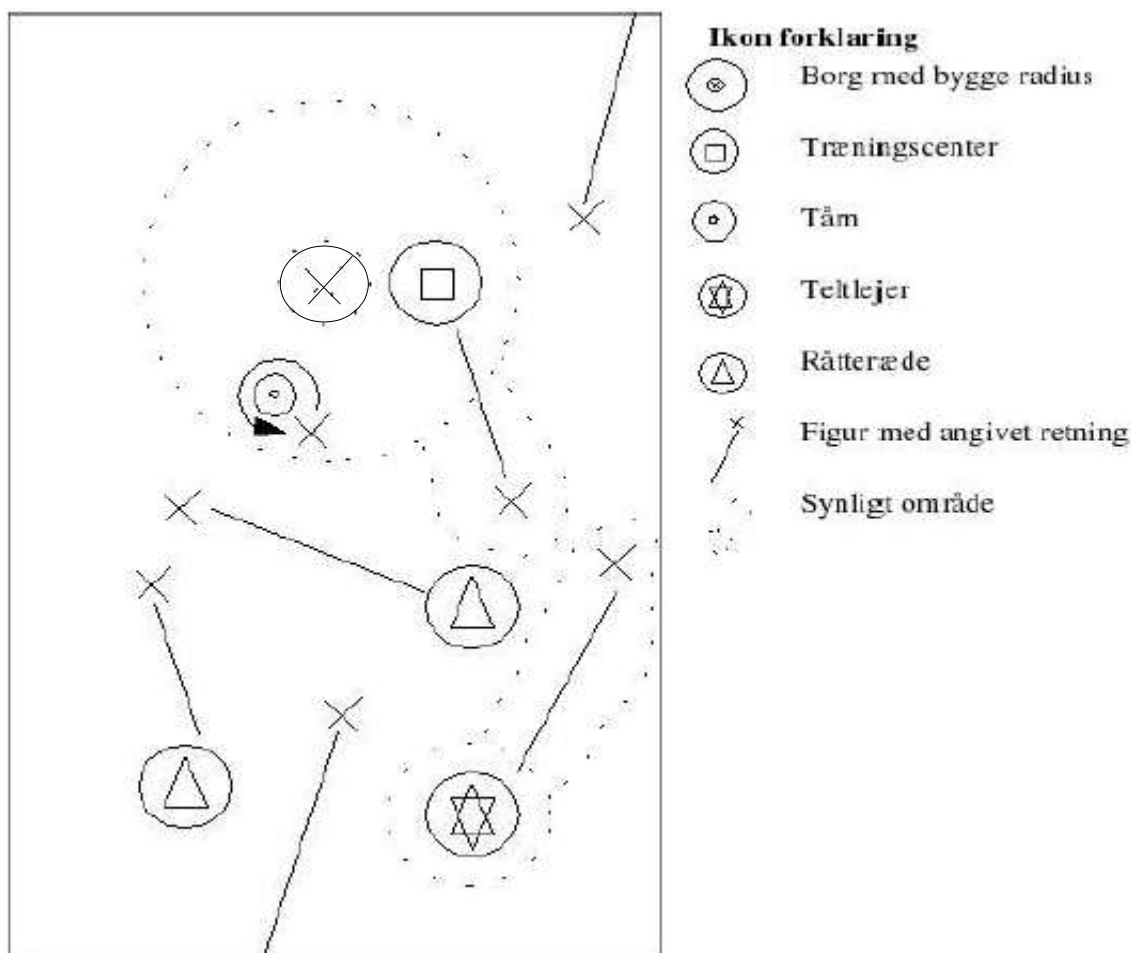


Rouler

Målet er at udvikle et spil baseret på modellen fra strategispillet Majesty, hvor man som spiller råder over én borg, hvorfra der indenfor en given radius kan bygges forskellige bygninger. Hver bygning kan udklække forskellige figurer. Figurerne bevæger sig uafhængigt af spilleren i en given retning med en fastsat hastighed. Denne retnings vektor ændres hvis figuren møder en konkret forhindring eks. en bygning eller hvis der inden for en radius optræder en fjende. Der findes dog en måde at påvirke figureres bevægelse, man kan udsætte missions flag med en af spilleren angivet værdi. Disse flag/positioner vil påvirke figurerne til at bevæge sig i disses retning hvis værdien er stor nok. En høj værdi giver en stor tiltrækningsradius.



Skitsen ovenfor beskriver en/et plade/ spillerkort med bygninger og figurer. Følgende figurer optræder i spillet:

- Orker, indkommer fra spillets side og bevæger sig i en tilfældig retning
- Råtter, ud går fra råttereder i en tilfældig retning.
- Rider, ud går fra træningscenter i en tilfældig retning.
- Spejdere, ud går fra teltlejer i en tilfældig retning.
- Vagter, ud går fra et tårn og bevæger sig i en cirkel omkring dette.

