Onderwijsgroep Professionele Opleidingen

Handelswetenschappen en bedrijfskunde

|  |
| --- |
|  |
| **Bram Van den Abbeele**  **Glenn van Wezel**  **Fabian Denoodt**  **Emmanuel Kanyinda Kashila**  **Luis Sapon Lucas** |
| Tweede bachelor - Toegepaste Informatica  Choose an item. |
|  |
|  |
| **Software Engineering 4**  Hans Vandenbogaerde |
|  |
| Academiejaar 2018-2019 |

|  |  |
| --- | --- |
| Opdracht 1 | |
| OneHouseStand (2.3) |

Inhoud

[Deel 1 4](bookmark://_Toc3905849#_Toc3905849)

[Scope of Work 4](bookmark://_Toc3905850#_Toc3905850)

[Scope of Product 5](bookmark://_Toc3905851#_Toc3905851)

[Stakeholders: 5](bookmark://_Toc3905852#_Toc3905852)

[Turtle diagram 7](bookmark://_Toc3905853#_Toc3905853)

[Gekozen proces: Een pand toevoegen 7](bookmark://_Toc3905854#_Toc3905854)

[AS-IS BPMN Toevoegen van een pand 8](bookmark://_Toc3905855#_Toc3905855)

[Start event: 8](bookmark://_Toc3905856#_Toc3905856)

[Tasks 8](bookmark://_Toc3905857#_Toc3905857)

[Functionele requirements 10](bookmark://_Toc3905858#_Toc3905858)

[Requirements gevonden uit nieuwe User Stories 11](bookmark://_Toc3905859#_Toc3905859)

[Persona’s 11](bookmark://_Toc3905860#_Toc3905860)

[Grequirements 12](bookmark://_Toc3905861#_Toc3905861)

[Het applicatie-process moet snel gaan 12](bookmark://_Toc3905862#_Toc3905862)

[De applicatie moet makkelijk bruikbaar zijn 12](bookmark://_Toc3905863#_Toc3905863)

[Turtle: Een reservatie plaatsen 13](bookmark://_Toc3905864#_Toc3905864)

[BPMN van proces ‘Een reservatie plaatsen’ 14](bookmark://_Toc3905865#_Toc3905865)

[Userstories 14](bookmark://_Toc3905866#_Toc3905866)

[Koppel requirements aan 5 userstories 14](bookmark://_Toc3905867#_Toc3905867)

[User rollen 15](bookmark://_Toc3905868#_Toc3905868)

[Persona’s 15](bookmark://_Toc3905869#_Toc3905869)

[Technieken 16](bookmark://_Toc3905870#_Toc3905870)

[Observatie 16](bookmark://_Toc3905871#_Toc3905871)

[Interview (met Glenn als screener) 17](bookmark://_Toc3905872#_Toc3905872)

[INVEST: 17](bookmark://_Toc3905873#_Toc3905873)

[INVEST 19](bookmark://_Toc3905874#_Toc3905874)

[Oude versie 19](bookmark://_Toc3905875#_Toc3905875)

[Nieuwe versie 19](bookmark://_Toc3905876#_Toc3905876)

[INVEST 20](bookmark://_Toc3905877#_Toc3905877)

[INVEST 20](bookmark://_Toc3905878#_Toc3905878)

[AcceptatieCriteria 20](bookmark://_Toc3905879#_Toc3905879)

[Business en informatie use cases 22](bookmark://_Toc3905880#_Toc3905880)

[**Use Case Diagrammen** 22](bookmark://_Toc3905881#_Toc3905881)

[Use case beschrijvingen 24](bookmark://_Toc3905882#_Toc3905882)

[Business use case: Pand verwerken 24](bookmark://_Toc3905883#_Toc3905883)

[Information use case: Pand status bekijken 26](bookmark://_Toc3905884#_Toc3905884)

[Business use case: Pand reviewen 29](bookmark://_Toc3905885#_Toc3905885)

[Information use case: Staat van matten opvragen 30](bookmark://_Toc3905886#_Toc3905886)

[Kandidaat-objecten 31](bookmark://_Toc3905887#_Toc3905887)

[Woordenboek 33](bookmark://_Toc3905888#_Toc3905888)

[Klassendiagram 34](bookmark://_Toc3905889#_Toc3905889)

[Toestandsdiagrammen 35](bookmark://_Toc3905890#_Toc3905890)

[Aanvraag 35](bookmark://_Toc3905891#_Toc3905891)

[Review 36](bookmark://_Toc3905892#_Toc3905892)

[Mat 37](bookmark://_Toc3905893#_Toc3905893)

[Iteratie 37](bookmark://_Toc3905894#_Toc3905894)

[Feedback medestudenten 38](bookmark://_Toc3905895#_Toc3905895)

[Validatie klassen diagram door user stories + use case beschrijving 39](bookmark://_Toc3905896#_Toc3905896)

[Use Cases 43](bookmark://_Toc3905897#_Toc3905897)

[Boundary klassen 52](bookmark://_Toc3905898#_Toc3905898)

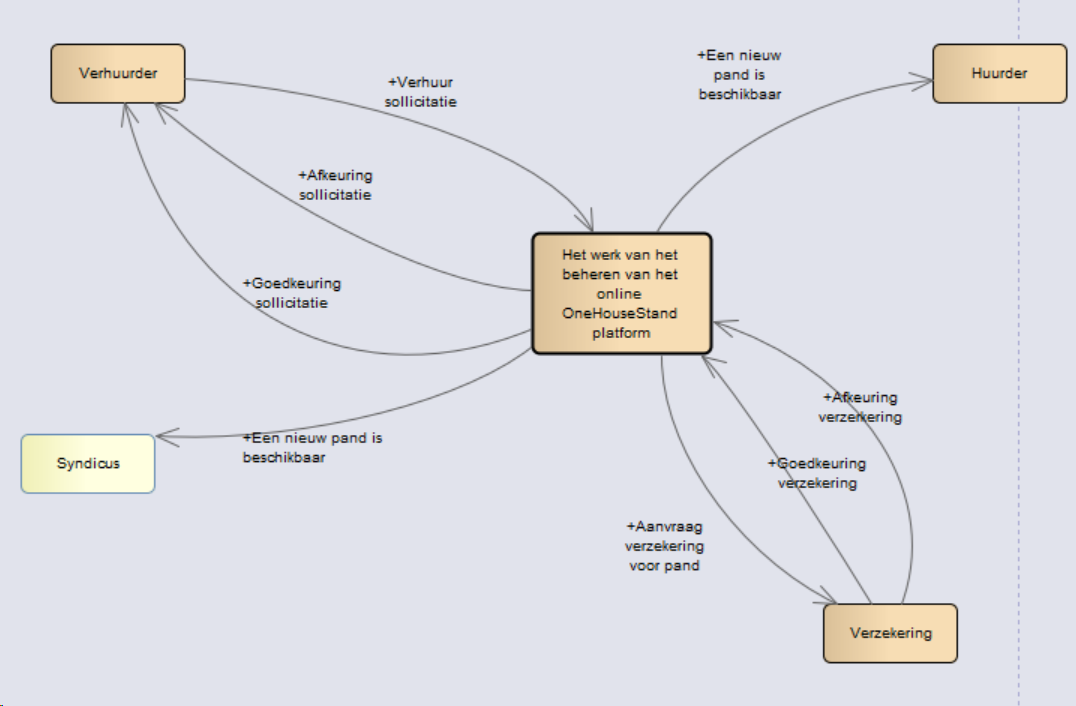
[Klassen diagram na feedback Hans 54](bookmark://_Toc3905899#_Toc3905899)

Software Opdracht 1

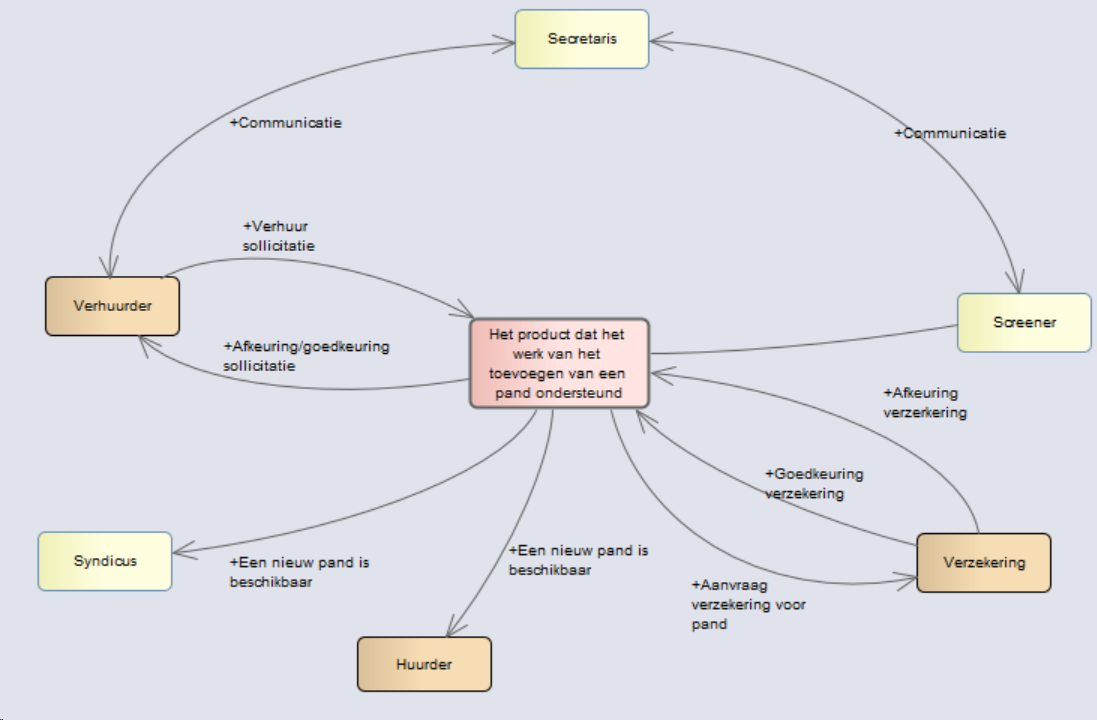
Deel 1

## Scope of Work

Gekozen proces: “Het werk van het toevoegen van een nieuw pand “



## Scope of Product



### Stakeholders:

* **Verhuurders** van panden (Omdat hij een aanvraag indient)
* **Huurders** van panden (Omdat als het pand goedgekeurd is en er dus meer kueuze is voor de huuder)
* **Screeners** (Hij zal het pand goed of afkeuren)
* **Secretaris** die met potentiële verhuurder communiceert
* **Syndicus** die matten van Verimpex installeert

Gekozen proces

Alle geïdentificeerde processen

Primaire (Case) processen:

* Een huis screenen
* Een pand beoordelen
* Een verzekering regelen
* Een nieuw pand toevoegen
  + Een pand screenen
* Een pand voorzien van matten
* Een start-up-circus ruimte verhuren
* Een pand verhuren
  + Een feedback bericht nalezen van een huurder op een pand
* Een pand overzien tijdens verhuur
* Een pand van de nodige Verimpex-matten voorzien

Support (Case) processen:

* Een nieuw personeelslid aanwerven
* Een foto / post toevoegen aan de facebook pagina
* Contact onderhouden met derde partij services
* Een mattenprobleem doorgeven aan Verimpex

Management processen:

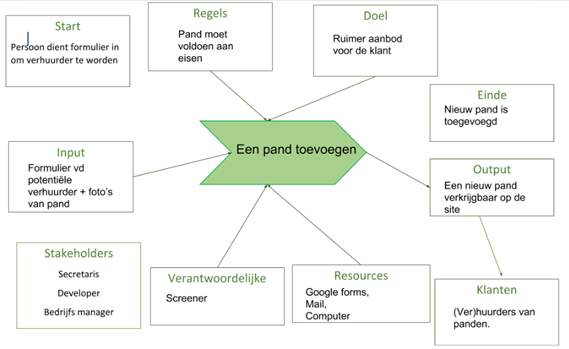
* Online platform beheren

Strategische processen:

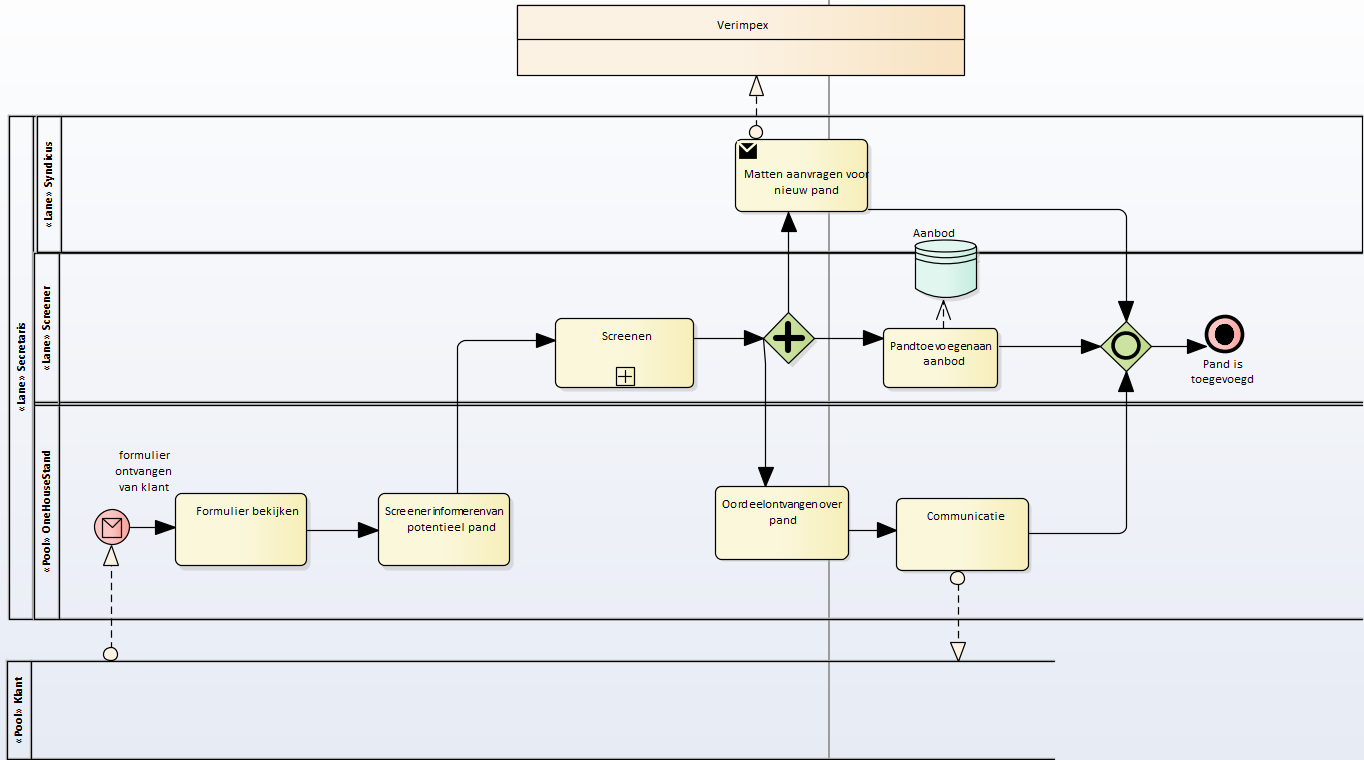
* Innoveren (Nieuwe features toevoegen aan website/ applicatie die wij gaan ontwikkelen voor hun)
* Ons nieuw verhuurders sollicitatie systeem implementeren
* De website vernieuwen

## Turtle diagram

### Gekozen proces: Een pand toevoegen



## AS-IS BPMN Toevoegen van een pand



### Start event:

Ontvangen van een formulier, het formulier waarmee een potentiele verhuurder hun pand aanbied aan OneHouseStand

### Tasks

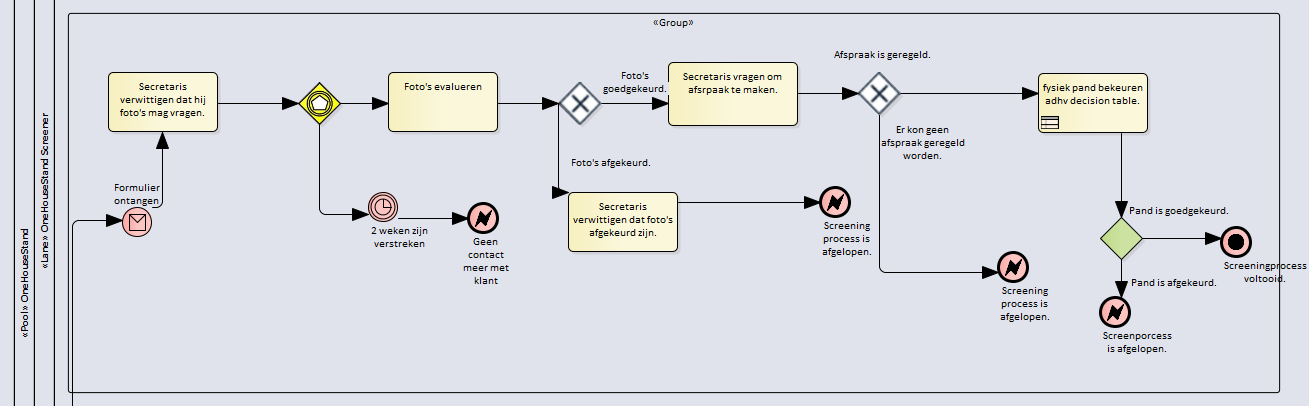
Eerst bekijkt de secretaris het formulier, daarna informeert hij/zij de screener.

