

TDP005 Projekt: Objektorienterat system

Designspecifikation

Författare

Ismail Safwat, saysa289@student.liu.se

1 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.0	Designspecifikation skapad	2022-11-24

2 Detaljbeskrivning

2.1 Player

Player (spelaren) ärvs av Sprite-klassen. Player-klassen har två funktioner: `move()` och `shoot()`. `move()` gör så att spelarobjektet kan röra på sig vänster (←) och höger (→) genom piltangenterna. Sprite-subklassen har en funktion som heter `wall()`, vilket hindrar objekt från att gå bortom spelplanen. `shoot()` tillåter spelaren att skjuta ett skott, vilket är en klass i sig.

I Sprite-subklassen finns en viktig funktion som är relevant för Player: `collision()`. `collision()` kommer att kolla om spelaren och en fiende har kolliderat, och om den returnerar `true` kommer `GameOverState` att kallas.

2.2 Enemy

Enemy ärvs av Sprite-subklassen och har två klassbarn. Enemy-klassen har två funktioner: `move()`, `hit()` och `killed()`. `move()` gör så att fiendeklassen rör sig genom banan i realtid i ett mönster. Mönstret är att den alltid går från vänster till höger eller vice versa tills den nuddar sidan av skärmen (dvs. `wall()` anropas av Sprite-klassen), vilket gör att den går ner ett steg och byter riktning och fortsätter så tills den nått spelaren. `hit()` är en funktion som håller koll hur många gånger en enemy har blivit träffad av ett Shoot-objekt. `killed()` kollar på `hit()` och returnerar en `bool` som säger om fienden ska dö eller inte beroende på vilket värde `int life` fiendet har. Värdet för `life` beror om klassbarnet är `Enemy1` eller `Enemy2`. Om fiendet är `Enemy1` är `life = 1` medan om fiendet är `Enemy2` är `life = 2`. Om `killed()` returnerar `true` kommer sprite-subklassens funktion `remove()` att kallas och förstöra Enemy-objektet.

3 Beskrivning av design

Designen bygger på två spelstadier: `GameState`, `GameVictoryState` och `GameOverState`. `GameState` körs så fort användaren öppnar spelet. Om det inte finns några Enemy-objekt kvar i spelet ska `GameVictoryState` ta över. Om spelaren förlorar (dvs. `collision()` kallas i Sprite-subklassen) kommer `GameOverState` att ta över och visa en Game Over screen.

Under `GameState` har vi spelvärlden, vilket består av sprites som representeras av subklassen `Sprite`. `Sprite` har tre klassbarn: `Player`, `Shoot` och `Enemy`. Alla dessa klassbarn ärver av `Sprite`. `Enemy` har i sin tur två klassbarn: `Enemy1` och `Enemy2`.

Fördelen med denna design är att den är mycket enkel och inte komplicerad. En nackdel är förstås att på grund av dess enkelhet finns det mycket som kan utarbetas. Vi skulle kunna lägga till en startmeny och en pausmeny, vilket skulle innebära flera spelstadier.

4 Klassdiagram

