|  |
| --- |
| Software Requirements Specification |
| HetFrietje | Bas van Schoubroeck | B2F1 Pi4 Inleveropdracht 1 |

Inhoud

[1 Requirements 2](#_Toc177905921)

[2 Diagrammen 4](#_Toc177905922)

[2.1 Usecase diagram 4](#_Toc177905923)

[2.2 Class diagram 5](#_Toc177905924)

[3 Usecase beschrijvingen 6](#_Toc177905925)

[4 Wireframes 8](#_Toc177905926)

[5 Source control 8](#_Toc177905927)

# Requirements

Hieronder zijn de opgestelde requirements, deze requirements hebben een code waarbij de eerste letter aangeeft waar de requirement voor is. Bijvoorbeeld wanneer de requirement begint met een ‘A’ dan heeft de requirement te maken met het Assortiment. ‘B’ voor Bestelling en ‘P’ voor Punten.

|  |  |  |  |
| --- | --- | --- | --- |
| **Code** | **Eis** | **(Niet / Wel) Functioneel** | **Prioriteit** |
| A01 | Een 'eigenaar' beheert het assortiment | Functioneel | 100 |
| A01-1 | Een 'eigenaar' plaatst producten in het assortiment | Functioneel | 100 |
| A01-2 | Een 'eigenaar' voegt een prijs toe aan een product | Functioneel | 100 |
| A01-3 | Een 'eigenaar' voegt een korting toe aan een product | Functioneel | 70 |
| A01-4 | Een 'eigenaar' voegt een foto toe aan een product | Functioneel | 50 |
| A01-5 | Een 'eigenaar' voegt een beschrijving toe aan een product | Functioneel | 90 |
| A01-6 | Een 'eigenaar' voegt keuze opties toe aan een product | Functioneel | 50 |
| A01-7 | Een 'eigenaar' voegt een categorie toe aan een product | Functioneel | 50 |
|  |  |  |  |
| A02 | Een 'eigenaar' wijzigt een product in het assortiment | Functioneel | 100 |
| A02-1 | Een 'eigenaar' wijzigt de prijs van een product | Functioneel | 100 |
| A02-2 | Een 'eigenaar' wijzigt de korting van een product | Functioneel | 70 |
| A02-3 | Een 'eigenaar' wijzigt de foto van een product | Functioneel | 50 |
| A02-4 | Een 'eigenaar' wijzigt de beschrijving van een product | Functioneel | 90 |
| A02-5 | Een 'eigenaar' wijzigt de keuze opties van een product | Functioneel | 50 |
| A02-6 | Een 'eigenaar' wijzigt de categorie van een product | Functioneel | 50 |
|  |  |  |  |
| A03 | Een 'eigenaar' verwijdert een product uit het assortiment | Functioneel | 90 |
| A03-1 | Een 'eigenaar' verwijdert een korting van een product | Functioneel | 70 |
| A03-2 | Een 'eigenaar' verwijdert een keuze optie van een product | Functioneel | 50 |
| A03-3 | Een 'eigenaar' verwijdert een categorie van een product | Functioneel | 50 |
|  |  |  |  |
| A04 | Een 'klant' bekijkt het assortiment | Functioneel | 100 |
| A04-1 | Een 'klant' bekijkt de prijs van een product | Functioneel | 90 |
| A04-2 | Een 'klant' bekijkt de korting van een product | Functioneel | 70 |
| A04-3 | Een 'klant' bekijkt de foto van een product | Functioneel | 50 |
| A04-5 | Een 'klant' bekijkt de beschrijving van een product | Functioneel | 90 |
| A04-6 | Een 'klant' kiest een keuze optie van een product | Functioneel | 50 |
| A04-7 | Een 'klant' bekijkt de categorieën van een product | Functioneel | 50 |
|  |  |  |  |
| A05-1 | Een 'klant' bekijkt welke producten in de aanbieding zijn. | Functioneel | 50 |
|  |  |  |  |
| B01 | Een 'eigenaar' bekijkt de geplaatste bestellingen | Functioneel | 100 |
| B01-1 | Een 'eigenaar' bekijkt de status van een geplaatste bestelling | Functioneel | 100 |
| B01-2 | Een 'eigenaar' wijzigt de status van een geplaatste bestelling | Functioneel | 100 |
|  |  |  |  |
| B02 | Een 'klant' plaatst een bestelling | Functioneel | 100 |
| B02-1 | Een 'klant' voegt producten toe aan zijn bestelling | Functioneel | 100 |
| B02-2 | Een 'klant' verwijdert producten uit zijn bestelling | Functioneel | 80 |
| B02-3 | Een 'klant' bekijkt de totale prijs van een bestelling | Functioneel | 100 |
| B02-4 | Een 'klant' bekijkt de prijs per product binnen zijn bestelling | Functioneel | 80 |
| B02-5 | Een 'klant' bekijkt de korting per product binnen zijn bestelling | Functioneel | 70 |
| B02-6 | Een 'klant' krijgt een nummer voor zijn geplaatste bestelling. | Functioneel | 100 |
|  |  |  |  |
| B03 | Een 'klant' bekijkt zijn geplaatste bestellingen | Functioneel | 100 |
| B03-1 | Een 'klant' bekijkt de status van zijn geplaatste bestellingen | Functioneel | 90 |
| B03-2 | Een 'klant' krijgt een melding wanneer de status van zijn bestelling is gewijzigd. | Functioneel | 70 |
| B03-3 | Een 'klant' plaatst zijn geplaatste bestelling opnieuw | Functioneel | 50 |
|  |  |  |  |
| P01-1 | Een 'klant' bekijkt hoeveel punten hij heeft | Functioneel | 30 |
| P01-2 | Een 'klant' bekijkt hoeveel punten hij verdient met het plaatsen van de huidige bestelling | Functioneel | 30 |

