

LOPPURAPORTTI				1 (6)
Laatija(t) Jaakko Ikäheimo, Arttu Rusanen, Konsta Holm	Tiedosto Loppuraportti_v1_0		Versio 1.00	
Hyväksynyt TOIMITUSRYHMÄ	Luotu 2020-04-23	Talletettu 2020-04-24 11.042	Tulostettu 2020-04-24	

PROJEKTI PELI/IOT (RYHMÄ 2)

LOPPURAPORTTI VERSIO 1.0

DOKUMENTIN VERSIOHISTORIA

VERSIONRO.	PÄIVÄMÄÄRÄ	MUUTOSPERUSTE	TEKIJÄ / HYVÄKSYJÄ
1.0	24.4.2020	Dokumentti valmis	Toimitusryhmä
0.2	23.4.2020	Kaikki kohdat täytetty	Toimitusryhmä
0.1	23.4.2020	Suuntaa näyttävä sisältö lisätty	Jaakko Ikäheimo
0.01		Dokumenttipohja	EeNo

LOPPURAPORTTI			2 (6)
Laatija(t) Jaakko Ikäheimo, Arttu Rusanen, Konsta Holm	Tiedosto Loppuraportti_v1_0	Versio 1.00	
Hyväksynyt TOIMITUSRYHMÄ	Luotu 2020-04-23	Talletettu 2020-04-24 11.042	
	Tulostettu 2020-04-24		

SISÄLLYSLUETTELO

1. PROJEKTIN KUVAUS	2
2. PROJEKTIN AIKATAULU JA TULOKSET	2
3. YLEISARVIO PROJEKTIN LÄPIVIENNISTÄ	3
4. KOKEMUKSET KÄYTETYISTÄ TYÖKALUISTA, LAITTEISTOISTA JA OHJELMISTOISTA	4
5. ARVIO PROJEKTIN ONNISTUMISESTA	5
6. HENKILÖKOHTAISET KOKEMUKSET JA OPPIMAT	5

1. PROJEKTIN KUVAUS

Projektin päätavoitteena oli suunnitella ja toteuttaa Nucleo-mikrokontrollerilla ja Raspberry-ytimellä järjestelmä, missä Nucleosta luettua anturidataa esitetään sopivassa formaatissa, jossain muodossa käyttäjälle. Projekti toteutettiin Oulun ammattikorkeakoulun sulautetun järjestelmän ohjelmistoprojektin kurssille, mikä toimii osana sulautetun järjestelmän sovellusprojektikonaisuutta.

Projektin luonteesta johtuen tilaajana toimi Oulun ammattikorkeakoulu. Projektin toimitusryhmänä toimivat ohjelmistokehittäjät Jaakko Ikäheimo, Arttu Rusanen ja Konsta Holm, jotka toimivat myös vuorotellen kokousten puheenjohtajana, sihteerinä ja asiantuntijana.

2. PROJEKTIN AIKATAULU JA TULOKSET

Kuten projektin alkuvaiheessa luodusta suunnitelmasta voidaan havaita, aloitettiin projektin tekeminen keväällä 2020, neljännen kouluperiodin kolmannella viikolla, ja lopetusajankohdaksi asetettiin periodin viimeinen viikko. Tämä aikataulu asetti projektin kestoksi noin 5 viikkoa, mikä aikataulutettiin taulukon 1 mukaisesti. Projekti pysyi suurimmalta osin loppuun

LOPPURAPORTTI				3 (6)
Laatija(t) Jaakko Ikäheimo, Arttu Rusanen, Konsta Holm	Tiedosto Loppuraportti_v1_0		Versio 1.00	
Hyväksynyt TOIMITUSRYHMÄ	Luotu 2020-04-23	Talletettu 2020-04-24 11.042	Tulostettu 2020-04-24	

asti aikataulussa. Loppuvaiheessa saattoi ilmetä hieman kiirettä, mutta tämä saatiin kuriin ahkeran ja aktiivisen tiimityöskentelyn avulla.

