

RALLY RACING-PELIN KEHITYSTÄ UNITY-PELIMOOTTORILLA

Ismo Männikkö
Opinnäytetyö
Kevät 2025
Tietojenkäsittelyn tutkinto-ohjelma
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Tietojenkäsittelyn tutkinto-ohjelma

Tekijä: Ismo Männikkö

Opinnäytetyön otsikko: Rally Racing-Pelin Kehitystä Unity-Pelimoottorilla

Työn ohjaaja: Minna Kamula

Työn valmistumislukukausi ja -vuosi: kevät 2025

Sivumäärä: 26

Tässä opinnäytetyössä oli tavoitteena suunnitella ja toteuttaa jo olemassa olevaan Rally Racing peliin parannuksia ja uusia ominaisuuksia käyttäen Unity pelimoottoria. Pelisuunnittelusta kerrotaan myös yleisemminkin.

Lähteinä on käytetty erinäisiä nettilähteitä, joita on etsitty käyttäen monia hakumoottoreita. Pelin toteutukseen on käytetty assetteja Unityn Asset Storesta, ja omia vuosien aikana hankittuja pelin teko taitoja.

Tuloksena on parempi versio pelistä, jossa on monia uusia ominaisuuksia ja joitain jo olemassa olleita ominaisuuksia on myös muunneltu paremmiksi. Parannettavat asiat ovat pääasiassa löydetty pelaamalla peliä ja miettimällä mikä tekisi siitä paremman. Samalla tavalla on myös keksitty suurin osa uusista ominaisuuksistakin.

Tämä opinnäytetyö keskittyy pääasiassa pelin ensimmäiseen kenttään aikataulun takia. Pelissä olisi kuitenkin vielä paljon paranneltavaa sen muissa kentissä. Peliin voisi myös lisätä täysin uusiakin kenttiä.

ABSTRACT

Oulu University of Applied Sciences
Degree Program in Business information systems

Author: Ismo Männikkö

Title of thesis: Rally Racing Game Development With Unity

Supervisor: Minna Kamula

Term and year when the thesis was submitted: Spring 2025

Number of pages: 26

This thesis is about improving an already existing game called Rally Racing using the game engine Unity. General game development is also addressed.

Various web sources have been used as source materials for this thesis. They have been found by using various search engines. The game uses assets from the Unity Asset Store. My own skills obtained over many years are also utilized.

The result is a better version of the game with more features and improvements to existing features. These features were invented by playing the game and thinking what would make it better.

This thesis focuses primarily on the first level of the map due to scheduling. The game does have other levels that could be improved too. New maps could also be added to the game.

SISÄLLYS

SANASTO	5
1 JOHDANTO	6
2 UNITY	7
2.1 Pelimoottori	7
2.2 Asset Store	7
2.3 Unity Editor	8
2.4 Versionhallinta	9
3 PELISUUNNITTELU	10
3.1 Ideointi	10
3.2 Genre	11
3.3 Pelillistäminen	11
3.4 Prototyypit	12
4 RALLY RACING-PELIN UUSIEN OMINAISUUKSIEN SUUNNITTELU	13
4.1 Alkutilanne	13
4.2 Oma ideointi	16
4.3 Asset Storen avulla ideointi	17
4.4 Muiden pelien avulla ideointi	18
4.5 Sopivuuden mietintää	18
5 RALLY RACING-PELIN UUSIEN OMINAISUUKSIEN TOTEUTUS	19
5.1 Assettien etsiminen	19
5.2 Assettien muokkaamisen	19
5.3 Skriptien tekeminen ja käyttö	20
5.4 Prefabien tekeminen ja käyttö	22
6 POHDINTA	24
LÄHTEET	25

SANASTO

3D	Kolmiulotteisuus pelimaailmassa.
3D Malli	3D Mallinnukset ovat kolmeulotteisia objekteja pelimaailmassa.
Android	Googlen kehittämä mobiili käyttöjärjestelmä, jolle tämä peli on suunniteltu.
Asset	Peliä varten käytettävä resurssi, kuten 3D-malli, kuva, tai ääni.
Collider	Collider on Unityssä oleva komponentti, jota käytetään pelissä fyysisten törmäyksien määrittämiseen.
C#	Microsoftin kehittämä ohjelmointikieli, jota käytetään paljon Unityssä.
Komponentti	Unityssä käytettävä asetteihin liitettävä asia, joka määrittelee jonkin osan sen toiminnallisuudesta. Esimerkiksi collider tai skripti.
Prefab	Valmiiksi tehty asetelma asetteja ja komponentteja, joita voi helposti käyttää monia kertoja.
Rigidbody	on Unityssä oleva komponentti, jota käytetään fysiikoiden simulointiin.
Skripti	Skripti on lista ohjeita, joilla pelilogiikkaa määritellään.

1 JOHDANTO

Tässä opinnäytetyössä käsitellään pelisuunnittelua ja jo olemassa olevien pelien parantamista eri tavoilla. Lisäksi toteutetaan jo olemassa olevan Rally Racing pelin parantamista lisäämällä siihen uusia asioita ja ominaisuuksia ja muuntamalla joitain jo olemassa olevia asioita.

Rally Racing on Super God Oy:n tekemä peli, ja he ovat tämän opinnäytetyön toimeksiantajat. Super God:in tunnetuin peli on Combat Kings, joka on hyvin erilainen peli kuin Rally Racing.

Rally Racing on 3D autolla ajelu peli, jossa ajetaan erilaisissa ympäristöissä ja yritetään kulkea mahdollisimman pitkä matkan ilman että polttoaine loppuu. Ajassa myös kerätään erilaisia poimittavia asioita, jotka nostavat pistemäärää tai antavat lisää polttoainetta.

Teoriaosassa tarkastellaan monia pelien suunnitteluun ja toteutukseen liittyviä asioita. Rally Racing on tehty Unity pelimoottorilla, joten tämä opinnäytetyö keskittyy siihen, eikä käsittele muita pelimoottoreita.

Minulle on annettu melko vapaat kädet peliä varten ideointiin ja toteutukseen, mutta tietenkin kaiken pitää sopia siihen millainen pelin on tarkoitus olla. Olen myös saanut toimeksiantajalta joitain toiveita mitä pelille pitäisi tehdä.

2 UNITY

Unity on Unity Technologies:in kehittämä pelimoottori, jolla voi tehdä sekä 3D, että 2D pelejä (Unity 2024.). Unityllä voi tehdä pelejä monille eri käyttöjärjestelmille, kuten Windows, Android, iOS, MacOS, ja Linux. Unityllä voi tehdä myös virtuaalitodellisuus ja lisätty todellisuus pelejä (Theo 2024.).

2.1 Pelimoottori

Pelimoottorilla tarkoitetaan ohjelmaa, joka hoitaa taustalla tapahtuvat asiat, kuten pelin renderöinnin ja äänen toiston. Monissa pelimoottoreissa tulee mukana myös monia muitakin ominaisuuksia, kuten pelin sisäisten tekoälyä käyttävien hahmojen ohjaus ja verkon yli monin pelaamisen pohjustavia ominaisuuksia. (Analytics Insight 2025.)

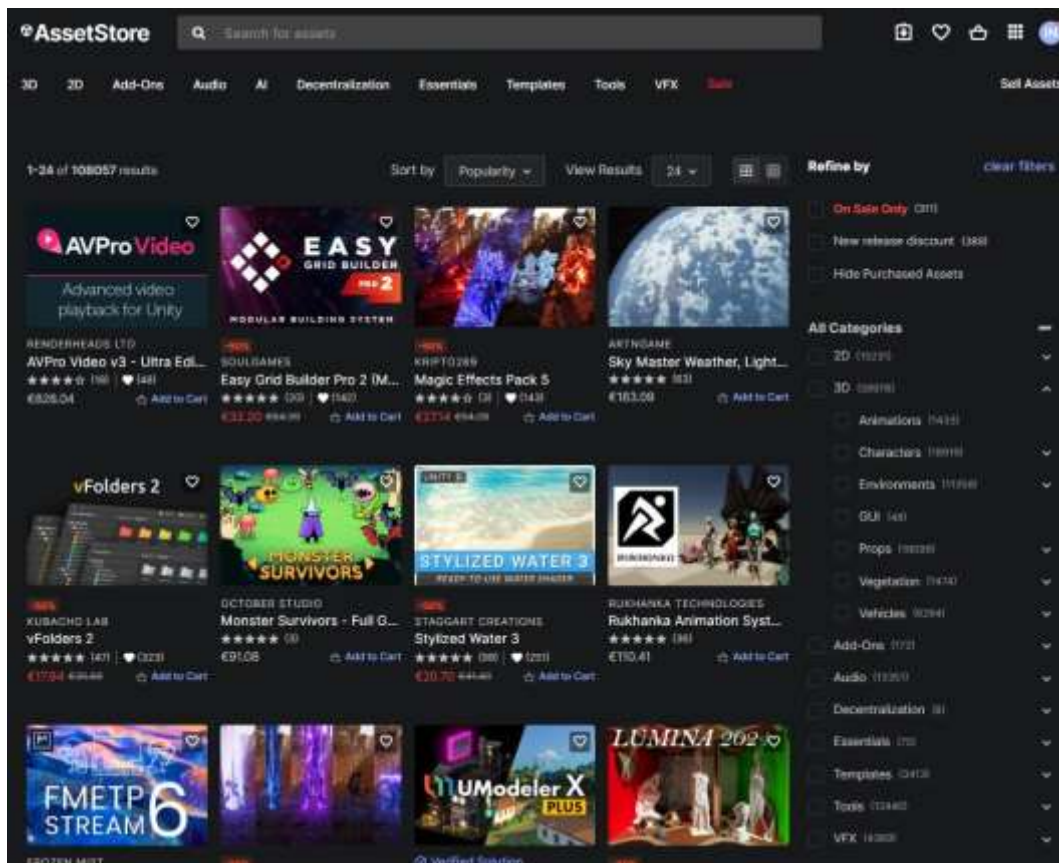
2.2 Asset Store

Asset Store on Unityn virallinen verkossa sijaitseva kauppapaikka, jossa voi myydä ja ostaa asetteja, sekä myös ladata ilmaisia asetteja. Osa aseteista on myös Unityn itsensä tekemiä (Ocean 2024.).

Asset Storen valikoima on laaja ja monipuolinen. Se sisältää kaikenlaisia asioita, joita peleissä tarvitsee olla, kuten grafiikoita, ääniefektejä, skriptejä, ja jopa valmiita monimutkaisempia pelimekaniikkoja ja myös kokonaisia pelipohjia, joiden päälle voi rakentaa itse oman pelinsä.

Asset Store on tällä hetkellä saatavilla vain selaimessa, mutta ennen se on ollut integroituna Unity Editoriin, ja sitä ennen vain erillisenä sivustona, kuten se jälleen on. (Ocean 2024.)

Asset Storen valikoimassa (kuva 1) kaikki on Unityn tarkistamia ennen julkaisua (Unity s.a. a.). Tämä tarkoittaa sitä, että assetit eivät ole aivan surkeita, eikä niiden mukana tule mitään haitallista, kuten haittaohjelmia sisältävää koodia.



Kuva 1 Unity Asset Store (Unity 11.2.2025)

2.3 Unity Editor

Unity Editor on ohjelma, jota käytetään Unity projektien tekemiseen. Kaikki Unityn sisäiset asiat tehdään Unity Editorilla, kuten asettien projektiin lisääminen ja niistä pelimaailman kokoaminen.