Vervolgens start de screener het Screener proces, dit is een sub proces en werd in een apart BPMN uitgewerkt vorig semester.

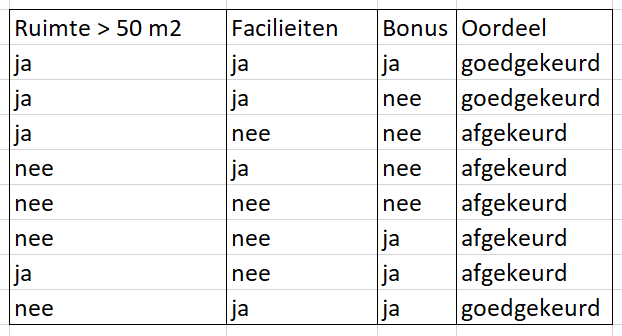
Na het screener wordt er een aanvraag verstuurd naar Verimpex voor matten te regelen voor het nieuwe pand (indien het pand goedgekeurd werd), tegelijkertijd wordt de secretaris geïnformeerd dat een pand beoordeeld werd.

Als het pand goedgekeurd werd, wordt het ten slotte toegevoegd aan het aanbod, terwijl de secretaris contact houdt met de nieuwe verhuurder.

Het proces eindigt wanneer het pand is toegevoegd, en contact niet langer nodig is met de nieuwe verhuurder (alle vragen zijn beantwoord en alles is duidelijk).



Decision table gebruikt voor het beoordelen van een pand:



# Functionele requirements

* Als klant kan ik een formulier over mijn pand indienen
* Als klant kan ik foto’s doorsturen van mijn pand
* Als screener wil ik een ingediend formulier kunnen bekijken
* Als screener wil ik de foto’s van een pand ontvangen
* Als screener wil ik de foto’s van een pand evalueren
* Als screener wil ik foto’s kunnen beoordelen
* Als screener wil ik de potentiele verhuurder kunnen vragen om een afspraak te maken
* Als screener wil ik een pand kunnen bekeuren
* Als screener wil ik de status pand kunnen doorgeven
* Als screener kan ik een pand toevoegen aan OneHouseStand
* !! start-up-circus kan een pand van OneHouseStand gebruiken
* !! Een huurder kan op de website van start-up-circus een pand van OneHouseStand aanvraagen
* Als klant wil ik de status van mijn pand kunnen bekijken
* ------------------------------------------------------------------
* Als huurder wil ik een boeking kunnen plaatsen
* Als screener wil ik feedback kunnen ontvangen
* Als screener wil ik kunnen zien of er problemen waren met de Verimpex(c) matten
* Als screener wil ik de syndicus kunnen een probleem met de matten doorsturen
* Als Screener wil ik de verhuurder de feedback kunnen doorsturen
* Als Syndicus wil ik een probleem met de matten kunnen ontvangen
* Als Syndicus wil ik Verimpex op de hoogte kunnen brengen van een probleem met de matten
* (door aanpassen van “pand toevoegen” bpmn is er een nieuwe activity => nieuwe requirement)
* Als Syndicus moet ik matten kunnen aanvragen aan Verimpex(c) voor nieuwe panden

## Requirements gevonden uit nieuwe User Stories

### Persona’s

Als screener wil ik het potentiële pand kunnen vergelijken met dat van de al bestaande (goedgekeurde) panden.

Als potentiële verhuurder wil ik een inbreng hebben in de doelgroep waar mijn huis voor bestemd is .  
Als potentiële verhuurder wil ik de status van mijn aanvraag/ sollicitatie kunnen bekijken.

Als potentiele huurder wil ik voordat ik mijn keuze maak een kijkje kunnen nemen in het pand

## Requirements

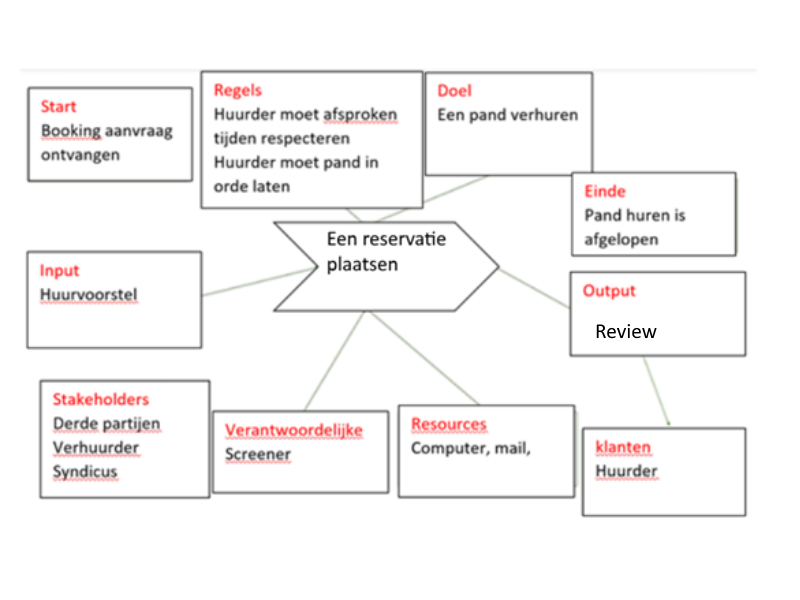
### Het applicatie-process moet snel gaan

|  |  |
| --- | --- |
| KEYWORD | INVULLING |
| TAG | Aanvraag snelheid |
| GIST | De mate waarin een gebruiker moet wachten tot zijn aanvraag verwerkt is. |
| STAKEHOLDER | Potentiële verhuurder |
| SCALE | Tijd die potentiële verhuurder moet wachten vanaf hij zijn aanvraag indient totdat de aanvraag verwerkt is en het eindoordeel van het pand ziet. |
| METER | Gemiddelde wachttijd van de laatste 20 potentiele verhuurders. |
| MUST | 1 maand |
| PLAN | 2 weken |
| STRETCH | 1 week |
| WISH | 3 dagen |
| PAST | 1 maand |
| TREND | 1 maand |
| RECORD | / |
| DEFINED | Average waiting time (AWT) |
| AUTHORITY | / |

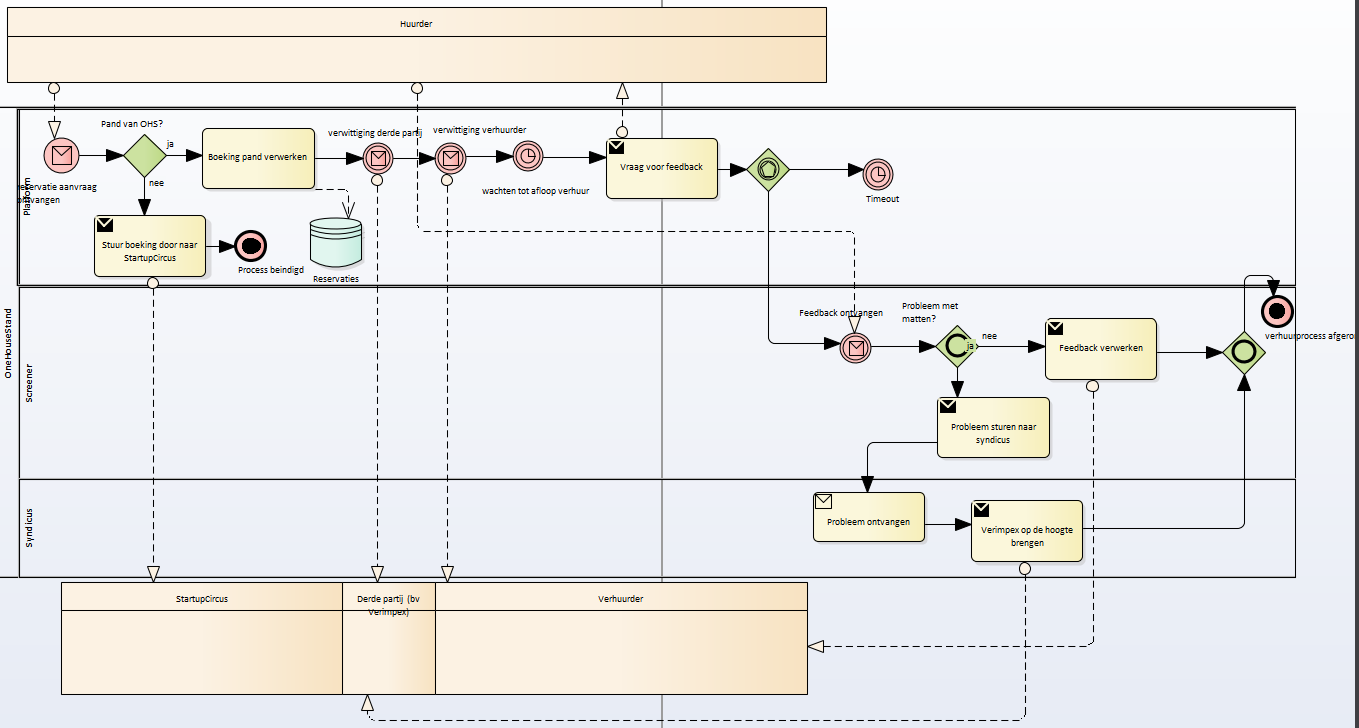
### De applicatie moet makkelijk bruikbaar zijn

|  |  |
| --- | --- |
| KEYWORD | INVULLING |
| TAG | REQ-01 Bruikbaarheid |
| GIST | De mate waarin het programma eenvoudig bruikbaar is voor nieuwe gebruikers, om zo snel mogelijk zijn functie te kunnen uitvoeren. |
| STAKEHOLDER | Gebruiker van de applicatie |
| SCALE | Aantal keer dat een actie ondernomen moet worden (geklikt) om tot een bepaalde pagina / info te komen. |
| METER | Gemiddeld aantal acties dat een gebruiker doet om zijn functie naar verhand uit te voeren. Gemiddede score van 25 gebruikers. |
| MUST | 500 acties |
| PLAN | 400 acties |
| STRETCH | 300 |
| WISH | 200 |
| PAST | / niet gemeten |
| TREND | / |
| RECORD | / |
| DEFINED | Number of Performed Actions (NPA) |
| AUTHORITY | / |

## Turtle: Een reservatie plaatsen



### BPMN van proces ‘Een reservatie plaatsen’



## Userstories

### Koppel requirements aan 5 userstories

1. Als verhuurder kan ik een pand voor verhuur aan OneHouseStand voorstellen
   1. Als klant kan ik een formulier over mijn pand indienen
   2. Als klant kan ik foto’s doorsturen van mijn pand
   3. Als klant wil ik de status van mijn pand kunnen bekijken
2. Als screener kan ik een pand aan OneHouseStand toevoegen
   1. Als screener wil ik een ingediend formulier kunnen bekijken
   2. Als screener wil ik de foto’s van een pand ontvangen
   3. Als screener wil ik de foto’s van een pand evalueren
   4. Als screener wil ik foto’s kunnen beoordelen
   5. Als screener wil ik de potentiele verhuurder kunnen vragen om een afspraak te maken
   6. Als screener wil ik een pand kunnen bekeuren
   7. Als screener wil ik de status pand kunnen doorgeven
   8. Als screener kan ik een pand toevoegen aan OneHouseStand
3. Als syndicus kan ik een pand van de nodige Verimpex-matten voorzien
   1. Als Syndicus wil ik een probleem met de matten kunnen ontvangen
   2. Als Syndicus wil ik Verimpex op de hoogte kunnen brengen van een probleem met de matten
4. Als huurder kan ik een geschikt pand voor mijn evenement zoeken en vastleggen
   1. Als huurder wil ik een boeking kunnen plaatsen
5. Als huurder kan ik na huur het gehuurde pand reviewen
   1. Als screener wil ik feedback kunnen ontvangen
   2. Als screener wil ik kunnen zien of er problemen waren met de Verimpex(c) matten
   3. Als screener wil ik de syndicus kunnen een probleem met de matten doorsturen
   4. Als Screener wil ik de verhuurder de feedback kunnen doorsturen
   5. Als Syndicus wil ik een probleem met de matten kunnen ontvangen
   6. Als Syndicus wil ik Verimpex op de hoogte kunnen brengen van een probleem met de matten
6. Als screener kan ik de informatie over het pand aanpassen op basis van de reviews door de huurder
   1. Als screener wil ik feedback kunnen ontvangen
   2. Als screener wil ik de syndicus kunnen een probleem met de matten doorsturen
   3. Als Screener wil ik de verhuurder de feedback kunnen doorsturen

## User rollen

* Verhuurder
* Screener
* Huurder
* Syndicus

### Persona’s

1. Screener:
   1. Jeff (20) full time screener van OneHouseStand, houd ervan om alles geordend en overzichtelijk te houden, zodat hij zo efficiënt mogelijk kan werken.  
        
      Hij houdt van overzicht omdat hij op die manier ook de verschillende panden kan vergelijken met elkaar, om zo een extra variabel in het selectieproces toe te voegen.

**User Story:**  
**Als screener wil ik het potentiële pand kunnen vergelijken met dat van de al bestaande (goedgekeurde) panden, omdat ik een overzicht kan hebben van dat de kwaliteit van de huizen zodat die niet achteruitgaat. En eventueel ook meer diversiteit in de huizen kan brengen.**

1. Potentiële verhuurder
   1. Bert (47) heeft onlangs geïnvesteerd in een uniek pand om dit te kunnen verhuren. Bert is niet meer mee met de laatste nieuwe technologieën, en dus is het belangrijk voor hem dat hij gebruik kan maken van een platform dat makkelijk te gebruiken is.  
        
      **User Story:**  
      **Als potentiële verhuurder wil ik dat de applicatie gebruiksvriendelijk is omdat ik niet meer van de jongste ben en sommige programma’s te moeilijk zijn voor mij.**
   2. Gert (53, koningskind). Nu zijn ouders gestorven zijn, heeft hun villa geërfd. Echter zou hij er zelf niet in willen wonen omdat het hem te veel aan zijn ouders doet denken en hij dan zelf te ver van zijn werk woont. Het huis verkopen wil hij ook niet doen vanwege de emotionele band. Gelukkig heeft Gert recent een artikel in de krant gelezen over OneHouseStand. Nu kan hij zijn huis op momenten verhuren en als hij wilt er zelf ook nog langs komen. Als bonus moet Gert dan ook geen leegstand betalen.  
        
         
        
      Als potentiële verhuurder verwacht Gert dus ook dat er geen sukkelaars in zijn huis komen en enkel deftige mensen die zijn kot niet op stelten zet. Gert moet dus de optie hebben om enkel een bepaalde doelgroep te accepteren. Bv: enkel voor werk.  
        
      **User Story:**  
      **Als potentiële verhuurder wil ik een inbreng hebben in de doelgroep waar mijn huis voor bestemd is omdat ik een veilig gevoel kan hebben over mijn pand.**
   3. Alfred (35) wil altijd geïnformeerd worden over hoe de situatie van bepaalde zaken is. Hij heeft er een hekel aan als hij niet op de hoogte wordt gebracht van iets. Onlangs heeft Alfred zijn pand te huur gesteld voor OneHouseStand en wilt de status van de verwerking van zijn sollicitatie bekijken.  
        
      **User Story:**  
      **Als potentiële verhuurder wil ik de status van mijn aanvraag/ sollicitatie kunnen bekijken omdat ik een idee heb van hoe lang ik nog zal moeten wachten voor ik het oordeel van mijn sollicitatie weet.**
2. Huurder:
   1. Jan (18) gaat binnenkort trouwen en overweegt om bij OneHouseStand een plek te huren. Hij zou echter op voorhand een kijkje willen nemen in het pand zodat hij zeker is van zijn keuze.

**User Story:**  
**Als potentiele huurder wil ik voordat ik mijn keuze maak een kijkje kunnen nemen in het pand zodat ik zeker ben dat ik de juiste keuze zal maken.**

## Technieken

### Observatie

Verhuren van pand niets op aan te merken/naast foto’s die niet onmiddellijk gevraagd worden.

Bij het huren van een pand verwacht je onmiddellijk te kunnen betalen met Paypal. Maar als je op “pand verkrijgen klikt” wordt het pand onmiddellijk in jouw naam geregistreerd en ontvang je onmiddellijk een email je voor van x tot x het pand gehuurd hebt. Er wordt niet naar een bevestiging gevraagd of je zeker bent dat je wilt reserveren. + het is niet duidelijk dat je later pas gaat moeten betalen.

