

TDP005 Projekt: Objektorienterat system

Kodgranskning

Författare

Ismail Safwat, saysa289@student.liu.se

1 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.0	Granskning av anmälargruppens kod	2022-12-08

2 Möte

Efter att jag fick reda på att vilken grups kod jag ska granska och vice versa kontaktade jag beställargruppen för att ordna en möte samt bad dem att skicka inbjudan till deras GitLab repo. Sedan gick både jag och beställargruppen varandras kod och antecknade feedback och förbättringsförslag som diskuterades i mötet. Vi träffades klockan 09:10 den 8:e december via Teams.

3 Granskat projekt

Spelet jag granskade gick ut på att spelaren kunde skjuta fiender och klara banan. Spelplanen visas i fågelperspektiv och spelaren kan röra sig över hela banan samt skjuta åt alla håll. Spelaren rör sig genom en serie rum (banor) som har olika layout och upplägg av hinder och eventuellt olika starka fiender.

3.1 Kommentarer

Det fanns inte tillräckligt kommentarer i klasser och variabler. Även om det finns lite kommentarer är det inte användbart för doxygen.

3.2 Sprite

Det ska finnas en gemensam basklass (Sprite) för alla figurer i programmet. Övriga sprite-figurer ska ärva från denna klass som det saknades i deras kod.

3.3 Main-klassen

Main-funktionen ska vara liten, huvuddelen av spelet ska finnas i de klasser ni designar.

3.4 Positiv feedback

3.4.1 CMake

CMake filen var välskriven och med kommentarer.

3.4.2 Variabelnamn

Beställargruppen hade valt vettiga namn för deras variabler som i sig förtydligar vad den innehåller.

4 Vårt projekt

4.1 Game-klassen

- Game klassen gör för mycket, det gör koden svår att förstå och är inte optimalt (klassupdelning).
- Många updates i Game, dessa skulle kunna vara färre om antingen game delas upp i fler klasser eller om man använder override mer.
- Skulle kunna finnas fler pure virtual funktioner i Game för att tilldela mer ansvar till underklasser.
- UpdateUniverse funktionen skulle kunna vara i en egen universe klass som är ovanför game klassen.
- Vad gör updateGUI?

Jag har försökt hålla allt som sker från rendering till avslutning inuti Game-klassen samt jag har lagt kommentarer för varje funktion då jag tyckte att det känns vettigt på det sättet. Pure virtual funktioner är relaterade till föregående problem så det kommer åtgärdas också.

updateGUI är tänkt för att skriva ut hälsopoäng för spelaren.

4.2 Kommentarer

- Lägg till fler kommentarer i koden så att det är tydligare vad som görs, lägg till kommentarer på variabler också.

Jag trodde att jag hade lagt tillräckligt kommentarer men ska lägga kommentarer på variabler med.

4.3 Textures

- Kan kanske göra en klass eller funktion för att ladda in texturer för att inte upprepa kod.

Eftersom det var två bilder jag använt så använde jag sf::Texture direkt men ska försöka åtgärda det här med.

4.4 Sprite

- ladda in bilder, filsökvägen är fel
- Vi har också haft problem med att ladda in bilder, använd : "../Textures/shapeshooter.png"

Detta kommer åtgärdas.

4.5 Formatering

- Cmake kan ha mer mellanrum och kommentarer

Detta ska åtgärdas direkt.

4.6 Positiv feedback

- Inga minnesläckor
- Bra kommentarer för funktioner i enemy