



**Faculteit Bedrijf en Organisatie**  
Valentin Vaerwyckweg 1  
9000 GENT

ACADEMIEJAAR 2022-2023  
OPLEIDING TOEGEPASTE INFORMATICA

## **PROJECTWERK**

Opleidingsonderdeel 'Software Development Project I'

*Splendor*

**GROEP: 71**  
**STUDENTEN:**

*CHANDER Charan*

*MEERPOEL Matisse*

*PRIELS Renzo*

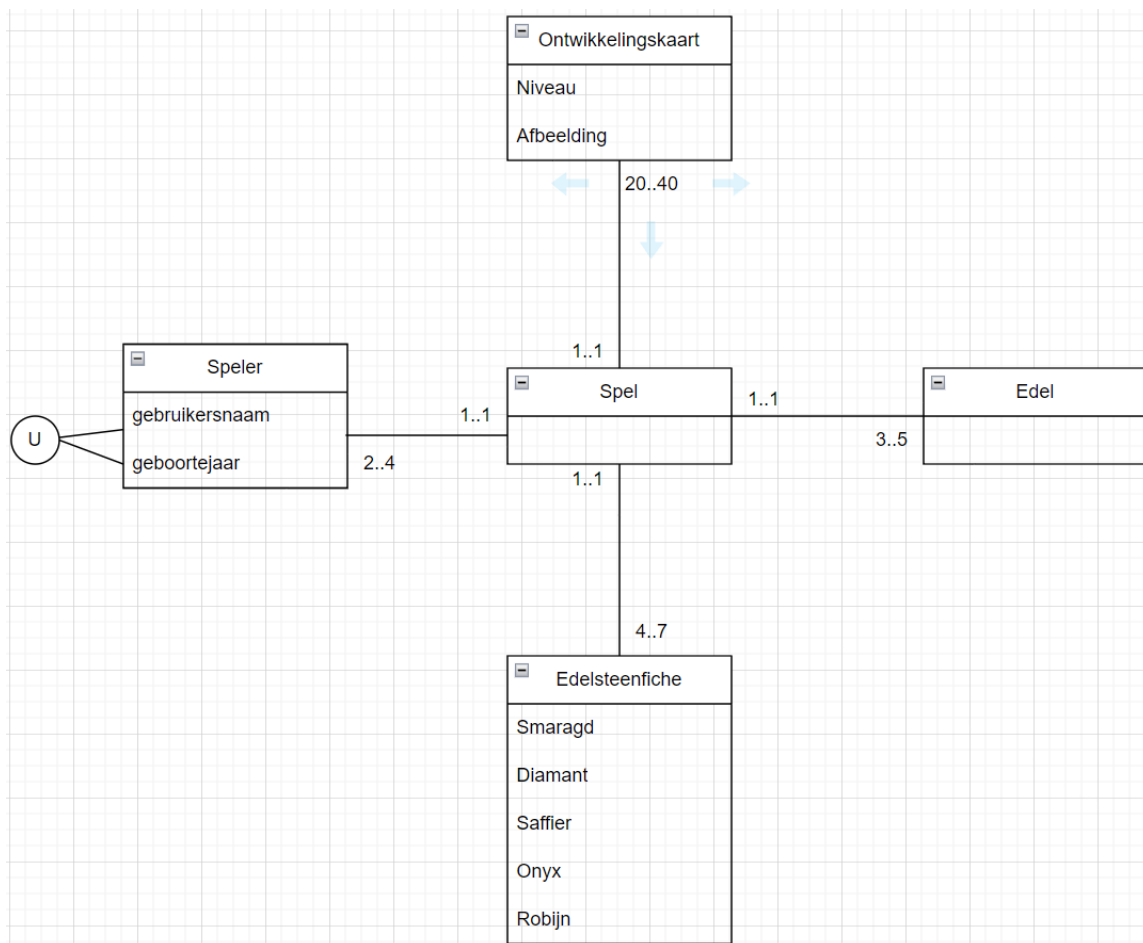
*Vandermeersch Sonia*

**BEGELEIDER:**

# **Inhoudsopgave**

ERD .....	3
Relationeel Model .....	4
Database script (DDL) .....	5
Analyse en Ontwerp .....	6
Use case 1: .....	6
Use Case 2: .....	10
Use Case 3: .....	13
Use Case 4: .....	15
Ervaring Groepsproject .....	22

# ERD



# Relationeel Model

```
Speler(gebruikersnaam,geboortejaar)
Spel()
Edel()
Ontwikkelingskaart(Niveau1, niveau2, niveau3)
Edelsteenfiche(Smaragd, diamant, saffier, onyx, robijn)|
```

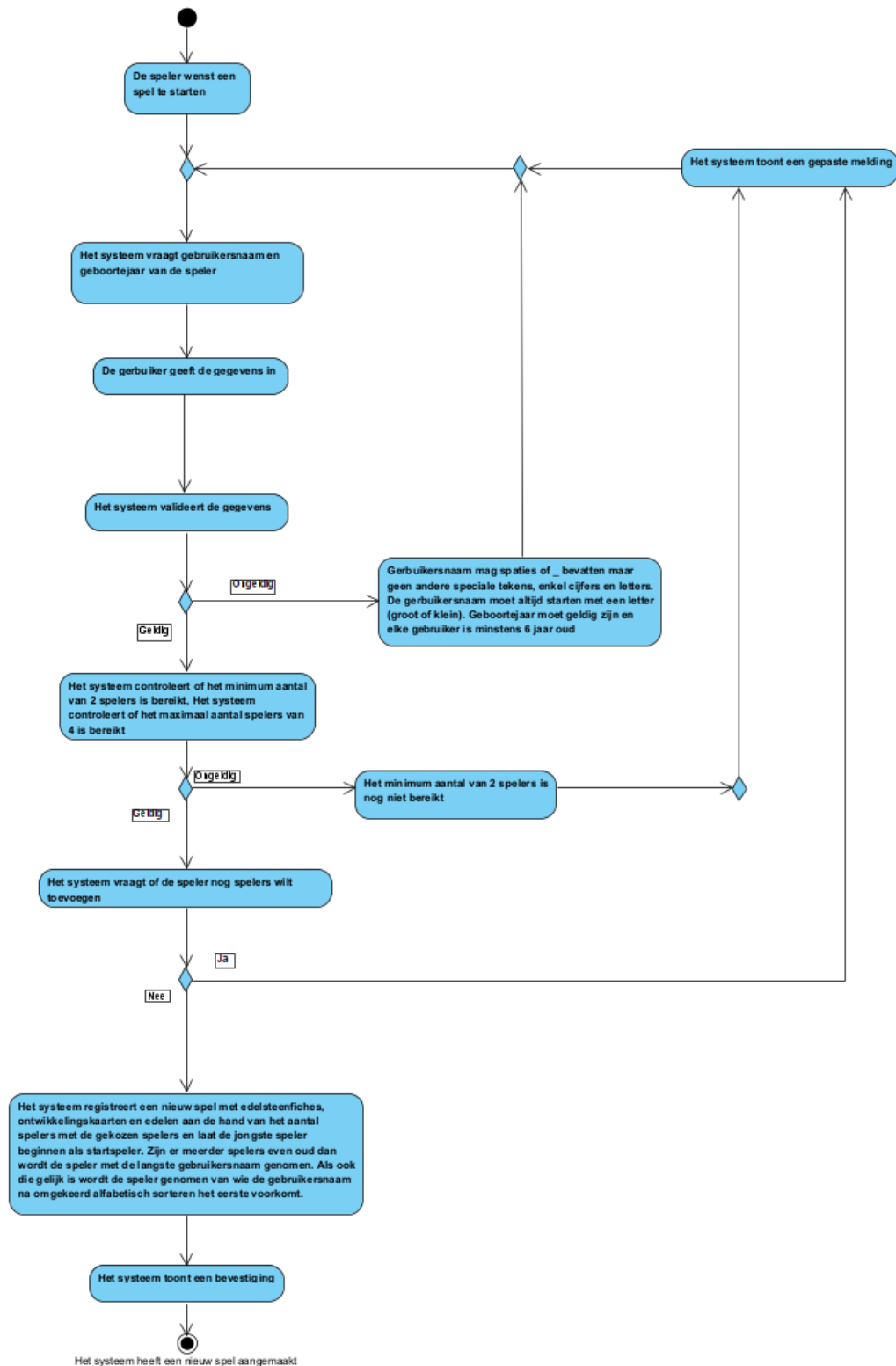
# Database script (DDL)

```
1 • ⊖ create table Edelsteenfiche(  
2     Smaragd int,  
3     Diamant int,  
4     Saffier int,  
5     Onyx int,  
6     Robijn int);  
7  
8 • ⊖ create table Ontikkelingska  
9     Niveau1 int,  
10    Niveau2 int,  
11    Niveau3 int);  
12  
13 • ⊖ create table Speler(  
14     naam varchar(50),  
15     geboortejaar varchar(50));|
```