**Als huurder wil ik een bevestiging online voordat er van mij verwacht wordt dat ik betaal voor het pand zodat ik niet voor een onaangename verassing kom te staan.**

**Requirement**: Als huurder wil ik online een bevestiging krijgen voordat er van mij wordt verwacht dat ik betaal.

**Als huurder wil ik online kunnen betalen, zodat ik zoveel mogelijk beschikbare mogelijkheden heb en er de juiste voor mij kan uitkiezen.**

**Requirement**: Als huurder wil ik online kunnen betalen

Vervolgens als je per ongeluk gereserveerd hebt ontvang je een mail met een bevestiging van je bestelling. Maar er is geen enkele manier om de bestelling te annuleren!! OneHouseStand zal dus tijd verliezen om mij te contacteren.

**Als huurder wil een nadat ik een locatie gehuurd heb, mijn reservering kunnen annuleren indien ik me bedenk of een vergissing gemaakt heb, Zodat ik niet onnodig hoef te betalen.**

**Requirement:** Als huurder wil ik mijn reservering kunnen annuleren bij een fout

### Interview (met Glenn als screener)

1. Hoe gaan jullie nu te werk?

Contactpersoon die huis wilt verhuren, bekijkt applicaties.

Beoordeeld op basis v criteria. Kijkt naar min grootte, checkt criteria.

Als ok --> kijkt naar foto’s + beoordeling --> keurt goed indien in orde

Verwittigt secretaris van status om verhuurder op de hoogte te brengen

1. Wie is er allemaal betrokken bij het screenen/bekijken/toevoegen van nieuwe panden?

Screener, secretaris als tussen persoon. Als pand goedgekeurd zal syndicus actie ondernemen om matten te plaatsen. Meet matten

**Als syndicus wil ik op de hoogte gehouden worden van nieuwe panden, zodat ik in deze panden afmetingen kan gaan nemen om door te geven aan Verimpex**

***Afgeleide requirements:***

* + - **Als syndicus wil ik berichten kunnen ontvangen die mij op de hoogte houden**

1. Wie van bovenste personen levert hierbij het meeste werk?

Screener, syndicus minste

1. Waar wordt momenteel het meeste tijd aan besteed?

Communicatie met de klant, veel tijd verloren

Klanten die interesse verliezen --> tijd verloren doordat geen antwoord meer krijgt

**Als screener wil ik dat de verhuurders in het begin een grote inspanning moeten doen om alle info in orde te krijgen en dit niet verspreid is over tijd... zodat de kans kleiner is dat ze later afhaken en tijd verloren gaat.**

***Afgeleide requirements:***

* + **Als verhuurder wil ik alle nodige informatie vanaf de eerste keer kunnen invullen in mijn aanvraag**
  + **Als screener wil ik alle nuttige informatie kunnen terugvinden in de aanvraag**

Fouten in applicaties

Potentiele verhuuders die meermaals hun pand registreren en telkens afgekeurd worden

--> **Als screener wil ik de pand-verhuur-aanvragen filteren op aantal keren geregistreerd, zodat mensen die meer dan 3 keer afgekeurd werden en met hetzelfde adres registreren nu onmiddellijk afgekeurd worden en zo tijd kan sparen.**

INVEST:

**--> Als screener wil ik dat mensen die meer dan 3 keer afgekeurd zijn en opnieuw hetzelfde adres gebruiken nu automatisch afgewezen worden, zodat er tijd bespaard wordt.**

***Afgeleide requirements:***

* + ***Als screener wil ik dat het systeem een persoon die al 3 keer hetzelfde pand heeft ingediend en afgewezen is, automatisch afwijst.***

**--> Als screener wil ik dat mensen die meer dan 3 keer afgekeurd zijn nog altijd kunnen registreren met een ander (bestaand) adress, zodat werkelijke aanvragen nog behandeld kunnen worden.**

***Afgeleide requirements:***

* + **Als screener wil ik dat mensen die automatisch afgewezen worden door het systeem wel een ander adres succesvol kunnen indienen**

In geval van bepaalde huurders die vaak misbruik plegen, of voorwerpen meenemen. Dit moet de verhuurder telkens vergoeden en hij is hier niet tevreden mee. Zou tof zijn om iets te hebben waar we een voor en achter kunnen hebben van de plek. Om zo de klant te kunnen aanklagen.

Of een klachtensysteem voor de verhuurder.

**Als verhuurder wil ik een klachtensysteem waar ik klachten op kan plaatsen van slechte huurders zodat als een huurder te vaak misbruik pleegt , die geblockt kan worden van het systeem.**

***Afgeleide requirements*:**

* + **Als verhuurder wil ik een klacht kunnen indienen tegen een huurder**

1. Zijn er veranderingen die uzelf zou willen zien aan het process, of de werkwijze?

Klachtensysteem,

Manier verloren tijd van opgevende verhuurders te verminderen, manier een meer gestructureerd centraal punt van aanvragen te hebben.

Als via mail afspraak wilt maken neemt dat telkens een aantal heen en weer berichten om een moment te vinden binnen de werkuren.

**Als screener wil ik mijn overzicht van open uren beschikbaar stellen voor mensen die voor potentiele verhuurders zodat zei onmiddellijk ik dag kunnen reserveren op een moment dat ons beiden past en er geen tijd moet verspild worden aan het heen en weer mailen.**

***Afgeleide requirement:***

* + **Als screener wil ik mijn overzicht van open uren beschikbaar stellen voor mensen die voor potentiele verhuurders zodat zei onmiddellijk ik dag kunnen reserveren op een moment dat ons beiden past en er geen tijd moet verspild worden aan het heen en weer mailen.**

Spam, gesprekken lopen dood, moeilijk om afspraak te regelen binnen werkuren

## INVEST

Independant, Negotiable, Valuable, Estimatable, Small, Testable

### Oude versie

* Als verhuurder kan ik een pand voor verhuur aan OneHouseStand voorstellen
* Als screener kan ik een pand aan OneHouseStand toevoegen
* Als syndicus kan ik een pand van de nodige Verimpex-matten voorzien
* Als huurder kan ik een geschikt pand voor mijn evenement zoeken en vastleggen
* Als syndicus kan ik een probleem met de matten (laten) doorgeven aan Verimpex
* Als huurder kan ik na huur het gehuurde pand reviewen
* Als screener kan ik de informatie over het pand aanpassen op basis van de reviews door de huurder
* Een OneHouseStand-ruimte via Start-Up-circus te huur aanbieden
* De matten in de verhuurde ruimtes kunnen bestellen en laten herstellen door Verimpex
* Start-Up-circus-rumtes via OneHouseStand kunnen verhuren

### Nieuwe versie

Als verhuurder kan ik een pand voor verhuur aan OneHouseStand voorstellen   
INVEST

* 1. Workflow Steps
     1. Als verhuurder wil ik de kenmerken van mijn pand kunnen voor verhuur aan OneHouseStand zodat OneHouseStand kan sneller een beslissing kan nemen (adhv de beslissingstabel)

--> Als screener wil ik dat een verhuurder de kenmerken van zijn pand kan invoeren zodat ik sneller een beslissing kan nemen (adhv de beslissingstabel)

* + 1. Als verhuurder wil ik foto’s van mijn pand kunnen meesturen met mijn voorstel aan OneHouseStand zodat zij een beter inzicht hebben op mijn pand

--> Als screener wil ik dat een verhuurder foto’s van zijn pand kan meesturen zodat ik een beter beeld heb van hoe het pand eruit ziet en een beslissing kan nemen.

Als screener kan ik een pand aan OneHouseStand toevoegen   
INVEST

* 1. Als screener kan ik een pand aan OneHouseStand toevoegen zodat klanten een ruimer aanbod hebben om uit te kiezen.

Als syndicus kan ik een pand van de nodige Verimpex-matten voorzien   
INVEST

* 1. Als syndicus wil ik een pand van de nodige verimpex-matten voorzien zodat de vloer beschermd blijft.

Als huurder kan ik een geschikt pand voor mijn evenement zoeken en vastleggen   
INVEST

* 1. Als huurder wil ik een geschikt pand voor mijn evenementen zoeken en vastleggen zodat de juiste locatie kan vinden voor mijn evenement of behoefte.
  2. Workflow Steps:
     1. Als huurder kan ik een geschikt pand voor mijn evenementen zoeken zodat ik zelf kan kiezen welk pand bij mijn behoeften en vereisten hoort.
     2. Als huurder kan ik een gevonden pand vastleggen op de juiste data zodat wanneer ik een pand gevonden heb ik dit kan reserveren en verzekerd ben dat niemand anders dit boekt.

Als syndicus kan ik een probleem met de matten (laten) doorgeven aan Verimpex   
INVEST

* 1. Als syndicus wil ik een probleem met de matten (laten) doorgeven aan Verimpex zodat alle mattenproblemen van Verimpex vermeld worden en zij deze kunnen oplossen

Als huurder kan ik na de huur het gehuurde pand reviewen   
INVEST

* 1. Als huurder wil ik na de huur het gehuurde pand reviewen zodat ik mijn bevindingen kan doorgeven en de verhuurder zich kan verbeteren naar de toekomst toe.

Als screener kan ik de informatie over het pand aanpassen op basis van de reviews door de huurder  
INVEST

* 1. Als screener wil ik de informatie over het pand aanpassen op basis van de reviews door de huurder zodat foutieve informatie bijgewerkt kan worden.

1. Een OneHouseStand-ruimte via Start-Up-circus te huur aanbieden

INVEST

* 1. Als verhuurder wil ik dat een OneHouseStand-ruimte via Start-Up-circus te huur aangeboden kan worden zodat mijn pand een groter doelpubliek bereikt en meer verhuurd kan worden.

1. De matten in de verhuurde ruimtes kunnen bestellen en laten herstellen door Verimpex

INVEST

* 1. Als verhuurder wil ik dat de matten in de verhuurde ruimtes door Verimpex besteld en hersteld kunnen worden door Verimpex zodat ik hier geen tijd aan verlies en mijn panden er altijd mooi uitzien.

Start-Up-circus-ruimtes via OneHouseStand kunnen verhuren  
INVEST

* 1. Als huurde wil ik dat Startup-circus-ruimtes verhuurd kunnen worden via OneHouseStand zodat ik meer keuze heb wanneer ik een pand aan het kiezen ben.

## AcceptatieCriteria

**Als verhuurder kan ik een pand voor verhuur aan OHS voorstellen**

Given De verhuurder er zijn 8 ongekeurde aanvragen

AND Het aanvraagformulier is ingevuld

When verhuurder dient aanvraag in

Then aantal ongekeurde aanvragen is 9

**Als huurder kan ik een geschikt pand voor mijn evenement zoeken en vastleggen**

Given het aantal reservaties vanaf vandaag is gelijk aan 31

When huurder klikt op pand reserveren

Then het aantal resersvaties vanaf vandaag is gelijk aan 32

**Als huurder kan ik na huur het gehuurde pand reviewen**

Given current date is groter dan einddatum van reservatie

AND huurder ontving een mail voor een review

AND het pand heeft 12 reviews

When huurder verzendt een review over het pand

Then het pand heeft 13 reviews

**Als screener kan ik een pand aan OHS toevoegen**

Given er is een aanvraag voor verhuur ingediend

AND de aanvraag is goedgekeurd

AND het aantal panden op OHS = 399

AND aantal niet-afgehandelde aanvragen = 10

When de screener dit pand toevoegd aan OHS

Then het aantal panden op OHS = 400 AND aantal niet-afgehandelde aanvragen = 9

**Als screener kan ik de informatie over het pand aanpassen op basis van de review door de huurder**

Given screener ontvangt nieuwe review

When screener valideert info van review

AND info is correct

THEN screener past info van pand aan adhv review

## Business en informatie use cases

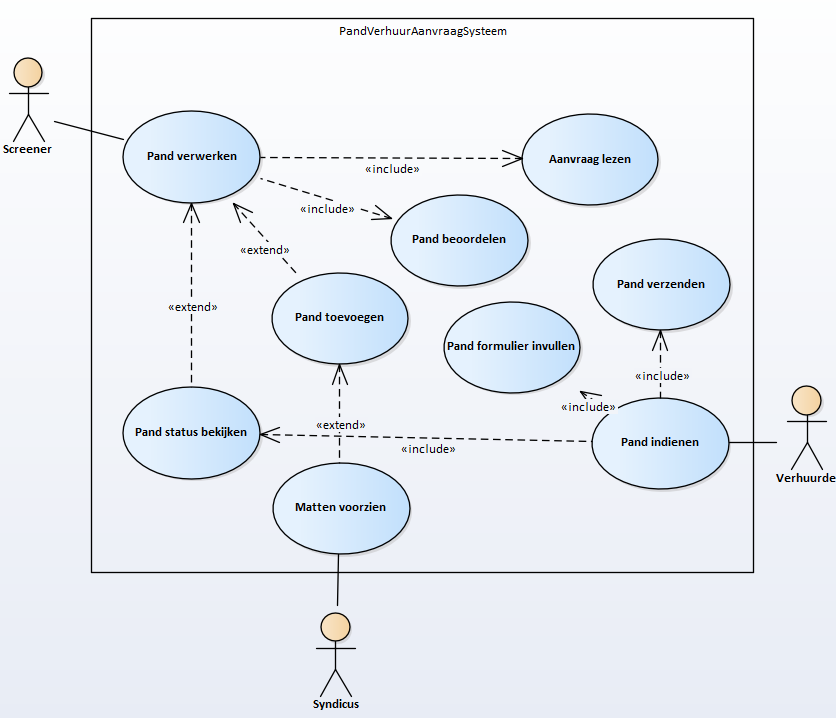
**Actoren**

* Verhuurder
* Screener
* Huurder
* Syndicus

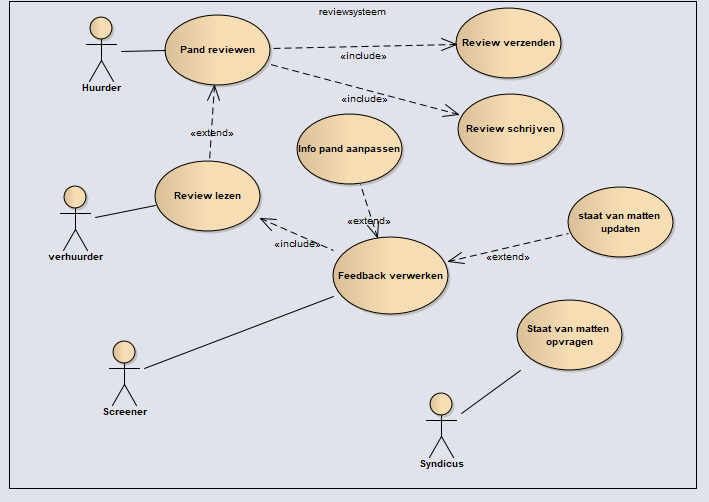
**“2 business use cases °°, 2 informatie use cases”**

