|  |  |
| --- | --- |
| Reflectie-verslag  Leertaak 2: Project Vossen & Konijnen | oVER  Het reflectieverslag van het project Vossen & Konijnen. Dit project wordt uitgevoerd in periode twee van het eerste jaar.  Door  Frank Noorlander, Rick van der Poel en Tsjeard de Winter |

Reflectie-verslag

Leertaak 2: Project Vossen & Konijnen



**Auteurs:** Frank Noorlander, Rick van der Poel en Tsjeard de Winter

**Opdrachtgever**: Dhr. Heijnen

**Organisatie**: Hanzehogeschool Groningen

**Datum** 24-01-2015

**Locatie**: Groningen

# Samenvatting

In dit document vindt u de documentatie van het project Vossen & Konijnen. Deze samenvatting geeft u een indruk van onze werkzaamheden tijdens het project.

Zoals in de probleemstelling al wordt beschreven, heeft de opdrachtgever ons van een incomplete applicatie voorzien welke wij als projectgroep hebben uitgebreid en verbeterd.

De nieuwe applicatie is voorzien van een uitgebreide GUI zodat de simulatie gemakkelijk en correct weergeven wordt op het scherm. In deze GUI is het ook mogelijk om parameters van de dieren en de omgeving te veranderen om de simulatie zo realistisch mogelijk te maken.

De nieuwe applicatie is ook voorzien van verschillende Junit tests. Met deze tests wordt het onderhouden en uitbreiden van de applicatie vereenvoudigd.

De nieuwe applicatie wordt ondersteund met verschillende UML-diagrammen die gemaakt zijn tijdens en na het project. Deze diagrammen geven u een goed beeld van de werkzaamheden die zijn verricht tijdens het project.

De nieuwe applicatie is tevens uitgebreid met verschillende dieren, jagers en voedselbronnen. Met deze uitbreiding is de simulatie een stuk realistischer geworden.

De packagestructuur van de nieuwe applicatie is nu volgens MVC standaarden opgezet. Met deze verandering is de overzichtelijkheid van de code enorm verbeterd.

Tot slot is de GUI van de nieuwe applicatie uitgebreid met verschillende grafieken en afbeeldingen. Deze uitbreiding zorgt ervoor dat u de simulatie gemakkelijker en beter kunt volgen.

Meer informatie over de hierboven besproken onderwerpen vindt u in de verschillende hoofdstukken in dit document.

Inhoudsopgave

[Samenvatting 2](#_Toc409969070)

[Verklarende woordenlijst 4](#_Toc409969071)

[Voorwoord 5](#_Toc409969072)

[1. Probleemstelling 7](#_Toc409969073)

[2. Analyse huidige situatie 8](#_Toc409969074)

[3. Uitbreiding 1 9](#_Toc409969075)

[4. MVC 10](#_Toc409969076)

[5. Uitbreiding 2 11](#_Toc409969077)

[6. Bonus uitbreidingen 12](#_Toc409969078)

[7. Conclusie 13](#_Toc409969079)

[8. Advies / aanbevelingen / actieplan 14](#_Toc409969080)

[9. Noten 15](#_Toc409969081)

[10. Literatuurlijst 16](#_Toc409969082)

[Bijlage 1 17](#_Toc409969083)

[Bijlage 2 18](#_Toc409969084)

[Register 19](#_Toc409969085)

# Verklarende woordenlijst

BlueJ – Programma om Java in te programmeren   
Eclipse – Programma om o.a. Java in te programmeren  
GUI – Graphical User Interface (Grafische gebruikers omgeving)  
V&K – Vossen & Konijnen

# Voorwoord

Voor u ligt het reflectieverslag van het project Vossen & Konijnen. Het project wordt uitgevoerd in de tweede periode van het eerste schooljaar, in opdracht van de Hanzehogeschool. De projectleden zijn: Frank Noorlander, Rick van der Poel en Tsjeard de Winter

Dit verslag zal een reflectie zijn op de werkzaamheden die we als projectgroep hebben verricht om dit project tot een succes te brengen. Meer informatie over het project vindt u in de volgende hoofdstukken.Inleiding

# Probleemstelling

Onze opdrachtgever; de Hanze Hogeschool heeft veel verzoeken gekregen om een applicatie te maken waarmee het de levenscyclus van dierenpopulaties kan simuleren. Omdat er tegenwoordig veel meer aandacht is voor de natuur en het behouden ervan zijn er verschillende partijen geïnteresseerd in zo’n applicatie.

De opdrachtgever heeft zelf al een poging gedaan om een dergelijke applicatie te maken. Deze applicatie genaamd ‘*Vossen & Konijnen’* voldoet echter niet aan de eisen om een goede simulatie te genereren.

Hieronder volgt een opsomming van de problemen met de huidige applicatie:

* De applicatie heeft geen GUI.
* De applicatie heeft geen Junit tests.
* Er zijn nog geen UML diagrammen van de applicatie.
* De applicatie is niet uitgebreid genoeg.
* De packagestructuur van de applicatie is niet volgens MVC.
* De applicatie heeft extra views nodig.
* De parameters van de diersoorten moeten in de GUI aangepast kunnen worden.
* De applicatie moet ondersteund worden met afbeeldingen en/of geluiden.

.