Asetteja itsejään ei kuitenkaan tehdä Unity Editorin sisällä, vaan ne tehdään erillisillä ohjelmilla, joista ne sitten lisäävät Unity projektiin Unity Editorin sisällä käyttöä varten. Editorissa voi kuitenkin kasata asetteja ja komponentteja yhteen, ja tehdä niistä prefabejä helpompaa uudelleenkäytettävyyttä varten (Unity s.a. c.).

Unity Editorin käyttöliittymää voi helposti muunnella käyttäjän tarpeisiin sopivamaksi (kuva 2). Käyttöliittymä koostuu paneelista, joita voi vapaasti siirrellä eri paikkoihin ikkunan sisällä, sekä irrottaa omiksi erillisiksi ikkunoikseen.



Kuva 2 Ruutukaappaus Unity Editorin päänäkymästä, jossa auki Rally Racing-pelin valikko. 11.2.2025.

2.4 Versionhallinta

Kuten melkein kaikissa ohjelmistonkehitysprojekteissa, myös pelejä tehdessä yleensä käytetään versionhallintaa. Versionhallinta mahdollistaa helpon muutosten seuraamisen ja hallinnoinnin, kuten aikaisempaan tilanteeseen palaamisen (Atlassian. s.a.).

Unityssä on oma sisäänrakennettu versionhallintajärjestelmänsä. Sen ilmaisessa versiossa on rajoituksena kolmen käyttäjän maksimi, joka voi aiheuttaa ongelmia sitä isommille ryhmille. (Unity s.a. b.)

On myös yleistä käyttää Unity-projektien versionhallintaa git-versionhallintaa, joka on myös käytössä tätä peliprojektia varten. Tätä varten on suositeltavaa lisätä tietyt kansiot projektista gitignore-tiedostoon, jotta ne jäävät pois versionhallinnasta (Farchy 27.8.2024.).

3 PELISUUNNITTELU

Pelisuunnittelu tarkoittaa pelin kaikkien osien sekä niistä koostuvan kokonaisuuden suunnittelua. Se sisältää sekä laajemman kuvan pelistä kokonaisuutena, että yksityiskohtien tarkemman mietinnän. (rajatpandey1220 2024.)

Pelisuunnittelu koostuu monista eri vaiheista (kuva 3), joista jotkin osat pitää tehdä järjestyksessä ja joitain pystyy tekemään yhtä aikaa. Joitakin osioita, kuten testausta, voidaan tehdä moneen kertaan pelin suunnittelun ja toteutuksen aikana.



Kuva 3 Pelin suunnittelu prosessi (rajapandey1220 2024.)

3.1 Ideointi

Pelisuunnittelun ensimmäisenä vaiheena on sitä varten idean keksiminen. Ilman ideaa ei peliä voi aloittaa suunnittelemaankaan. Ideoiden keksimistä varten voi

tehdä monia asioita, kuten pelata muita pelejä ja keksiä niistä ideoita omaa peliä varten (Samad 30.9.2019.). Myös oikeasta maailmasta voi keksiä ideoita.

3.2 Genre

Pelien genret ovat kategorioita, jotka ryhmittävät pelejä niiden pelimekaniikkojen perusteella. Useimmilla genreillä on myös aligenrejä, jotka tarkentavat pelien ryhmittelyä. Kaikki pelit eivät välttämättä sovi suoraan yhteenkään genreen, ja jotkin sopivat useampaan. (Dodge 2022.)

Genrejä on olemassa monia, kuten toimintapelit, pulmapelit, hiekkalaatikkopelit, kauhupelit ja rallipelit. Monilla genreillä on myös paljon aligenrejä, kuten räiskintäpelit ja selviytymispelit. Joidenkin genrejen eroavaisuudet eivät välttämättä ole kovin isoja, kuten ensimmäisen persoonan ja kolmannen persoonan räiskintäpelit, joissa ei välttämättä ole mitään muuta eroa kuin pelaajan näkemä kuvakulma. Jotkin pelit antavat pelaajan vapaasti vaihtaa ensimmäisen ja kolmannen persoonan väliltä.

3.3 Pelillistäminen

Pelillistäminen on pelimäisyyden tuomista pelien ulkopuolelle saadakseen käyttäjät olemaan aktiivisia ja motivoimaan tekemään tiettyjä haluttuja asioita. Pelillistämisenä käytetään pelimekaniikkoja ja pelisuunnittelua tehdäkseen tehtävistä mukavampia. (Barney 2023.)

Yleisimmin pelillistämisenä käytettyjä pelimekaniikkoja ovat pisteet, palkkiot ja listaus parhaimmista pelaajista järjestyksessä. Yleistä on myös edistyminen ja tasot, jotka saavat käyttäjän tuntemaan saavuttaneensa jotain, ja saa käyttäjän haluamaan jatkaa kyseisen järjestelmän käyttämistä. (Barney 2023.)

3.4 Prototyypit

Prototyyppi on hyvin yksinkertainen testiversio pelistä, jossa on vain tärkeimmät pääominaisuudet, eivätkä nekkään yleensä ole loppuun asti tehtyjä. Prototyypin tarkoituksena on testata pelin tärkeimpiä osia selvittääkseen onko peli edes hyvä idea, ja millaisia ominaisuuksia ja muuta sisältöä se tarvitsee, jotta siitä tulisi hyvä (Brazie s.a.).