### **Use Case Diagrammen**

* Pand reservatie systeem



* Review Systeem

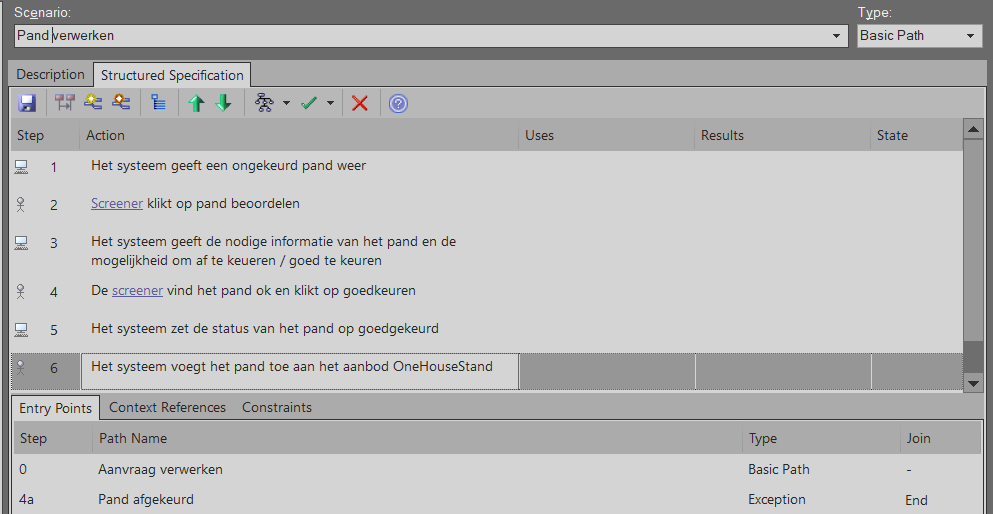


## Use case beschrijvingen

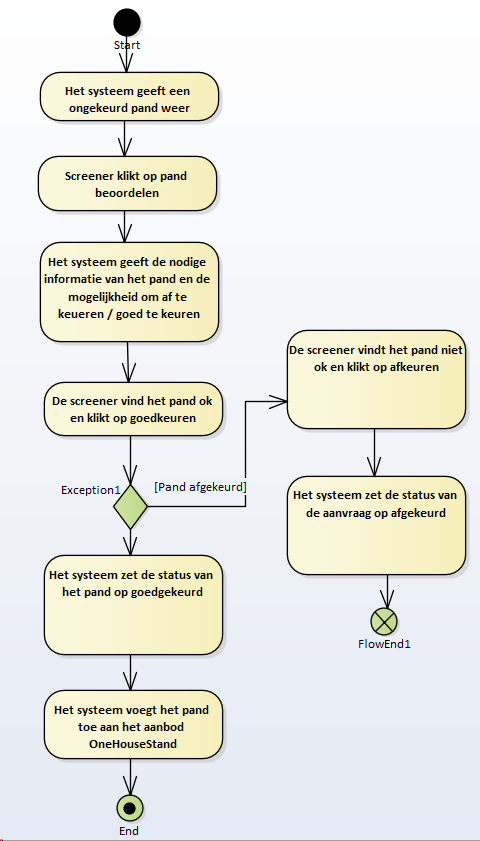
### Business use case: Pand verwerken

|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Pand verwerken | |
| **Versie** | 0.1 | |
| **Primaire actor** | Screener | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - er is een onverwerkt pand | |
| **Eindconditie bij succes** | - pand is goedgekeurd  - pand is toegevoegd | |
| **Eindconditie bij falen** | - pand is afgekeurd | |
| **Hoofdscenario** | stap | actie |
|  | **1.** | Het systeem geeft een ongekeurd pand weer |
|  | **2.** | Screener klikt op pand beoordelen |
|  | **3.** | Het systeem geeft de nodige informatie van het pand en de mogelijkheid om af te keuren / goed te keuren |
|  | **4.** | De screener vind het pand ok en klikt op goedkeuren |
|  | **5.** | Het systeem zet de status van de pand op goedgekeurd |
|  | **6.** | Het systeem voegt het pand toe aan OneHouseStand |
| **Uitbreidingen** | **4.1** | *De screener vindt het pand niet ok en klikt op afkeuren* |
|  | **4.2** | Het systeem zet de status van de pand op afgekeurd |
| **Speciale vereiste** | NVT | |

#### Enterprise architect use case beschrijving



#### Activity diagram



#### Features

**Feature**: Pand verwerken

Een screener kan een pand verwerken

**Scenario A**: Pand is goedgekeurd

* GIVEN: er is een ongekeurd pand
* WHEN screener klikt op goedgekeurd
* THEN het systeem zet de status van het pand op “goedgkeurd”
* AND het systeem voegt het pand toe aan OneHouseStand

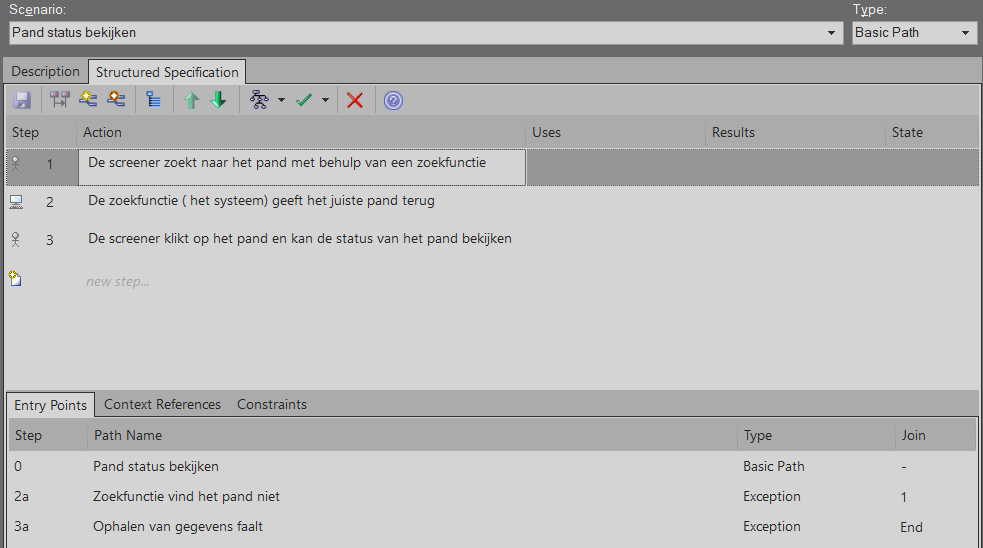
**Scenario B**: Pand is afgekeurd

* GIVEN: er is een ongekeurde pand
* WHEN Screener klikt op afgekeurd
* THEN het systeem zet de status van de pand op “afgekeurd”

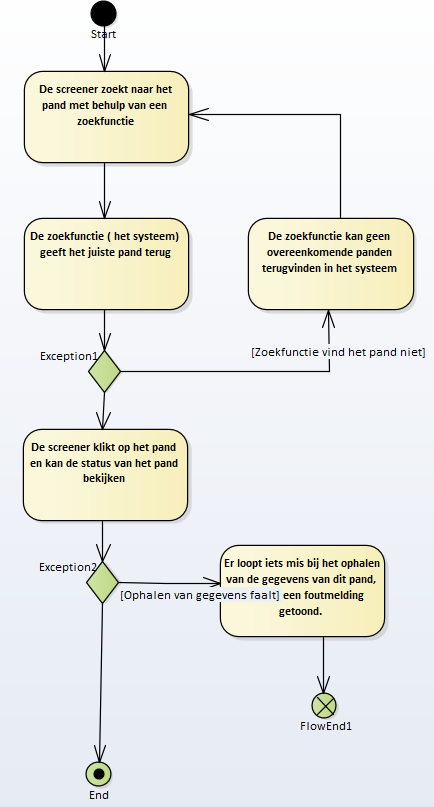
### Information use case: Pand status bekijken

|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Pand status bekijken | |
| **Versie** | 0.1 | |
| **Primaire actor** | screener | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - de screener is de huidige gebruiker van het systeem  - de screener is de huidige gebruiker van het systeem | |
| **Eindconditie bij succes** | - de status van de aanvraag wordt weergegeven | |
| **Eindconditie bij falen** | - de status van het pand kan niet weergegeven worden | |
| **Hoofdscenario** | stap | actie |
|  | **1.** | De screener zoekt naar het pand met behulp van een zoekfunctie  User story gevonden: als screener kan ik een specifiek pand zoeken ahdv een pand id zodat ik sneller mijn werk verrichten  Requirement: een screener kan een pand zoeken adhv een pand id |
|  | **2.** | De zoekfunctie geeft het juiste pand terug |
|  | **3.** | De screener klikt op het pand en kan de status van een aanvraag bekijken |
| **Uitbreidingen** | 2.1 | De zoekfunctie kan geen overeenkomende panden terugvinden in het systeem |
|  | 3.1 | Er loopt iets mis bij het ophalen van de gegevens van dit pand, er wordt een foutmelding getoond. |
| **Speciale vereiste** | NVT | |

#### Implementatie in enterprise architect



#### Activity diagram



#### Features

**Feature**: Pand status bekijken

Een screener kan de status van een aanvraag bekijken, zodat hij weet welke panden nog behandeld moeten worden.

**Scenario**: de status kan succesvol opgevraagd worden  
Given de screener is aangemeld in het systeem

AND bevindt zich op de pagina met de aanvragen

AND in het zoekveld is waarde “1” ingevuld

When de screener klikt op zoeken

Then gegevens van het pand worden weergegeven

**Scenario**: het pand kan niet gevonden worden  
Given de screener is aangemeld in het systeem

AND bevindt zich op de pagina met de aanvragen

AND in het zoekveld is waarde “0” ingevuld

When de screener klikt op zoeken

Then de aanvraag kan niet gevonden worden door het systeem, er wordt een foutmelding getoond.

**Scenario**: er loopt iets fout bij het ophalen van de gegevens van een pand  
Given de screener is aangemeld in het systeem

AND bevindt zich op de pagina met de aanvragen

AND in het zoekveld is waarde “1” ingevuld

AND screener heeft geen internet

When de screener klikt op zoeken

Then er gaat iets mis en de gegevens kunnen niet gevonden worden, er wordt een foutmelding getoond.

### Business use case: Pand reviewen

|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Pand reviewen | |
| **Versie** | 0.1 | |
| **Primaire actor** | huurder | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - huurder heeft het pand gehuurd | |
| **Eindconditie bij succes** | - Review is ingediend | |
| **Eindconditie bij falen** | - Review kon niet ingediend worden | |
| **Hoofdscenario** | stap | actie |
|  | 1. | De huurder drukt op de knop om een review te schrijven |
|  | **2.** | Het systeem geeft een pand review formulier weer |
|  | **3.** | De huurder vult het formulier in |
|  | **3.** | De huurder klikt op verzenden |
|  | **4.** | Het systeem merkt op dat de velden correct ingevuld zijn en slaagt de review op in het OneHouseStand opslagsysteem |
| **Uitbreidingen** | 4.1 | Het systeem merkt op dat alle gegevens niet correct ingevuld zijn en geeft een foutmelding weer. |
| **Speciale vereiste** | NVT | |

**Feature**: Pand reviewen

Een huurder kan na de huur van een pand, het gehuurde pand reviewen zodat hij zijn bevindingen kan doorgeven en de verhuurder zich kan verbeteren naar de toekomst toe.

**Scenario**: review succesvol ingediend  
Given de huurder begeeft zich op het formulier om een review te schrijven

AND heeft alle vragen volledig ingevuld

When huurder klikt op verzenden

Then de review is verzonden naar OneHouseStand

AND de review is opgeslagen

**Scenario**: review niet succesvol ingediend wegens info niet volledig ingevuld  
Given de huurder begeeft zich op het formulier om een review te schrijven

AND heeft niet alle vragen beantwoord

When huurder klikt op verzenden

Then de huurder krijgt een foutmelding dat de gegevens niet correct ingevuld zijn

### Information use case: Staat van matten opvragen

|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Staat van matten opvragen | |
| **Versie** | 0.1 | |
| **Primaire actor** | Syndicus | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - een pand heeft een review gekregen en de review ging over de matten  - de status van de matten is aangepast | |
| **Eindconditie bij succes** | - De syndicus heeft succesvol de statussen van de betroffen matten kunnen zien | |
| **Eindconditie bij falen** | - De link naar de statussen werkte niet  - De statussen konden niet getoond worden | |
| **Hoofdscenario** | stap | actie |
|  | **1.** | Het systeem stuurt de syndicus een melding dat de status aangepast is van een mat van een pand |
|  | **2.** | De syndicus klikt op de link binnen het bericht |
|  | **3.** | Het systeem navigeert naar de link en geeft de statussen weer van de matten van het pand |
|  |  |  |
| **Uitbreidingen** | 3.1 | 3.1 het systeem probeert naar de linkt te navigeren, maar kan de pagina niet vinden. Er wordt een foutmelding weergegeven dat de pagina niet gevonden is. |
| **Speciale vereiste** | NVT | |

**Feature**: Staat van matten opvragen

Een syndicus kan de staat van de matten opvragen van een specifiek pand, zodat hij ze later hersteld kunnen worden indien er een probleem is

**Scenario**: bericht succesvol verstuurd met correcte link  
Given een pand heeft een review gekregen

AND de status van het pand is aangepast

AND het systeem heeft een bericht verstuurd naar de syndicus

AND het bericht bevat een werkende link

When de syndicus klikt op de link

Then de syndicus ziet op het scherm de status van de matten van het pand

**Scenario**: bericht niet succesvol verstuurd met incorrecte link  
Given een pand heeft een review gekregen

AND de status van het pand is aangepast

AND het systeem heeft een bericht verstuurd naar de syndicus

AND het bericht bevat een foutieve link

When de syndicus klikt op de link

Then de syndicus ziet op het scherm “de pagina kon niet gevonden worden”

## Kandidaat-objecten

Voor we kunnen beginnen aan het maken van het klassendiagram moeten we uiteraard eerst weten welke klassen hierop moeten komen. Om dit te kunnen bepalen gaan we op zoek naar kandidaat-objecten. Kandidaat-objecten zijn alle zelfstandige naamwoorden die voorkomen in de analyse van dit project. Voor elk van deze woorden bepalen we of dit relevant is voor het probleem, of het op zichzelf bestaat en gaan we uiteindelijk de woorden die dubbel voorkomen één keer schrappen. Als we dit doen komen we tot het onderstaande resultaat.

|  |  |  |  |
| --- | --- | --- | --- |
| **Kandidaat-objecten** | **Relevant voor probleem? Nodig voor de oplossing?** | **Bestaat het op zichzelf? Heeft eigen attributen/operaties?** | **Dubbels schrappen** |
| Screener (10) | Screener (10) | / | / |
| Huurder (8) | Huurder (8) | / |  |
| Foto's (2) | Foto's (2) | / | / |
| Klant (4) | Klant (4) | Klant (4) | Klant (4) |
| Verhuurder (9) | Verhuurder (9) | / | / |
| Rating (3) | Rating (3) | / | / |
| Panden (4) | Panden (4) | Panden (4) | / |
| Voortzetting (1) | / | / | / |
| Werk (2) | / | / | / |
| Secretaris (14) | / | / | / |
| Secretaresse (3) | / | / | / |
| Afspraak (3) | Afspraak (3) | / | / |
| Hoogte (1) | / | / | / |
| Status (4) | Status (4) | / | / |
| Aanvraag (12) | Aanvraag (12) | Aanvraag (12) | / |
| Sollicitatie (4) | Sollicitatie (4) | Sollicitatie (4) | / |
| Idee (1) | Idee (1) | / | / |
| Oordeel (4) | Oordeel (4) | / | / |
| Beeld (1) | / | / | / |
| Rangorde (2) | / | / | / |
| Verhuurders (3) | Verhuurders (3) | Verhuurders (3) | Verhuurder (3) |
| Agenda (3) | Agenda (3) | / | / |
| Overzicht (2) | / | / | / |
| Screeningen (1) | / | / | / |
| Toegang (1) | / | / | / |
| Dag (1) | / | / | / |
| Huurder (3) | / | / | / |
| Programma (2) | / | / | / |
| Tijd (1) | / | / | / |
| Kwaliteit (2) | / | / | / |
| Huizen (3) | Huizen (3) | Huizen (3) | / |
| Diversiteit (1) | / | / | / |
| Applicatie (3) | / | / | / |
| Programma's (1) | / | / | / |
| Inbreng (2) | / | / | / |
| Doelgroep (1) | Doelgroep (1) | / | / |
| Huis (4) | Huis (4) | Huis (4) | / |
| Gevoel (1) | / | / | / |
| Klant (4) | Klant (4) | Klant (4) | / |
| Klanten (2) | Klanten (2) | Klanten (2) | / |
| Applicatie (3) | Applicatie (3) | Applicatie (3) | / |
| Applicatie process (2) | Applicatie process (2) | / | / |
| Velden (1) | / | / | / |
| Aanvragen (2) | Aanvragen (2) | Aanvragen (2) | / |
| Knoppen (1) | / | / | / |
| Invoering (1) | / | / | / |
| Pand (10) | Pand (10) | Pand (10) | Pand (21) |
| Beheer (1) | / | / | / |
| Reservatie (6) | Reservatie (6) | Reservatie (6) | Reservatie (6) |
| Afbeelding (2) | Afbeelding (2) | / | / |
| Contactinformatie (1) | Contactinformatie (1) | / | / |
| Beslissing (4) | Beslissing (4) | / | / |
| Manager (2) | / | / | / |
| Technologieën (2) | / | / | / |
| Platform (3) | / | / | / |
| Potentiele  Verhuurder (8) | Potentiele  Verhuurder (8) | / | / |
| Email (6) | / | / | / |
| Eindoordeel (4) | Eindoordeel (4) | / | / |
| Opmerking (3) | Opmerking (3) | / | / |
| Aanpassing (3) | Aanpassing (3) | / | / |
| Beslissingstabel (8) | Beslissingstabel (8) | / | / |
| Aanvraagformulier (3) | Aanvraagformulier (3) | Aanvraagformulier (3) | / |
| Afbeeldingen (4) | Afbeeldingen (4) | / | / |
| Systeem (2) | / | / | / |
| Waarde (2) | / | / | / |
| Punten (3) | Punten (3) | / | / |
| Eindresultaat (4) | Eindresultaat (4) | / | / |
| Commentaar (2) | Commentaar (2) | / | / |
| Syndicus (8) | Syndicus (8) | / | / |
| Review (9) | Review (9) | Review (9) | Review (9) |
| Eindscore (4) | Eindscore (4) | / | / |
| Start-up-circus (4) | Start-up-circus (4) | / | / |
| Verimpex (4) | Verimpex (4) | / | / |
| 3e partij (2) | 3e partij (2) | / | / |
| Mat (9) | Mat (9) | / | / |

## Woordenboek

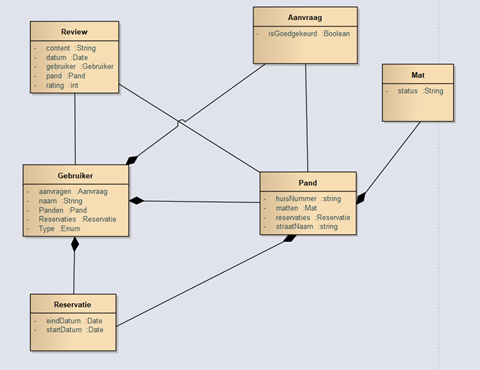
Nu we de kandidaat-objecten hebben bepaald, kunnen me met de woorden die overblijven het woordenboek maken. Dit woordenboek is, zoals de naam al doet vermoeden, een verklaring van de woorden die belangrijk zijn en we zullen gaan gebruiken in de verdere uitwerking van dit project. Het is heel belangrijk dat dit woordenboek goed bijgehouden wordt, en iedereen zich hieraan houdt. Indien we dit niet zouden doen kunnen woorden verkeerd geïnterpreteerd worden. Dit moeten we ten alle kosten vermijden, want dit zou betekenen dat we later zeker fouten zouden gaan maken.

