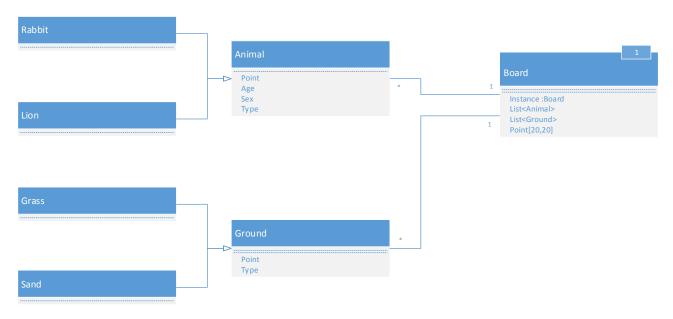
## SavannaGame

## UseCase FURPS

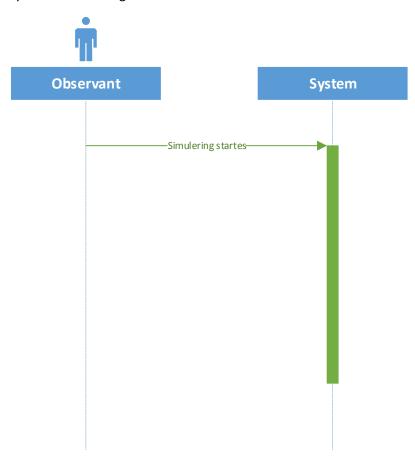
// Functionality		
Use Case Name	Start Simulation	
Scope	Funktionsstadie, ingen grafisk brugerflade.	
Level	User goal or sub function	
Primary Actor	Observant	
Stakeholders and interests	None	
Preconditions	What must be true on start and worth telling the reader	
Success guarantee	En simulering af mindst 50 runder	
Main success scenario	<ol> <li>Observant starter simuleringen</li> <li>Simulering kører I minimum 50 runder</li> </ol>	
Variations list	Varying I/O methods and data formats	
Frequency of occurrence	Influences investigation, testing and timing of implementation	
Miscellaneous	Such as open issues	
// Useability Se Scope // Reliability		
// Performance		
Et minimum af nedbrud ;)		
// Supportability		
+		
Dette er en Consol applikation		

// Fully dressed	
Use Case Name	Start Simulering
Scope	Funktionsstadie, ingen grafisk brugerflade.
Level	User goal.
Primary Actor	Observant
Stakeholders and Interests	Klassens elever: ønsker at se et funktionelt program der kører uden bugs Klassens Lærer: ønsker at se elevens (mig) lærte viden inden for fagene programering og systemudvikling Udvikleren (mig): ønsker at vise et funktionelt program der lever op til de krav stillet
Preconditions	- ingen
Postconditions	<ul> <li>Simulering kørt i minimum 200 runder</li> <li>Kaniner og løver har flyttet sig på spilpladen</li> <li>Kaniner og løver har formeret sig i simuleringen</li> <li>Kaniner og løver har spist i simuleringen</li> <li>Kaniner og løver er døde i løbet af simuleringen</li> </ul>
Success Guarantee	- Ingen fejl meldinger
Main Success Scenerio	<ol> <li>Observant starter simuleringen.</li> <li>Simuleringen kører</li> <li>Dyr flytter rundt på spilpladen</li> <li>Kanin møder en kanin af modsatte køn og formere sig, der tilføres 4 nye kaniner til simuleringen</li> <li>Løve møder en løve af modsatte køn og formere sig, der tilføres 2 nye løver til simuleringen</li> <li>Løve møder en kanin, kaninen bliver spist, løven tager på i vægt og forlænger derved sin levetid, kaninen slettes fra simuleringen</li> <li>Kanin møder græs, græsset bliver spist, kaninen tager på i vægt og forlænger derved sin levetid, græsset slettes fra simuleringen</li> </ol>
Basic Flow	Simuleringen kører automatisk og der er derfor ingen interaktion med observanten
Variations list	Simuleringenkøressomenconsolapplikation
Frequency of Occurrence	Influences investigation, testing and timing of implementation
Miscellaneous	<ul> <li>Simuleringen stopper pgafejlmeddelser</li> <li>Simuleringen bliver meget langsom ved mange dyr</li> </ul>

## Domain Model



# System Sekvens Diagram



#### **GRASP**

Arkitekturen er lag-opdelt med en decideret Controller klasse, så der på sigt let ville kunne indføres en grafisk grænseflade.