PROJEKTIN VAIHE	AIKATAULU	VAIHESEEN KÄYTETÄÄN PÄIVIÄ
Projektin suunnitteluvaihe	Viikko 13	3
Järjestelmän suunnitteluvaihe	Viikko 13	4
Moduulitoteutusvaihe	Viikko 14-15	14
Moduulitestausvaihe	Viikko 14-15	2
Integroititoteutusvaihe	Viikko 16	7
Integroititestaustaihe	Viikko 16	2
Projektin lopetusvaihe	Viikko 17	2
	YHTEENSÄ	32

Taulukko 1. Projektin aikataulus

Projektin tulokset, kehittäjien saaman kokemuksen lisäksi, saatiin saavutettua aikataulussa. Tulokset kattavat niin projektin aikana eri ympäristöille toteutetut koodit, niiden sisältämän dokumentaation, kuin myös projektin aikana palaverista laaditut muistiot sekä muut asiakirjat. Projektin päätuloksena voidaan kuitenkin pitää koko tuotettua ohjain-pelijärjestelmää mitä varten piti tuottaa ohjelmakoodit monessa eri kehitysympäristössä, ja mikä saatiin toteutettua aikataulun mukaisesti ja ilman suurempia ongelmia.

3. YLEISARVIO PROJEKTIN LÄPIVIENNISTÄ

Projektin aikainen ajallinen käyttö ja alussa laadittu aikataulu kulkivat suurimmaksi osaksi käsi kädessä ilman viivästyksiä. Läpiviennissä ei ollut uusien teknologioiden opetteluun lisäksi hirveästi vastoinkäymisiä. Projekti sujui

LOPPURAPORTTI				4 (6)
Laatija(t) Jaakko Ikäheimo, Arttu Rusanen, Konsta Holm	Tiedosto Loppuraportti_v1_0		Versio 1.00	
Hyväksynyt TOIMITUSRYHMÄ	Luotu 2020-04-23	Talletettu 2020-04-24 11.042	Tulostettu 2020-04-24	

koko kehitystiimin osalta suhteellisen sulavasti myöskään ilman konflikteja tai muita sivuärsykyitä. Läpivientiä edesauttoi huomattavasti projektin sisäisen kommunikoinnin vapaamuotoisuus ja sujuvuus, mitä suoritettiin kehittäjäryhmän viestintäkanavalla lähes päivittäin. Tämän lisäksi pidettiin viikoittain formaalimuotoisemmat viikkopalaverit, missä pyrittiin selvittämään mahdollisia ongelmia ja niiden vaatimia toimia.

4. KOKEMUKSET KÄYTETYISTÄ TYÖKALUISTA, LAITTEISTOISTA JA OHJELMISTOISTA

Vaikka kehitysryhmä olikin jo ennen projektin toteuttamista käyttänyt Nucleota ja sille tarjottua mbed-ohjelmistokehitysympäristöä, oli koodin toimivaksi toteaminen ja testaaminen laitteessa projektin hitain prosessi, sillä laitteessa suoritettu ohjelmakoodi piti aina erikseen kääntää ja siirtää laitteeseen. Kehitysympäristön vaihto webpohjaisesta työpöytäapplikaation kuitenkin nopeutti tätä prosessia huomattavasti, sillä se tarjosi kaiken tarvittavan kehitystyötä varten, kuten esimerkiksi monitorin sarjayhteyden tarkastelemiseksi.

Ohjaimen Raspilla (Raspberry Pi) suoritettavan ajuriohjelmiston kehitystyö ja siinä käytetyt työkalut sekä ympäristö oli erittäin helppo valita ja käyttää, sillä niiden suoritus onnistui suoraan kehittäjän läppäriltä Ethernet-liitännän ja VNC-apuohjelman avulla itse Raspissa. Ajurin kehittämiseen käytetyllä Python-ohjelmointikielellä oli myös erittäin helppo testata ja toteuttaa Nucleolta luetun ohjaindatan lukeminen, jatkokäsittely ja esittäminen.

Projektissa käytetyistä teknologioista positiivisimmat kokemukset tulivat suurimmaksi osin JavaScriptin opiskelun ja käytön myötä. JavaScriptissä käytettiin Phaser-kirjastoa, jonka avulla voi helposti luoda selaimessa pyöriviä

LOPPURAPORTTI				5 (6)
Laatija(t) Jaakko Ikäheimo, Arttu Rusanen, Konsta Holm	Tiedosto Loppuraportti_v1_0	Versio 1.00		
Hyväksynyt TOIMITUSRYHMÄ	Luotu 2020-04-23	Talletettu 2020-04-24 11.042		
		Tulostettu 2020-04-24		

pelejä. Phaserin perusteiden opettelu oli suhteellisen helppoa, minkä takia se sopi hyvin ensimmäisen itseohjelmoidun pelin luomiseen. Tiled-ohjelmaa käytettiin kartan suunnitteluun ja luomiseen, sillä sitä oli helppo ja hauska käyttää Phaser-yhteensopivien karttojen ja karttaobjektien luomiseksi.

