

Functioneel Ontwerp

ANALYSIS

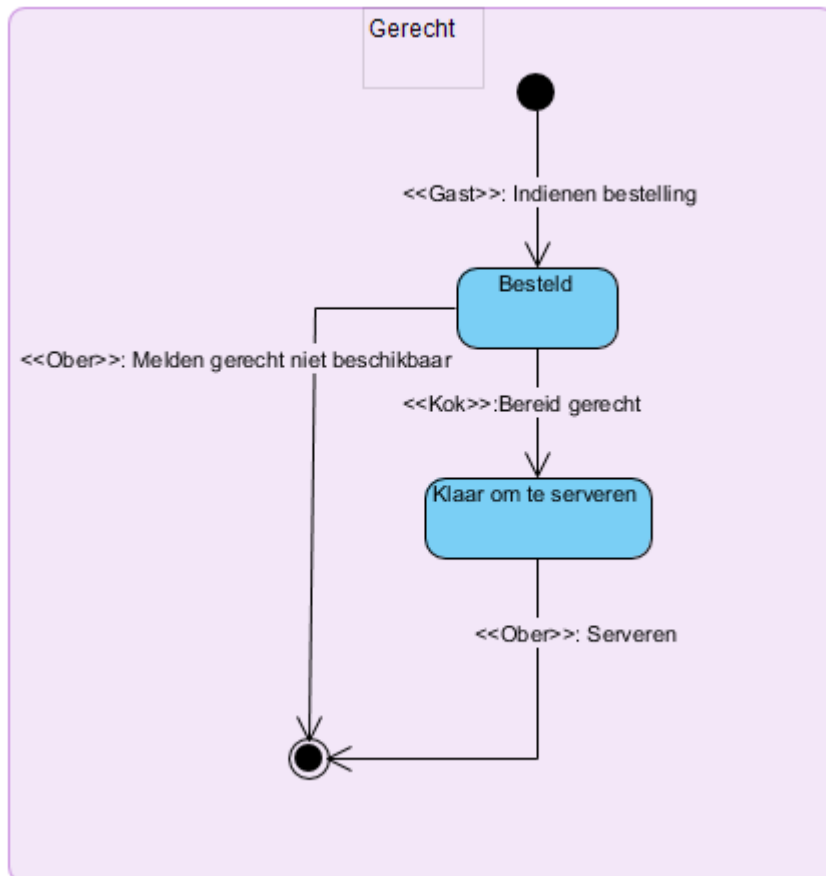
KELVIN DE VRIES, LENNART MEIJER, RUBEN OOSTERINK

Inhoudsopgave

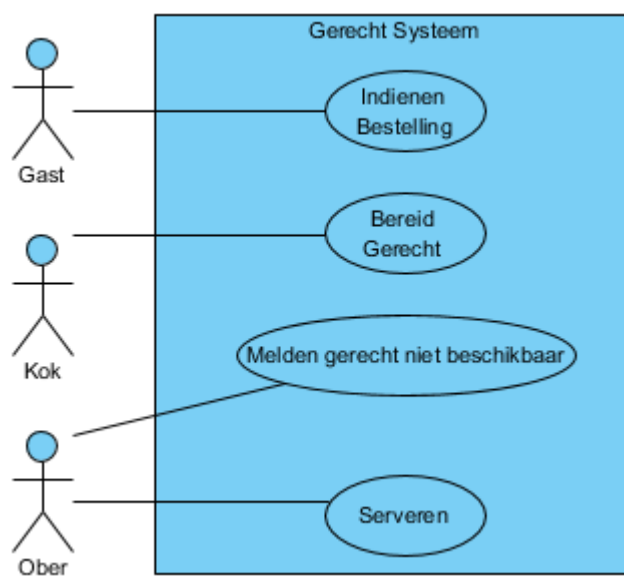
OBJECT gerecht	2
Toestand diagram	2
Use case diagram	2
Samenvatting	3
Use Case Template	4
Wire Frames	8
OBJECT tafel	10
Toestand diagram	10
Use case diagram	10
Samenvatting	11
Use case template	12
Volledigheid	15
Wire Frame	16