Teltlejligheder kan placeres vilkårligt i et synligt område. Tårne og træningscentre kan kun bygges inden for bygge radius af borgen.

Gameplay

Der kan spilles på to måder enten online, spiller mod spiller, eller singleplayer hvor modstanden styres af computeren.

Udgangspunktet for et spil er en spilleplade hvor kun en egens borg og den del af kortet som ligger inde for bygge radius er synlig. Spillet udvikler sig ved at man som spiller bygger nye bygninger som kan producere figurer. De synlige felt udvider sig når man bygger bygninger og når figurerne bevæger sig, når en figur bevæger sig synlig gøres en radius rundt om denne. Når en figur møder en fjende stopper begge deres bevægelser og de begynder at slå. Spillet er slut når modstanderens borg er destrueret. Bygninger og figurer har en pris, og kunsten er at balancere udgifterne mellem disse således at der kan opbygges et fornuftigt forsvar og en effektiv angrebsstyrke. Hver spiller har som udgangspunkt en given portion penge, som beskrevet bruges til at købe huse og figurer. Måden at få nye penge er opkrævning af skatter fra de enkelte huse/bygninger. Summen af penge i et hus er afhængigt af antallet af figurer knyttet til dette.

Følgende mønstre ønskes brugt i systemet

- Strategy, hvor hook metoderne er bevægelses mønstre
- MVC, for at separerer det grafiske fra det funktionelle
- Singleton, der skal kun køre en spilmaskine . Denne styrer tråde og dermed hvornår figurerne skal bevæge sig.
- Factory, til oprettelse af figurer og bygninger.

Følgende emner ønskes medtaget i rapporten:

- Tråde i Klient/Server model således at flere kan spille samtidig. Samt mulighed for at figurer kan bevæge sig individuelt. Enten dette eller der skal laves en visitor til gennemløb af de enkelte figurer. Sidstnævnte er måske det smarteste da det vil kræve færre opdateringer af spillepladen, end hvis den skulle opdateres hver gang en figur har flyttet sig.
- Datastrømme, highscore gemmes fil Følgende Factories skal oprettes:

Komponenter i systemet

Spilmasking

Komponenter:

Opbyg-bygninger

Tidsstyret fjende generering

Figurstyring

Bevægelse

Kamp

Client/Server

Kommunikationsprotokol, eller RMI

GUI

Opbygning af spilplade
Spillepladen
Synligt lag
figurbygnings lag
Geo kort lag

Figurer
Bygninger

FactoryFactory

Actor/ figurfactories

Enemy/ fjendefactories laver:

Orkfactory

Ratfactory

(Defence/ attack)/(forsvar/ angreb) factories laver:

Knight/ ridder

Scout/ spejder

Guard/ vagt

Building/ bygningsfactories

Enemy/fjendefactory laver:

Den/ råtte rede

(Defence/ attack)/(forsvar/ angreb) factories laver:

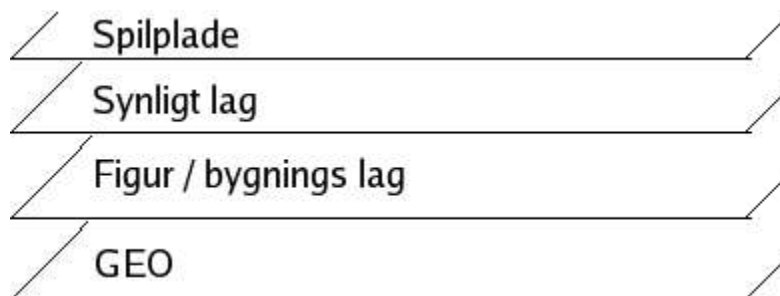
Tentcamp/teltlejer

Guild/træningscenter

Tover/tårn

Som det fremgår af ovenstående tekst er jeg endnu ikke helt kommet til en konklusion i henhold til om figurer skal styres af spilmaskinen eller om de skal virke selvstændigt, og have denne funktionalitet implementeret i et selvstændigt komponent. Ifølge den objektorienterede tankegang burde det nok laves i et separat komponent og lade spilmaskinen importere dette.

I GUI listen ovenfor opstiller jeg nogle forskellige lag i spillepladen, som tilsammen skal danne spillepladen. Spillepladen, det synlige lag og Geo kortet(tegning af skov, bjerge og andre geografiske egenskaber) er alle fysiske lag(eks Bitmap, jpeg, PNG ...), hvor figur/ bygningslaget er virtuelt (et array, en liste eller noget tilsvarende). Det synlige lag består kun af to farver synlig og usynlig. Det synlige laves af de figurer og bygninger som er oprettet i systemet. Når den samlede spilleplade generes vises kun de af fjendens figurer og bygninger som ligger inden for spillerens eget synlige felt.



Hvad angår implementeringen af Klient/Server modellen vælger jeg formentligt RMI da det vil spare den tid det vil tage at "opfinde" en protokol mellem spillerne.