Als projectgroep hebben we de taak gekregen om de huidige applicatie van de opdrachtgever aan te vullen en te verbeteren om tot een realistisch resultaat te komen. De meegegeven applicatie is momenteel erg simpel, dat betekend dat we ons momenteel alleen zullen gaan richten op een bosgebied met daarin vossen en konijnen. Later tijdens dit project zal de applicatie echter uitgebreid worden met meer functionaliteiten.

Als de applicatie een succes wordt, kan het gebruikt kunnen worden om veel verschillende dierenpopulaties te simuleren en daarmee een enorme schat aan informatie te verkrijgen. Deze informatie kan op zijn beurt helpen de natuur en haar inwoners te behouden.

# Analyse huidige situatie

In de huidige versie van het programma dat hebben gekregen van de opdrachtgever is het mogelijk om via een visuele simulator het leven van vossen en konijnen te simuleren.

De vossen en konijnen hebben de volgende eigenschappen:

* Beide dieren hebben een maximale leeftijd. Als deze leeftijd overschreden wordt zullen de dieren verdwijnen uit het programma.
* Beide dieren hebben de mogelijkheid om te paren.
* De vossen kunnen de konijnen opeten als deze naast elkaar komen te staan in de simulator.
* De vossen kunnen dood gaan als ze te lang geen konijnen gegeten hebben.

Zoals hier boven al blijkt, is de applicatie nog erg kaal. De enige overlevingscondities waar de applicatie momenteel rekening mee houdt zijn de leeftijd en de honger. Er wordt bijvoorbeeld geen rekening gehouden met weersomstandigheden, andere dieren die hoger in de voedselketen staan zoals beren of jagers, vervuiling enzovoort.

Ook heeft de huidige applicatie nog geen GUI. Door het gebrek van een GUI is het erg moeilijk om de simulatie te volgen en dat is juist de bedoeling van de applicatie.

Het doel van deze applicatie is om de levenscyclus van dierenpopulaties zo goed en realistisch mogelijk te simuleren. Om dat doel te kunnen realiseren moet de applicatie uitgebreid worden met nieuwe elementen.

1. Uitbreiding 1

# We hebben de originele versie van de applicatie verbeterd door een aantal wijzigingen door te voeren en daarna nieuwe onderdelen toegevoegd. In dit hoofdstuk worden de wijzigingen besproken die we in de eerste week hebben toegepast. **Applicatie aangepast zodat het gebruikt kan worden buiten BleuJ** We zijn als eerst begonnen met het aanpassen van de applicatie zodat het ook in andere programma’s gebruikt kan worden en het niet volledig afhankelijk is van BleuJ. Door het toevoegen van een main methode kan het programma ook geopend worden in andere programmeer programma’s zoals bijvoorbeeld Eclipse. **Ontwikkeling van een simpele GUI** De applicatie is uitgebreid met een simpele GUI waardoor de gebruiker acties kan uitvoeren binnen in de applicatie. Er is een menu aangemaakt waar basis functionaliteit ingezet kan worden zoals bijvoorbeeld het afsluiten van de applicatie. Daarnaast hebben we voor het uitvoeren van de stappen twee knoppen gemaakt en ook nog reset knop toegevoegd die de applicatie terug zet naar een begin status.

Figuur - Simulatie GUI

**Unittests**Om wijzigingen in het programma te controleren op fouten hebben we ook een aantal unittests aangemaakt. Deze testen kunnen worden uitgevoerd als de applicatie is gewijzigd om te controleren of bepaalde functionaliteit van de applicatie nog steeds goed werkt. Slaagt de applicatie er niet in om door de test te komen dan weten we dat er iets mis is met de code. Omdat je met unittests een specifiek onderdeel controleert kun je ook snel het probleem terug vinden.

We hebben de volgende unittests gemaakt waarbij vooral het gedrag van de dieren wordt getest:  
  
- Worden de dieren ouder per stap?   
- Wordt een konijn opgegeten als het naast een vos staat?  
- Sterft een vos ook als er niets te eten is doordat het ingesloten zit?  
  
**Klasse- en sequentiediagram na deze uitbreidingen**  
Hieronder in ziet u een klassendiagram van de applicatie na het toevoegen van de uitbreidingen.   
  


Figuur – Klassendiagram van de V&K applicatie

Naast een klasse-diagram hebben we ook een sequentie-diagram gemaakt. In dit diagram ziet u de interacties tussen verschillende objecten van de applicatie.   


Figuur - Sequentie-diagram

**Toevoeging van een Actor interface, beren en jagers**Omdat het programma met meerdere diersoorten en zelfs met mensen moet kunnen werken hebben we de structuur van de applicatie moeten aanpassen zodat dit makkelijker gedaan kan worden in de toekomst. Om dit te realiseren hebben we een Interface genaamd Actor toegevoegd.  
   
Vervolgens hebben we de klasse Animal abstract gemaakt en ook een abstracte klasse Human toegevoegd. Deze klassen implementeren de klasse Actor en zijn op hun beurt weer de super klasse geworden van de wezens die daar bij horen. Om het duidelijk de veranderingen weer te geven hebben we een stukje van een klassendiagram gemaakt die hieronder te zien is.   
  
  
****

Figuur - klassendiagram (Actoren)

1. MVC
2. Uitbreiding 2
3. Bonus uitbreidingen
4. Conclusie
5. Advies / aanbevelingen / actieplan
6. Noten
7. Literatuurlijst

Bijlage 1

Bijlage 2

Register