Break A Brick

Individuellt Mjukvaruutvecklingsprojekt

# Abstrakt

Det här är en rapport på mitt spel som jag har gjort under dessa 10 veckor. Det är ett spel i 2d av typen breakout och arkanoid. Spelet är skrivet i C# med hjälp av ramverket XNA 4.0. Då XNA var helt nytt för mig måste jag säga att det ändå varit rätt lätt att sätta sig in i och jag har fått lite mer kunskap om spelutveckling och programmering än jag hade innan.

I rapporten kommer jag ta upp det som har gått bra och det som har gått mindre bra i mitt projekt.

# Bakgrund

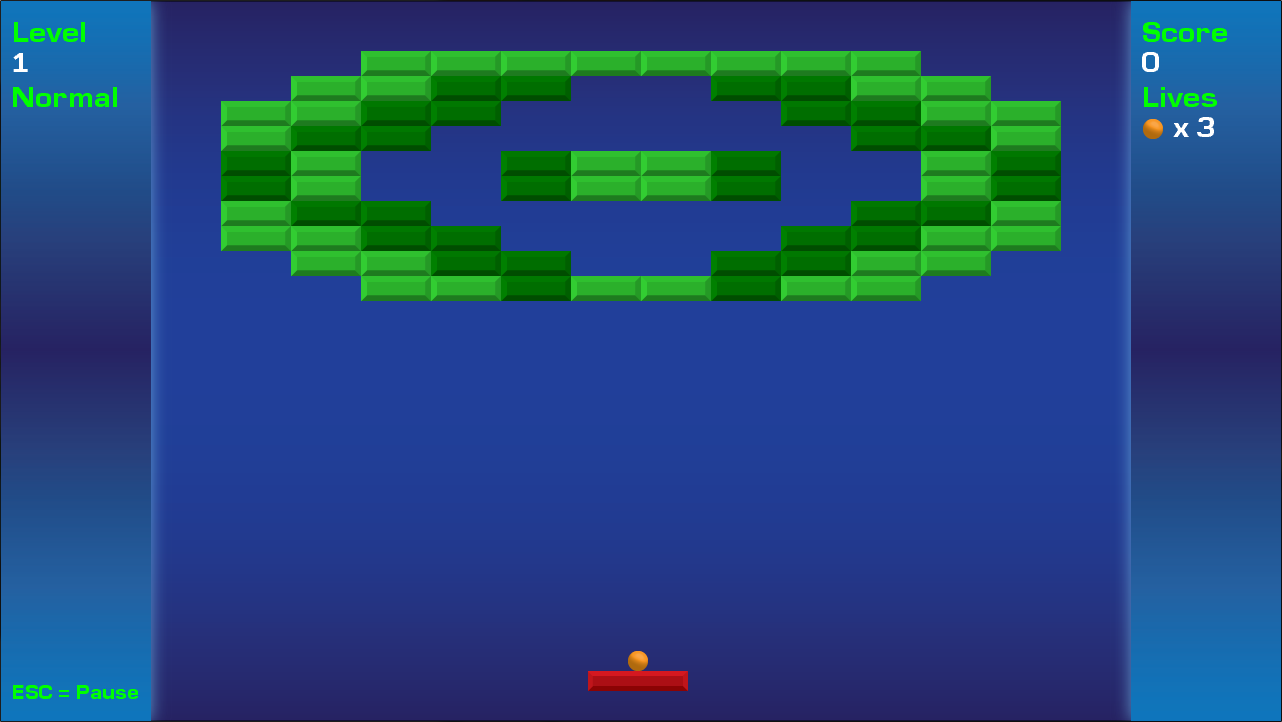
Då jag har ett intresse för spel så hade jag tänkt att jag ville göra ett spel som projekt. Sen tog det stopp och jag hade ingen aning om vad för spel det skulle bli. Eftersom det skulle bli mitt första spel så visste jag att jag behövde komma på något simpelt.

Till att börja med så läste jag igenom ett par gamla rapporter om andra projekt i samma kurs. Då fick jag syn på några som hade gjort spel i ramverket XNA 4.0, som faktiskt är gjort för spelutveckling. Det passade ju bra för språket man skriver i är C# (c-sharp) och det är det språket som jag har tyckt mest om under det här läsåret. När det kom till spelval så hade jag sett många säga att när de stöter på ett nytt språk som spelprogrammerare gör de ett breakoutspel som övning. Då både spelutveckling och ramverket XNA 4.0 var nytt för mig så tyckte jag det passade bra som mitt projekt.

Att bara göra själva grundspelet räckte dock inte för mig utan jag ville göra ett så komplett spel jag kunde. Därför har jag ett menysystem, inställningar, ljud (som gör stor skillnad), flera nivåer, poäng och powerups. Nästan allt i spelet är musstyrt förutom att kunna pausa spelet som man gör med escape-knappen.

Den grafik som finns i spelet är gjord av mig i Adobe Illustrator och kodningen gjordes i Visual Studio 2012.

Det har varit roligt att arbeta på ett SCRUM-liknande sätt med projektet. Alltså att man har ett antal krav man ska uppfylla i applikationen under hela kursen, sen för varje vecka så ska man försöka bryta ut uppgifter att göra utifrån de kraven (komma på nya också), skatta hur långt tid man tror det tar för att i slutet av veckan ha en levererad produkt som man bygger på vecka för vecka. Man arbetar alltså iterativt tills produkten är klar och får en struktur på arbetsättet.