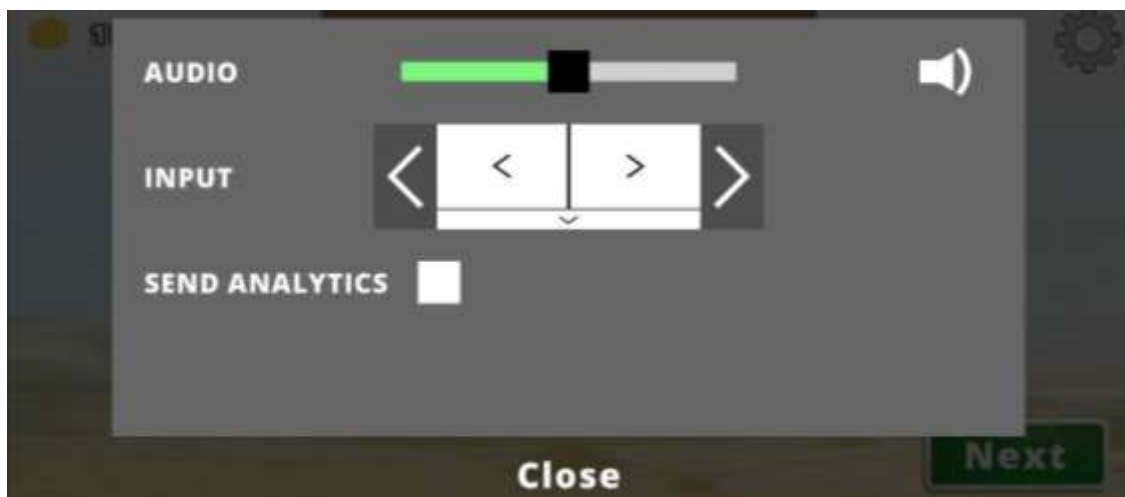
Valmiimmaksi tehtyjä prototyyppejä voi myös käyttää ulkoiseen pelitestaukseen, jossa ihmiset, jotka eivät osallistu pelin tekemiseen eikä suunnitteluun testaavat peliä ja antavat siitä palautetta. Nopean muutettavuutensa takia tällaisia prototyypien testauksia saattaa voida tehdä mahdollisesti jopa useamman kerran viikossa pelin teon alussa. (Brazie s.a.)

4 RALLY RACING-PELIN UUSIEN OMINAISUUKSIEN SUUNNITTELU

4.1 Alkutilanne

Rally Racing-peli on hyvin yksinkertaisessa tilassa, jossa siinä on olemassa pelikästään tärkeimmät ominaisuudet ja pohjustava idea minkälainen pelistä on tarkoitus tulla. Peli on käytännössä prototyypitasolla, tai vain vähän siitä pidemmällä.

Pelissä on jo valikko, joka sisältää auton valinnan ja sen päivittämisen (kuva 2) ja myös pelin asetukset. Asetuksissa on äänen voimakkuuden säätö ja auton ohjausnäppien valinta ja analyttikoiden lähettämiseksi asetukset (kuva 4).



Kuva 4 Rally Racing pelin asetukset. 28.2.2025.

Pelissä on neljä kenttää, jotka ovat kaikki melko erilaisia, mutta niissä on sama tarkoitus, eli päästä mahdollisimman pitkälle ennen kuin polttoaine loppuu. Pelissä on kaksi autoa, joiden välillä pelaaja voi valita. Toinen auto pitää ostaa pelin sisäisellä rahalla, jota kerätään kentistä. Autoja voi myös päivittää samalla rahalla, kuten parantaa sen moottoria tai renkaita.

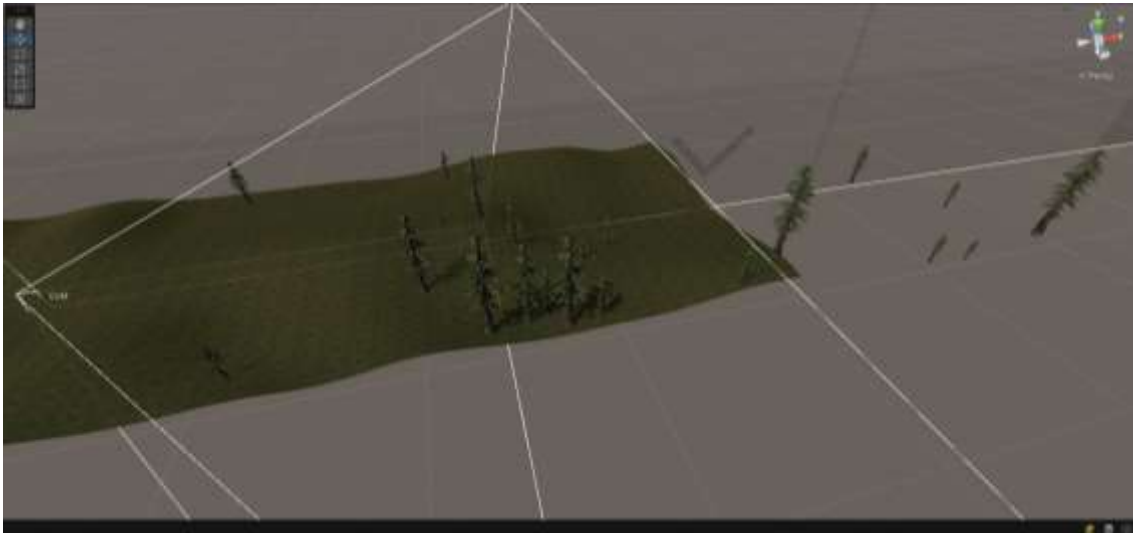
Autojen ja kenttien valikoimaa ei ole tarkoitus muuttaa tätä opinnäytetyötä varten. Opinnäytetyö on myös rajattu pääasiassa pelin ensimmäisen kentän sisältöihin.

Proceduraarinen maasto

Pelissä käytetään matemaattisesti laskettua kohinaan perustuvaa pelimaailman luontia (Zenva 2023.), johon pelitason lataamisen aikana satunnaisesti lisätään asetteja täyttämään pelitaso.

Tämän avulla jokainen pelikerta samassa tasossa on erilainen kuin yksikään aikaisempi pelikerta, pitäen kuitenkin pelikentän yleisesti samankaltaisena, esim. metsänä.

Kentän maasto luodaan vasta kun pelaaja menee riittävän lähelle sitä (kuva 5). Näin säästetään tarvittavaa laskentaa pelin pyörittämiseen, joka helpottaa pelin pyörittämistä puhelimella.