# Diagrammen

Momenteel zijn er 2 diagrammen gemaakt, een usecase diagram en een class diagram. Er is (nog) geen sequence diagram gemaakt omdat het nog niet helemaal duidelijk is hoe ik deze moet maken met het MVC-design.

## Usecase diagram

In het onderstaand usecase diagram heeft de actor ‘Eigenaar’ toegang tot dezelfde usecases als de actor ‘Klant’. Bovendien is usecase 4 geen extent omdat deze ook kan worden uitgevoerd zonder eerst producten toe te voegen aan een winkelwagen. In dit geval zou de klant dan een voorheen geplaatste order opnieuw plaatsen.

Afbeelding met tekst, diagram, schermopname, lijn

Automatisch gegenereerde beschrijving

## Class diagram

Voor het class diagram zijn alleen de classes van de entiteiten beschreven. De classes voor de koppeltabel en het MVC-patroon zitten niet in dit diagram.

Afbeelding met tekst, schermopname, Lettertype, diagram

Automatisch gegenereerde beschrijving

# Usecase beschrijvingen

Voor deze opdracht zijn voor twee usecases beschrijvingen gemaakt.

|  |  |  |  |
| --- | --- | --- | --- |
| Naam | Verwerkt bestelling | | |
| Code | UC6 | |
| Beschrijving | Een ‘eigenaar’ verwerkt een ingestuurde bestelling van een klant. | | |
| Actor | Eigenaar | | |
| Preconditie | De eigenaar is ingelogd en een bestelling is geplaatst. | | |
| Postconditie | De bestelling is als ‘Afgehandeld’ geregistreerd in het systeem | | |
| Stappen | 1. De gebruiker navigeert naar de ‘bestellingen’ pagina |  |
|  |  | 2. Het systeem toont de ‘bestellingen’ pagina met daarop alle ingestuurde bestellingen |
|  | 3. De gebruiker voert een besteltijd in |  |
|  | 3a. De gebruiker voert geen besteltijd in |  |
|  | 4. De gebruiker drukt op ‘Bevestig’ |  |
|  |  | 5. Het systeem verandert de status van de bestelling naar ‘Aan het bereiden’ |
|  |  | 5a. Het systeem geeft een melding dat er geen besteltijd is ingevuld. |
|  | 6. De gebruiker drukt op ‘Ophalen’ |  |
|  |  | 7. Het systeem verandert de status van de bestelling naar ‘Klaar om op te halen’ |
|  | 8. De gebruiker drukt op ‘Afhandelen’ |  |
|  |  | 9. Het systeem verandert de status van de bestelling naar ‘Afgehandeld’ |
|  |  | 10. Het systeem verwijdert de bestelling van de pagina |
| Hoofdscenario | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 |  |
| Alt scenario |  |  |
| Fout scenario | 1, 2, 3a, 4, 5a | De gebruiker voert geen besteltijd in |