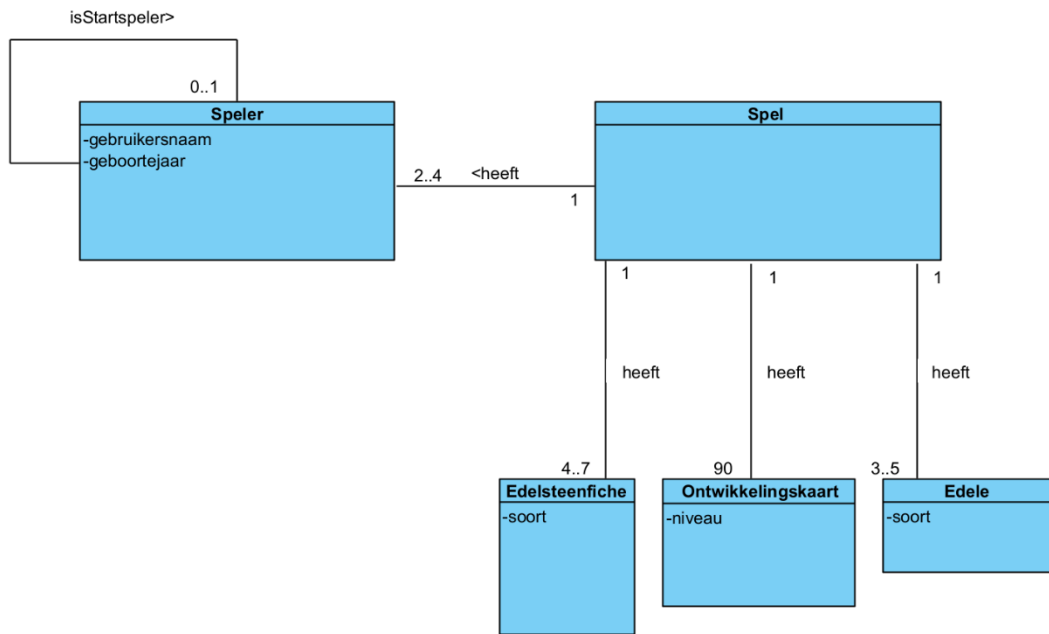
# Analyse en Ontwerp

## Use case 1:

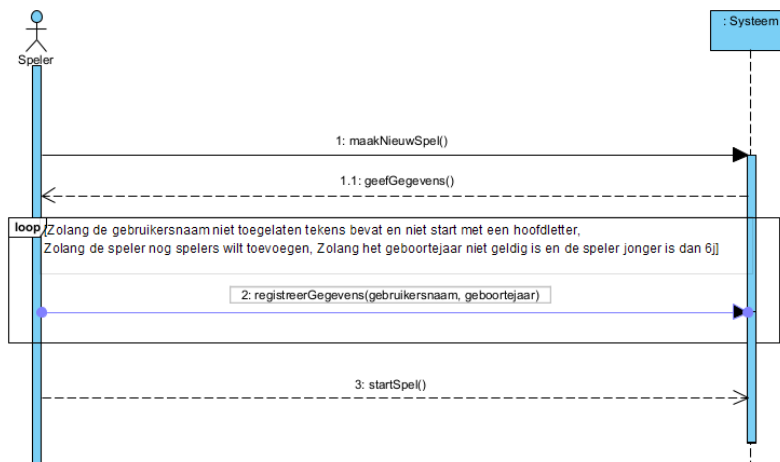
- Activiteitendiagram



- **Domeinmodel**



- **SSD**



Contract: Nieuw spel starten

Operation:	maakNieuwSpel()
Cross reference:	Maak nieuw spel
Precondities:	-
Postcondities:	Het spelobject werd gemaakt Daarna werden de drie stapels met kaarten gemaakt De stapels werden gelinkt met het spel