Na feedback werd dit woordenboek nog gewijzigd, omdat er enkele woorden niet gedetailleerd en exact genoeg uitgewerkt waren.

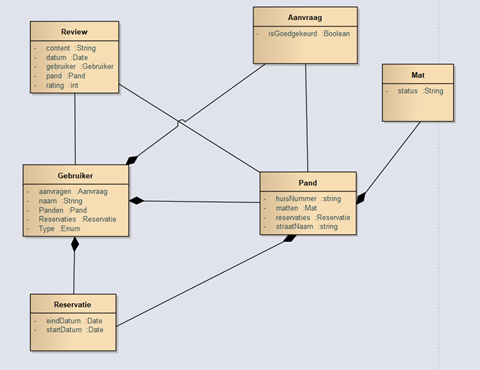
* Klant:
  + Deze kan gesplitst worden in 2 types:
  + Huurder
    - Een huurder kan een pand reserveren voor een bepaalde termijn. Na de huur van het pand kan hij het pand een review geven, dat de screener zou kunnen bekijken, en zo informatie van het pand aanpassen.
  + Verhuurder
    - Een verhuurder kan een pand toevoegen aan het aanbod van OneHouseStand.
    - Dit pand kan vervolgens gescreend worden door een screener. Het zal dan een status “goedgekeurd” of “afgekeurd” krijgen. De goedgekeurde panden zullen zichtbaar zijn op het aanbod van onehousestand.
* Reservatie:
  + Een pand kan gehuurd worden door een huurder, een reservatie zal ontstaan wanneer een huurder een pand huurt.
  + De verhuur van een pand, bevat de startdatum, einddatum, over welk pand het gaat en wie dit pand zal huren met hoeveel personen
* Pand :
  + Een pand kan ingediend worden door een potentiële verhuurder.
  + Een Screener zal over dit pand beslissen en zal bepalen of dit pand toegevoegd zal worden aan het aanbod van OneHouseStand en gehuurd kan worden.
  + Indien een pand goedgekeurd wordt zal de status veranderen naar “goedgekeurd”.
  + De status van een pand kan zijn “ongekeurd, afgekeurd, goedgekeurd”.Goedgekeurde panden zullen zichtbaar zijn in het aanbod van OHS en zullen gehuurt kunnen worden.
* Review:
  + Verhuurder kan na de huur van een pand het gehuurde pand reviewen
  + Dit is gewoon een tekst bericht en zal een screener kunnen bekijken en doorsturen naar een syndicus indien nodig
  + De screener kan de review ook gebruiken om de info van een pand te updaten. (waaronder de status van de mat
* Mat:
  + Een mat behoort tot een pand en heeft een status, wordt gelegd door een syndicus.
  + Als een huurder een pand gehuurd heeft, kan hij dat pand een review geven. Een screener kan de review verwerken, indien er in de review staat dat de matten   
    “versleten” zijn kan de screener de status van de mat aanpassen.
  + Een syndicus kan vervolgens een mat herstellen (of vervangen) en de status veranderen in “goed”

## Klassendiagram

Nu we het woordenboek gemaakt hebben kunnen we beginnen aan het klassendiagram. Er is veel gewerkt aan het klassendiagram, en er zijn veel wijzigingen aan uitgebracht. We beginnen met de eerste versie van het klassendiagram, en werken daarna verder om uiteindelijk tot de finale versie te komen.



Eerste versie van het klassendiagram.



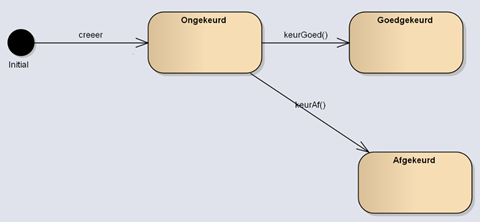
We voegen extra klassen toe, waaronder screener en syndicus.

## Toestandsdiagrammen

Een toestandsdiagram geeft gedrag weer die een object van een klasse kan hebben. Deze zijn belangrijk om later te weten in welke toestand welke operaties uitgevoerd mogen worden, en hoe de transitie verloopt tussen de verschillende toestanden. Na het maken van de toestandsdiagrammen moet er geïtereerd worden. Dit wil zeggen dat we opnieuw het klassendiagram erbij gaan nemen en nu rekening houdend met de toestandsdiagrammen operaties en attributen gaan toevoegen.

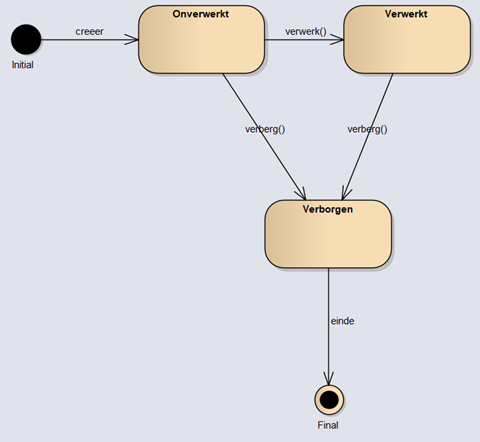
### Aanvraag

Bij het aanmaken van een aanvraag is de status “ongekeurd”. De screener zal deze keuren, en ofwel goedkeuren of afkeuren. Afhankelijk hiervan zal de status aangepast worden naar “goedgekeurd” of afgekeurd”.



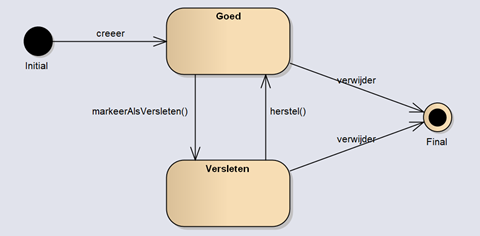
### Review

Bij de creatie van een review is de status ervan “onverwerkt”. De screener kan een review verwerken, dit wil zeggen dat hij de review zal lezen en aan de hand hiervan info over het pand kan aanpassen. Als dit gebeurt wordt de status “verwerkt”. Een Review kan ook verborgen worden, bijvoorbeeld als inhoud ervan incorrect is. Een review kan enkel verwijderd worden vanuit de “verborgen” status.



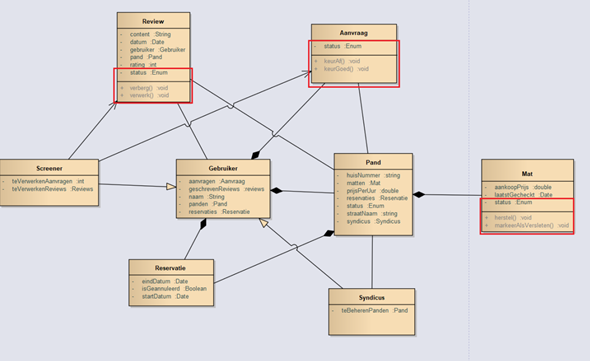
### Mat

Bij de creatie van een mat is de status “goed”. Aan de hand van reviews kan dan de mat gemarkeerd worden als “versleten”. Een mat kan hersteld worden, dit zorgt ervoor dat de status van “versleten” naar “goed” wijzigt. Vanuit beide toestanden kan dit object verwijderd worden.



### Iteratie

Als we na het maken van de toestandsdiagrammen nog eens terugkijken naar het klassendiagram kunnen we opmerken dat er verschillende attributen en operaties nog aangevuld moeten worden. Dit doen we nu. Het gaat voornamelijk over het attribuut status en de operaties die dienen om de status te wijzigen.



## Feedback medestudenten

Bij deze opdracht werd ons twee partners toegewezen. Deze zijn StartupCircus en Verimpex. We hebben ons klassendiagram aan hen voorgesteld, en gevraagd om feedback. Deze hebben we ook gekregen en kan u hieronder lezen. We hebben de feedback ontvangen van: Jonas (StartupCircus) en Mattias, Tibo (Verimpex).

“Reservatie: moet andere relatie krijgen, zoals review.“

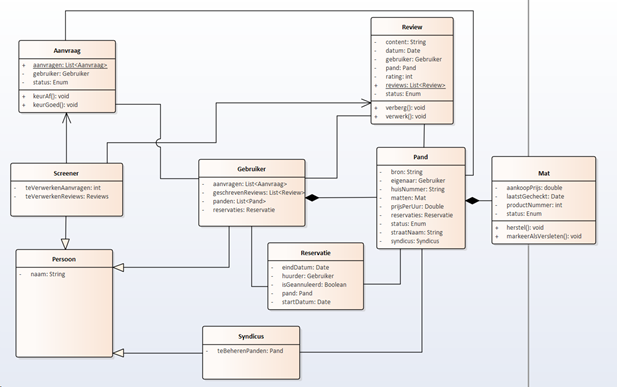
“Jullie hebben geen manier om alle aanvragen en alle reviews bij te houden. Jullie missen aanvragen en reviews.”

“Een pand zou een bron moeten hebben, zo kunnen we onderscheid houden tussen de panden van Startup Circus en OneHouseStand.”

“Mat moet in dagelijkse leven ook id hebben.”

“Een screener kan zelf verhuuraanvraag indienen en zijn eigen verhuuraanvraag goedkeuren.”

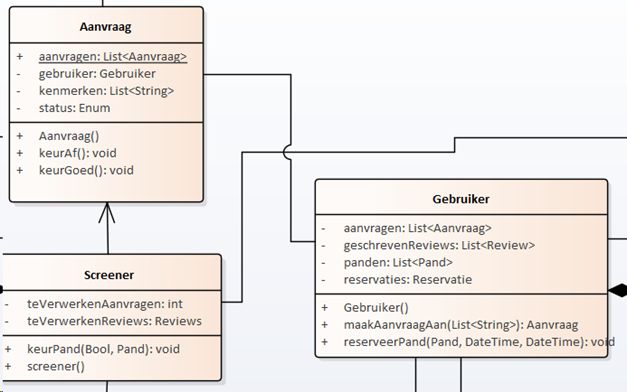
Met hun feedback zijn we gekomen tot het volgende klassendiagram:



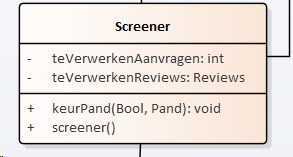
## Validatie klassen diagram door user stories + use case beschrijving

Als screener wil ik dat een verhuurder de kenmerken van zijn pand kan invoeren zodat ik een beslissing kan nemen.

Als screener wil ik dat een verhuurder foto’s van zijn pand kan meesturen zodat ik een beter beeld heb van hoe het pand eruit ziet en een beslissing kan nemen.

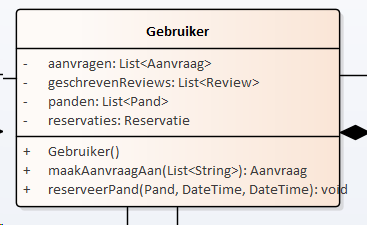


Als screener kan ik een pand aan OneHouseStand toevoegen zodat klanten een ruimer aanbod hebben om uit te kiezen.

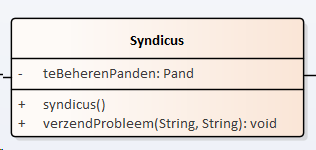


Als huurder kan ik een geschikt pand voor mijn evenementen zoeken zodat ik zelf kan kiezen welk pand bij mijn behoeften en vereisten hoort.

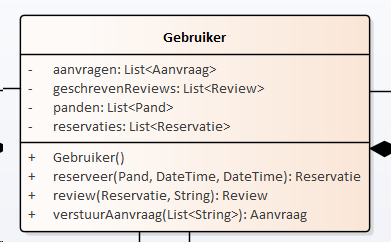
Als huurder kan ik een gevonden pand vastleggen op de juiste data zodat wanneer ik een pand gevonden heb ik dit kan reserveren en verzekerd ben dat niemand anders dit boekt.



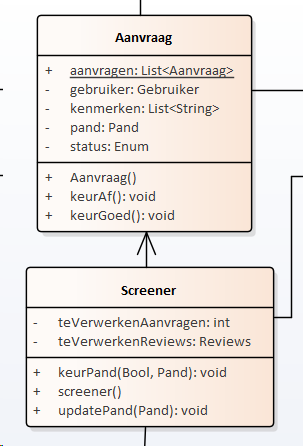
Als syndicus wil ik een probleem met de matten (laten) doorgeven aan Verimpex zodat alle mattenproblemen van Verimpex vermeld worden en zij deze kunnen oplossen



Als huurder wil ik na de huur het gehuurde pand reviewen zodat ik mijn bevindingen kan doorgeven en de verhuurder zich kan verbeteren naar de toekomst toe.

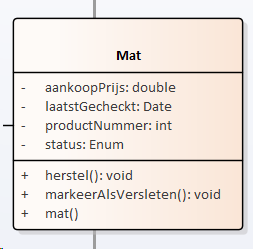


Als screener wil ik de informatie over het pand aanpassen op basis van de reviews door de huurder zodat foutieve informatie bijgewerkt kan worden.

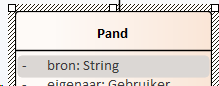


Als verhuurder wil ik dat een OneHouseStand-ruimte via Start-Up-circus te huur aangeboden kan worden zodat mijn pand een groter doelpubliek bereikt en meer verhuurd kan worden. API

Als verhuurder wil ik dat de matten in de verhuurde ruimtes door Verimpex besteld en hersteld kunnen worden door Verimpex zodat ik hier geen tijd aan verlies en mijn panden er altijd mooi uitzien.

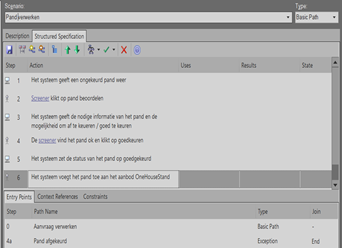
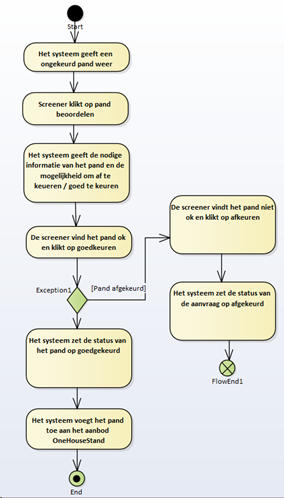


Als huurder wil ik dat Startup-circus-ruimtes verhuurd kunnen worden via OneHouseStand zodat ik meer keuze heb wanneer ik een pand aan het kiezen ben.



## Use Cases

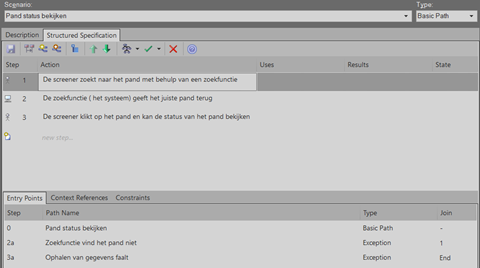
Om een specifiek proces beter uit te werken hebben we ook vier use case diagrams en beschrijvingen gemaakt. Eentje voor het verwerken van een aanvraag tot het verhuren van een pand, voor de status te kunnen bekijken van een aanvraag, voor het reviewen van een pand en ten slotte om te status van de matten te kunnen bekijken.



