedoston määrityksien mukaan. Tämän jälkeen alkaa sovellus kuuntelemaan selaimen yhdistämistä sovellukseen. Tähän käytettävä web-osoite projektissa oli ” http://localhost:8081/index.html”.Text

Description automatically generated

(Liite kuvassa projektin ”package.json” tiedosto. Komento ”npm start” käynnistää objektin ”scripts” kohdasta ”start” komennon.)

Käyttäjän ympärille luodaan sylinterimäisesti paneeleita joihin voidaan luoda erilaisia toimintoja. Tässä projektissa nämä ovat infopaneelit ja yleinen paneeli asetuksia varten ja matkustamista varten.

React-360 toimintojen luonti on yksinkertaista. Tarvitaan vain kolmea tiedostoa muokkailla, kyseisessä projektissa. Ensimmäisessä (client.js) määritellään haluttujen paneeleitten pinta-alat ja paikat, näitä määrityksiä voidaan sitten käyttää paneeleitten luonnissa. Tässä projektissa nämä ovat infopaneelit ja matkustamista varten oma paneeli. Framework myös määrittää sylinterimäisesti taustakuvaksi 360 asteen taustakuvan. Klikkaamalla ja raahaamalla saadaan myös näkymää muutettua. Toinen (index.js) määrittää, mitä halutaan kyseisille pinnoille renderöidä. Tämä tiedosto ottaa haluttuja tietoja tourData.js datatiedostosta, nämä tiedot ovat: taustakuvan, infopaneeleitten, matkustamispaneelin tiedot. Datatiedoston muodostuu objektista. Objektin avaimena on paikannimi. Objektin sisällä ensimmäisenä on paikannimi (placeName) tämä on avain string-muodossa, tähän kytkettynä on paikannimi string-muodossa (kirjoitettuna tekstiksi, jotta ohjelma tunnistaa sen olevan tekstiä, eikä esimerkiksi numeroita). Tämän jälkeen tulee ”mainInfo” avaimena ja tähän on kytkettynä pääinfo string-muodossa. Tämän jälkeen viereiset paikat ”adjacentPlaces” avaimena, tähän on kytkettynä stringejä kokoelmassa(tekstiä kokoelmassa, jotka kuvaavat viereisten paikkojen nimiä). Tämän jälkeen on ”info” avaimena, johon kytkettynä infoista objekteja kokoelmassa(objekti muodostuu kahdesta ”text”, ”img” stringistä, jotka ovat avaimia ja datana on ”text” kohdalla nimi viereisestä tilasta ja ”img” kohdalla on kuvatiedoston nimi). Tämä rakenne mahdollistaa dynaamisen tilojen käyttämisen, tämä datarakenne määrittää mihin tarvittavat asiat kohdistuvat.

Projektissa on infopaneeleita, mitä klikkaamalla saadaan lisätietoa kyseisestä asiasta. Kyseisiä paneeleita voidaan päättää ovat taustan kanssa samassa kohdassa vai ovatko niitten edessä ja myös sijaintia X-, Y- ja Z- akseleilla, elikkä kolmiulotteisesti voidaan päättää paneelin sijainnin. Tein päätöksen datatiedoston muodosta, joka määrittelee onko kyseinen infopaneeli käytössä vai ei. Ohjelma etsii onko kyseistä infopaneelia vai ei ja tämän mukaan näyttää sen käyttäjälle. Sitten on matkustamista varten oma paneeli. Tätä klikkaamalla saadaan näkyviin kyseisen paikan lähipaikat/paikat johon voidaan kyseistä paikasta matkustaa. Kyseistä lähipaikkaa klikkaamalla taustakuva vaihtuu ja uudet tiedot paikasta päivittyvät. Infopaneelit ja musiikki/taustaselostus tarvittaessa. Musiikin/taustaselostuksen saa matkustamispaneelista klikkaamalla pois halutettaessa.