# Positiva Erfarenheter

När jag började med projektet så ville jag få in ett system som hanterade olika lägen i spelet, en så kallad state manager. Sättet jag har gjort det på kallas polymorfism. Jag gjorde en basklass kallad skärm som alltid ritas ut för användaren. När jag vill visa en annan skärm så skapar jag en till klass som ärver ifrån skärm och har egna metoder. Basklassen är alltså din skärm och klasserna är det som ska visas på skärmen. Ett annat ställe jag använde polymorfism är med mina powerups. Jag har en basklass powerup och alla olika powerups ärver ifrån den för att få grundstrukturen och sen gör varje powerup det som det ska göra. Jag har alltså lärt mig mer om polymorfism som är ganska nyttigt i spel eftersom det ofta finns liknande saker (fiender) i spel men som har olika beteenden. Allmänt har jag fått lite mer kött på benen när det gäller c# (c-sharp).

I början av projektet blev det mycket inlärning om XNA genom att läsa mycket och göra tutorials. Jag började med att faktiskt läsa en bok där man fick göra ett spel under tiden man läste. Själva spelet var inte så komplicerat men man fick lära sig grunderna i XNA för tillverkning av 2d spel vilket var det jag behövde. Jag fick också mycket inlärt från många sidor och youtube-filmer med tutorials. Det är viktigt att faktiskt få in hur man gör saker istället för att bara kolla upp hur jag kan göra det här för att sedan inte veta vad man gjorde. Kan man grunderna så kan man bygga på och puzzla ihop de olika delarna för att få det man är ute efter.

Något som var roligt var när jag skulle börja med ljudet i min applikation. Från början hade jag inte ens tänkt på ljud eftersom jag inte jobbat med ljud inom programmering förut. De flesta spel har faktiskt ljud (om inte alla) så det blev ju självklart att man skulle ha med ljud. Som tur är gör XNA det lätt för en att lägga till ljud i sitt spel. Med XNA följer det med ett program XACT som just hanterar ljud. Man helt enkelt gör ett projekt, lägger in de ljud man vill ha, ändra volym, ändra pitch, om den ska loopa, olika variationer på ljudet, sparar och lägger in den i projektet. Sen i koden så skriver man bara vilket ljud som ska spelas genom så kallade cues, ett ljud är en cue och varje cue har ett namn som man ger när man skapar ljud-projektet. Så det var roligt att leta ljud och försöka hitta något som passar de olika funktionerna i spelet. Man ska absolut inte underskatta ljud när det kommer till spel för ett minsta lilla ljud när man skjuter eller trycker någonstans gör det så mycket bättre, att man får den där feedbacken när man gör något.

# Negativa Erfarenheter

Det har varit lite saker som ändå har gått lite mindre bra och till att börja med har testningen inte varit den bästa och är nog det jag är minst nöjd med. Testning är något som jag inte haft mycket erfarenhet av när det kommer till mjukvaruutveckling och det lilla jag gjort var i kursen iterativ mjukvaruutveckling där vi gjorde några simpla white- och black-box tester. Tyvärr gav det inte mig tillräckligt mycket för att ge mig på enhetstester i mitt projekt och det blev i princip bara black-box tester, alltså att man har en input och man har ett förväntat output och sedan dokumenterar man om man fick rätt output eller inte. Som resultat går mina test inte särskilt djupt och jag är därför inte säker på vilka olika buggar som kan finnas. Till nästa projekt borde jag studera mer om hur man gör bra tester för att få en allmänt bättre applikation.

Jag har också en tendens att skjuta upp saker och det har också påverkat mitt projekt lite. Istället för att arbeta direkt på mina iterationer (de uppgifter som ska göras varje vecka) i början på veckan så tar jag ledigt för att sedan börja senare. Det leder till att jag ibland inte hinner med det jag skulle vilket jag egentligen borde gjort om jag inte tog ledigt i början på veckan. Detta är något som jag har haft problem med länge fast som sakta har blivit lite bättre med åren. Trots det har jag faktiskt lyckats att få ihop mitt spel med de krav jag hade och samtidigt komma upp i den tidsbudget vi hade (nästan).

Min kod, själv tycker jag att till en början var min kodstruktur helt okej men ju längre jag kom i projektet ju sämre blev min kod. Saker som kanske borde varit en metod eller klass ligger på flera ställen i koden. Mycket med dåliga namn på variabler och metoder som gjorde att till och med jag som skrev koden hade svårt att hitta runt. Dåligt kommenterat. Saker som jag själv tycker borde varit i en klass som jag bara kunde skapa ett nytt objekt av hade jag redan gått för långt med och det skulle ta för mycket tid att strukturera om och behöva sätta sig in i igen. Mot slutet så satte jag mig ner och kollade igenom koden för att lägga till eventuella kommentarer, ta bort onödig kod, och försöka allmänt förbättra den. Detta tycker jag nu är väldigt viktigt för att jag vill kunna skriva sån kod som är lätta att hantera och som någon annan programmerare skulle kunna läsa och förstår utan för mycket svårigheter.

# Sammanfattning

Allt som allt så är jag nöjd med slutresultatet av mitt projekt, jag hann med det jag skulle göra och kunde naturligtvis har utvecklat vissa krav för att göra ett ännu fylligare spel. Jag har lärt mig mycket nytt och fått testa på att utveckla mitt första spel, vilket var väldigt roligt faktiskt. Jag har lärt mig att ljud är viktigt när det kommer till spel som jag håller med om för så fort jag lade till ljud så blev mitt spel genast roligare.

Själva ramverket kan man säga håller på att dö ut då det inte längre utvecklas eller stöds på senare platformar. Men det ger mig möjlighet att prova på andra sätt att göra spel som jag tror är något jag vill göra. Jag tror inte jag kommer fortsätta på det här spelet utan det blir nog ett nytt spel i så fall.