

TDP005 Projekt: Objektorienterat system

Reflektionsdokument

Författare

Ismail Safwat, saysa289@student.liu.se

1 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.0	Skapade reflektionsdokumentet	2022-12-09

2 Vad har jag fått ut av kursen, vad har jag lärt mig?

Att utveckla ett tvådimensionellt spel i C++ är en spännande och givande upplevelse. Genom processen att utveckla ett spel lär man sig en mängd olika koncept som relaterar till programmering, matematik och speldesign.

I den här sammanfattningen kommer att de huvudämnena som jag lärt mig under spelutvecklings-processen diskuteras. Det är också viktigt att ta hänsyn till den övergripande spelstrukturen. Detta inkluderar att bestämma vilken typ av spel som ska utvecklas, spelets mekanik och spelets design.

Det är viktigt att överväga vilken typ av spel som kommer att vara roligast och mest engagerande för spelarna. Detta inkluderar att överväga vilken typ av spelar interaktioner, målen för spelet och hur spelet ska se ut. Dessutom bör spelmekaniken vara väl genomtänkt och planerad för att spelet ska vara underhållande. Detta inkluderar att överväga spelets regler och handlingar, typerna av karaktärer och föremål och det övergripande spelet.

Slutligen bör spelet utformas på ett estetiskt tilltalande sätt. Detta inkluderar färgerna, grafiken och spelets övergripande layout. När spelstrukturen har bestämts måste programmeringstekniker användas för att utveckla spelet.

2.1 Utmaningar

Att utveckla ett spel i C++ kan vara en utmanande uppgift, eftersom det kräver en djup förståelse av språket och datorns kapacitet. Den här sammanfattningen kommer att fokusera på de svårigheter jag stötte på och hur jag övervann dem.

2.1.1 Svårigheter som uppstått

Den största svårigheten jag stötte på var själva språket. C++ är ett kraftfullt och mångsidigt språk, men det kan vara svårt att lära sig och förstå. Det är viktigt att ha en god förståelse för språket innan du försöker utveckla ett spel med det, eftersom detta kommer att göra utvecklingsprocessen mycket smidigare.

En annan svårighet jag stötte på var spelets komplexitet. Ett spel kan vara ganska komplext, eftersom det kräver många olika element för att implementeras. Detta inkluderar bland annat grafik, kollisionsdetektering och så vidare. Alla dessa element måste implementeras korrekt för att spelet ska vara spelbart.

2.1.2 Hur jag övervann dem

För att övervinna svårigheterna med att lära mig och förstå C++ studerade jag språket mycket. Jag sökte hjälp av mina klasskamrater, tittade på tutorials om språket och tränade helt enkelt på att skriva kod för att få en bättre förståelse av språket.

För att övervinna komplexiteten i spelet bröt jag upp spelet i mindre delar och arbetade med varje del en i taget. Detta gjorde att jag kunde fokusera på ett specifikt element i spelet åt gången, vilket gjorde det lättare att förstå och implementera.

Återigen att utveckla ett spel i C++ kan vara en utmanande uppgift, och det finns många svårigheter som måste övervinnas. Språket i sig kan vara svårt att lära sig och förstå, och själva spelet kan vara ganska

komplext. Men genom att studera språket och dela upp spelet i mindre delar kunde jag övervinna dessa svårigheter i något viss mån och utveckla ett funktionellt tvådimensionellt spel i C++.

2.2 SFML

Att lära mig att utveckla ett spel i C++ med SFML var en givande och lärorik upplevelse mig. De har en ganska lättläst och bra dokumentation och tutorials i deras webbsida som var till stor hjälp för mig.

2.3 Kravspecifikation

Att skriva kravspecifikationen i början av kursen bidrog mycket då man har en bra bild på vad man kommer implementera även om det inte blir exakt som man sätter sitt mål i dokumentet.

2.4 Designspecifikation

Precis som kravspecifikationen är det mycket viktigt att lägga tid för att skriva en vettig designspecifikation då är det ännu viktigare jämfört med kravspecifikation. Det spelade stor roll för min del då jag brukade titta på designspecifikation då och då medan jag skrev koden. Som nämnts tidigare kom en hel del ändringar från första versionen på designspecifikationen.

2.4.1 Doxygen och dokumentation

Jag har tagit hänsyn på kommentarer i min kod och det gynnade i slutet av arbetet. Då behövde jag bara generera dokumentationen på ett smidigt sätt för att få en relativt omfattande dokumentation av alla mina kod och kommentarer.

2.5 Granskning av beställargruppens kod

Ett annat moment som var väldigt viktig och jag fick lära mig nya saker. Efter mötet med beställargruppen fick jag ändra några saker i koden.