Kuva 5 Proceduraarinen kentän luonti Unity Editorin sisällä. Pelaajan auto on kuvassa vasemmalla. 28.2.2025.

Toimeksiantajan toiveet

Peliin haluttiin paljon sisällön lisäystä, pääasiassa pelin kenttiin. Tarkoituksena on lisätä peliin tekemistä pelaajalle ja tehdä siitä mielenkiintoisempi ja monipuolisempi. Myös pelin sisäisiin pelatessa aukeaviin auton parannuksiin haluttiin pieniä säädöksiä.

Tarkoituksena on pääasiassa keskittyä pelin ensimmäiseen kenttään, joka on metsä kenttä (kuva 6). Sinne haluttiin monenlaisia metsään sopivia asioita, kuten lisää puita, pensaita, kiviä ja mökkejä.



Kuva 6 Rally Racing pelin metsä kenttä ennen parannuksia. 28.2.2025.

Pelissä on myös turbonappi, jota painamalla auto liikkuu hetken ajan nopeampaa ja sen taakse ilmestyy liekki efekti (kuva 7). Tämä oli saatavilla heti pelin alussa, ja haluttiin siirtää olemaan saatavilla pelaajalle vasta sen ostettua pelin sisäisillä pisteillä.



Kuva 7 Näyttökuva pelin turbo ominaisuudesta käytössä. 28.2.2025.

Auto on myös tarkoitus tehdä hajotettavaksi, jos pelaaja kolaroi liikaa. Tätä varten auton räjähdys sen hajotessa ja on riittävä. Autolle ei siis rakenneta monimutkaista hajoamissysteemiä, joka näyttäisi auton hajoamisen sen 3D mallissa.

Pelin kentän valinta valikkoon (kuva 8) on tarkoitus lisätä tähtiä merkitsemään kuin hyvin pelaaja on kentän pelannut parhaimmillaan. Tämä pohjautuu pääasiassa siihen, kuinka pitkälle pelaaja on kentässä selvinnyt.

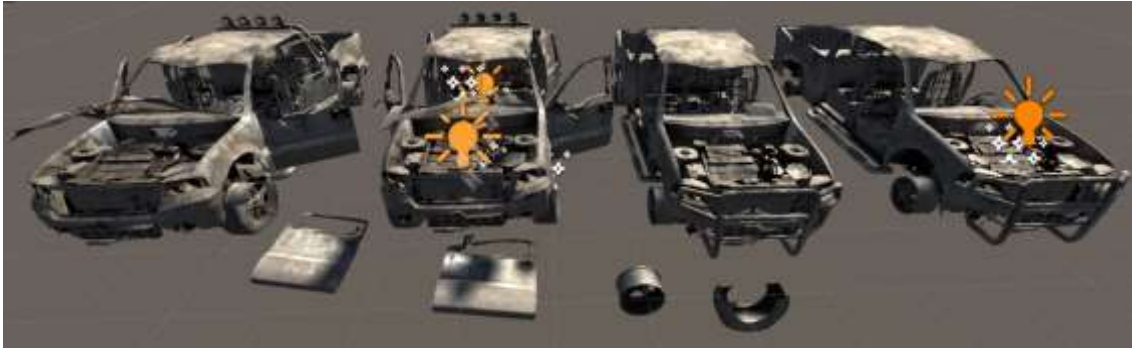


Kuva 8 Kentän valinta valikko. 28.2.2025.

4.2 Oma ideointi

Osa peliin lisättävistä ominaisuuksista on minun tai pelin omistajan keksimiä. Tätä varten pääasiassa vain mietitään mitä itse haluaisi tämänkaltaisessa pelissä tehdä tai nähdä ja mistä itse tykkäisi ja mistä ei tykkäisi. Jos projektia tehdään ryhmässä, on tietenkin kannattava kuunnella kaikkien mielipiteitä.

Koska tarkoituksena on pääasiassa keskittyä pelin ensimmäiseen kenttään, joka on metsä, on ideointia varten pääasiassa mietitty asioita, jotka sopisivat metsään ja joita olisi siellä kiva tai erikoista nähdä. Keksittyjä lisättäviä asetteja on sekä luonnollisia, kuten puita ja pensaita ja kiviä, että ihmisten tekemiä, kuten teltoja ja leirejä ja autojen romuja (kuva 9).

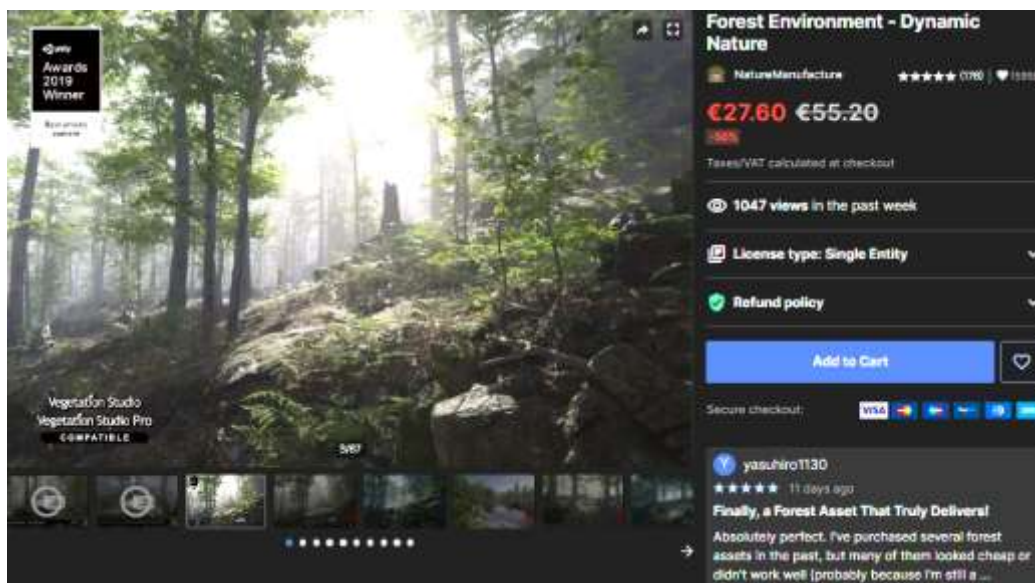


Kuva 9 Autojen romu asetteja. 31.3.2025

4.3 Asset Storen avulla ideointi

Asset Storea voi käyttää myös asettien selaamiseen, jolloin sieltä löytyviä asetteja voi käyttää uusien ominaisuuksien ideoimiseen. Samalla voi myös listata asetteja, joita voisi käyttää omassa pelissä, ja mahdollisia muutoksia, joita niihin olisi tekemässä omaa peliä varten.