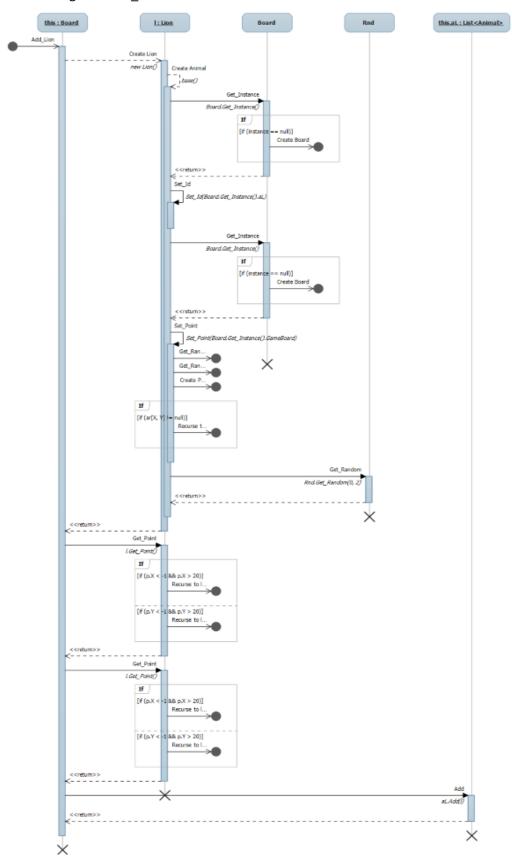
Programmet er modul opdelt for at øge muligheden for at udskifte enkelte komponenter nemt.

For at opnå samhørighed er metoder lagt i de klasse hvor den største del information til metoden allerede ligger.

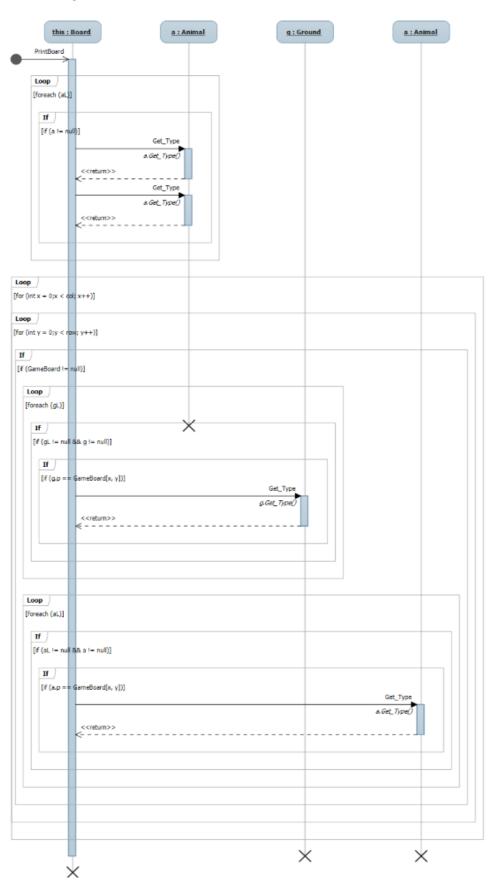
De fleste metoder ligger i Board klassen da den indeholder lister med dyr og grund tiles.

Controlleren og board klassen er begge singletin klasser for at lette tilgangen til dem ellers er variabler private og metoderne public bortset fra enkelte metoder som er private.

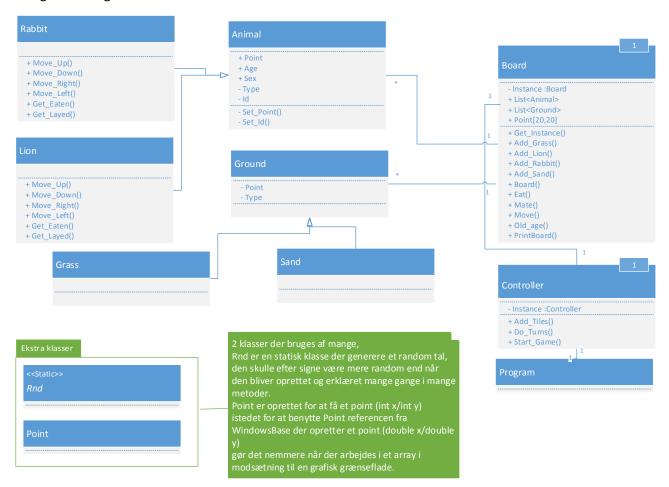
# SekvensDiagram: Add\_Lion



## SekvensDiagram: PrintBoard



#### DesignClassDiagram



#### Eks på kode: