Haaga-Helia
University of Applied Sciences

Ohjelmistokehityksen teknologioita - *S*eminaarityö

<Snake Game>

<7 käyttöliittymät ja pelit>

Alex Kiippa

14.4.2022

Sisältö

Tiivistelmä

Tässä ohjemointiprojektissa olen toteuttanut yksinkertaisen käärmepelin C++, sekä Python-kielellä. Projektin tavoitteena oli harjoitella kummankin kielen käyttöä pelinkehityksessä. C++-versiossa käytin perinteisiä konsolipohjaisia kirjastoja. Ajatuksena oli luoda myös graafisesti miellyttävä pelikokemus muiden kirjastoiden avulla, kuten Raylib, GLFW ja SFML, mutta kirjastojen implikointi koodiin kävi hyvin mutkikkaaksi jostain syystä. Tämä johti siihen, että käänsin koodin Python-kielellä ja hyödynsin pygame-kirjastoa joka antaa mukavan pelaamiskokemuksen.

Projektin edetessä C++-versiossa rakentui pelilogiikka, käyttöliittymä ja vaikeustason asettaminen. Python-versiossa keskityin graafisen käyttöliittymän suunnitteluun ja toteutukseen, pelin logiikan päivittämiseen ja ruudun päivitykseen.

Tärkeimmät tulokset ovat toimivat käärmeenpeli-prototyypit kummassakin ohjelmointikielessä. C++-projekti tarjoaa mahdollisuuden harjoitella perusohjelmointitaitoja, kun taas Python-projekti näyttää, miten graafisen käyttöliittymän ja kuvien käyttö voi tehdä pelin visuaalisesti houkuttelevammaksi.

1 Johdanto/Ylätason esittely

Tämä ohjelmointiprojekti lähti liikkeelle halusta luoda yksinkertainen, mutta visuaalinen käärmepeli, joka tarjoaa peliviihdettä samalla kun harjoitellaan ohjelmointitaitoja. Aluksi toteutin projektin C++-kielellä perinteisellä konsolipohjaisella lähestymistavalla, ja myöhemmin päätin uudelleenimplementoida sen Pythonissa käyttäen pygame-kirjastoa. Projektin tavoitteena oli tarjota tapaa harjoitella ja soveltaa ohjelmointitaitojaan pelinkehityksessä.

2 Käytetyt tekniikat

2.1 Kirjastot

Projektissa käytetyt tekniikat vaihtelivat C++-version konsolipohjaisista kirjastoista, kuten "iostream" ja "windows.h", Python-version graafiseen käyttöliittymään pygame-

kirjaston avulla. Käytin myös kuvia animoidun käärmeen ja hedelmän esittämiseen, mikä toi visuaalista eloa peliin.

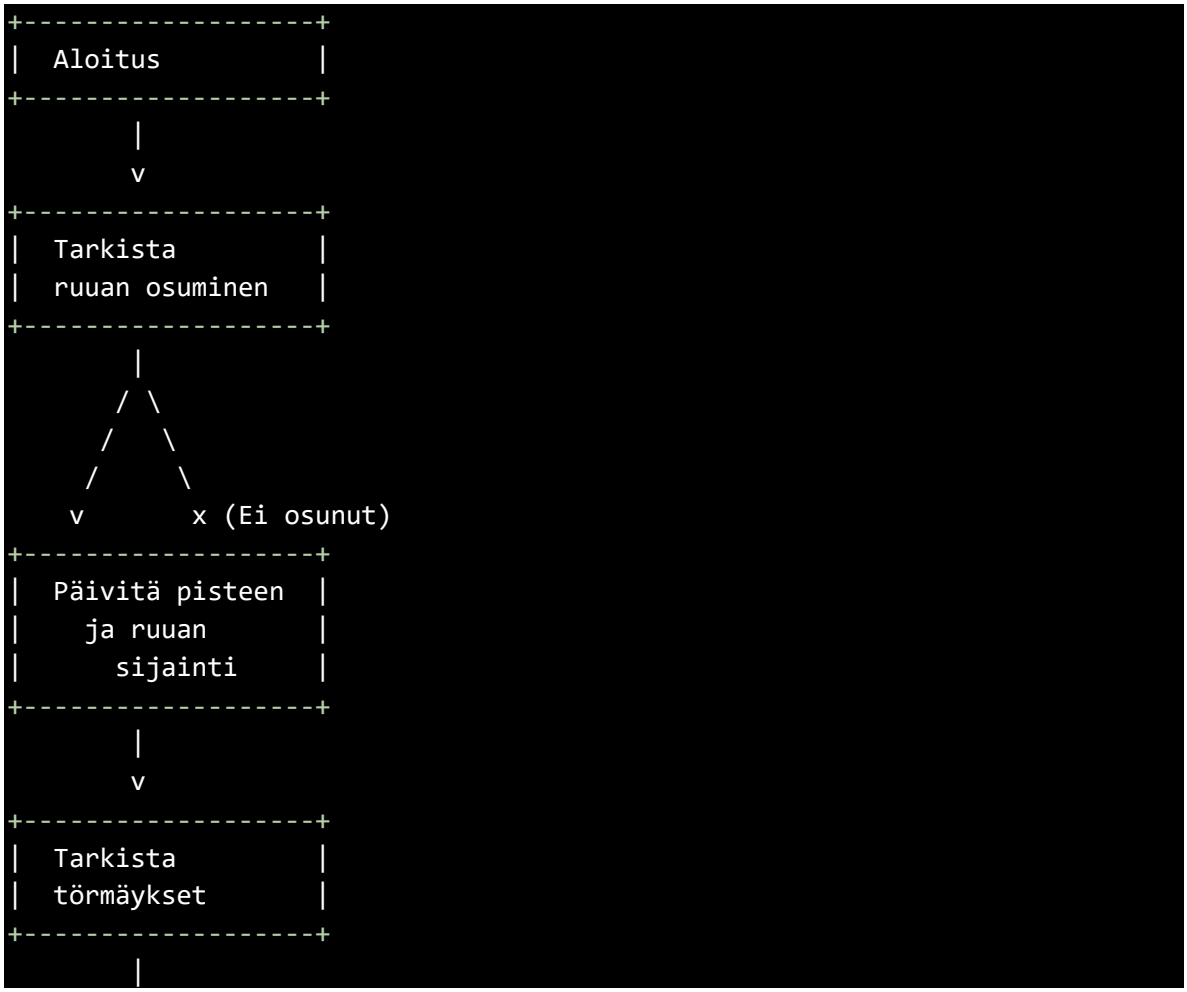
2.2 Automatisoidun testauksen työkalut

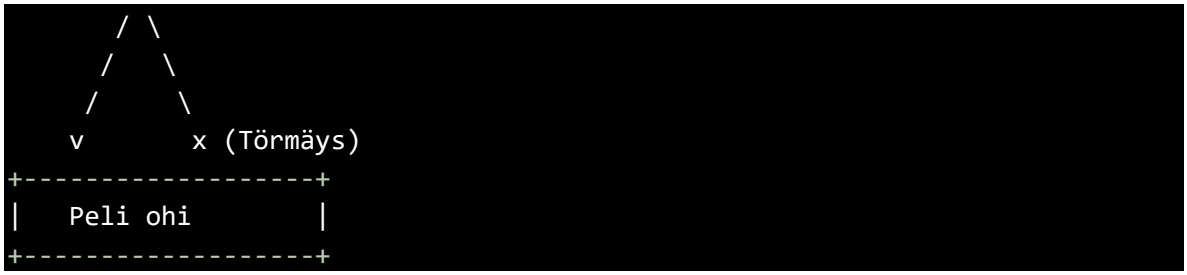
Projekti ei itsessään vaatinut laajaa automatisoitua testausta, koska pelin yksinkertainen rakenne mahdollisti tehokkaan manuaalisen testauksen. Käytetyt testaustavat ovat olleet pääasiassa käyttöliittymäkeskeisiä ja toiminnallisuuden perusteella tehtyjä. Jatkokehitystä miettien, jos peli kehittyy, niin automatisoidut testaukset voisivat olla harkinnassa.

2.3 ChatGPT

Projektissa hyödynsin monipuolisesti myös ChatGPT:tä. Käytin chattibottia uudelleenkirjoittamaan muuttujien nimiä, kommentoimaan koodia paremmin ja parantamaan koodin yleistä laatua, joka auttoi selkeyttämään koodin rakennetta ja ymmärryttävyyttä.

3 Arkkitehtuurikaavio / Algoritmin vuokaavio





4 Yhteenveto

4.1 Yhteenveto

Tässä ohjelmointiprojektissa olen onnistuneesti päässyt toteuttamaan yksinkertaisen käärmepelin sekä C++- että Python-ohjelmointikielellä. Käytetyt kirjastot, kuten perinteiset konsolipohjaiset C++-kirjastot ja graafinen pygame-kirjasto Pythonissa, osottavat monipuolisuuden ohjelmoinnissa.

4.2 Tulokset

Projektin tuloksena on kaksi toimivaa käärmepeliä. Molemmat versiot tarjoavat pelattavan prototyypin ja havainnollistavat eri kielten vahvuuksia pelinkehityksessä.

4.3 Johtopäätökset

Vaikka projekti onnistui saavuttamaan toimivan pelin, avautuu useita ideoita jatkokehitykselle. Yksi suunta voisi olla pelin ominaisuuksien laajentaminen, kuten äänitehosteet, pistetaulu ja erilaisia pelattavia ympäristöjä.

4.4 Arviointia

Arvioin, että projektin tavoitteet ovat saavutettu onnistuneesti. Pelit toimivat odotetusti, ja projektin myötä olen syventänyt osaamistani ohjelmoinnissa ja pelinkehityksessä. Manuaalisen ja automatisoidun testauksen roolin pohtiminen on avannut näkökulmia laadunvarmistukseen ja antanut eväitä tuleviin projekteihin.

4.5 Pohdinta

Oppimiskokemuksena projekti oli loistava. Uuden ohjelmointikielen opettelu uudessa aiheessa toi oman haasteensa ja aikaisempi kokemus Arduinon ohjelmoinnista C++-kielellä oli luonut pienen käsityksen syntaxista. Projekti oli hauska oppimismahdollisuus ja koen, että pelikehittämiseen on syntynyt suurempi into.

Kiitos!

Lähdeluettelo