Contract: Nieuw spel starten

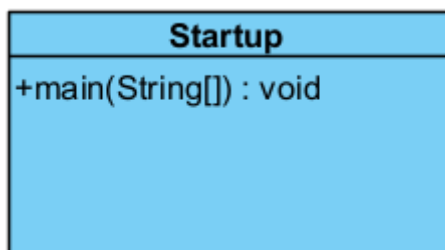
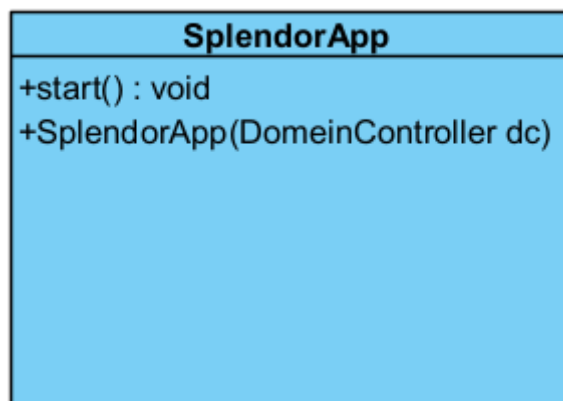
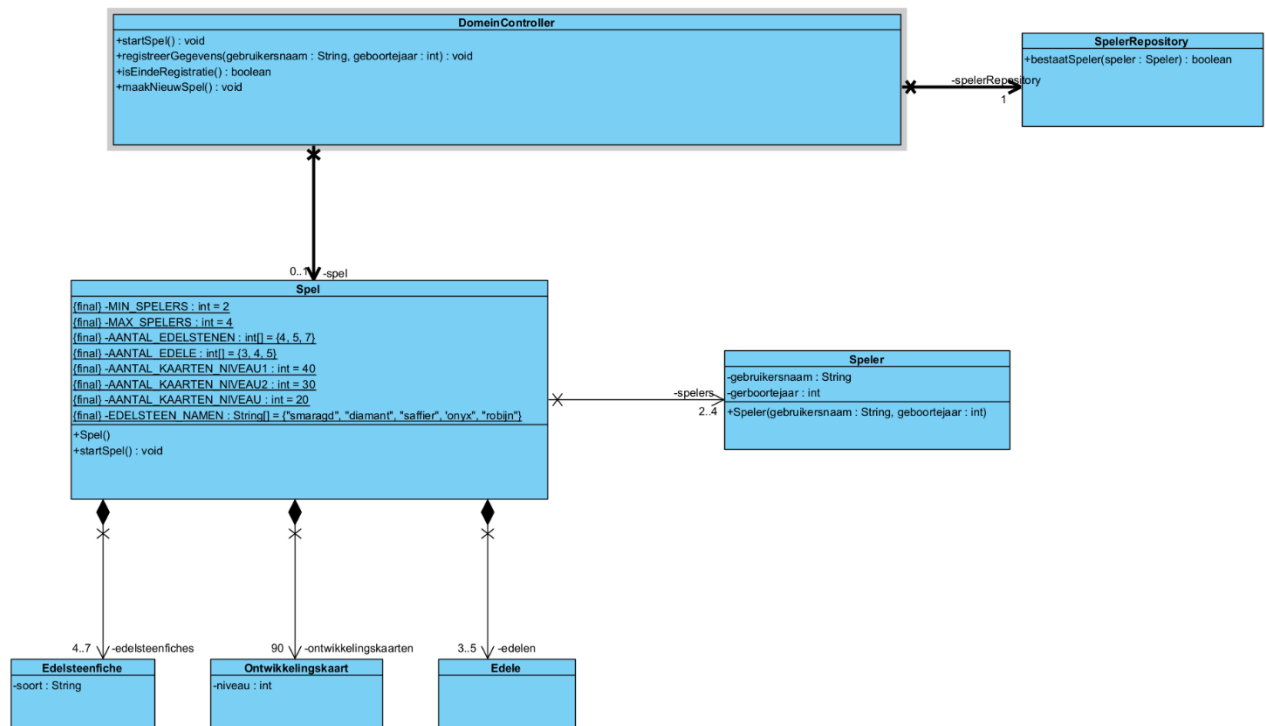
Operation:	registreerGegevens()
Cross reference:	Maak nieuw spel
Precondities:	Er bestaat een spelobject
Postcondities:	Er werd een speler aangemaakt De speler werd gelinkt aan het spel

Contract: Nieuw spel starten

Operation:	startSpel()
Cross reference:	Maak nieuw spel
Precondities:	Er bestaat een spelobject Er werd een speler aangemaakt
Postcondities:	De edelsteenfiches werden aangemaakt volgens het aantal spelers De edelen werden aangemaakt volgens het aantal spelers Deze werden beide toegevoegd aan het spel