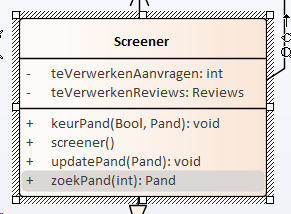
|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Aanvraag verwerken | |
| **Versie** | 0.1 | |
| **Primaire actor** | screener | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - er is een onverwerkte aanvraag | |
| **Eindconditie bij succes** | - pand is goedgekeurd  - pand is toegevoegd | |
| **Eindconditie bij falen** | - pand is afgekeurd | |
| **Hoofdscenario** | stap | actie |
|  | **1.** | Het systeem geeft een ongekeurde aanvraag weer |
|  | **2.** | Screener klikt op aanvraag beoordelen |
|  | **3.** | Het systeem geeft de nodige informatie van het pand en de mogelijkheid om af te keuren / goed te keuren |
|  | **4.** | De screener vind het pand ok en klikt op goedkeuren |
|  | **5.** | Het systeem zet de status van de aanvraag op goedgekeurd |
|  | **6.** | Het systeem voegt het pand toe aan OneHouseStand |
| **Uitbreidingen** | **4.1** | *De screener vindt het pand niet ok en klikt op afkeuren* |
|  | **5.1** | Het systeem zet de status van de aanvraag op afgekeurd |
| **Speciale vereiste** | NVT |  |

|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Aanvraag status bekijken | |
| **Versie** | 0.1 | |
| **Primaire actor** | screener | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - de screener is de huidige gebruiker van het systeem  - de screener is de huidige gebruiker van het systeem | |
| **Eindconditie bij succes** | - de status van de aanvraag wordt weergegeven | |
| **Eindconditie bij falen** | - de status van het pand kan niet weergegeven worden | |
| **Hoofdscenario** | stap | actie |
|  | **1.** | De screener zoekt naar het pand met behulp van een zoekfunctie  User story gevonden: als screener kan ik een specifiek pand zoeken ahdv een pand id zodat ik sneller mijn werk verrichten  Requirement: een screener kan een pand zoeken adhv een pand idc |
|  | **2.** | De zoekfunctie geeft het juiste pand terug |
|  | **3.** | De screener klikt op het pand en kan de status van een aanvraag bekijken |
| **Uitbreidingen** | 2.1 | De zoekfunctie kan geen overeenkomende panden terugvinden in het systeem |
|  | 3.1 | Er loopt iets mis bij het ophalen van de gegevens van dit pand, er wordt een foutmelding getoond. |
| **Speciale vereiste** | NVT |  |

Use case beschrijving in EA:



Hierbij hebben we ook ontdekt dat screener een zoekpand methode nodig heeft, al zullen we later merken dat deze klasse overbodig blijkt te zijn.

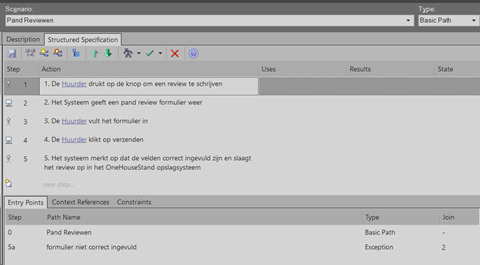


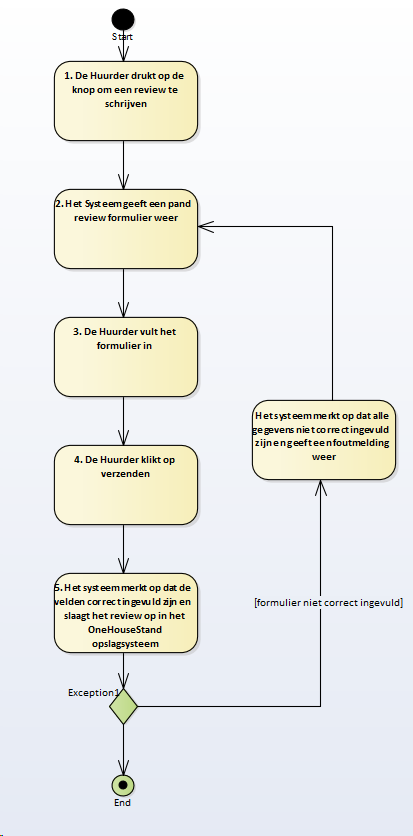
**Use case beschrijvingen**

(Schrijven)

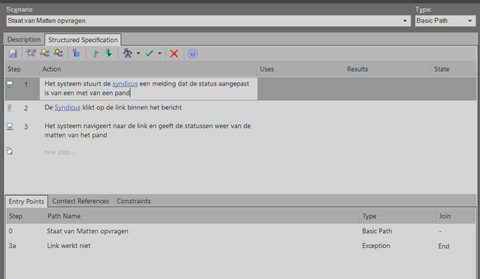
|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Pand reviewen | |
| **Versie** | 0.1 | |
| **Primaire actor** | huurder | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - huurder heeft het pand gehuurd | |
| **Eindconditie bij succes** | - Review is ingediend | |
| **Eindconditie bij falen** | - Review kon niet ingediend worden | |
| **Hoofdscenario** | stap | actie |
|  | 1. | De huurder drukt op de knop om een review te schrijven |
|  | **2.** | Het systeem geeft een pand review formulier weer |
|  | **3.** | De huurder vult het formulier in |
|  | **4.** | De huurder klikt op verzenden |
|  | **5.** | Het systeem merkt op dat de velden correct ingevuld zijn en slaagt de review op in het OneHouseStand opslagsysteem |
| **Uitbreidingen** | 5.1 | Het systeem merkt op dat alle gegevens niet correct ingevuld zijn en geeft een foutmelding weer. |
| **Speciale vereiste** | NVT |  |

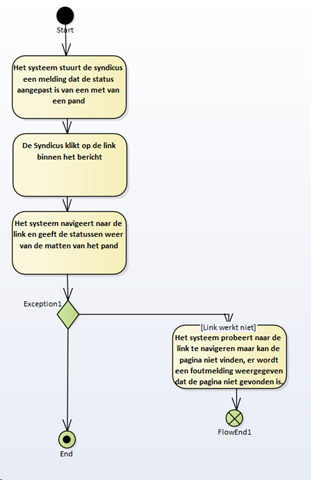
Use case beschrijving uitgewerkt in EA:



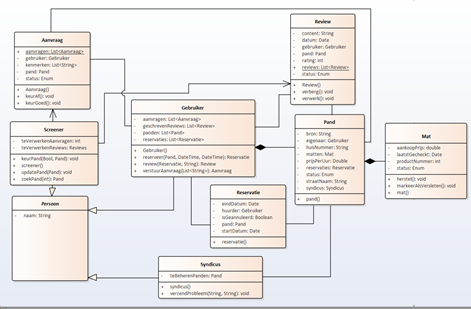


|  |  |  |
| --- | --- | --- |
| **USE CASE titel** | Staat van matten opvragen | |
| **Versie** | 0.1 | |
| **Primaire actor** | Syndicus | |
| **Secundaire actoren** | NVT | |
| **Preconditite** | - een pand heeft een review gekregen en de review ging over de matten  - de status van de matten is aangepast | |
| **Eindconditie bij succes** | - De syndicus heeft succesvol de statussen van de betroffen matten kunnen zien | |
| **Eindconditie bij falen** | - De link naar de statussen werkte niet  - De statussen konden niet getoond worden | |
| **Hoofdscenario** | stap | actie |
|  | **1.** | Het systeem stuurt de syndicus een melding dat de status aangepast is van een mat van een pand |
|  | **2.** | De syndicus klikt op de link binnen het bericht |
|  | **3.** | Het systeem navigeert naar de link en geeft de statussen weer van de matten van het pand |
|  |  |  |
| **Uitbreidingen** | 3.1 | 3.1 het systeem probeert naar de linkt te navigeren, maar kan de pagina niet vinden. Er wordt een foutmelding weergegeven dat de pagina niet gevonden is. |
| **Speciale vereiste** | NVT |  |



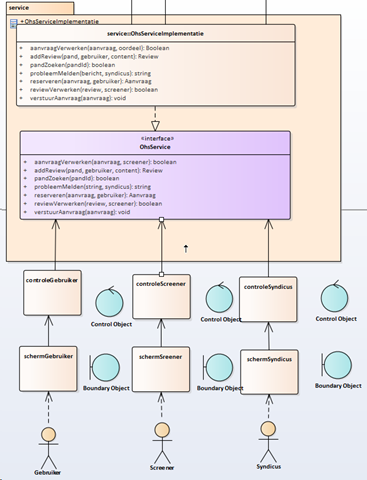


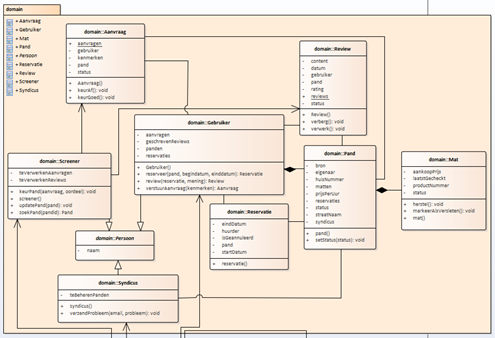
Dit klassendiagram is het resultaat van deze uitgewerkte use cases.

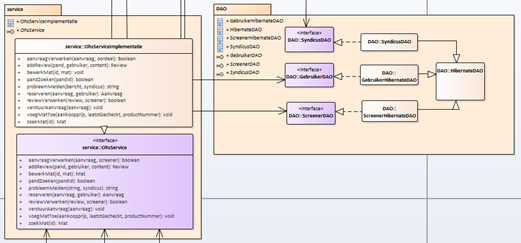


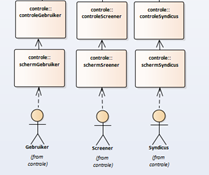
## Boundary klassen

Om ons klassendiagram te vervolledigen voegen we hier de boundary klassen aan toe. Dit geeft weer hoe de communicatie verloopt van aan de user interface tot aan de domeinklassen.



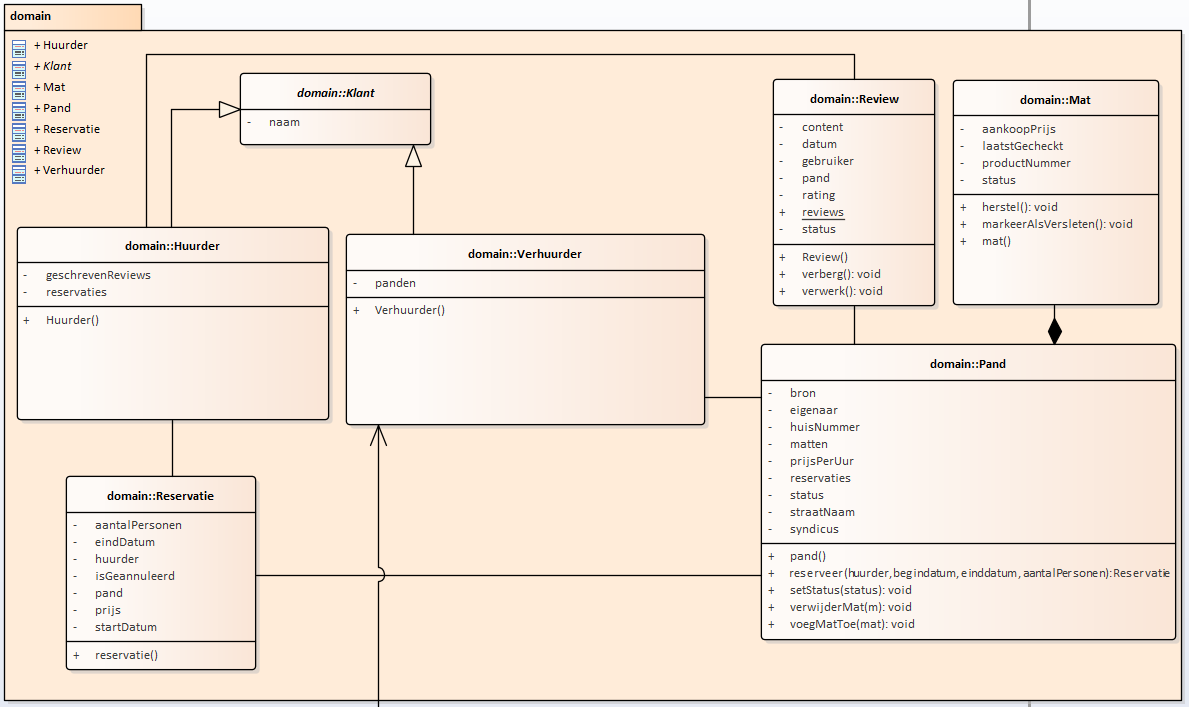






Klassen diagram na feedback Hans

Zoals je kan zien waren er oorspronkelijk heel wat klassen bijgekomen die eigenlijk helemaal niet nodig zijn. Dit hebben we ingezien dankzij de informatie die meneer Vandenbogaerde ons vertelde.



Er zijn een aantal grondige aanpassingen gebeurd hierdoor;

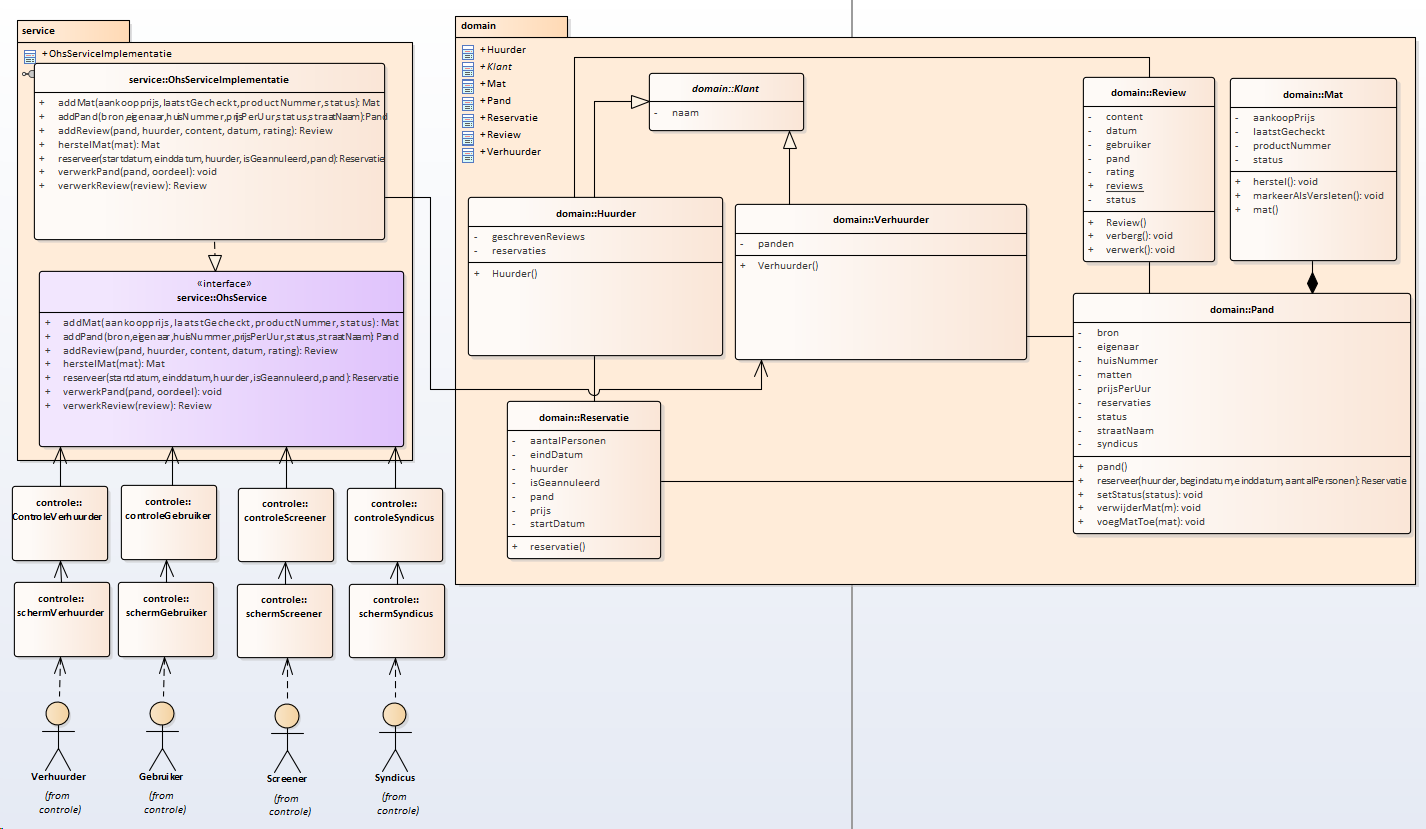
* Klasse **Aanvraag**: deze klasse was eigenlijk wat verwarrend en kon gemerged worden met Pand.

Dit zorgt ervoor dat een verhuurder een pand kan indienen, en een personeelslid van OHS dit pand kan screenen. Het pand zal dus nu een status krijgen en niet meer Aanvraag. Opties: “Ongekeurd, Goedgekeurd, Afgekeurd”. Enkel de goedgekeurde panden zullen zichtbaar zijn in het aanbad van OHS.

* Klassen **Screener** + **Syndicus:** oorspronkelijk bestonden deze klassen doordat we deze 2 rollen hadden, maar later werd duidelijk dat een klasse eigenlijk pas nodig is wanneer je er info van wilt bijhouden. Van syndicus en screener houden we niet echt iets bij, van daar dat we ze verwijderd hebben.
* **Gebruiker:** deze klasse had niet echt de beste naam. Deze klasse hield eigenlijk info bij van de “**Huurder**” rol en de “**Verhuuder**” rol. Dit hebben we gesplitst in de klassen “Huurder” en “Verhuurder”.