OBJECT gerecht

Toestand diagram



Use case diagram



Samenvatting

Use case naam	Samenvatting
Indienen Bestelling	Het systeem laat een overzicht van alle gerechten [out](vereist attribuut beschikbaar[in]). De actor kiest een gerecht en het systeem verwerkt de bestelling.
Bereid Gerecht	Het systeem laat een overzicht zien van alle gerechten die besteld zijn [out]. De Actor kiest het gerecht dat bovenaan de lijst staat, maakt deze en zet in het systeem dat deze klaar is[in]. Het systeem neemt alle gebruikte ingrediënten op en zet deze op de bestellijst [out]
Melden Gerecht niet beschikbaar	Het systeem geeft aan dat het gerecht niet beschikbaar is [out]. Het systeem stelt een ander gerecht voor [out] (vereist attribuut beschikbaar [in]) die de gast kan kiezen.
Serveren	Het systeem geeft aan dat het gerecht klaar is om te geserveerd te worden [out](vereist attribuut klaar[in]). De ober serveert gerecht en geeft in systeem aan dat het gerecht is geserveerd [out].

Use Case Template

Use Case nummer	Ger-01
Use Case naam	Indienen Bestelling
Actor	Gast
Samenvatting	Zie Samenvattingen
Pre-Conditie	Gerecht moet beschikbaar zijn
Main Scenario	<ol style="list-style-type: none">1. Systeem laat overzicht gerechten zien2. Actor kiest gerecht3. Systeem verwerkt de bestelling4. Einde use case
Post-Conditie (Main Scenario)	Systeem is klaar om nieuwe bestelling op te nemen

Use Case nummer	Ger-02
Use Case naam	Bereid Gerecht
Actor	Kok
Samenvatting	Zie Samenvattingen
Pre-Conditie	Er zijn bestellingen
Main Scenario	<ol style="list-style-type: none"> 1. Systeem laat overzicht zien van alle bestelde gerechten 2. Actor kiest bestelling om te maken 3. Systeem laat alle ingrediënten zien voor gerecht 4. Actor maakt gerecht 5. Actor geeft aan dat gerecht klaar is 6. Systeem zet gerecht op klaar 7. Einde use case
Post-Conditie (Main Scenario)	Systeem heeft gerechten op klaar gezet en is klaar om nieuwe gerechten te ontvangen

Use Case nummer	Ger-03
Use Case naam	Melden gerecht niet beschikbaar
Actor	Ober
Samenvatting	Zie Samenvattingen
Pre-Conditie	Gerecht is niet beschikbaar

Main Scenario	<ol style="list-style-type: none"> 1. Systeem geeft aan dat gerecht niet beschikbaar is 2. Systeem geeft een ander gerecht om te bestellen 3. Actor selecteer een ander gerecht 4. Einde use case
Post-Conditie (Main Scenario)	Systeem is klaar om nieuw gerecht voor te stellen

Use Case nummer	Ger-04
Use Case naam	Serveren
Actor	Ober
Samenvatting	Zie Samenvattingen
Pre-Conditie	Gerechten zijn op klaar gezet
Main Scenario	<ol style="list-style-type: none"> 1. Systeem geeft een overzicht welke gerechten klaar zijn 2. Actor selecteert een gerecht <ol style="list-style-type: none"> a. Actor brengt gerecht b. Actor geeft aan in systeem dat gerecht is geserveerd 3. Systeem zet gerecht op geserveerd 4. Einde use case
Post-Conditie (Main Scenario)	Systeem is klaar om nieuwe gerechten te serveren

Volledigheid

Nummer	Voorbeeld scenario	Vraag aan de klant
1	Tijdens de avond heeft de kok geen gerechten meer in het systeem staan die gemaakt moeten worden. De kok kan dan niet zijn werk doen en hier stopt het systeem dan ook.	<p>Wat als er geen gerechten zijn besteld?</p> <p>Zou het niet logischer zijn om er een alternatieve route in het systeem te zetten voor het geval geen gerechten beschikbaar om te maken.</p>
2	Stel dat je iets wilt bestellen maar het gerecht kan niet besteld worden omdat het niet beschikbaar is. Nu gaat het direct naar besteld en eindigt daarna, een nieuw toestand zoals niet beschikbaar zou handiger zijn.	<p>Wat gebeurt er als het gerecht niet beschikbaar is. Eindigt het toestandsdiagram meteen of gaat het gerecht naar een nieuwe toestand?</p>
3	Stel het gerecht ligt te lang op klaar om te serveren en is niet meer goed. Het gerecht is van toestand verandert en kan niet meer geserveerd worden. De klant moet hier dus een melding van krijgen en het gerecht moet opnieuw gemaakt worden	<p>Wat gebeurt er als het gerecht te lang ligt en niet meer goed is. Wordt het gerecht dan opnieuw gemaakt of krijgt de klant een melding dat het gerecht opnieuw gemaakt wordt?</p>