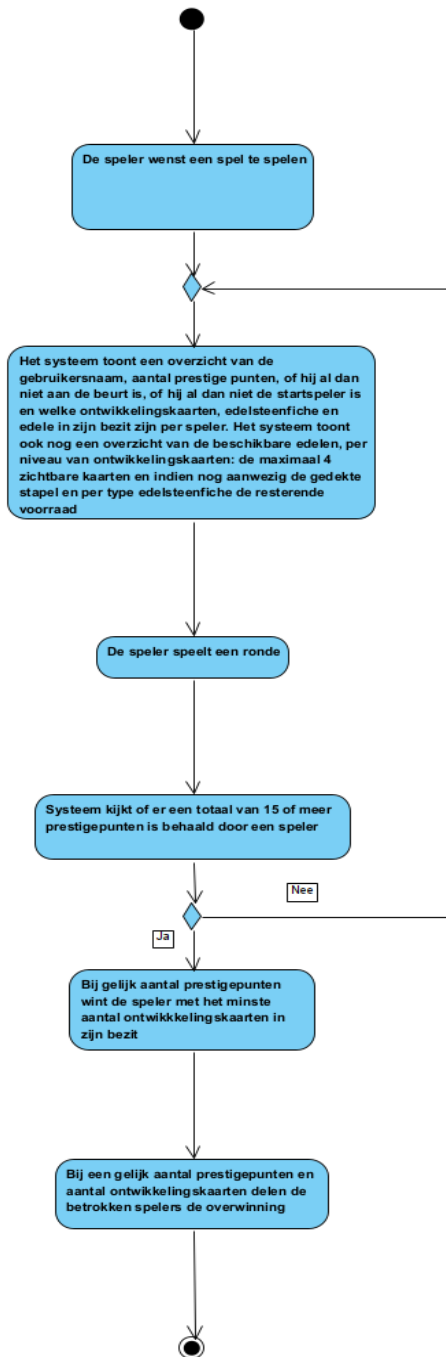


- **DCD**

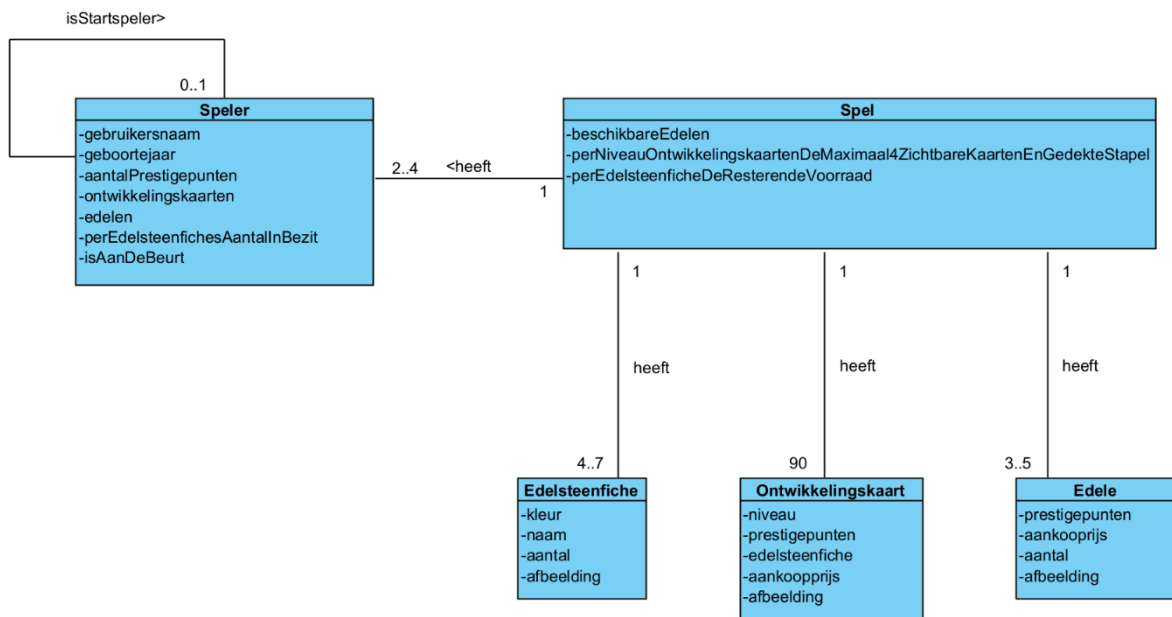


## Use Case 2:

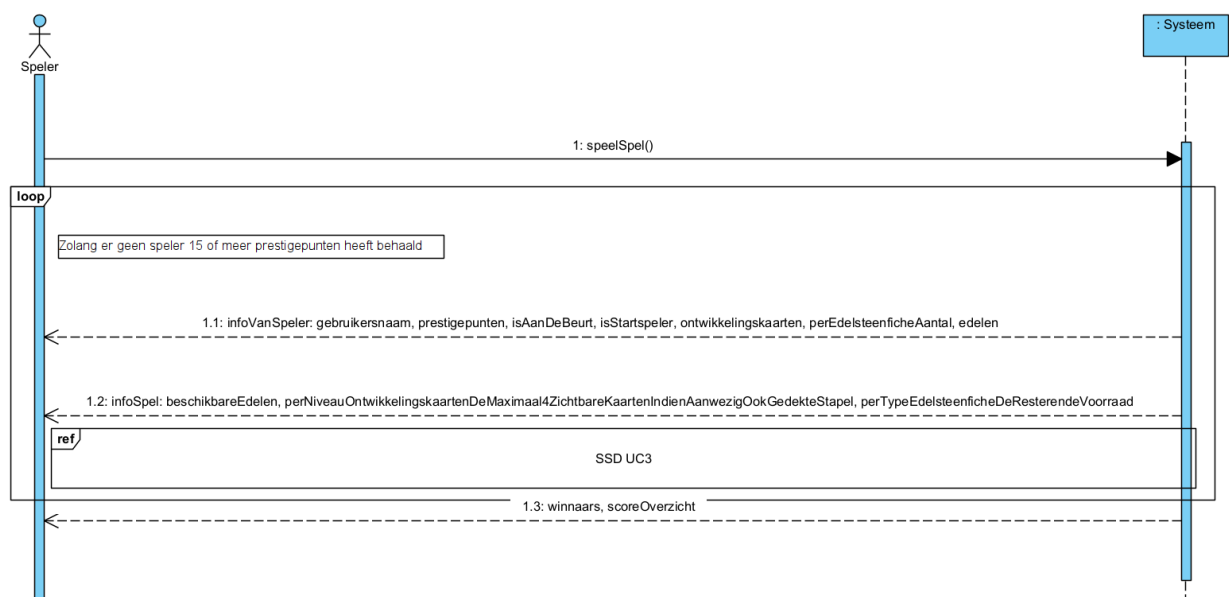
- Activity Diagram Use case 2



- **Domeinmodel Use Case 2:**

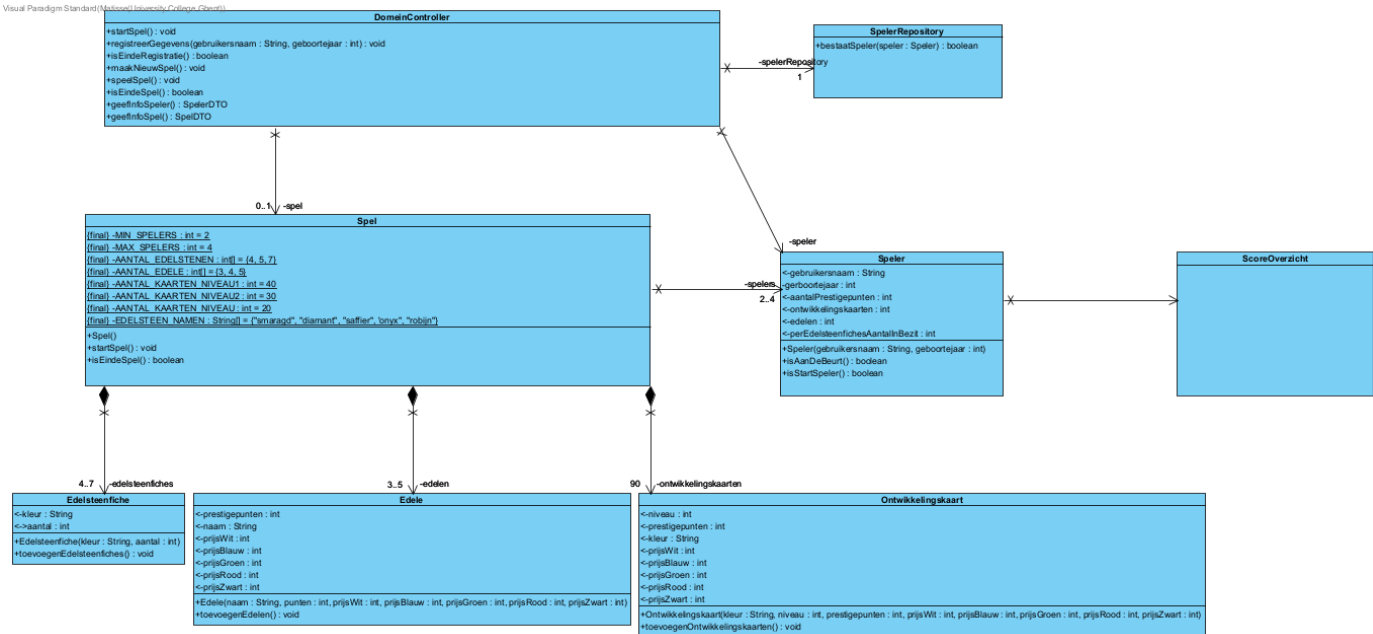