**Nieuwe Service Klasse:**

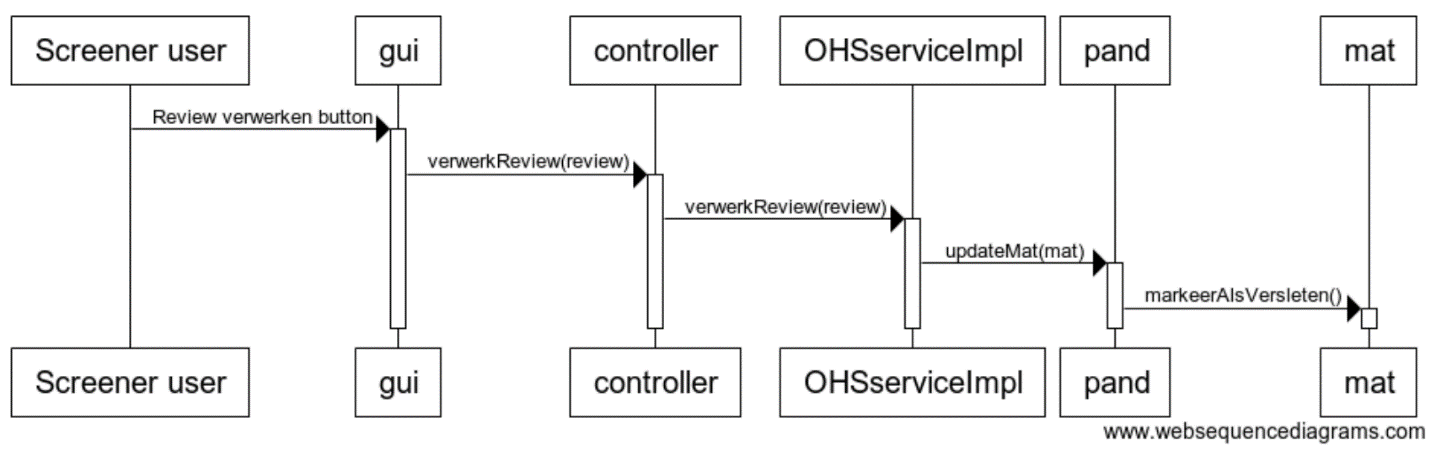
Uit de aanpassingen in het domain model volgden ook kleine aanpassingen in de service klasse.



Hieruit volgt voornamelijk dat er een apart scherm zal zijn voor verhuurder en huurder rollen. (deze waren oorspronkelijk in 1 controler).

**Sequentie diagrammen:**

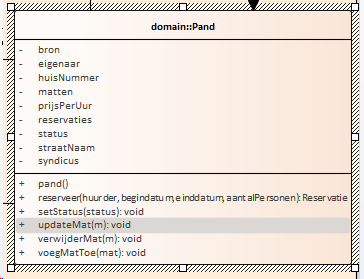
**Onderwerp: Een review verwerken.**

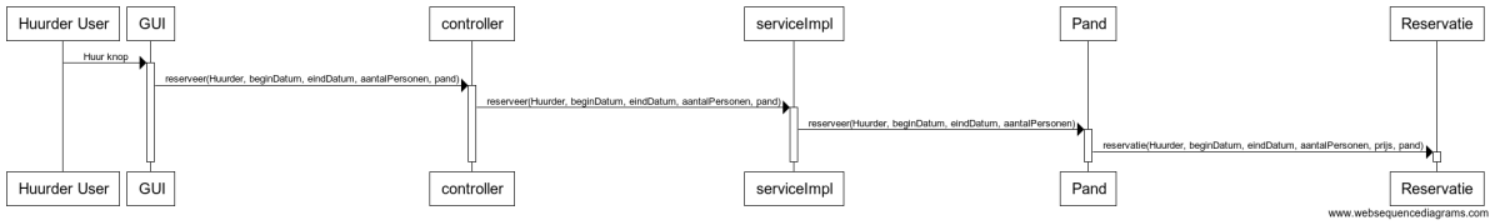


Een screener user kan een review bekijken, als hij in de review leest dat de matten verouderd zijn, kan hij de matten markeren als verouderd. Dit zal updateMat(mat) oproepen op pand, wat vervolgens de mat zal markeren als versleten.

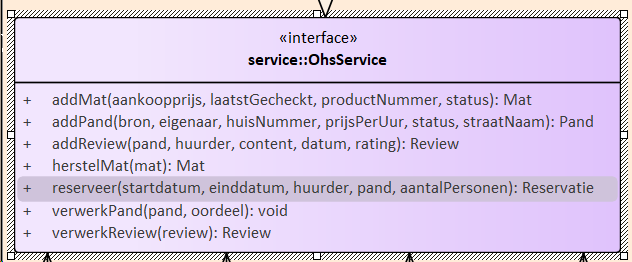
Dit sequentie diagram maakte duidelijk dat er een methode ontbrak binnen om de mat te updaten. “updateMat”.

Nieuwe methode:



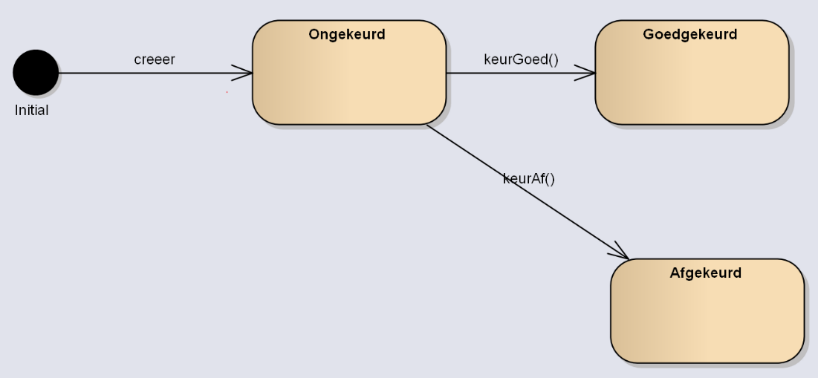
**Onderwerp: Een verhuur aanvraag indienen.**

Aanpassingen op service klasse:

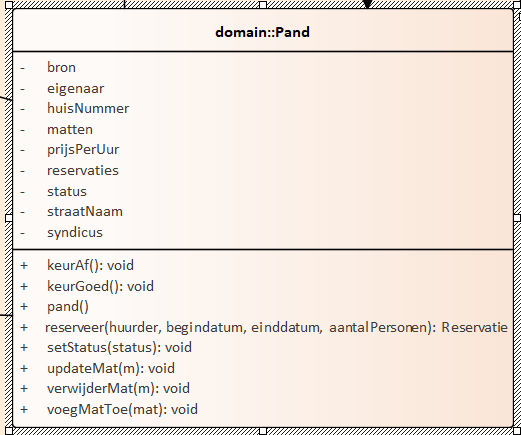


**Toestandsdiagrammen:**

Door de aanpassingen op het klassendiagram werden deze ook herbekeken.

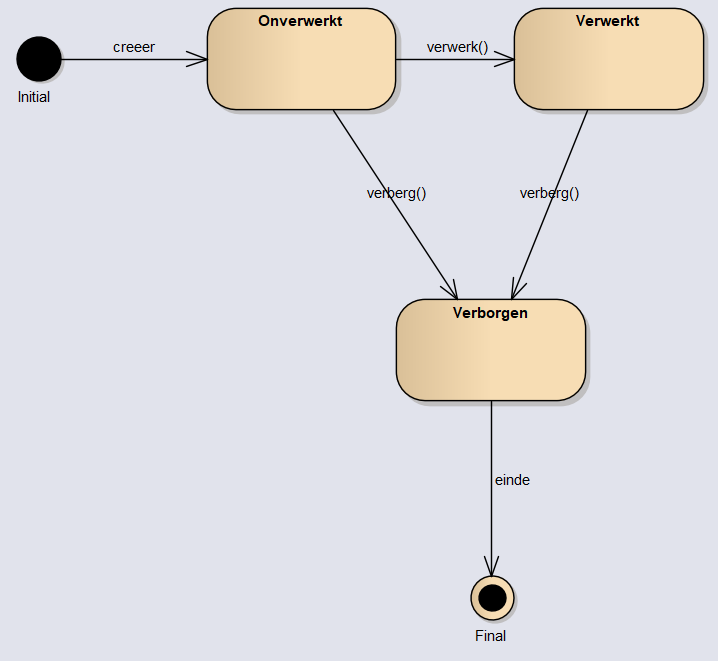
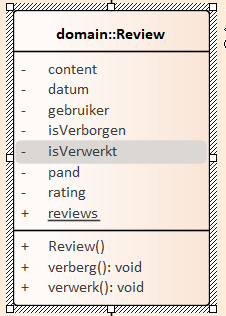
Pand:

Een pand had oorspronkelijk wel de methode “setStatus”, maar welk het had methoden “keurGoed” en “keurAf”. Dit is toegevoegd aan de klasse Pand.

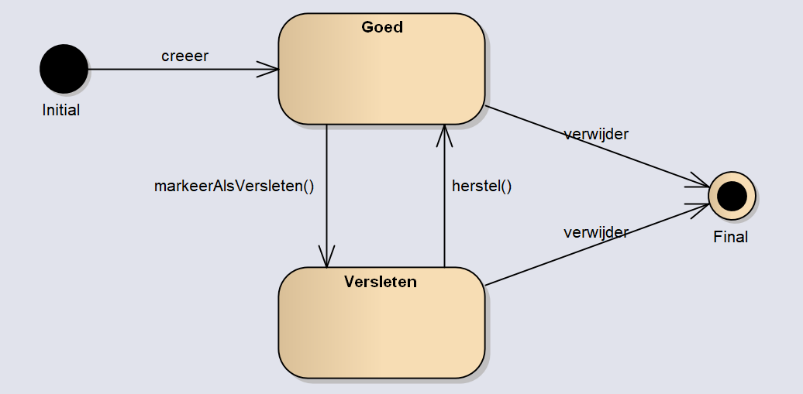


Review:

Klasse review kreeg de Booleans “IsVerborgen” en Booleans “IsVerwerkt”.



Mat:



Deze zorgde voor geen aanpassingen op het klasse diagram.

# Deel 2: zie project dependency injection

# Deel 3

**Identificeren van het design doel:**

* **Usability**
  + User error protection

Dit zouden we bereiken door bij systeem fouten een foutmelding weer te geven. De fouten zouden opgeslagen worden en doorgestuurd naar OneHouseStand zodat zei er rekening mee kunnen houden, en zo bugs te fixen.

* + Learnability

Dit zal gebeuren door duidelijk weer te geven wat je waar moet doen, en zo veel mogelijk toeters en bellen te vermijden.

* **Security**
  + Authenticity

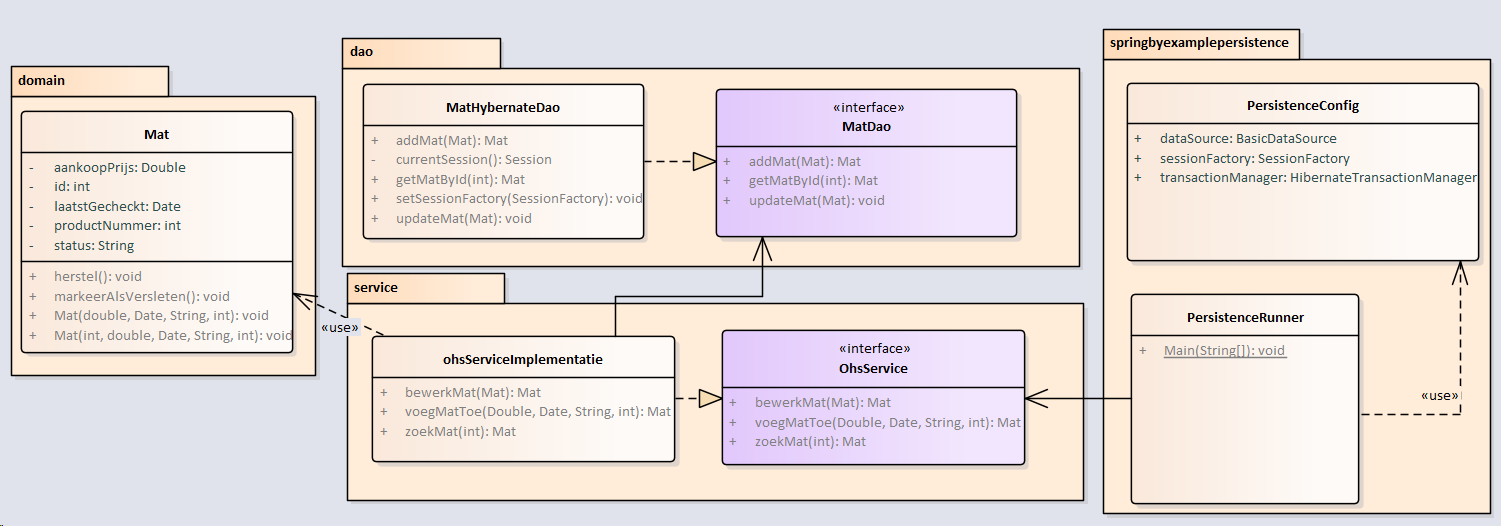
Een gebruiker zal zich moeten inloggen om acties te kunnen uitvoeren op het systeem.

* **Maintabability:**
  + Testability

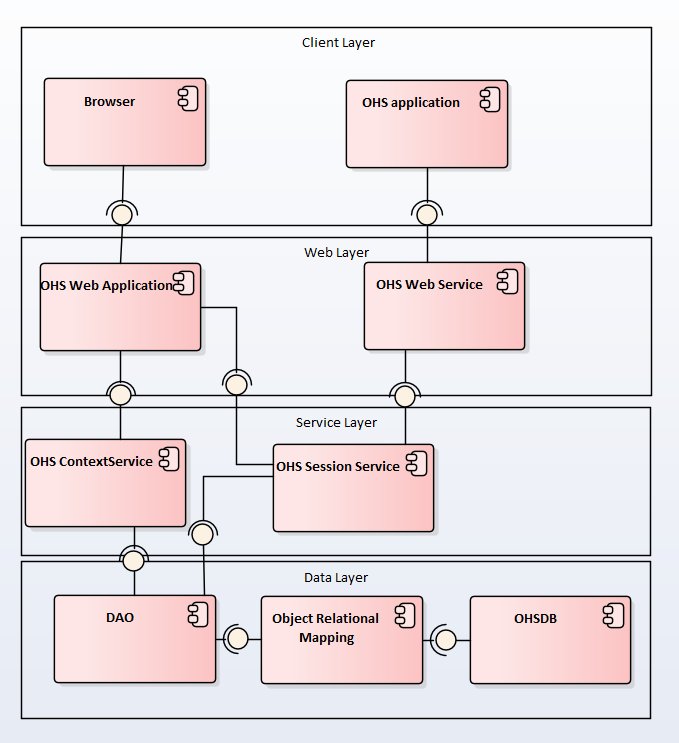
Dit doormiddel van TDD, acceptatie testen, unit testen.

**Uml package diagram**

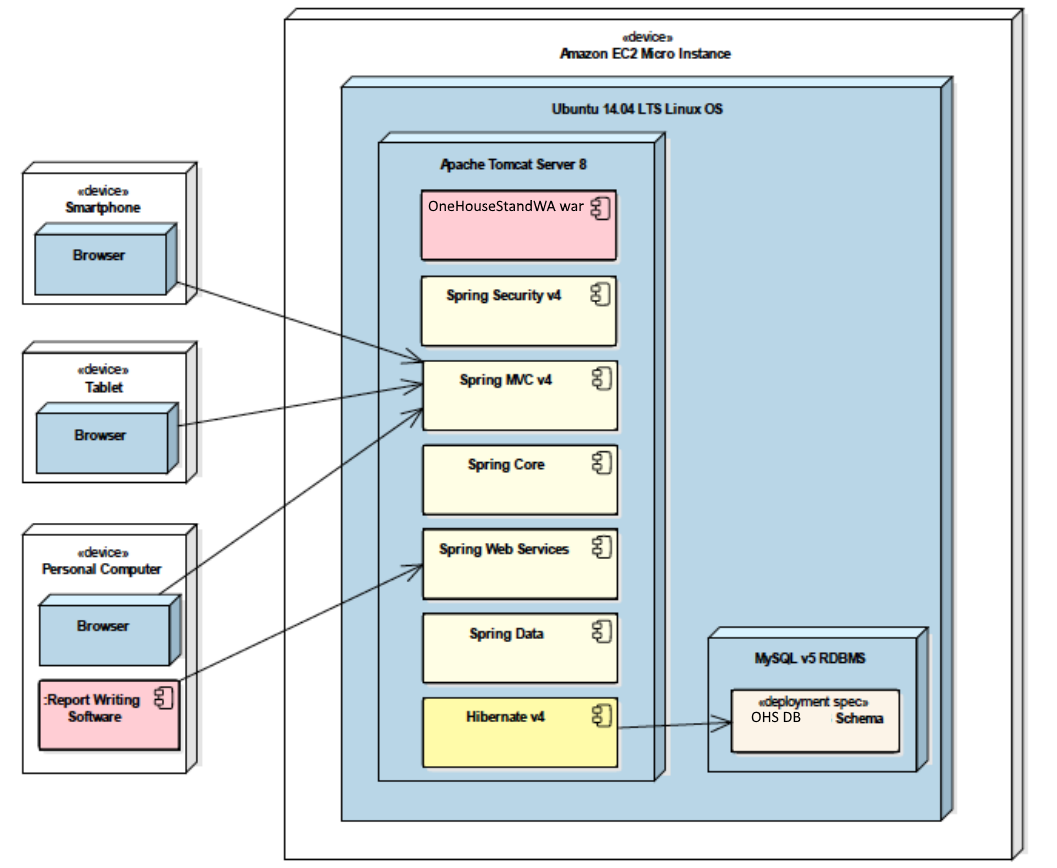
Dit onderdeel hebben we “verfijnd” door 1 onderdeel van het oorspronkelijk klassendiagram uit te werken. Het systeem zal toelaten een Mat toe te voegen, verwijderen en te bewerken.