Asset Storesta voi löytää myös valmiiksi tehtyjä paketteja (kuva 10) erilaisia paikkoja varten. Vaikkei sellaista kokonaista pakettia hankkisikaan, voi siitä silti valikoida, tai sen pohjalta keksiä asioita, jotka sopisivat omaan peliin.



Kuva 10 Metsä paketti Asset Storessa. Paketti sisältää monia metsään kuuluvia asioita, kuten puita ja kiviä ja oksia. (NatureManufacture 26.4.2025.)

4.4 Muiden pelien avulla ideointi

Peliä suunnitellessa voi myös olla hyvä idea katsoa mitä muut samankaltaiset pelit sisältävät, ja miettiä voisiko jotkin ominaisuudet sieltä sopia hyvin omaan peliin. Muista peleistä voi myös saada ideoita uusille ominaisuuksille, joita ei niissäkään ole.

Muita pelejä voi myös käyttää pelitestaamiseen, joka voi olla hyvä keino löytää siinä olevia ongelmia, joita sitten vältellään omaa peliä tehdessä. Ongelmia voi vältellä joko korjaamalla sen, joka niitä aiheuttaa, tai muuntamalla jotain muuta, niin ettei ongelma enää ilmene.

4.5 Sopivuuden mietintää

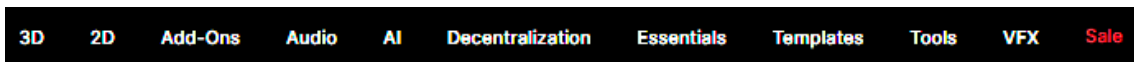
Kun ideoita on kertynyt paljon kannattaa ne käydä läpi ja rajata niistä käyttöön vain ne, jotka parhaiten sopivat peliin. Lisäksi kannattaa myös jättää pois ideat, joiden toteutus on liian vaikeaa, aikaa kuluttavaa, kallista tai vain ei toteuttamisen arvoista jostakin muusta syystä.

Ideota kannattaa myös testata pikaisilla prototyypeillä. Monet ideat kuulostavat hyviltä, mutta ovatkin toteutettuna oikeasti huonoja. Prototyyppeihin on myös helppo tehdä isoja muutoksia, joiden avulla voi ideasta ainakin yrittää tehdä paremman ja peliin sopivamman.

5 RALLY RACING-PELIN UUSIEN OMINAISUUKSIEN TOTEUTUS

5.1 Asettien etsiminen

Kun uusi ominaisuus on keksitty, sille etsitään Unity Asset Storesta sopivia asetteja, kuten 3D-malleja ja ääniä. Asset Storessa voi hakea asetteja, sekä käyttää monenlaisia rajoituksia löytämään asetteja esimerkiksi niiden tyyppin tai hinnan perusteella. Asetteja voi myös selata kategorioittain (kuva 11).



Kuva11 Asset Storen pääkategoriat. (Unity 28.4.2025)

Ideaan asetteja etsiessä kannattaa ajatella kaikkea mitä se idea tarvitsee, eikä pelkästään sen tärkeimpiä osia. Esimerkiksi tarvittavat ääniefektit ovat helppo unohtaa asioille, joille ääni ei ole keskeinen osa. Myös erilaiset visuaaliset efektit kuten auton renkaiden pyörinnästä lentävä maan pöly on helppo unohtaa.

Kun sopivat assetit on löydetty, ne tarvittaessa ostetaan, jos ne eivät ole ilmaisia. Assetit sitten lisätään omaan Unity projektiin, jossa niitä voi käyttää vapaasti. Monissa aseteissa tulee mukana jonkinlaisia esimerkkejä niiden käyttötavoista, joita voi hyödyntää tutkimalla niistä, miten asettia käytetään tarkoitetulla tavalla.

Tätä peliä varten käytetään pääasiassa asetteja Asset Storesta. Niitä etsiessä on aina ensin katsottu kaikki ilmaiset vaihtoehdot ja vasta sitten maksullisia, jos on tarvinnut. Maksullisia kuitenkin tarvittiin monia.

5.2 Asettien muokkaamien

Asetteja voi myös lisenssistä riippuen muokata itse sopimaan omaan peliin paremmin (Unity 2024). Suurin osa aseteista antaa muokkausoikeudet. Joskus niitä tarvitsee myös korjata, jos ne ovat esimerkiksi suunniteltu eri versiolle

Unitystä, tai niistä puuttuu jotain, kuten colliderit, joita ilman assetista voisi vapaasti kulkea läpi, joka ei esim. kiven kohdalla ole järkevää ollenkaan.

Koska kyseessä on puhelimille suunniteltu peli, monia asetteja joutuu hieman yksinkertaistamaan, jotta ne eivät veisi niin paljon tallennustilaa ja puhelimen suoritusnopeutta. Useista hankituista aseteista on myös poistettu joitain osia, jotta ne sopisivat paremmin tähän peliin ja veisivät vähemmän tallennustilaa.