Wire Frames

Home Bestelling invoeren Bestelling Status Tafel Toevoegen Tafel status

Tafelnummer

De domtoren

HU Special

Vegan deluxe

Vegan deluxe

Bestel Annuleer

Ger-01

Home Tafel status

Tafelnummer

De domtoren

HU Special

Vegan deluxe

Vegan deluxe

Gerecht is niet beschikbaar kies een ander gerecht

Bestel Annuleer

Ger-03

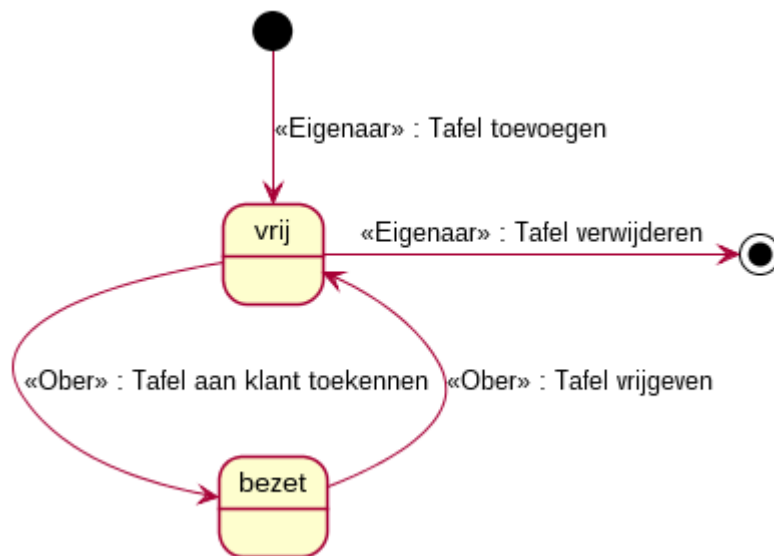
Status gerecht

Tafel:1	De domtoren	Klaar om te serveren
Tafel: 4	De domtoren	Wordt Bereid
Tafel: 3	De domtoren	In afwachting
Tafel: 5	De domtoren	Wordt Bereid

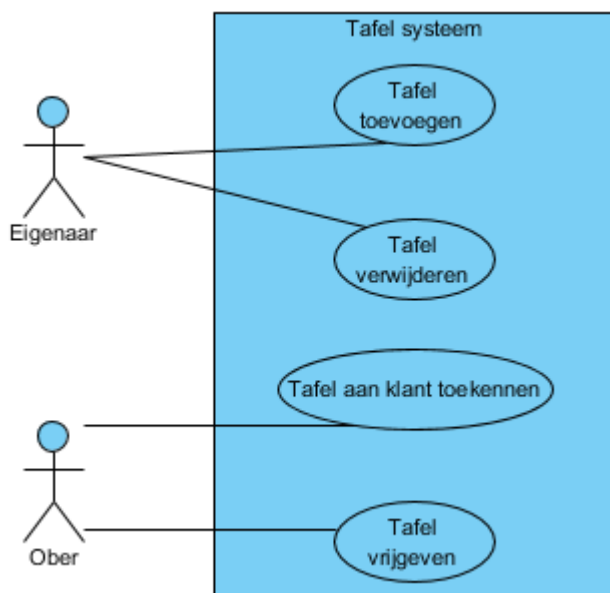
Ger-04

OBJECT tafel

Toestand diagram



Use case diagram



Samenvatting

Use Case	Samenvatting
Tafel toevoegen	Het systeem berekent het eerstvolgende vrije <u>tafelnummer</u> [in], kent dit aan de nieuwe tafel toe en toont dit nummer [out] aan de actor. Per default is de tafel niet <u>bezet</u> [in].
Tafel aan klant toekennen	Het systeem toont een overzicht van vrije <u>tafelnummers</u> [out] (vereist attribuut <u>bezet</u> [out]). De actor selecteert een nummer en het systeem markeert de tafel als <u>bezet</u> [in].
Tafel vrijgeven	Het systeem toont een overzicht van bezette <u>tafelnummers</u> [out] (vereist <u>bezet</u> [out]). De actor selecteert een nummer en het systeem markeert de tafel als <u>niet bezet</u> [in] (vrij).
Tafel verwijderen	Het systeem toont een overzicht van vrije <u>tafelnummers</u> [out] (vereist <u>bezet</u> [out]). De actor selecteert een nummer en het systeem verwijdert de geselecteerde tafel uit het systeem.

Use case template

Use Case nummer	Ta-01
Use Case naam	Tafel toevoegen
Actor	Eigenaar
Samenvatting	Zie Samenvattingen
Pre-Conditie	n.v.t.
Main Scenario	<ol style="list-style-type: none"> 1. Actor voegt nieuwe tafel toe 2. Systeem berekent eerstvolgend tafelnummer 3. Systeem kent deze nummer toe aan tafel 4. Systeem toont tafel nummer aan actor 5. Einde use case
Post-Conditie (Main Scenario)	Systeem heeft een nieuwe tafel toegevoegd

Use Case nummer	Ta-02
Use Case naam	Tafel aan klant toekennen
Actor	Ober
Samenvatting	Zie Samenvattingen
Pre-Conditie	Tafel is niet bezet

Main Scenario	<ol style="list-style-type: none"> 1. Het systeem toont een overzicht met alle tafels 2. Actor selecteert een tafel en selecteert dat deze bezet is. 3. Systeem verandert de status van de tafel 5. Einde use case
Post-Conditie (Main Scenario)	Systeem is klaar om een nieuwe tafel toe te kennen

Use Case nummer	Ta-03
Use Case naam	Tafel vrijgeven
Actor	Ober
Samenvatting	Zie Samenvattingen
Pre-Conditie	Tafel is bezet
Main Scenario	<ol style="list-style-type: none"> 1. Het systeem toont een overzicht met bezette tafels 2. Actor selecteert een tafel 3. Systeem geeft tafel vrij 4. Einde use case
Post-Conditie (Main Scenario)	Systeem is klaar om een nieuwe tafel vrij te geven

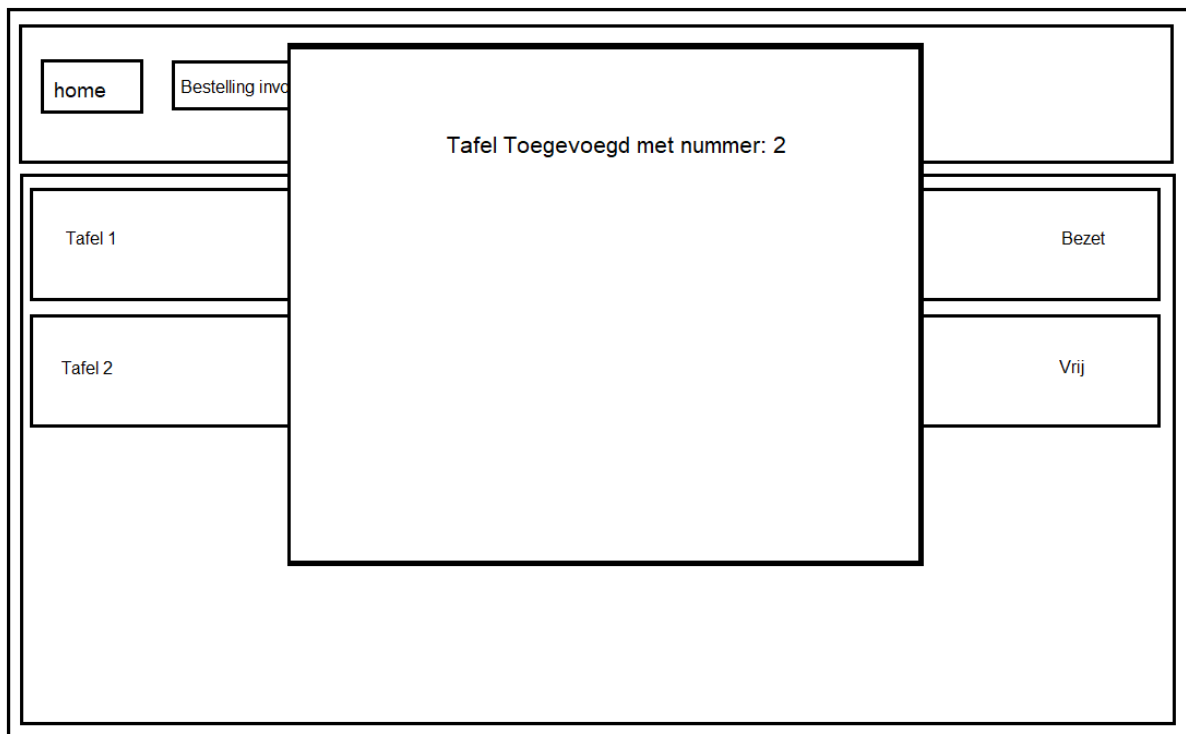
Use Case nummer	Ta-04
Use Case naam	Tafel verwijderen
Actor	Eigenaar
Samenvatting	Zie Samenvattingen
Pre-Conditie	
Main Scenario	<ol style="list-style-type: none"> 1. Systeem toont overzicht van vrije tafelnummers 2. Actor selecteer tafel nummer 3. Systeem verwijdert tafelnummer 4. Einde use case

Post-Conditie (Main Scenario)	Systeem is klaar om nieuwe tafel te verwijderen
-------------------------------	---

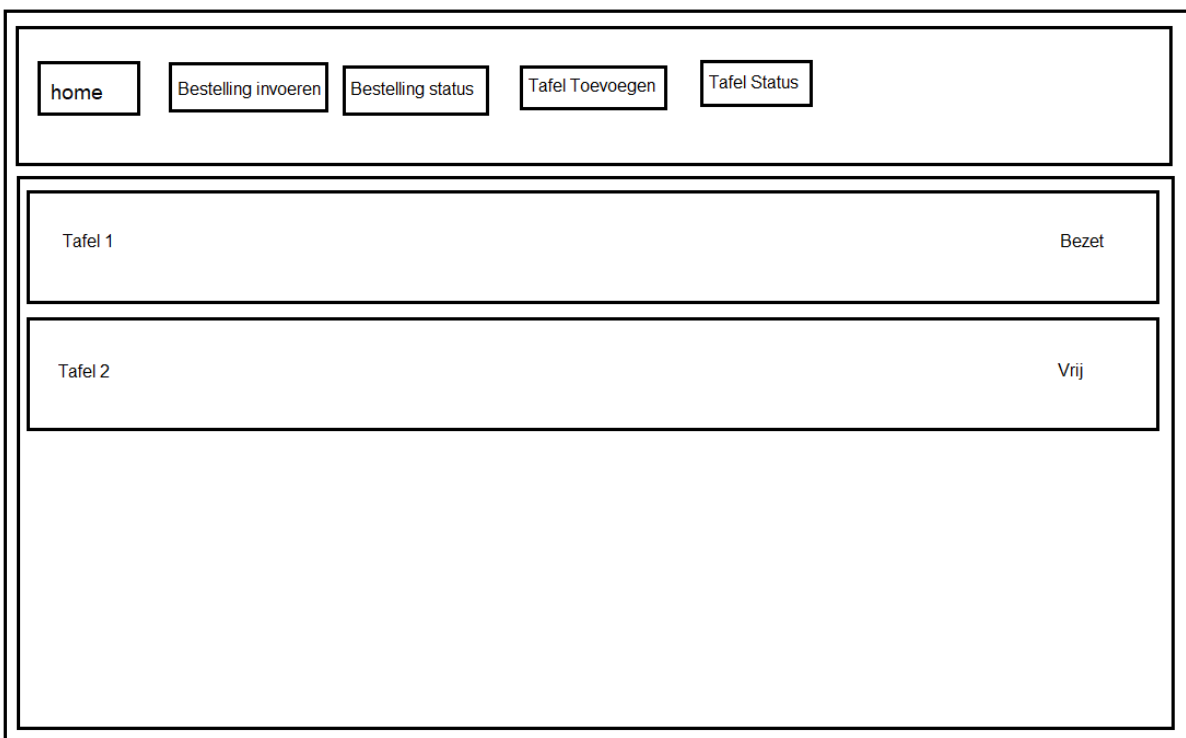
Volledigheid

Nummer	Voorbeeld scenario	Vraag aan de klant
1	Stel er zijn geen tafels beschikbaar en de actor kan dan geen tafel aan de klant toekennen. Het blijkt dus dat er te weinig tafels beschikbaar/ toegevoegd zijn.	Worden er nieuwe tafels toegevoegd als dit voorkomt of worden de klanten geweigerd omdat er geen tafels beschikbaar zijn?
2	Op een avond zijn er niet genoeg obers en moet de eigenaar helpen. De eigenaar moet dan een tafel toekennen aan een klant en een tafel vrijgeven.	Kan de eigenaar ook tafels toekennen en vrijgeven of is dat alleen een functie van de ober?

Wire Frame

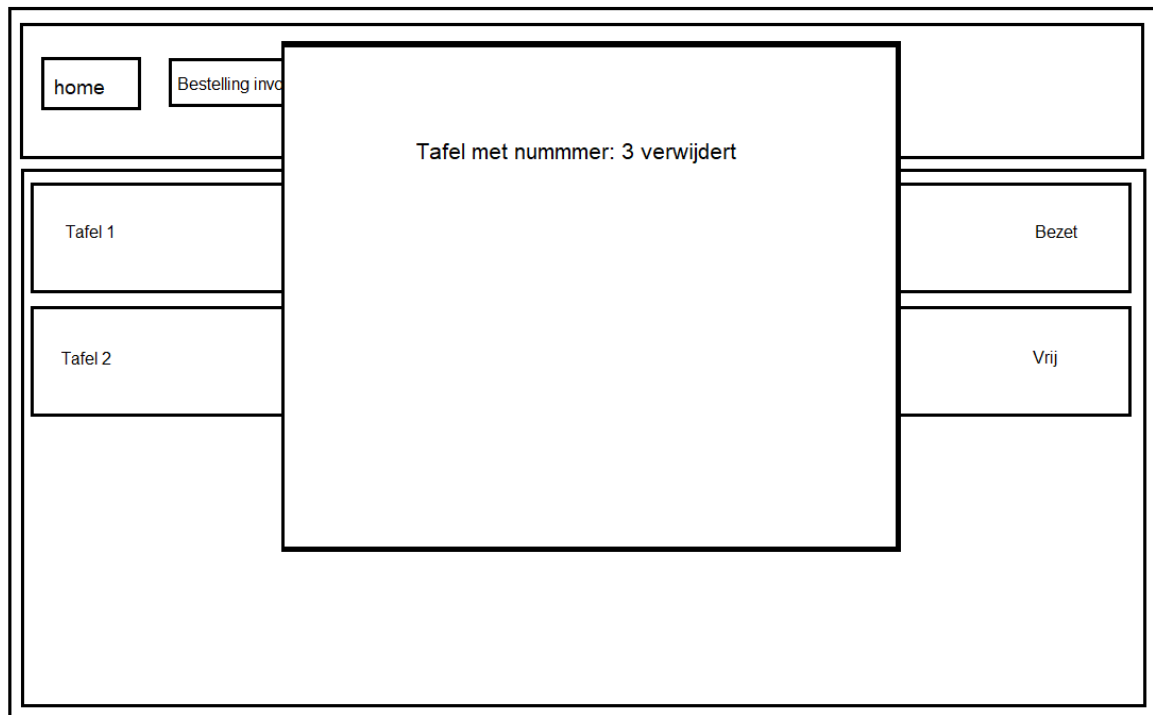


Ta-01 stap 4



Ta-02

Ta-02 stap 1 en Ta-03



Ta-04 stap 3