|  |  |
| --- | --- |
| Leertaak 2  CASUS “Sushirestaurant” | Lars van scheijndel  sanne van der linden  Sander Reintjes  Patrick Druart  casper plate  TICT-V1AUI-17 |

# Inleiding

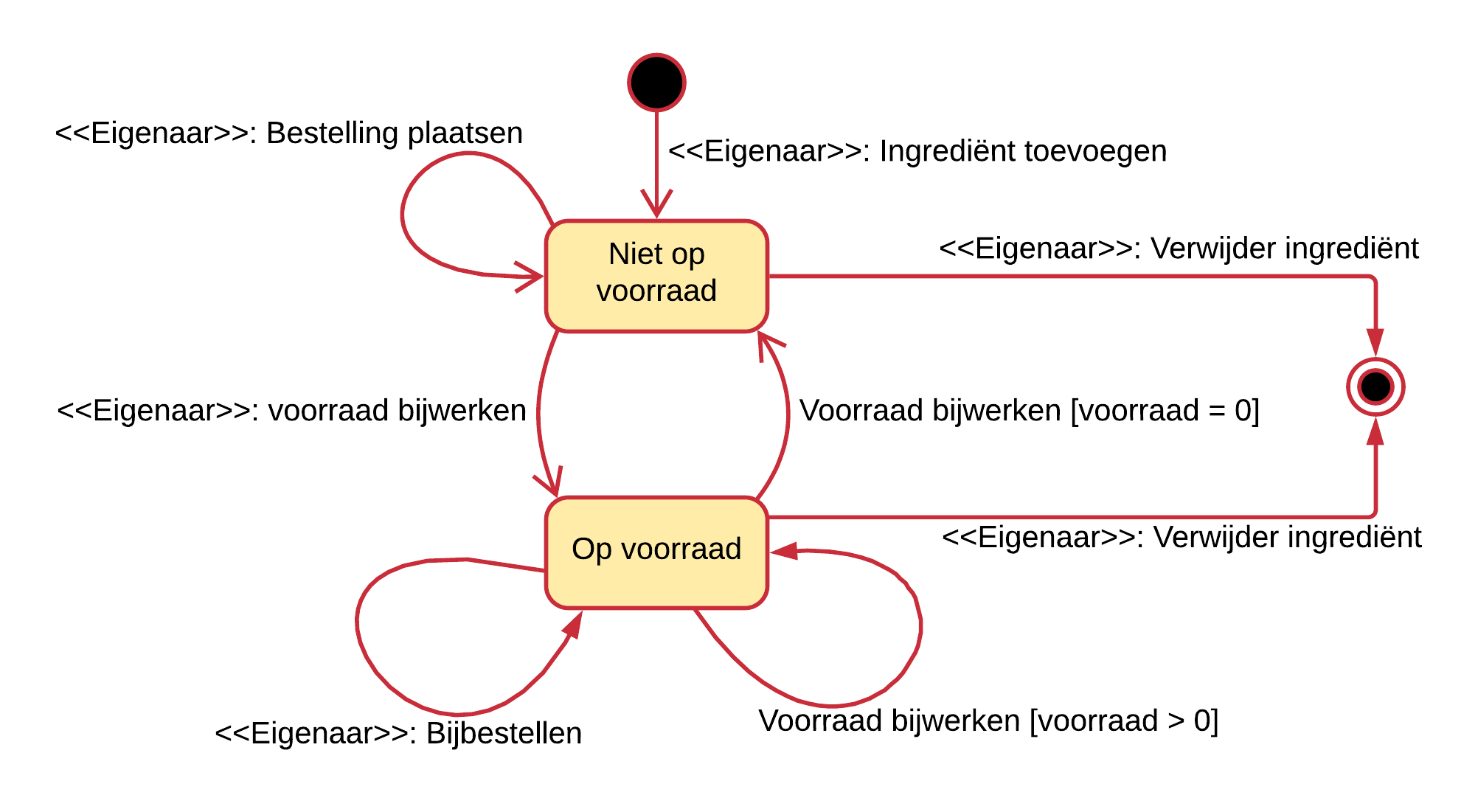
Dit document bevat de antwoorden voor de leertaken die gebaseerd zijn op de casus “Sushirestaurant”. De leertaken dienen als een vervanging van een formatieve toets.

Sushirestaurant "Tokyo" is een familiebedrijf in Utrecht dat sushi- en teppan-yaki gerechten serveert. Sushi staat bekend om de vele verschillende, kleine gerechten die samen een volledige maaltijd vormen. Gasten mogen gedurende hun verblijf verschillende bestellingen van gerechten doorgeven. Hiervoor legt de bediening op elke tafel een bestellijst. Als de gasten zijn gaan zitten noteert de ober op de bestellijst hoeveel personen het gezelschap telt. Bij deze papieren bestellijst zitten heel veel beperkingen en de eigenaren willen hier graag van af. Aan mij is nu de taak om een nieuw systeem te ontwerpen.

De eigenaren van “Tokyo” willen graag investeren in een softwaresysteem waarbij gasten via een tablet vanaf hun tafel de gewenste gerechten kunnen bestellen. In de keuken komt dan een groot touchscreen te hangen waarop de kok en de obers de openstaande bestellingen kunnen zien en afhandelen. In de toekomst moet het ook mogelijk zijn om met dit systeem de voorraad en bestellingen van ingrediënten bij te houden en om de menukaart te beheren.

# Toestandsdiagram Ingrediënt

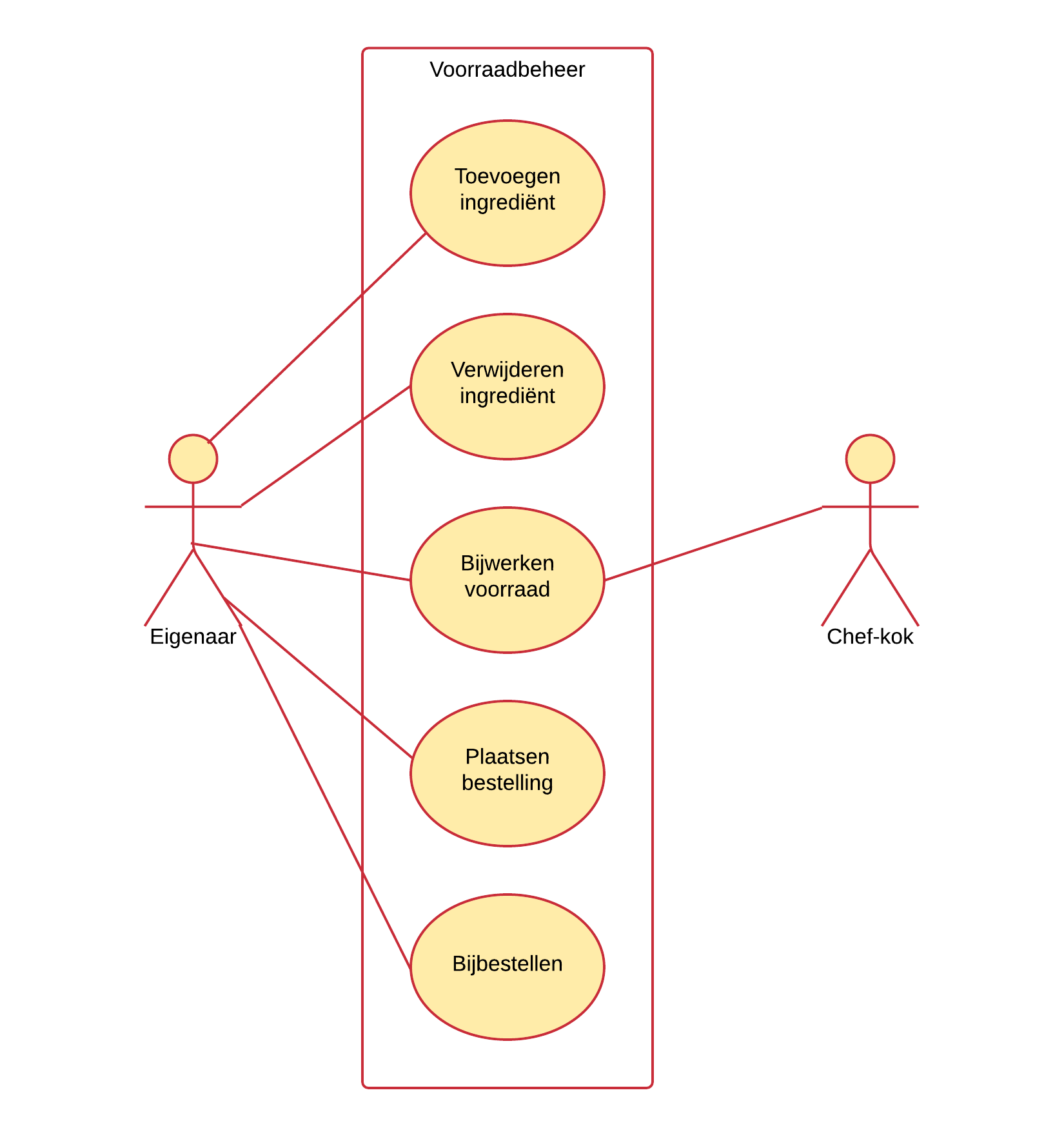
Voor het ontwikkelen van het voorraadbeheersysteem is onderstaand UML-toestandsdiagram gemaakt met als kernobject ingrediënt.



Eens in de zoveel tijd bespreken de chef-kok en de eigenaren of ze nog wel de juiste gerechten aanbieden. Als ze dan beslissen dat er een nieuw gerecht aan de menukaart toegevoegd wordt zijn er soms ook nieuwe ingrediënten nodig. Wanneer dit ingrediënt dan aan het systeem toegevoegd wordt bevindt het zich in de toestand niet op voorraad. De eigenaar kan dan beslissen om het te bestellen en vervolgens de toestand naar op voorraad te zetten. Als er tijdens een bespreking besloten wordt dat een gerecht van de kaart wordt geschrapt dan kan het zijn dat een ingrediënt niet meer nodig is. Het ingrediënt kan dan uit het systeem verwijderd worden door de eigenaar.

# UseCase Diagram Ingrediënt

Aan de hand van het eerdergenoemde toestandsdiagram kunnen we onderstaand UseCase diagram afleiden. Het diagram bevat vier UseCases. Toevoegen ingrediënt, Verwijderen ingrediënt, Bijwerken voorraad en Plaatsen bestelling. De chef mag alleen de voorraad bijwerken. De eigenaar mag alles.



Bij dit UseCase diagram hoort onderstaande samenvatting.

|  |  |
| --- | --- |
| UseCase | Samenvatting |
| Toevoegen ingrediënt | Er wordt een nieuw ingrediënt aangemaakt. Het systeem vraagt om de **naam**, **minimale voorraad** en zet de **huidige voorraad**. |
| Verwijderen ingrediënt | Het systeem toont een lijst van de ingrediënten met de **namen** of vraagt om de **naam**. Het ingrediënt wordt uit het systeem verwijderd. |
| Bijwerken voorraad | Het systeem vraagt om de **naam** en toont de **huidige voorraad** en **minimale voorraad**. Het systeem vraagt dan om de **huidige voorraad** en **minimale voorraad** van **naam** bij te werken. |
| Plaatsen bestelling | Het systeem berekend welke ingrediënten er besteld moeten worden aan de hand van de **huidige voorraad** en **minimale voorraad** en toont de **naam**, **huidige voorraad** en **minimale voorraad** vervolgens de actor om een verzend bevestiging. |
| Bijbestellen | Het systeem vraagt om de **naam** en zet het ingrediënt op de bestellijst. Na ok van de actor wordt de bestelling verzonden. |

# UseCase Templates Ingrediënt

Aan de hand van de samenvatting zijn onderstaande UseCase templates ontstaan.

|  |
| --- |
| ID: UC1 |
| Naam: Toevoegen Ingrediënt |
| Hoofdactor: Eigenaar |
| Preconditie: N.V.T. |
| Samenvatting: Er wordt een nieuw ingrediënt aangemaakt. Het systeem vraagt om de **naam**, **minimale voorraad** en zet de **huidige voorraad**. |
| Scenario:  1. Systeem vraagt om naam en minimale voorraad.  1.1 Actor voert de gegevens in.  2a. Systeem voegt het ingrediënt toe (als het nog niet bestaat). |
| Postconditie: Ingrediënt is toegevoegd. |
| Alternatiefscenario:  2b. Systeem geeft een waarschuwing: ingrediënt bestaat al (als het ingrediënt al bestaat) |
| Postcondities: Er is geen ingrediënt toegevoegd. |

|  |
| --- |
| ID: UC2 |
| Naam: Verwijderen Ingrediënt |
| Hoofdactor: Eigenaar |
| Samenvatting: Het systeem toont een lijst van de ingrediënten met de **namen** of vraagt om de **naam**. Het ingrediënt wordt uit het systeem verwijderd. |
| Preconditie: N.V.T. |
| Scenario:  1. Systeem toont een lijst met ingrediënten.  1.1. Actor zoekt een ingrediënt op.  2a. Systeem toont informatie over het ingrediënt.  2.1 Actor verwijdert ingrediënt.  3. Systeem verwijdert ingrediënt. |
| Postconditie: Ingrediënt is verwijderd. |
| Alternatiefscenario:  2b. Systeem kan geen ingrediënt vinden. |
| Postconditie: Er is geen ingrediënt verwijderd. |

|  |
| --- |
| ID: UC3 |
| Naam: Bijwerken voorraad |
| Hoofdactor: Eigenaar  Tweede actor: Kok |
| Samenvatting: Het systeem vraagt om de **naam** en toont de **huidige voorraad** en **minimale voorraad**. Het systeem vraagt dan om de **huidige voorraad** en **minimale voorraad** bij te werken. |
| Preconditie: N.V.T. |
| Scenario:  1. Het systeem toont de lijst met ingrediënten.  1.1 Actor kiest een ingrediënt.  2. Systeem toont informatie over het ingrediënt.  2.1 Actor werkt de gegevens bij.  3. Systeem voert aanpassingen door. |
| Postconditie: De voorraad is bijgewerkt. |

|  |
| --- |
| ID: UC4 |
| Naam: Plaatsen bestelling |
| Hoofdactor: Eigenaar |
| Samenvatting: Het systeem berekend welke ingrediënten er besteld moeten worden aan de hand van de **huidige voorraad** en **minimale voorraad** en toont de **naam**, **huidige voorraad** en **minimale voorraad** vervolgens de actor om een verzend bevestiging. |
| Preconditie: N.V.T. |
| Scenario.  1. Systeem berekent de bestellijst.  1.1 Actor bevestigt.  2. Systeem verzendt de bestelling. |
| Postconditie: Er is een bestelling geplaatst. |

|  |
| --- |
| ID: UC5 |
| Naam: Bijbestellen |
| Hoofdactor: Eigenaar |
| Samenvatting: Het systeem vraagt om de **naam** en zet het ingrediënt op de bestellijst. Na ok van de actor wordt de bestelling verzonden. |
| Preconditie: N.V.T. |
| Scenario:  1. Systeem toont lijst met ingrediënten.  1.1 Actor zoekt een ingrediënt op.  2. Systeem toont informatie over het ingrediënt.  2.1 Actor zet het ingrediënt op de bestellijst.  2.2 Actor bevestigt de bestellijst.  3. Systeem verzendt de bestelling. |
| Postcondities: Er is een bestelling geplaatst. |

# Volledigheidsanalyse Ingrediënt

## Check toestandsdiagram

De gevonden toestanden zijn:

* Niet op voorraad.
* Op voorraad.

Aan de hand van de opdracht die van de eigenaren is ontvangen heb ik geen ontbrekende toestanden en/of triggers gevonden.

## Check UseCase Diagram

Aan de hand van het UseCase Diagram kan de chef alleen voorraadbijwerken. Maar misschien dat het makkelijk is als de chef ook kan bestellen of ingrediënten kan beheren.

## Check UseCase Template

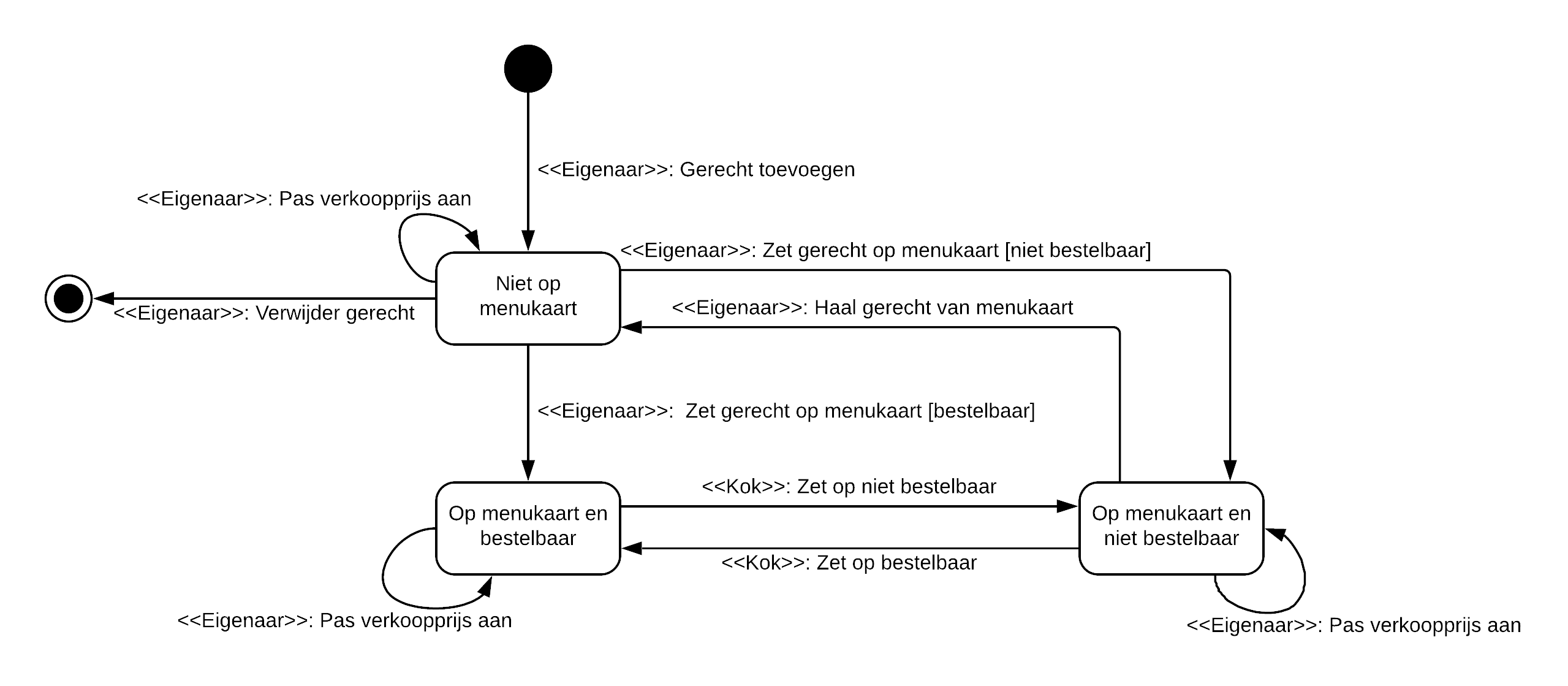
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Ingrediënt  Toevoegen | Ingrediënt  Verwijderen | Bestelling  Plaatsen | Bijbestellen | Voorraad  Bijwerken |  |
| Naam | In | In/Out | Out | In | In/Out | Ok |
| MinVoorraad | In | - | Out | - | In/Out | Ok |
| HuidigeVoorraad | In | - | Out | - | In/Out | Ok |

## Issues

|  |  |  |
| --- | --- | --- |
| Nr. | Scenario | Vraag |
| 1 | De chef kan alleen voorraad bijwerken. | Mag hij ook bestellen en/of ingrediënten toevoegen/verwijderen? |

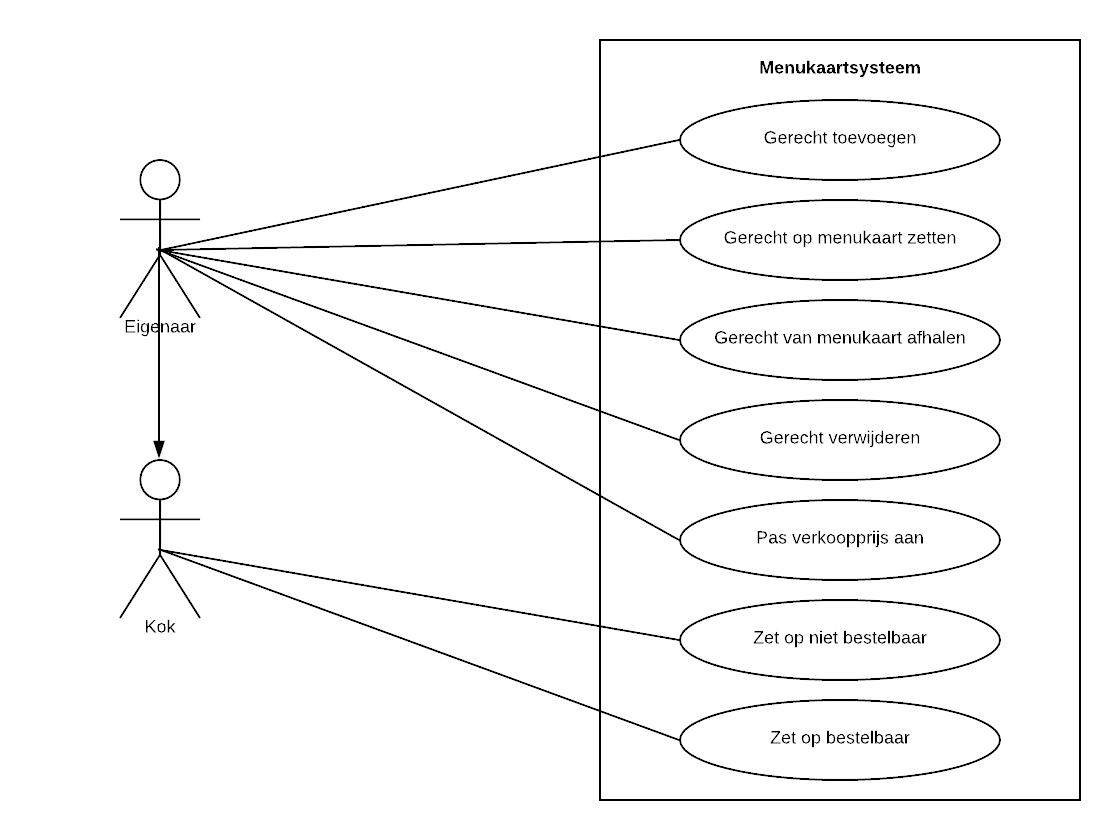
# Toestandsdiagram Gerecht

Voor het kernobject gerecht hebben wij drie toestanden gevonden. De eerste (waar het gerecht in terecht komt wanneer deze wordt toegevoegd) is “niet op de menukaart”. Een gerecht kan alleen uit het systeem verwijderd worden wanneer het zich in deze toestand bevindt ter beveiliging. Vanuit deze toestand kan de eigenaar beslissen om het gerecht op de menukaart te zetten. Een gerecht kan zich in twee verschillende toestanden vinden als het op de menukaart staat. Deze toestanden zijn “op menukaart en niet bestelbaar” en “op menukaart en bestelbaar”. Een gerecht kan op de menukaart staan maar het kan zijn dat de ingrediënten opraken. Als iemand net op dit moment het gerecht aan het bekijken was en het verdwijnt ineens van het scherm denkt die persoon ook van wat gebeurt er nu. Daarom is het belangrijk dat het gerecht eerst naar niet bestelbaar wordt gezet voordat het van de menukaart wordt gehaald.



# UseCase Diagram Gerecht

Als we nog een kijken naar het toestandsdiagram kunnen we zien dat er zeven handelingen gedaan kunnen worden. Deze staan in onderstaand UseCase diagram.



Bij het UseCase diagram op de vorige pagina hoort de onderstaande tabel met samenvattingen.

|  |  |
| --- | --- |
| UseCase | Samenvatting |
| Gerecht toevoegen | Het systeem vraagt de actor om de naam(i), prijs(i), verwachte bereidingstijd(i) en de ingrediënten(i) en voegt het gerecht toe. |
| Gerecht op menukaart zetten | Het systeem toont de lijst met gerechten niet op de menukaart. De actor selecteert het gerecht(o) en voegt deze toe aan de menukaart. |
| Gerecht van menukaart afhalen | Het systeem toont de lijst met gerechten op de menukaart. De actor selecteert het gerecht(o) en haalt deze van de menukaart af. |
| Gerecht verwijderen | Het systeem toont de lijst met gerechten niet op de menukaart. De actor selecteert het gerecht(o) en bevestigd. Het systeem verwijdert het gerecht. |
| Pas verkoopprijs aan | Het systeem toont de lijst met gerechten. De actor selecteert het gerecht(o) en past de verkoopprijs(i) aan. |
| Zet op niet bestelbaar | Het systeem toont de lijst met gerechten. De actor selecteert het gerecht(o) en zet deze op niet bestelbaar. |
| Zet op bestelbaar | Het systeem toont de lijst met gerechten. De actor selecteert het gerecht(o) en zet deze op bestelbaar. |

# UseCase Templates Gerecht

|  |  |
| --- | --- |
| **Id 01** | **Use case: Gerecht Toevoegen** |
| **Actor** | Eigenaar |
| **Samenvatting** | De actor voegt in het systeem(i) een nieuw gerecht toe met de daarbij behorende ingrediënten en een unieke naam en de prijs. |
| **Preconditie** | Ingrediënten zijn in het systeem bekend |
| **Main Scenario** | 1. Systeem toont de invoervelden en een lijst van ingrediënten 2. Actor geeft de naam, prijs, verwachte bereidingstijd en ingrediënten op 3. Systeem toont een overzicht van de ingevulde invoervelden 4. Actor controleert gegevens en drukt op invoeren 5. Systeem controleert gegevens 6. Systeem slaat gegevens op |
| **Postconditie** | Er is een nieuw gerecht aangemaakt |

|  |  |
| --- | --- |
| Trigger: | In stap 4; [naam, prijs of de ingrediënten zijn onjuist] |
| Stappen: | 1. Actor verandert de onjuiste gegevens in het invoerveld 2. Systeem gaat verder met stap 5 |

|  |  |
| --- | --- |
| Trigger: | In stap 5; [een van de ingevulde waardes is ongeldig ] |
| Stappen: | 1. Systeem toont foutmelding bij het ingevulde invoerveld met een ongeldige waarde 2. Actor verandert de ongeldige waardes 3. Systeem gaat verder met stap 5 |

|  |  |
| --- | --- |
| **Id 02** | **Use case: Gerecht Op Menukaart Zetten** |
| **Actor** | Eigenaar |
| **Samenvatting** | De actor voegt het betreffende gerecht toe aan de menukaart, zodat klanten dit gerecht kunnen bestellen. |
| **Preconditie** | Het gerecht staat in het systeem. |
| **Main Scenario** | 1. Systeem toont lijst van gerechten die niet op de menukaart staan 2. Actor kiest uit de lijst het gerecht dat hij wilt toevoegen. 3. Systeem toont het geselecteerde gerecht ter controle. 4. Actor controleert gegevens van het geselecteerde gerecht. 5. Systeem controleert of het gerecht nog niet op de menukaart staat. 6. Systeem voegt het gerecht toe aan de menukaart. 7. Systeem geeft feedback aan de gebruiker dat het gelukt is. |
| **Postconditie** | Het gerecht is in te zien op de menukaart en te bestellen door klanten. |

|  |  |
| --- | --- |
| Trigger: | In stap 4; [gegevens van gerecht zijn onjuist] |
| Stappen: | 1. Systeem gaat terug naar stap 1. |

|  |  |
| --- | --- |
| Trigger: | In stap 5; [gerecht staat al op de menukaart] |
| Stappen: | 1. Systeem ziet dat dit gerecht al op de menukaart staat. 2. Systeem toont foutmelding aan de actor. 3. Systeem gaat verder met stap 1. |

|  |  |
| --- | --- |
| **Id 03** | **Use case: Gerecht Van Menukaart Afhalen** |
| **Actor** | Eigenaar |
| **Samenvatting** | De actor haalt het betreffende gerecht van de menukaart(i), hierna kunnen de klanten het gerecht niet meer zien en bestellen. |
| **Preconditie** | Het gerecht is toegevoegd aan het systeem en staat op de menukaart. |
| **Main Scenario** | 1.   Systeem toont de lijst met gerechten die op de menukaart staan  2.   De actor kiest het gerecht uit de lijst  3.   Het systeem toont het gekozen gerecht  4.   De actor controleert of het juiste gerecht geselecteerd is  5.   Het systeem slaat de wijzigingen op. |
| **Postconditie** | Er is een gerecht van de menukaart afgehaald. |

|  |  |
| --- | --- |
| Trigger: | In stap 4; [gegevens van gerecht zijn onjuist] |
| Stappen: | Systeem gaat terug naar stap 1. |

|  |  |
| --- | --- |
| **Id 04** | **Use case: Gerecht Verwijderen** |
| **Actor** | Eigenaar |
| **Samenvatting** | De actor verwijdert het gerecht uit het systeem. |
| **Preconditie** | Het gerecht staat niet op het menu. |
| **Main Scenario** | 1.     Het systeem toont de lijst met gerechten.  1.1.  De actor kiest het gerecht.  2.     Het systeem toont het gekozen gerecht.  2.1.  De actor controleert of het juiste gerecht is weergegeven.  2.2.  De actor bevestigt het verwijderen.  3.     Systeem verwijdert het gerecht.  4.     SYsteem slaat gegevens op |
| **Postconditie** | Er is een gerecht verwijderd. |

|  |  |
| --- | --- |
| Trigger: | In stap 2.1; [gegevens van gerecht zijn onjuist] |
| Stappen: | 3. Systeem gaat terug naar stap 1. |

|  |  |
| --- | --- |
| **Id 05** | **Use case: Zet Op Niet Bestelbaar** |
| **Actor** | Kok, Eigenaar |
| **Samenvatting** | De actor zet in het systeem een gerecht op niet bestelbaar wat zorgt dat het gerecht niet langer bestelbaar is. |
| **Preconditie** | Het gerecht is bestelbaar en staat op het menu. |
| **Main Scenario** | 1.     Het systeem toont een lijst met alle bestelbare gerechten.  2.     De actor kiest het gerecht uit de lijst met te bestellen gerechten.  3.     Het systeem toont het gerecht.  4.   De actor controleert het getoonde gerecht  5.     De actor zet het gerecht op niet bestelbaar. |
| **Postconditie** | Het gerecht is niet langer bestelbaar maar staat nog wel op het menu. |
|  |  |

|  |  |
| --- | --- |
| Trigger: | In stap 4; [het verkeerde gerecht is geselecteerd] |
| Stappen: | 1. De actor selecteert het verkeerde gerecht om op niet bestelbaar te zetten 2. De actor annuleert het niet bestelbaar maken. 3. Systeem gaat verder vanaf stap 1 |

|  |  |
| --- | --- |
| **Id 06** | **Use case: Zet Op Bestelbaar** |
| **Actor** | Kok, Eigenaar |
| **Samenvatting** | De actor zet een gerecht dat als niet bestelbaar op de menukaart stond, als bestelbaar. |
| **Preconditie** | Het gerecht staat op de menukaart. |
| **Main Scenario** | 1. Systeem toont de lijst met gerechten. 2. De actor kiest het gerecht. 3. Het systeem toont de gegevens van het gerecht. 4. Actor controleert gegevens van het gerecht. 5. Systeem controleert of alle ingrediënten aanwezig zijn. 6. Systeem zet gerecht op bestelbaar. 7. Systeem toont melding aan actor dat het gerecht op bestelbaar staat. |
| **Postconditie** |  |

|  |  |
| --- | --- |
| Trigger: | In stap 4; [gegevens van gerecht zijn onjuist] |
| Stappen: | 1. Systeem gaat terug naar stap 1. |

|  |  |
| --- | --- |
| Trigger: | In stap 5; [Niet alle ingrediënten zijn voldoende aanwezig] |
| Stappen: | 1. Systeem toont controlevraag of het gerecht ook met te weinig ingrediënten bestelbaar mag worden. 2. Actor kiest ja.    1. Systeem gaat verder met stap 5. 3. Actor kiest nee.    1. Systeem gaat verder met stap 1. |

|  |  |
| --- | --- |
| **Id 07** | **Use case: Pas Verkoopprijs Aan** |
| **Actor** | Eigenaar |
| **Samenvatting** | De actor laat het systeem de verkoopprijs van een gerecht aanpassen naar het gewenste bedrag. |
| **Preconditie** | N.V.T. |
| **Main Scenario** | 1.    Het systeem toont de lijst met gerecht.  2.    De actor selecteert een gerecht.  3.    Het systeem toont het gerecht.  4.1  De actor controleert het gerecht.  4.2  De actor vult de nieuwe prijs in.  5.    Het systeem slaat de nieuwe prijs op. |
| **Postconditie** | De verkoopprijs van het gerecht is aangepast. |

|  |  |
| --- | --- |
| Trigger: | In stap 4.2; [gegevens van gerecht zijn onjuist] |
| Stappen: | 5. Systeem gaat terug naar stap 1. |

# Volledigheidsanalyse Gerecht

## Check toestandsdiagram

Door de casus door te lezen zijn wij op dit diagram gekomen. Daarom zijn wij van mening dat we niets over het hoofd hebben gezien kijkend naar wat de eigenaren van ons hebben gevraagd. Ook zijn wij na lang brainstormen zelf ook geen issues tegen het lijf gelopen.

## Check UseCase diagram

Op het moment mag de chef alleen gerechten op bestelbaar of niet bestelbaar zetten. Op sommige moment kan het gewenst zijn dat de chef ook een gerecht op of van de menukaart mag zetten/halen. Mag de chef ook de prijs aanpassen of gerechten toevoegen/verwijderen?

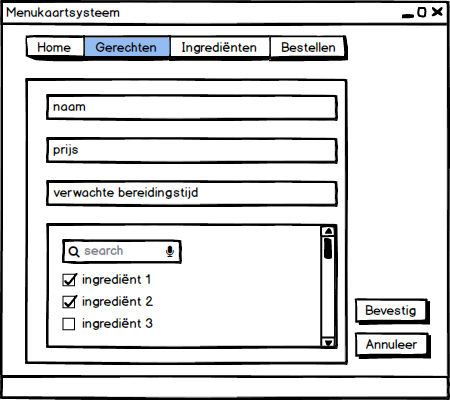
## Check UseCase templates

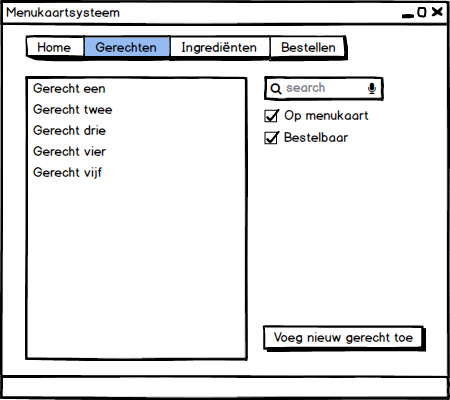
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Gerecht Toevoegen | Gerecht Op Menukaart Zetten | Gerecht Van Menukaart Afhalen | Gerecht Verwijderen |
| naam | In | Out | Out | Out |
| prijs | In | Out | Out | - |
| verwachteBereidingstijd | In | Out | Out | - |
| ingrediënten | In | Out | Out | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Zet Op Niet Bestelbaar | Zet Op Bestelbaar | Pas Verkoopprijs aan |  |
| naam | Out | Out | Out | Ok |
| prijs | Out | Out | In/Out | Ok |
| verachteBereidingstijd | Out | Out | Out | Ok |
| ingrediënten | Out | Out | Out | Ok |

# Wireframes Gerecht

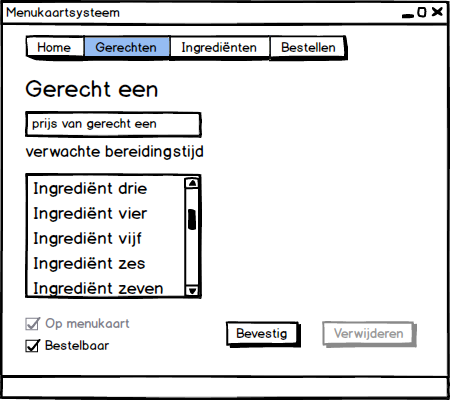
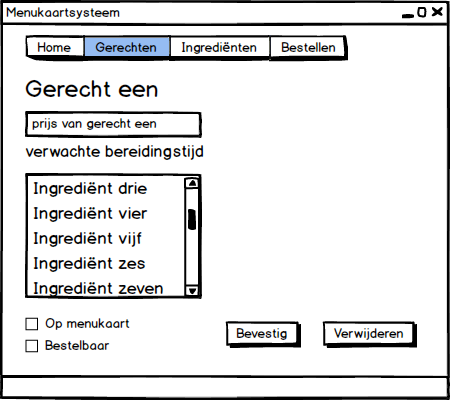
In dit hoofdstuk worden de wireframes van kernobject gerecht behandeld. Om te beginnen hebben we het overzicht. Dit is waar alle UseCases van gerecht beginnen.

Vervolgens krijgen we de UseCase waar het allemaal begint. Gerecht toevoegen, een gerecht kan toegevoegd worden door op de knop ‘Voeg nieuw gerecht toe rechts onderin het venster te klikken. Hier zijn vervolgens drie invoer velden en een lijst met ingrediënten. Ook weer rechts onderin staat de knop bevestigen. Deze knop voegt het gerecht toe en brengt de actor terug naar het overzicht.



Door in het overzicht op een gerecht te klikken kom je bij het informatie venster van dat gerecht.



De andere zes UseCases vinden allemaal plaats in hetzelfde venster. Bij dit venster kan je komen door op de knop bewerken te drukken van het laatst behandelde scherm.

Zoals je kan zien, ziet het scherm er soms anders uit. Dit komt omdat een gerecht alleen verwijdert kan worden als hij niet op de menukaart staat. Daarom is in de rechter afbeelding de verwijderen knop disabled.

De regel die de prijs van het gerecht bevat is veranderd in een input box, deze maakt het mogelijk om de prijs aan te passen.

Ook zijn er twee checkboxes, Op menukaart en Bestelbaar. Als beide checkboxes niet gevinkt zijn, staat het gerecht niet op de menukaart. Is Op menukaart gevinkt staat het gerecht op de menukaart maar is hij niet bestelbaar. Zijn beide boxes gevinkt dan staat het gerecht op de menukaart en is hij bestelbaar. Zoals je in het rechter venster kan zien is Op menukaart disabled. Dit komt omdat het gerecht niet van de menukaart gehaald mag worden als hij nog bestelbaar is.

Door op bevestigen te klikken worden de wijzigen doorgevoerd en word de actor weer terug naar het overzicht gebracht.