- **SSD use case 2**



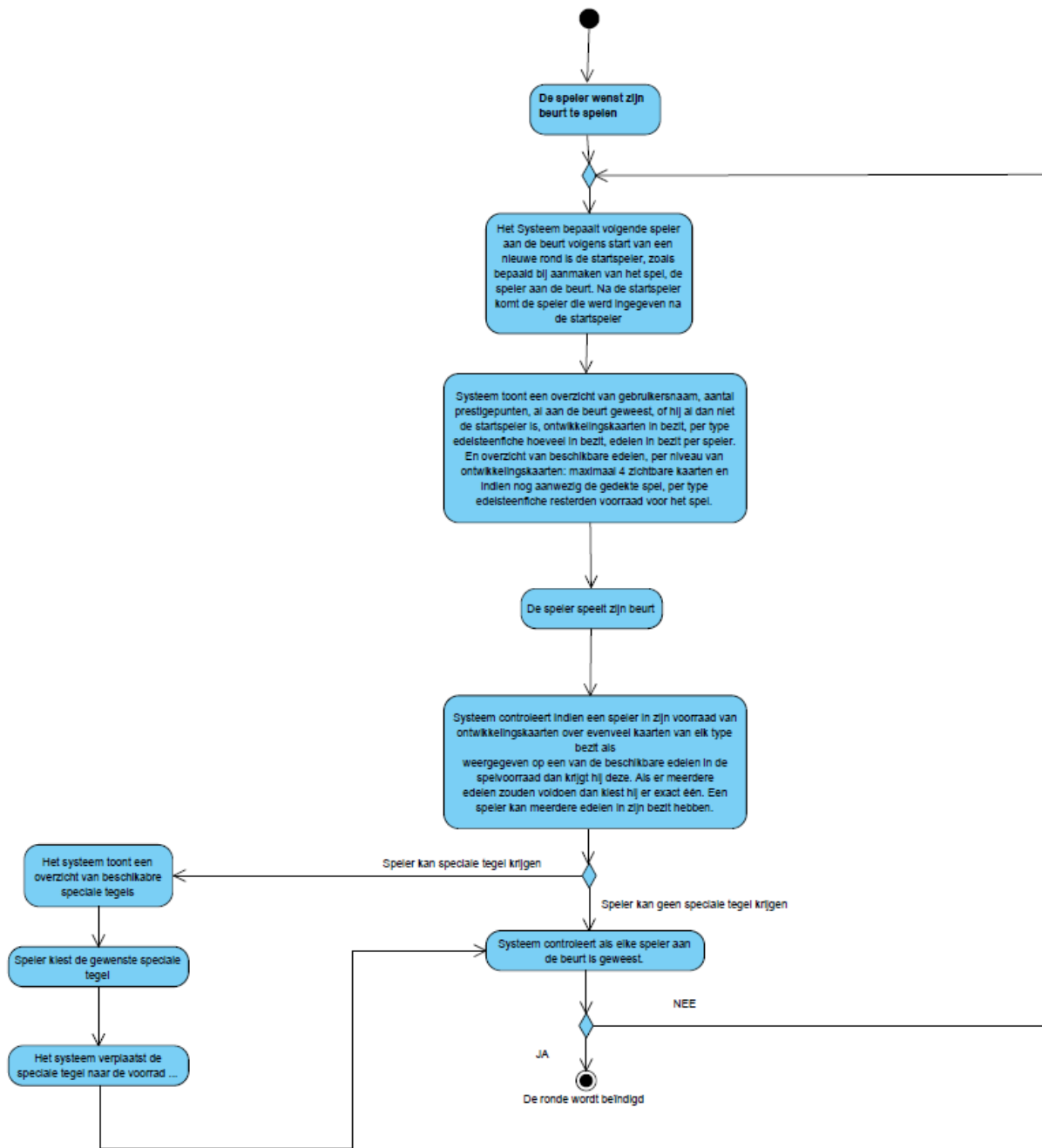
## • DCD Use Case 2

Visual Paradigm Standard (UML2002, UML2004, C++11, Java1.6)

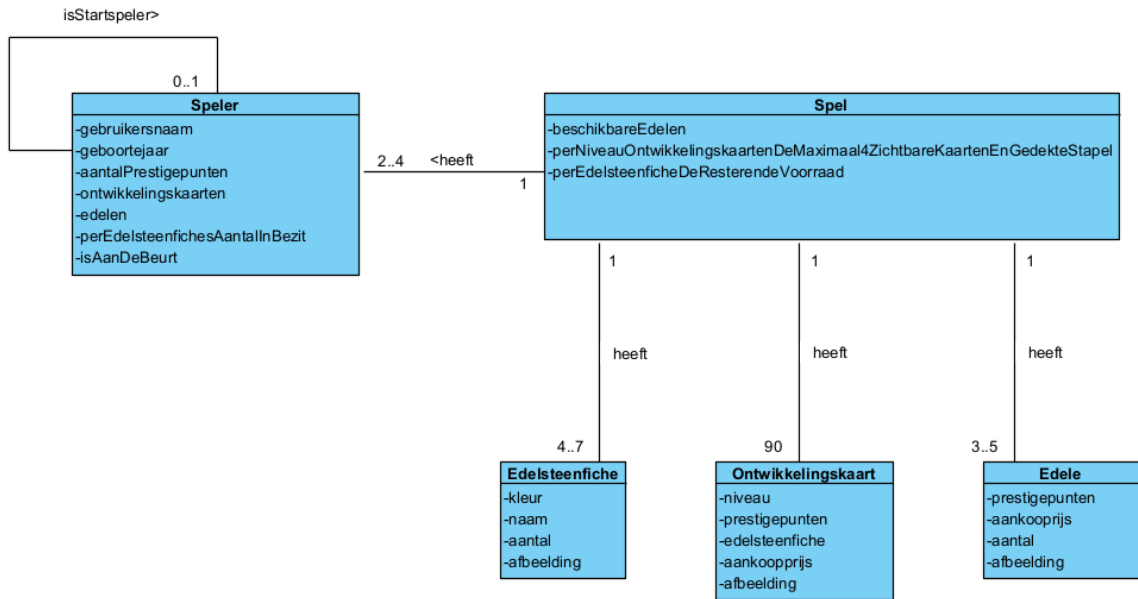


## Use Case 3:

- Activity Diagram Use Case 3:



### Domeinmodel Use Case 3:



### • SSD & OC Use Case 3:



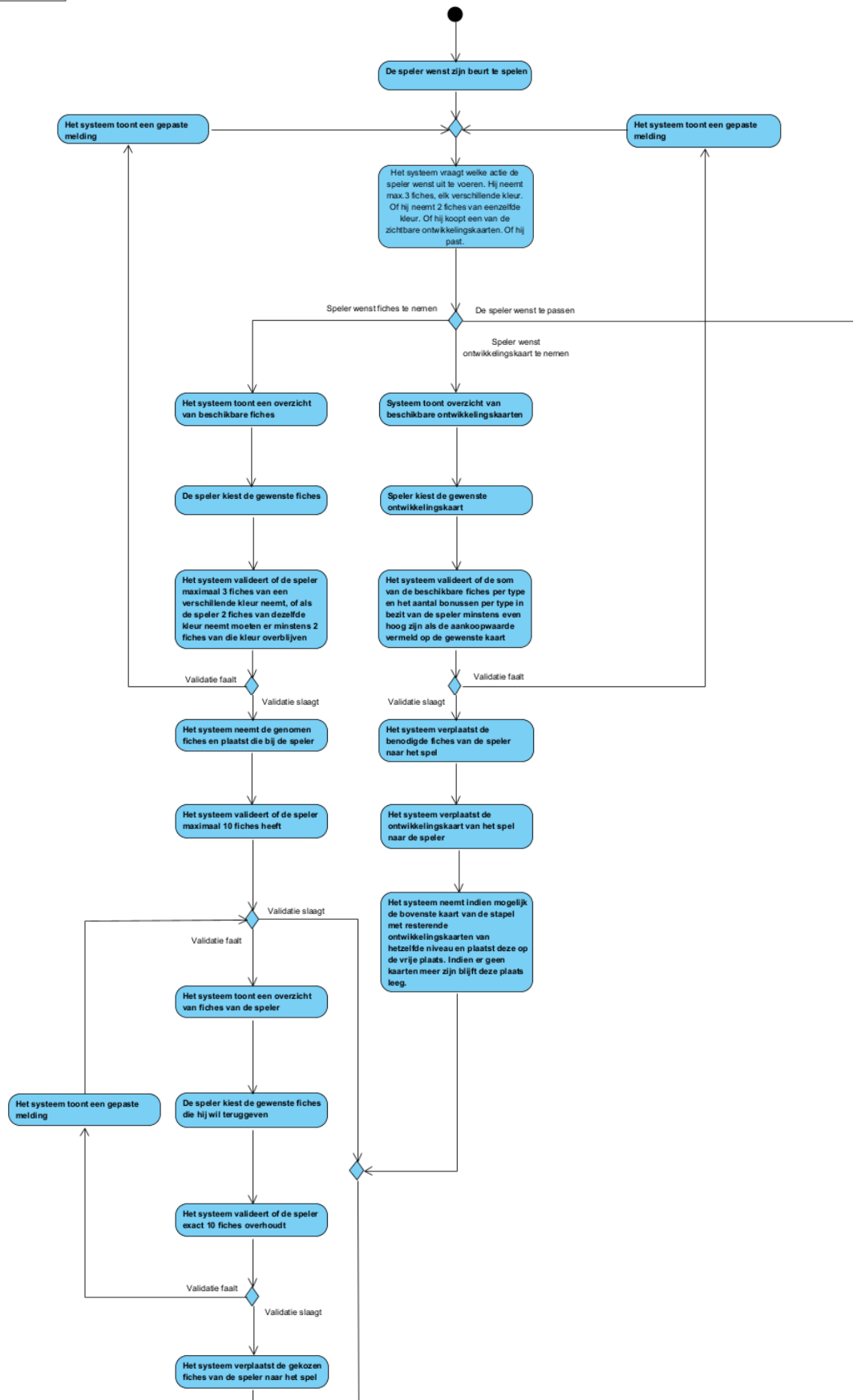
Contract: Edele kiezen

Operation:	koopEdele(gekozenEdele)
Cross reference:	Speel ronde
Precondities:	- Speler bezit over evenveel ontwikkelingskaarten van elke type om een beschikbare edele te kopen. - Ontwikkelingskaarten in bezit moeten overeen komen met het aantal dat op de edele staat.
Postcondities:	- De speciale tegel wordt toegevoegd aan de voorraad van de speler

#### **Use Case 4:**

## Activity Diagram Use Case 4:

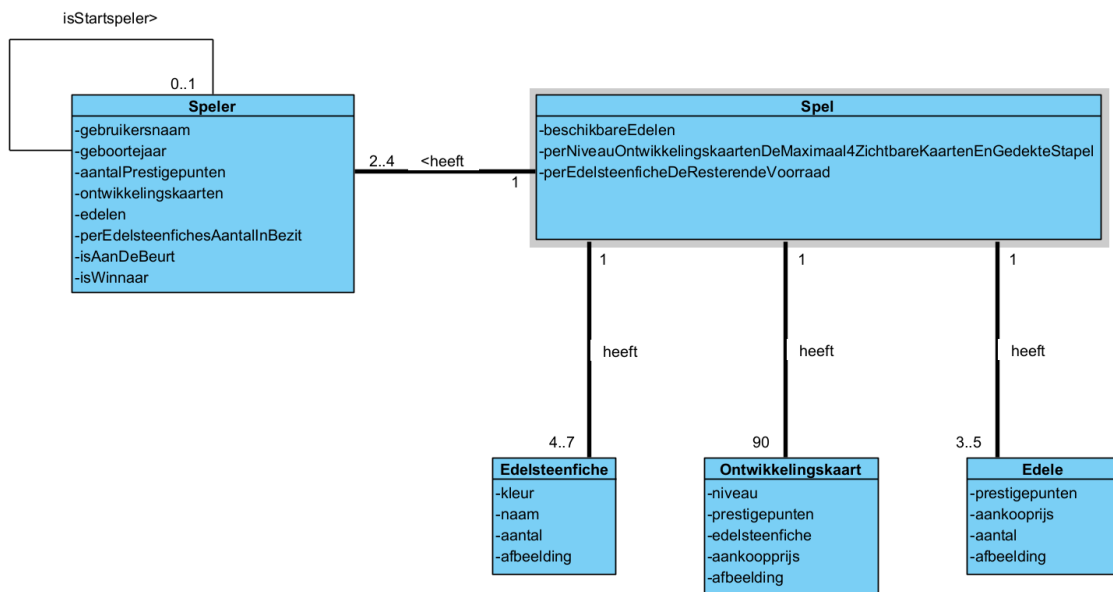
act [Activity Diagram Use Case 4]



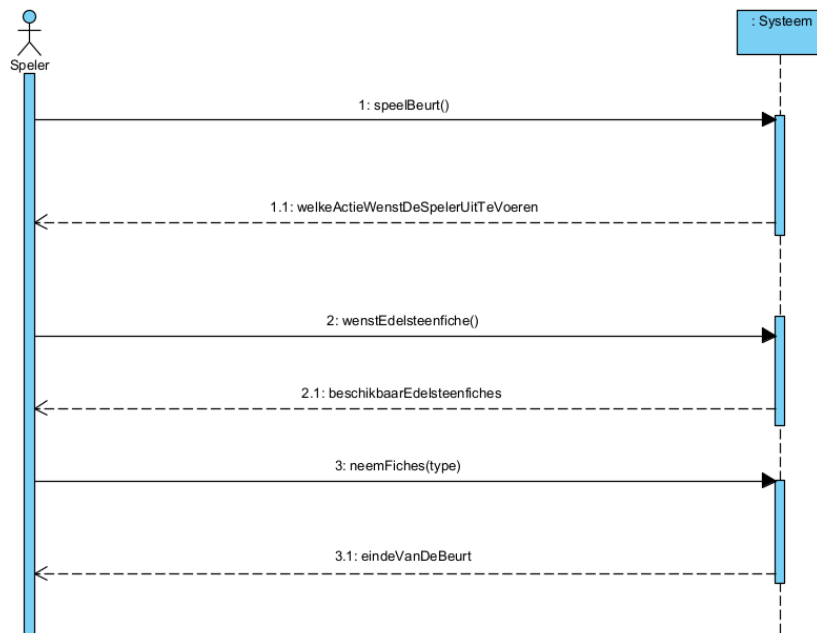




### Domeinmodel Use Case 4:



- **SSD & OC ( + Alternatief Verloop) Use case 4:**



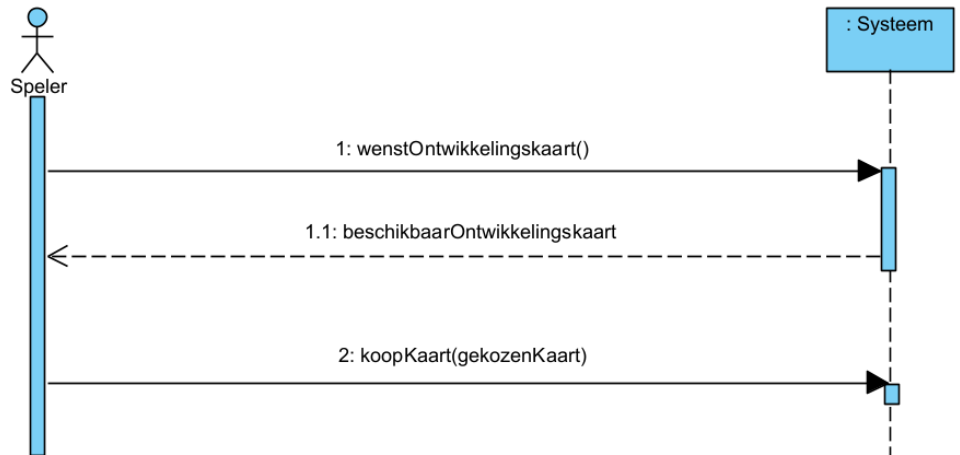
Contract: Speel een beurt

Operation:	speelBeurt()
Cross reference:	Speel beurt
Preconditions:	-Er wordt een ronde gespeeld
Postconditions:	-Het systeem heeft de wijziging in deze beurt bij de speler en het spel is geregistreerd -Beurt van de speler kan geindigd worden en de volgende speler kan beginnen spelen