Kaikki hankitut assetit veivät alun perin yhteensä yli kahdeksan gigatavua tallennustilaa, joka on aivan liikaa puhelinpelin varten. Niistä poistettiin osat, joita ei tätä peliä varten tarvita. Modeleiden tekstuurit, eli kuva, joka levitetään modelin ympärille antamaan sille väritystä ja kuviointeja ja muuta sellaista, olivat monissa aseteissa myös paljon isompi resoluutioisia kuin puhelinpelin varten tarvitsee, joten ne pienennettiin sopivaan kokoon.

5.3 Skriptien tekeminen ja käyttö

Skriptit kirjoitetaan erillisellä koodieditorilla. Sitä varten on monia vaihtoehtoja, joista osa on ilmaisia ja osa maksullisia. Periaatteessa koodia voisi kirjoittaa millä tahansa tekstieditorilla, mutta sitä varten suunnitellut ohjelmat helpottavat ja nopeuttavat sen kirjoittamista, sekä tekevät sen lukemisesta helpompaa esimerkiksi värjäämällä tiettyjä osia koodista, kuten muuttujien tyypit.

Unityssä käytetään pääasiassa C# koodikieltä. Unityssä on myös saatavilla sen oma visuaalinen koodaus, mutta sitä ei tätä projektia varten käytetä. Alla oleva esimerkki (koodi 1) hoitaa auton hajoamisen pelatessa. Koodin alussa määritellään tarvittavat muuttujat, kuten auton health-määrä. Funktio "OnCollisionEnter" kutsutaan joka kerta kun auto törmää johonkin, ja se laskee törmäysnopeuden ja vähentää auton health-muuttujasta sen verran. Sitten se tarkistaa menikö auto törmäyksessä hajalla, eli menikö health noltaan tai sen alle. Auton hajotessa skripti toistaa räjähdys efektin ja äänen ja aloittaa kahden sekunnin ajastimen kentän lopettamista varten. Koska nopeus on suhteellinen törmäytyyn asiaan, voi laskennan tulos olla negatiivinen, joka korjaisi autoa. Tätä varten on olemassa funktio "forceNonNegative", joka kertoo negatiiviset luvut miinus yhdellä, jotta ne muuttuvat positiivisiksi.

Koodi alkaa.

```
using System.Collections;
using UnityEngine;

public class CarDestruction : MonoBehaviour
{
    public float health = 100;
    private ResultScreen rs;
    public ParticleSystem explosion;
    public AudioSource explosionSound;

    void Start()
    {
        rs = GameObject.Find("ResultActiveScript").GetComponent<ResultScreen>();
    }

    float forceNonNegative(float input)
    {
        if (input < 0)
            return input * -1;
        else
            return input;
    }

    void OnCollisionEnter(Collision other)
    {
        Vector3 velocity = other.relativeVelocity;
        Vector3 collisionNormal = (other.transform.position - transform.position).normalized;
        float collisionSpeed = Vector3.Dot (collisionNormal, velocity);
        health -= forceNonNegative(collisionSpeed);

        if (health <= 0)
        {
            explosion.Play();
            explosionSound.Play();
            StartCoroutine(endTimer());
        }
    }

    IEnumerator endTimer()
    {
        yield return new WaitForSecondsRealtime(2);
        rs.EnableResultScreen("Car Crashed!");
        yield return null;
    }
}
```

Koodi päättyy.

Koodi 1 Rally Racing peliä varten itse kirjoittamani "CarDestruction.cs" skriptitiedosto.

5.4 Prefabien tekeminen ja käyttö

Unityssä prefabit ovat hyvä keino tallentaa tietty asetelma asetteja ja niiden ominaisuuksia helppoa uudelleenkäytettävyyttä varten. Prefabien teko tapahtuu ensin tekemällä haluttu asetelma, ja sitten vetämällä se hiirellä listasta kentän sisäisiä asioita projektin asettien listaan.

Tässä pelissä käytetään prefabeja kaikkiin pelin sisäisiin objekteihin. Prefabeja laitetaan kenttään satunnaisesti kentän latauksen yhteydessä käyttäen sitä varten kirjoitettua skriptiä, joka antaa säätää asioita kuten niiden tiheyttä kentässä ja kuinka kauas kentässä pitää päästä, jotta niitä alkaa ilmestymään.

Prefabit voivat olla joko yksittäisiä asioita, kuten puu, tai laajempia asetelmia, kuten leiri (kuva 12). Prefabeja voi myös käyttää toisissa prefabeissa. Prefabeista voi myös tehdä eri versioita, kuten eri värisiä tai pienillä muutoksilla, kuten ovi, joka on yhdessä versiossa auki ja toisessa kiinni ja kolmannessa hajalla.



Kuva 12 Aseteista kasattu leiri prefabi. 31.3.2025

Monet tätä peliä varten hankitut assetit sisälsivät jo valmiiksi prefabit käyttöä varten. Osaa näistä täytyi kuitenkin muuttaa, ja muutaman joutui tekemään itse täysin koska se oli liian erilainen siihen, mitä tähän peliin tarvitsi. Osassa oli ominaisuuksia, jotka vievät puhelimelta liikaa tehoja, kuten liian monimutkaisia

liekkiefektejä. Nämä ominaisuudet joko poistettiin tai korvattiin vähemmän tehoja vaativalla versiolla, riippuen kuinka tärkeä se oli tätä peliä varten.

Pelin metsä kentässä on käytetty prefabejä kaikkia siellä olevia asioita varten. Ne tekevät kentästä paljon paremman näköisen (kuva 13) ja lisäävät sinne paljon lisää tekemistä, kuten rampeilta hyppimistä (kuva 14) ja löydettäviä asioita kuten leirejä ja mökkejä.



Kuva 13 Metsä kenttä parannuksien jälkeen



Kuva 14 Ramppi metsässä

6 POHDINTA

Opinnäytetyön päätarkoituksena oli parantaa jo olemassa olevaa Rally Racing peliä. Peliin suunniteltiin ja toteutettiin uusia ominaisuuksia ja parannuksia. Osa näistä oli toimeksiantajan toiveita, ja loput minun itse keksimiäni käyttäen päätekstissä mainittuja ideoiden keksintä keinoja.

Tuloksena on parempi version pelistä, jossa on enemmän tekemistä pelaajalle. Tulokset vastaavat toimeksiantajan työnantoa, ja keskittyy niihin osiin pelistä mihin oli tarkoituskin.

Pelissä olevia muita kenttiä voisi myös parantaa, mutta tätä opinnäytetyötä varten on keskitytty vain ensimmäiseen kenttään, kuten toimeksiantaja toivoi. Muuten projektissa olisi kulunut aivan liian kauan opinnäytetyötä varten.

Tämä projekti oli hyvin sopiva pelien suunnittelun ja toteutuksen ammattialalle. Siinä tarvittiin pelin testauksen, suunnittelun, ja toteutuksen taitoja. Mielestäni tämä peli voisi hyötyä paljon jostakin isosta uudesta pelimekaniikasta, mutta semmoinen ei ollut osana toimeksiantoa, joten sitä ei tätä opinnäytetyötä varten ole toteutettu. Ensimmäinen mieleeni tuleva ehdotus sellaisesta olisi useamman pelaajan yhtäaikainen pelaaminen, jossa voi kolaroida toisten pelaajien autoihin häiriköidäkseen heitä.

Tätä projektia tehdessä opin paremmin selvittämään kuinka jonkun muun tekemä systeemi toimii, ja lisäämään siihen uusia ominaisuuksia ja muuntamaan siinä jo olleita ominaisuuksia. Varsinkin systeemit, joiden osia on ympäriinsä projektia vaativat paljon aikaa etsiä missä kaikki sen palaset ovat ja kuinka ne toimivat yhdessä. Niitä muokatessa tarvitsee monesti muokata monia osia, eikä vain yhtä.

LÄHTEET

Analytics Insight. 2025. Game Engines Explained: How Unreal, Unity, and Others Build Worlds. Manjeera Trinity Corporate. Luettavissa: <https://www.analyticsinsight.net/games/game-engines-explained-how-unreal-unity-and-others-build-worlds>. Luettu: 12.2.2025

Atlassian. s.a. What is version control? Atlassian Pty Ltd. Luettavissa: <https://www.atlassian.com/git/tutorials/what-is-version-control>. Luettu: 12.2.2025

Barney, N. 2023. gamification. TechTarget Inc. Luettavissa: <https://www.techtarget.com/searchHRSoftware/definition/gamification>. Luettu: 11.2.2025.

Brazie, A. s.a. Video Game Prototype: A Beginner's Guide. Funsmiths LLC. Luettavissa: <https://gamedesignskills.com/game-design/video-game-prototype/>. Luettu: 13.2.2025

Dodge, D. 2022. The Definitive Guide to Video Game Genres and Game Types. CodaKid Inc. Luettavissa: <https://codakid.com/video-game-genres/>. Luettu: 11.2.2025.

Farchy, A. 27.8.2024. Git and Unity: A Comprehensive Guide to Version Control for Game Devs. Virtual Maker Corporation. Luettavissa: <https://www.virtualmaker.dev/blog/git-and-unity-a-comprehensive-guide-to-version-control-for-game-devs>. Luettu: 12.2.2025.

NatureManufacture. 26.4.2025. Asset Store Forest Environment - Dynamic Nature ruutukaappaus. Unity Technologies. <https://assetstore.unity.com/packages/3d/vegetation/forest-environment-dynamic-nature-150668#content>. Luettu: 28.4.2025.

Ocean. 2024. What is the Unity Asset Store and how do I purchase Assets?. Unity Technologies. Luettavissa: <https://support.unity.com/hc/en-us/articles/210142503-What-is-the-Unity-Asset-Store-and-how-do-I-purchase-Assets>. Luettu: 12.2.2025

rajatpandey1220. 2024. Game Design. Sanchhaya Education Pvt. Ltd. Luettavissa: <https://www.geeksforgeeks.org/game-design/>. Luettu: 11.2.2025.

Samad K. 30.9.2019. How to come up with great game idea?. GENertainment Studios. Luettavissa: <https://blogs.geniteam.com/game-idea/>. Luettu: 31.3.2025

Theo. 2024. What platforms are supported by Unity?. Unity Technologies. Luettavissa: <https://support.unity.com/hc/en-us/articles/206336795-What-platforms-are-supported-by-Unity>. Luettu: 12.2.2025

Unity. 11.2.2025. Asset Store verkkosivun ruutukaappaus. Unity Technologies. <https://assetstore.unity.com/?orderBy=1>. Luettu: 11.2.2025.

Unity. 28.4.2025. Asset Store verkkosivun kategoria listan ruutukaappaus. Unity Technologies. <https://assetstore.unity.com>. Luettu 28.4.2025.

Unity. 2024. END-USER's Rights and Obligations. Unity Technologies. Luettavissa: <https://unity.com/legal/as-terms>. Luettu: 14.4.2025.

Unity s.a. a. Start publishing on the Asset Store. Unity Technologies. Luettavissa: <https://assetstore.unity.com/publishing/publish-and-sell-assets>. Luettu: 11.2.2025

Unity s.a. b. Unity Version Control. Unity Technologies. Luettavissa: <https://unity.com/solutions/version-control>. Luettu: 12.2.2025

Unity s.a. c. Prefabs. Unity Technologies. Luettavissa: <https://docs.unity3d.com/6000.0/Documentation/Manual/Prefabs.html>. Luettu: 18.2.2025

Unity. 2024. Start learning Unity. Unity Technologies. Luettavissa: <https://learn.unity.com/tutorial/start-learning-unity>. Luettu: 12.2.2025

Zenva. 2023. What is Procedural Generation – Complete Guide. Zenva Pty Ltd. Luettavissa: <https://gamedevacademy.org/what-is-procedural-generation/>. Luettu: 10.3.2025.