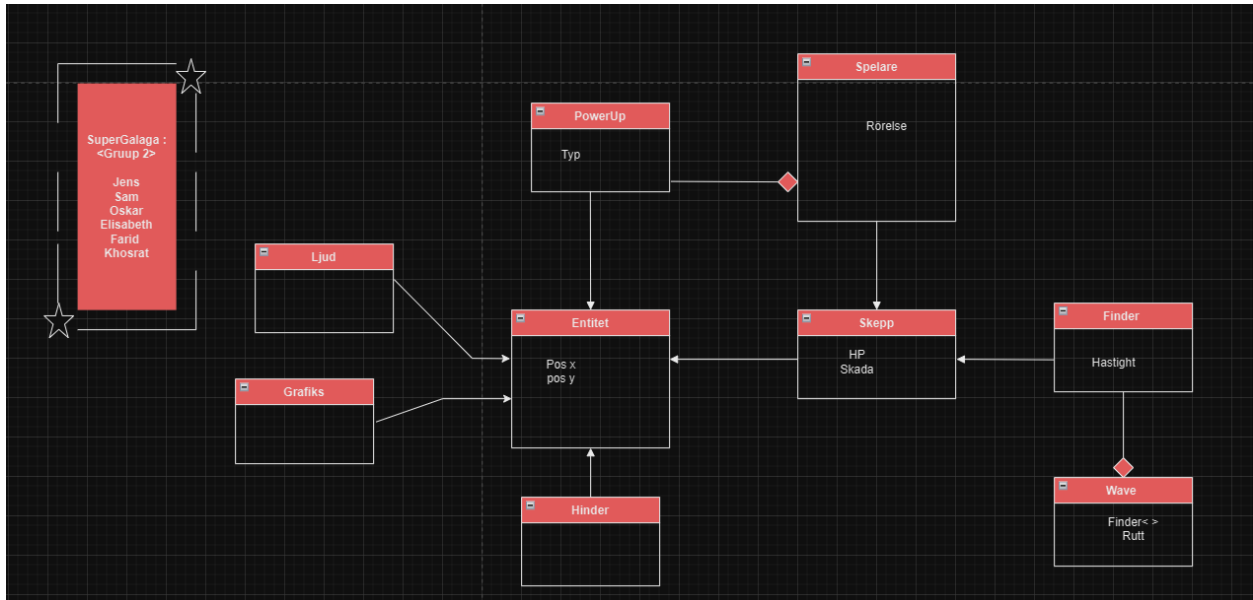


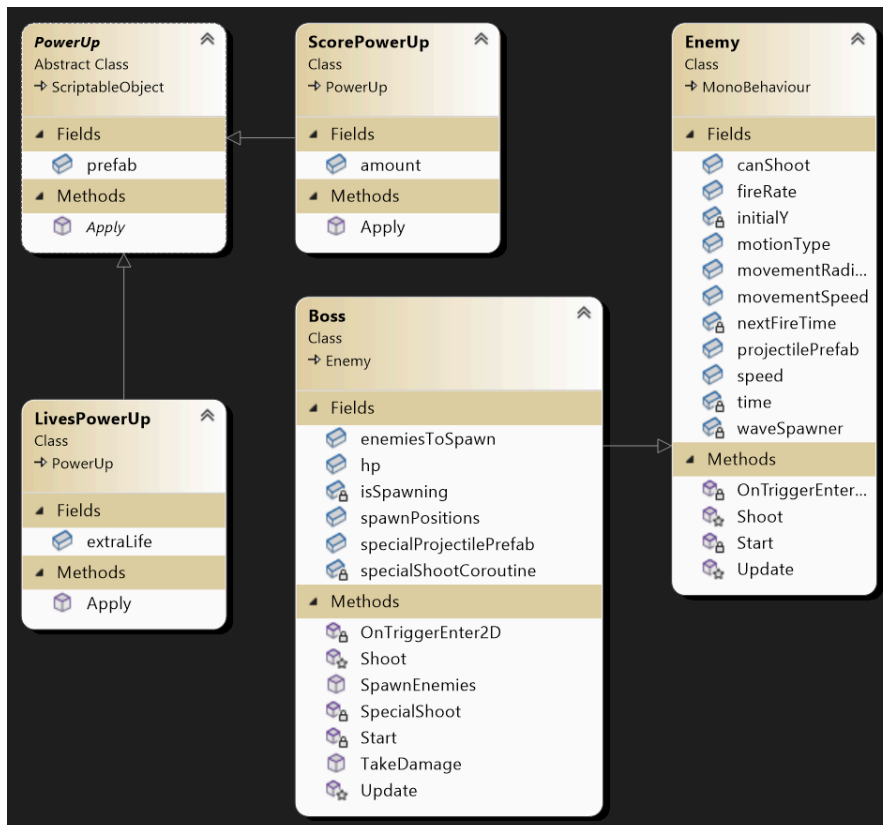
Vårt spel ska bli en horisontell shmup i klassisk stil. Spelarens skepp är till vänster, och fiender flyger in från höger i vågor med olika formation och rörelse. En del fiender skjuter projektiler, andra inte. I slutet av varje bana möter man en boss. På vägen går det att plocka upp powerups som ger extraliv eller poäng. Vid spelets slut får spelaren skriva in sitt namn och spara sin poäng i highscore-listan. Vi väljer att inte försöka följa Galagas speldesign slaviskt, men behåller att spelaren är låst i x-led till skärmens vänstra kant och endast kan röra sig i y-led.

En central fråga är hur man bäst skapar vågor av fiender som i en traditionell shmup, snarare än att de flyger in en och en hipp som happ. Fiendeobjekt ska läggas in i vågor, och en våg ska kunna tilldelas olika rörelsemönster, formationer, och beteenden.

	Krav	Prio	Klart
1	Projektet skall bestå av ett 1-spelarspel med grafiskt gränssnitt.	1	
2	Gränssnittet skall styras med tangentbord och/eller mus.	1	
3	Spelet skall ha en "meny", med några alternativ (start, avslut osv.).	1	
4	Spelet skall åtminstone ha två "banor".	1	
5	Minst två sorters fiender.	1	
6	Minst två powerups.	1	
7	Spara och ladda highscores från fil.	1	
8	Ljudeffekter/Musik	3	
9	Scrollande bakgrund med parallax.	2	
10	Boss-fiende.	3	



Ovan var hur vi tänkte oss spelets grundstruktur. Vi hörde från andra grupper att Winforms var olämpligt för scrollande spel, så vi bestämde oss att pröva Unity. Eftersom Unity var nytt för oss alla fick vi börja med att bekanta oss med det, och då blev det tydligt att en “vanlig” klasstruktur med arv och liknande inte var optimalt för ett spel av vår typ.



De flesta klasser i Unity ärver från en klass kallad MonoBehaviour, som redan där ger en mycket bred funktionalitet. På objekten kan man koppla på olika bitar kod och resurser som grafik och ljud. Till exempel. En fiende kan ha ett script som beskriver dess beteende. Vill man göra en fiende som har ytterligare ett beteende så lägger man på ytterligare ett script. Det blir liknande men inte samma. Vi har ändå några exempel där vi fått användning av vanligt arv, se till vänster:

Bifogas även diagrammet i sin helhet.