Contract: Edelsteenfiches kiezen

Operation:	neemFiches(type)
Cross reference:	Speel beurt
Preconditions:	Speler is aan de beurt
Postconditions:	-De gekozen edelsteenfiches worden uit de voorraad van het spel weggehaald -De gekozen edelsteenfiches worden in de voorraad van de speler toegevoegd

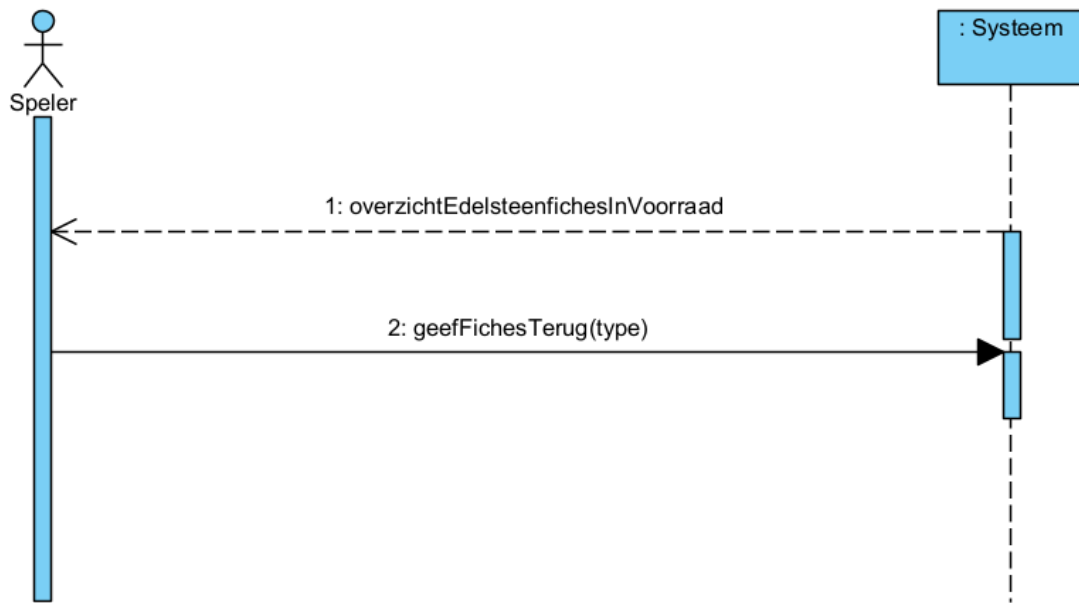
## SSD & OC UC4 (Alternatief Verloop #1)



Contract: Een ontwikkelingskaart kopen

Operation:	koopKaart(gekozenKaart)
Cross reference:	Speel beurt
Precondities:	De speler voldoet aan de aankoopprijs van de ontwikkelingskaart
Postcondities:	<ul style="list-style-type: none"> <li>- Het aantal edelsteenfiches dat nodig waren om de ontwikkelingskaart te kopen werden verwijderd uit de voorraad van de speler</li> <li>- De ontwikkelingskaart dat de speler gekocht heeft werd toegevoegd aan de voorraad van de speler</li> <li>- De ontwikkelingskaart dat gekocht werd werd verwijderd van het bord en als het nog mogelijk was werd daar een nieuwe ontwikkelingskaart van de stapel gelegd</li> </ul>

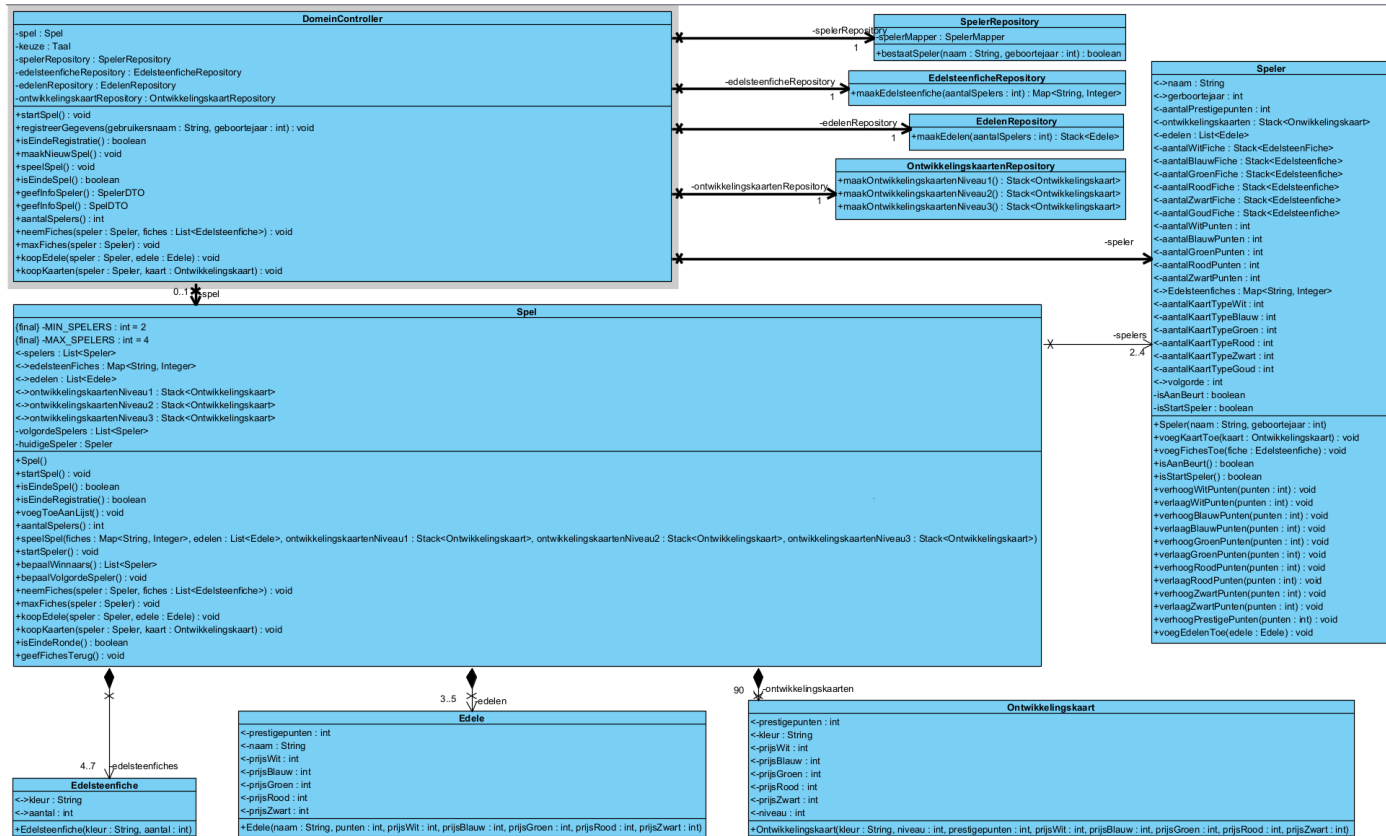
### SSD & OC UC4 (Alternatief Verloop #2)



Contract: Teruggeven  
teveel edelsteenfiches

Operation:	geefFichesTerug(type)
Cross reference:	Speel beurt
Precondities:	-Speler heeft meer dan 10 edelsteenfiches in zijn bezit op het einde van zijn beurt
Postcondities:	-De speler heeft exact 10 edelsteenfiches in zijn bezit - De teruggegeven fiches worden verplaatst van de voorraad van de speler naar de voorraad van het spel.