NPM:mää ja parcel-bundleria käytettiin pelin kehityksen rinnalla tuloksien reaaliaikaiseen seuraamiseen sekä tarvittavien pakettien lataamiseen. Projektin lopussa otettiin myös käyttöön Heroku-pilvipalvelu sen tarjoaman Node-ympäristöisen palvelimen ja pelin saatavuuden parantamiseksi sekä sen GitHub-integraation tuomien automaattisten integrointimenetelmien, joiden todettiin helpottavan pelin kehitysprosessia.

5. ARVIO PROJEKTIN ONNISTUMISESTA

Projekti onnistui toimitusryhmän mielestä erittäin hyvin sen monimutkaisuuteen nähden. Alussa suunniteltu arkkitehtuuri ja pääominaisuudet saatiin toteutettua niin kuin pitikin. Lisäominaisuuksiakin saatiin toteutettua kuten moninpelitila ja siinä ohjaimen tarkastelua varten tarvittava katsojatila.

Projektissa olisi voinut olla vielä enemmän pelin lisäominaisuuksia, kuten vihollisia, fysiikkaa hyödyntäviä esteitä ja mahdollisuus esimerkiksi vihollisen vahingoittamiselle. Nämä kuitenkin nähtiin projektin onnistumisen kannalta välttämättömiksi, joten niitä ei toteutettu ainakaan vielä tässä vaiheessa.

6. HENKILÖKOHTAISET KOKEMUKSET JA OPPIMAT

Jaakko: Projekti toi kokonaisuutena todella hyvän näkökulman sille miten erilaiset laitteet voivat muodostaa modulaarisen mutta samalla erittäin

LOPPURAPORTTI			6 (6)
Laatija(t) Jaakko Ikäheimo, Arttu Rusanen, Konsta Holm	Tiedosto Loppuraportti_v1_0	Versio 1.00	
Hyväksynyt TOIMITUSRYHMÄ	Luotu 2020-04-23	Talletettu 2020-04-24 11.042	

toimivan järjestelmäkokonaisuuden. Omassa osuudessani jouduin pohtimaan paljon, miten saada ohjaimen käyttäjän tuottama sensoridata järkevästi välitettyä USB-liitännän kautta Nucleolta Raspberryille, esitettyä sen järkevästi käyttäjälle, ja miten saada laite kommunikoimaan suhteellisen tehokkaasti toteutetun webbipelin kanssa. Tämän lisäksi webbipelin serverin ja moninpeliominaisuuden toteuttaminen opetti paljon usean Internetiin kytketyn laitteen välisestä reaaliaikakommunikaatiosta palvelimeen muodostetun socket-yhteyden avulla. Webbipelin fysiikkamoottorin testaus ja säätäminen opetti myös, miten ihan yksinkertaisiakin fysiikan kaavoja tai kuvaajia pystytään hyödyntämään pelin käyttäjäkokemuksen ja realistisen pelifysiikan tasapainottamiseksi.

Konsta: Projekti innosti käymään ulkoisesta lähteestä oman kurssinsa HTML5 Pelin kehityksestä Phaserilla joka avasi hyvin paljon kyseisen pelimoottorin mahdollisuuksista ja kaikkea mitä HTML5 pelin kehittämiseen tarvitaan. Opin myös käyttämään eri sovelluksia kuten Tilediä pelin kartan luomiseen ja suunnitteluun sekä Shoeboxia spritesheettejen muokkaamiseen. Projektissa tuli myös nähtyä sulautettujen järjestelmien käytön oikeassa projektissa Nucleolla ja Raspberry:llä. Päällimmäisenä käteen jäi JavaScriptin luku ja kirjoittaminen sekä siihen liittyvien luokkien, asettejen ja filejen hallinta. Lisäksi web-kehittäjälle tärkeitä työkaluja tuli käytettyä kuten npm ja siihen liittyvät paketit.

Arttu: Opin projektin aikana yleisesti lukemaan ja kirjoittamaan JavaScript-koodia. Opin tekemään simppeleitä pelejä käyttäen Phaser-kirjastoa. Seurasin sivusta, miten voit tehdä ohjaimen käyttäen Nucleota ja tietokoneella Python-koodia serial-välityksellä. Opettelin myös ihan yleisesti, miten tehdään ryhmä projekteja käyttäen GitHubia.