# Lähteet

Vttresearch. (2020). *virtuaalinen ja lisätty todellisuus*. Retrieved from vttresearch: https://www.vttresearch.com/fi/palvelut/virtuaalinen-ja-lisatty-todellisuus

Intran opinnäytetyöohjeesta löytyvät yksityiskohtaiset ohjeet tekstiviitteiden ja lähteiden merkitsemiseen: Opinnot & valmistuminen / Opinnäytetyöohje / Opinnäytetyön kirjoittaminen / >> Tekstiviitteet ja lähdeluettelo

Lähdeluettelon otsikossa on tyyli Lähteet-otsikko ilman numerointia.Lähteiden nimet kirjoitetaan käyttäen tyyliä Lähdeluettelo1. Lähdeluettelossa riviväli on 1.

Jos käytät työssäsi liitteitä, ohje liitteiden merkitsemiseen löytyy intran opinnäytetyöohjeesta: Opinnot & valmistuminen / Opinnäytetyöohje / Opinnäytetyön kirjoittaminen / >> Raportin osat.

Laita kukin liite omalle sivulleen. Oikeaan yläkulmaan tulee teksti LIITE 1, LIITE 2 jne, tyylinä Liite-otsikko.

Liite1 (”https://portaali.fi/info/”, Portaali),

Liite 2 (”https://www.vttresearch.com/fi/palvelut/virtuaalinen-ja-lisatty-todellisuus”, Vttresearch),

Liite 3 (“https://www.panoramic-photo-guide.com/finding-the-nodal-point.html”, panoramic-photo-guide),

Liite 4 (“https://www.thetalab.ricoh/en/article/7074/” ricoh),

LIITE 1

ASETTELUMALLISSA KÄYTETTÄVÄT TYYLIT

|  |  |
| --- | --- |
| Kansiteksti | nimiölehdessä olevat tekstit  fonttikoko 14, Arial, keskitetty teksti |
| Normaali (Normal) | leipäteksti, Times New Roman tai Arial,  fonttikoko 12, riviväli 1,5, molempien reunojen tasaus, tavutus |
| Otsikko1 (Heading1) | pääotsikot  fonttikoko 14, suuraakkoset |
| Otsikko2 (Heading2) | väliotsikot  fonttikoko 12 |
| Otsikko3 (Heading3) | alaotsikot  fonttikoko 12 |
| Tiivistelmä | tiivistelmässä olevat tekstit  fonttikoko 12, riviväli 1, molempien reunojen tasaus, tavutus |
| Sisällysotsikko | sisällysluettelon otsikko SISÄLLYS,  fonttikoko 14, suuraakkoset |
| Taulukko- ja kuvaotsikot | Taulukon otsikko tulee taulukon yläpuolelle ja kuvan /kuvion /kaavion otsikko sen alapuolelle, fonttikoko 12, riviväli 1 |
| Lähdeluettelo | lähdeluettelossa mainittavat kirjallisuus- ym. lähteet, fonttikoko 12, riviväli 1 |
| Lähteet-otsikko ilman numerointia | Lähteet-sivun otsikko, LÄHTEET  fonttikoko 14, suuraakkoset |
| Liite-otsikko | Liite-sivun oikeaan yläkulmaan tuleva otsikko LIITE 1, LIITE 2 jne  fonttikoko 14, suuraakkoset, tasaus oikealta |

Perehdy opinnäytetyön ohjeeseen SAMKin www-sivulla. Opinnäytetyön ohjaajasi neuvoo työn sisällön ja rakenteen suunnittelussa. Kirjastosta saat apua lähdemateriaalin etsintään. Opinnäytetyön asetteluun liittyvistä asioista voit kysyä opinnäytetyön ohjaajalta tai tutkimusviestinnän tai –menetelmien opettajalta. Kysymyksiä ja vastauksia löytyy myös [kirjaston blogista](http://samkontkysymyksia.blogspot.com/). Menestystä opinnäytetyöhön!

POISTA TÄMÄ SIVU JA KAIKKI OHJETEKSTIT OMASTA OPINNÄYTETYÖSTÄSI.

Tallenna työsi eri nimellä kuin tämä mallipohja. Ole huolellinen työn eri versioiden tallentamisessa ja nimeämisessä. Ohjeen valmiin opinnäytetyön julkaisemiseen löydät intrasta Opinnot & valmistuminen/ Opinnäytetyöohje/ Opinnäytetyön valmistumisvaihe/ >> Jakelu ja julkaiseminen