## • DCD Use Case 4:



# Ervaring Groepsproject

## Matisse Meerpoel:

Voor mij was dit de tweede keer dat ik deelnam aan Software Development Project I. Het is en blijft een grootschalig project. In mijn opinie verliep het project anders, meer bepaald op vlak van gepresteerde werk. Maar in zijn geheel vond ik SDP van dit jaar een serieuze uitdaging tegenover vorig academiejaar. De complexiteit van het te maken spel lag veel hoger.

Dit jaar zaten we in een groep van 3 personen en moesten we het spel Splendor maken. Het volledige project ging gepaard met struikelblokken, maar met wat moeite geraakte we elke keer verder. Zoals eerder vermeld heb ik dit jaar meer werk kunnen presteren, namelijk het opzetten van onze Databank met de nodige verbindingen met onze code, een werkende taalapplicatie geschreven met de werking van de resourcebundles, in het begin ook geholpen samen met Charan om de basis van onze applicatie te schrijven. Verder in de loop van het project wel een groot bijdrage kunnen leveren op vlak van Javafx.

Op vlak van Software & Analyse was ik in het begin niet actief, vanaf UC3 en UC4 heb ik meer werk kunnen inleveren. Voor dit deel heb ik dan ook vaak samengewerkt met Renzo.

Het gedeelte groepswork ging ook wel vlot, met hier en daar soms moeilijkheden met het plannen om verder te werken aan ons project. Het resultaat van dit project heb ik ook te danken aan de groepsleden, Charan en Renzo, ze hebben allebei een grote bijdrage kunnen leveren voor elk segment van dit project.

Om te concluderen was dit project wel complex. Het spel Splendor was zeker geen makkelijke opdracht om te maken en vroeg enorm veel inspanning. Uiteindelijk zijn we wel tot een redelijk resultaat gekomen met dit project, ondanks de moeilijkheidsgraad.

## Renzo Priels

Dit project was een serieuze uitdaging voor mij. Zelf ben ik niet zo ervaren met deze grote projecten dus dit was even wennen. In het begin was het allemaal een beetje zoeken hoe alles zou verlopen, maar doorheen het project is dit wel positief geëvolueerd. Het nadeel dat onze groep vanaf het begin zou hebben is dat we maar met drie aan dit project zouden werken, wat voor ons voor veel meer werk zorgde.

In het begin van het project was ik degene die zich vooral bezighield met de analyse en ontwerp van het project voor de simpele reden dat ik niet zo ervaren ben in het programmeren van een spel. Op aangeven van de lector ben ik rond het einde van UC2 ook begonnen aan het onderdeel programmeren om dit toch ook onder de knie te krijgen.

Met de nodige assistentie van mijn mede groepsleden heb ik ook mijn bijdrage kunnen leveren in het zelf schrijven van methodes en bepaalde lijntjes code. Zelf begon ik ook meer en meer te begrijpen hoe de logica in elkaar zat en hoe ik zelf sommige dingen beter zou kunnen schrijven.

Uiteindelijk heb ik ook nog een redelijke bijdrage kunnen leveren in het onderdeel van JavaFX. Hier heb ik mij vooral beziggehouden met het ontwerp van de panelen en niet echt met de logica erachter te krijgen.

Op het einde van het project wisten we alle logica juist te schrijven maar we kregen dit gewoonweg niet goed verwerkt in de JavaFX wat wel spijtig is want dat had de kers op de taart kunnen zijn.

## Charan Chander

Software Development Project heet lang in mijn achterhoofd gezeten, vooral omdat ik dit het 1<sup>ste</sup> jaar niet heb opgenomen wegens schrik van het niet te kunnen halen. Deze schrik had ik gekregen nadat ik mijn resultaat van OOSD 1 had gekregen, ik begreep het concept achter Java nog niet helemaal en hierdoor wou de logica ook niet klikken.

In het 2<sup>de</sup> jaar heb ik dan besloten om het project op te nemen, dit keer had ik Java een stuk beter onder de knie en had ik het gevoel dat ik het zou aankunnen.

Het project begon helaas al slecht, onze 4<sup>de</sup> teamgenoot besloot om zich uit te schrijven. Dit hield zich in dat we dit project met 3 zouden moeten doen in plaats van met 4. Desondanks dat we met 3 waren hebben we de hoop niet opgegeven en wouden we het project nog steeds volledig afronden.

Vanaf het eerste contactmoment had ik al door dat het voor serieus was, we moesten de leerstof van zowel software analysis en database samenleggen om aan onze diagrammen te komen. Dit was voor mij persoonlijk een minpunt omdat dit niet mijn sterkste kant is, gelukkig heeft Renzo mij kunnen helpen en uitleg gegeven waar nodig.

Mijn eerste diagrammen waren niet perfect, dit had ik zien aankomen omdat het niet mijn sterkste kant is. Naarmate het einde van het project begon ik het onder de knie te krijgen, dit kon ik vooral zien in het AD van UC4. Dit was zonder problemen gelukt. Ik heb het onder de knie kunnen krijgen door de link te proberen leggen met programmeren. Eens ik de link had gelegd ging het beter.

Eens we al onze diagrammen hadden afgerond en aangepast op de feedback mochten we beginnen aan het programmeren, hier keek ik naar uit. Om al ons werk en vooruitgang samen te zien komen in de code, natuurlijk liep het niet altijd vlotjes en zaten we af en toe achter. Desondanks de trage vooruitgang in het begin, hielden we de hoop vol. De doorbraken die we bereikten gaven ons moed en zo energie om door te zetten door de trage tijden.

Het grootste deel waar ik aan heb gewerkt, is de logica. Dit is mijn favoriete deel van het programmeren, ik ben geen van GUI. Bij logica is het 0 of 1, ja of nee. Bij GUI is dit niet zo, je moet lang zitten zoeken naar de juiste waarden en dit kan frustrerend zijn.

Uiteindelijk heb ik spijt dat we geen scenebuilder hebben gebruikt, dit had het GUI proces gemakkelijker kunnen maken denk ik. Door mijn gelimiteerde kennis van JavaFX hebben we jammer genoeg onze GUI niet kunnen afwerken, dit betekent dat ons uiteindelijke project niet 100% werkt. Maar de logica werkt wel 100% waar ik wel blij over ben.

Uiteindelijk vind ik het spijtig dat we ons project niet 100% hebben kunnen afronden, dit komt eender door onze gelimiteerde kennis van JavaFX, onze slechte planning en tijdsindeling. We hebben onze resterende tijd overschat en zijn dus zo te laat in gang geschoten. Aan deze punten ga ik zeker werken bij mijn volgende projecten, ik zal een planning maken en zal er op staan dat de rest van mijn groep dat ook doet.

Om te concluderen vond ik dit een waardige uitdaging, ik kan mij er bij neer leggen dat het niet 100% is gelukt.