**Code is te vinden in project “Persistentie”**

**Component diagram:**

Overzicht van wat de verschillende lagen zullen zijn. De klanten van OHS zullen werken het systeem kunnen gebruiken vanaf een Browser, het beheer van het systeem zal gebeuren via een desktop applicatie. Dit zal voornamelijk de screener en syndicus zijn.



**Uml Deployment Diagram**

**Transacties:**

1. Reserveren
   * Want we hebben een lock nodig hiervoor dat 1 pand niet 2x op het zelfde moment gereserveerd kan worden.

**Isolatie:**

1. Reserveren:
   * Zodat maar 1 pand op het zelfde moment gereserveerd kan worden.
   * Dit zou een **Repeatable read** zijn want dit **vermijdt dirty reads + repeatabale reads**

Geen fix op phantom reads, maar dat is geen probleem doordat je reserveert maar 1 element op zelfde moment zou reserveren. En zo hoeft er geen lock op heel de tabel bestaan.

Beveiliging:

* RBAC (reader, writer, admin)
  + De huurder, verhuurder en syndicus rollen zullen writers zijn.
  + De screener rol zal admin zijn.

Globale control flow:

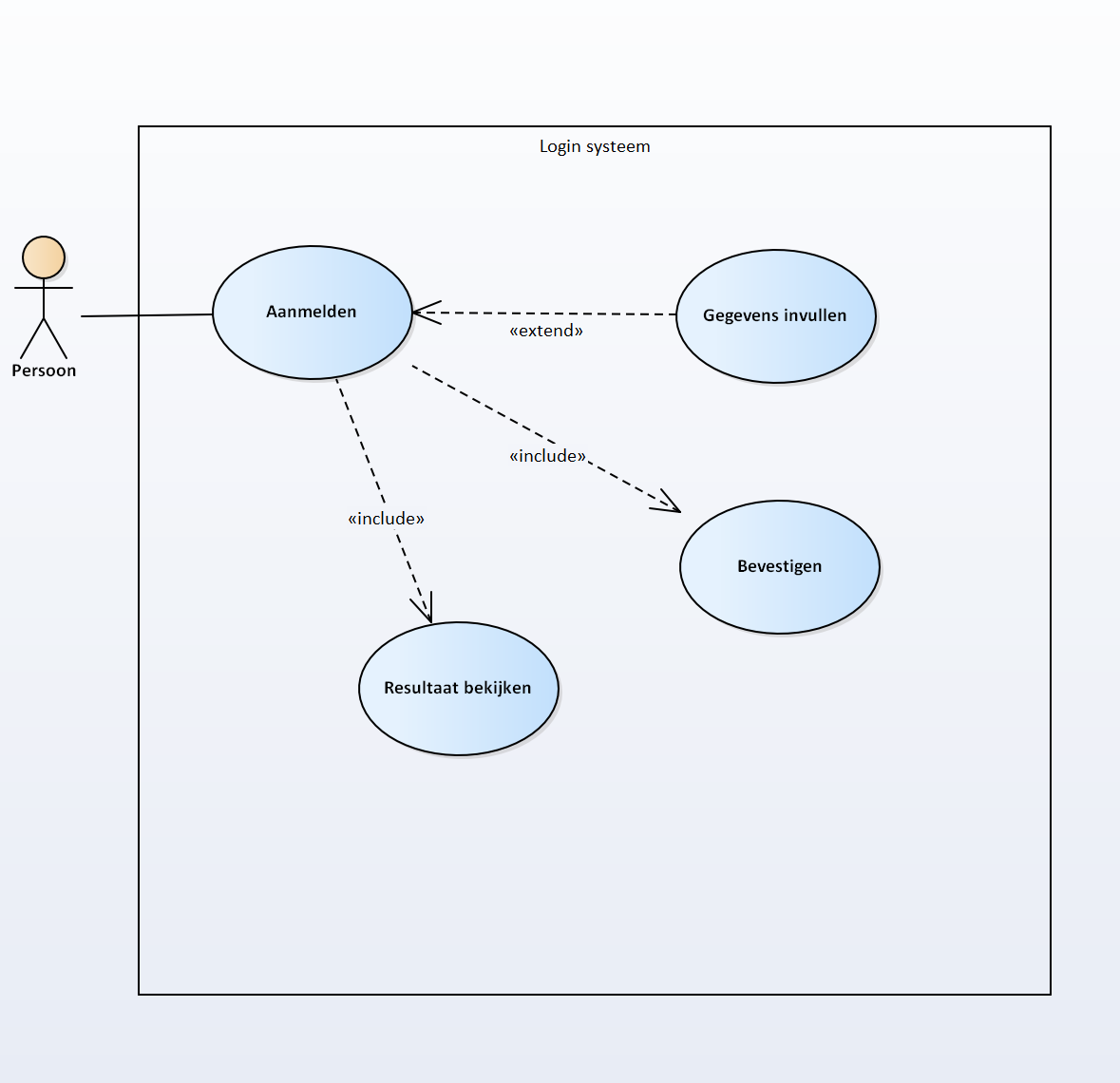
* Eventdriven:

Het systeem wacht op een input van de gebruiker en antwoord hier dan op.

Een voorbeeld binnen het systeem zal het toevoegen van een pand zijn, wanneer de verhuurder op “bevestig” klikt zal het systeem het pand verzenden.

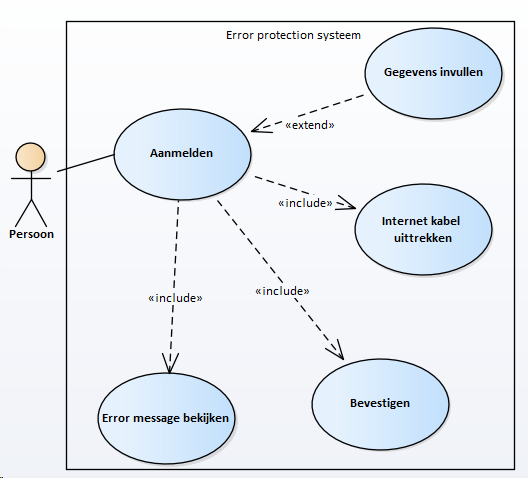
**Technische use cases:**

Login:



|  |  |  |
| --- | --- | --- |
| **Use Case titel** | Login | |
| **Versie** | 0.1 | |
| **Primarie actor** | Persoon | |
| **Secundaire actoren** | NVT | |
| **preconditie** | De persoon staat op de login pagina | |
| **Eindconditie bij succes** | De persoon is ingelogt en bevindt zich op het startscherm dat hen toebehoord | |
| **Eindconditie bij falen** | Persoon kon niet inloggen (foutmelding) | |
| **Hoofdscenario** | Stap | actie |
|  | 1. | De Persoon vult de login gegevens in |
|  | 2. | De Persoon klikt op login |
|  | 3. | Het Systeem ziet dat de gegevens correct zijn, en brengt de persoon naar zijn startscherm |
| **Uitbreidingen** | 1.1 | De Persoon vult de gegevens niet in |
|  | 3.1 | Het Systeem ziet dat de gegevens niet correct zijn en geeft een foutmelding weer |
| **Speciale vereiste** | NVT |  |

Error protection:



Wanneer een fout ontstaat zal een foutmelding weergegeven worden en zal het systeem de fout verzenden naar OHS. Indien er geen internet verbinding is (zoals hier het geval) zal de fout lokaal bijgehouden worden en later verzonden wanneer mogelijk.

**Design Pattern**

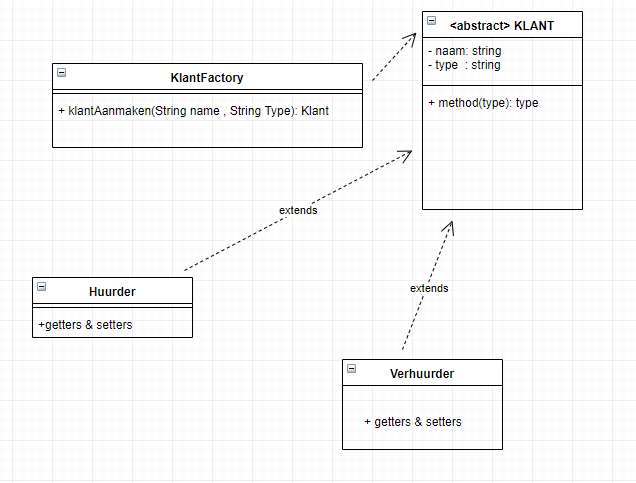
Eerste gekozen pattern is **Factory Design Pattern.**

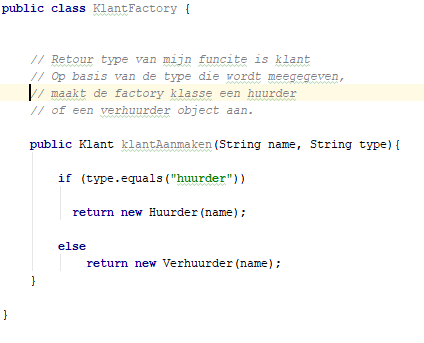
***Waarom ?***

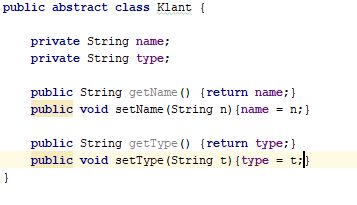
Zoals t e zien is hieronder hebben we twee klassen die dezelfde “superKlasse” hebben.

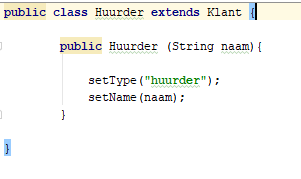
Namelijk , **Huurder** en **Verhuurder die** allebei linkt hebben met superKlasse **Klant.**

Dankzij de KlantFactoryKasse kunnen wij dus objecten instantieren zonder exact vastleggen van welke type ze zullen zijn. Dat is het principe van Factory Pattern.









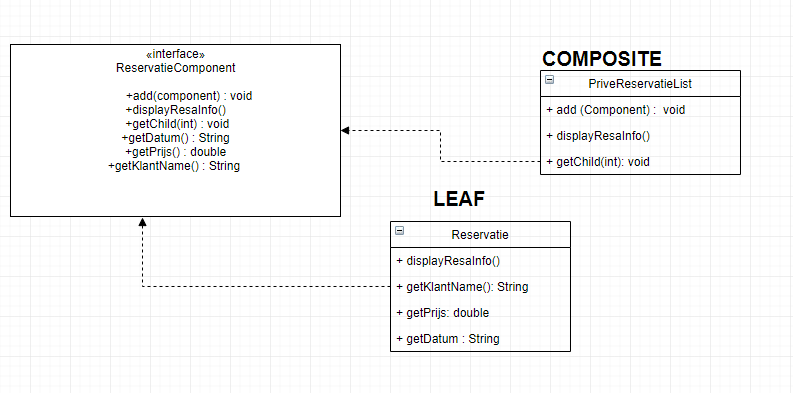
Als 2nd pattern gaan wij een kijkje nemen op de **COMPOSITE PATTERN**.

Om principe van deze pattern op onze case te kunnen toepassen moeten wij eerste een kleine uitbreiding bedenken want volgens ons klassendiagram maken wij geen verschil tussen reservaties.

Als de klant een reservatie maakt, gaat het systeem gewoon samenhagende gegevens bewaren.

Als wij verder kijken zouden wij bijvoorbeeld “privé” evenementen apart kunnen bjihouden. Composite pattern kan hier in actie komen.

Illustratie :



**Composite** : verzameling van reservaties voor privé evenementen

**Leaf** : een reservatie (eenheid)

This enables clients to work through the Component interface to treat Leaf and Composite objects uniformly: Leaf objects perform a request directly, and Composite objects forward the request to their child components recursively downwards the tree structure. This makes client classes easier to implement, change, test, and reuse.

Composite should be used when clients ignore the difference between compositions of objects and individual objects

## Reflectie

### Individuele reflectie

**Fabian:**

Algemeen verloop ging vrij goed, we waren altijd goed op tijd begonnen waardoor we vaak eerder klaar waren dan andere teamgenoten. Dit komt hoofdzakelijk doordat ik vaak afspraken regelde en op voorhand duidelijk maakte van wat er in de vergadering bereikt moest worden, om op schema te blijven. Vervolgens waren er enkele teamleden die niet altijd aanwezig waren, zij stelden dan wel voor om thuis een deel zelf te maken. Dit was erg handig op het einde als Emanuel het design pattern gedeelte helemaal zelf gemaakt heeft en de rest van het team zich kon focusen op de paper/presentatie.

**Glenn:**

In het geheel vond ik dat er in dit deel goed samengewerkt werd, we spraken vaak af en werkten goed door.

We begonnen altijd goed op tijd aan elk deel en zaten voor het de meerderheid van de tijd voor op schema, toch begonnen we enkele fouten hierdoor, omdat we de scope van de opdracht veel te groot geïnterpreteerd hadden, en zaken hermaakten die niet nodig waren.

Na het duidelijk werd dat we veel te veel werk aan het maken waren, werd ik ziek en was ik ziek voor een goeie 2 weken, inclusief de week vakantie die daarop volgde, toch werkte ik nog mee tijdens deze periode.

Mijn bijdrage aan dit gedeelte vond ik ook voldoende, ik maakte alles in Enterprise Architect omdat ik de meeste kennis en ervaring met dit programma heb.

**Bram:**

De eerste opdracht verliep vrij vlot. Er werd in het algemeen goed samengewerkt met iedereen. In het begin waren we al snel goed op weg, en konden we het goed met elkaar vinden. We hebben heel veel gewerkt, onder meer een volledige zaterdag (van 9u tot 22u) om al een goede voorsprong te hebben op de andere teams.

We hadden in al ons enthousiasme al te veel gewerkt en bepaalde onderdelen die vorig semester wel nodig waren en nu niet, gemaakt. Dit was jammer want het kostte ons veel tijd.

Al bij al kan ik zeggen dat ik zeker tevreden ben met deze opdracht. Ik merk dat bepaalde teams op de laatste dag nog een vrij groot deel van de opdracht moeten afwerken, terwijl wij enkel nog de paper eens moeten nalezen en de PowerPoint afwerken.

Ik was de secretaris, en buiten één keer heb ik altijd op het correcte moment het weekverslag doorgestuurd.

**Luis:**

Al bij al vond ik het tof en productief om in deze groep te werken. Ten eerste werd er altijd doelmatig gewerkt, op voorhand was altijd alles duidelijk van wat we willen bereiken en zo stonden we altijd op schema. We hadden meestal een voorsprong vergeleken met de andere groepen.

Ten tweede was het een open groep dat makkelijk was om te communiceren wat belangrijk is in een groepsproject. Ten slotte waren we vrij flexibel, als ik bijvoorbeeld er niet kon zijn tijdens een afspraak dan kon ik taken inhalen.

### Teamreflectie

Als team kunnen we concluderen dat deze opdracht alweer een succes was. Hier en daar waren af en toe wel wat momenten waarop enkele leden de focus verloren, maar uiteindelijk kwam alles goed. Iedereen heeft goed gewerkt, al waren er uiteraard bepaalde leden die erbovenuit schoten. We hebben heel regelmatig afgesproken (vaak 3 keer per week) om aan de opdrachten te werken. Dit gebeurde zowel online als persoonlijk.

Het gebeurde niet heel veel dat iedereen tezamen aanwezig was, maar dit werd rechtgezet door deze persoon een taak te geven. Zo was bijvoorbeeld Emmanuel enkele keren afwezig, maar heeft hij als compensatie het onderdeel design patterns gemaakt.

Fabian was teamleider en zorgde ervoor dat alle taken op tijd klaar waren en iedereen wist wat te doen. Glenn is de eindredacteur en zorgt ervoor dat de paper goed ingediend wordt. Bram is de secretaris en zorgt voor de weekverslagen. Luis en Emmanuel zijn de reserves.