|  |  |  |  |
| --- | --- | --- | --- |
| Naam | Voegt producten toe aan winkelwagen | | |
| Code | UC3 | |
| Beschrijving | De gebruiker voegt producten uit het assortiment toe aan zijn winkelwagen. | | |
| Actor | Klant/Eigenaar | | |
| Preconditie | UC2: De gebruiker bekijkt het assortiment. | | |
| Postconditie | De gebruiker heeft producten in zijn winkelwagen | | |
| Stappen | 1. De gebruiker navigeert naar de ‘bestellen’ pagina. |  |
|  |  | 2. Het systeem toont de ‘bestellen’ pagina met daarop alle producten in het assortiment. |
|  | 3. De gebruiker vult de hoeveelheid in van het product |  |
|  | 3a. De gebruiker past de hoeveelheid aan naar een getal lager dan 1. |  |
|  | 4. De gebruiker drukt op ‘Toevoegen’ |  |
|  |  | 5. Het systeem toont een pagina om het product samen te stellen |
|  | 6. De gebruiker kiest een ‘Maat’ uit van het product |  |
|  | 6a. De gebruiker kiest een saus voor bij het product |  |
|  | 6b. De gebruiker past de hoeveelheid aan van het product |  |
|  | 6c. De gebruiker past de hoeveelheid aan naar een getal lager dan 1. |  |
|  | 7. De gebruiker drukt op ‘Toevoegen aan winkelwagen’ |  |
|  |  | 8. Het systeem voegt het product toe aan de winkelwagen. |
|  |  | 8a. Het systeem toont een melding dat de hoeveelheid niet lager dan 1 kan zijn. |
|  |  | 9. Het systeem keert terug naar de ‘bestellen’ pagina |
|  |  | 10. Het systeem toont een melding dat het product is toegevoegd aan de winkelwagen |
|  | 11. De gebruiker sluit de melding |  |
| Hoofdscenario | 1, 2, 3, 4, 5, 6, 6a, 7, 8, 9, 10, 11 |  |
| Alt scenario | 1, 2, 4, 5, 6, 6a, 6b, 7, 8, 9, 10, 11 | De gebruiker verandert de hoeveelheid van het product later |
|  | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 | Het gekozen product heeft geen ‘saus’ optie |
| Fout scenario | 1, 2, 3a, 4, 8a, 11 1, 2, 3, 4, 5, 6b, 7, 8a, 11 | De gebruiker vult een hoeveelheid in dat lager is dan 1 |

# Wireframes

De wireframes zijn gemaakt met het programma ‘Figma’. Dit is een gratis programma om designs mee te maken. De wireframes hebben navigatie functionaliteit ingebouwd dus wanneer het bijgeleverde Figma bestand wordt bekeken kan er net zoals een website door genavigeerd worden.

Het bijgeleverde Figma bestand heeft de naam: ‘Wireframes HetFrietje.fig’.

Om Figma te downloaden kun je de volgende link openen: <https://www.figma.com/downloads/>

# Source control

Voor dit project is voor de source control gebruik gemaakt van GitHub. De repository heeft de HetFrietje. De sourcecode en gemaakte bestanden zijn ook te vinden op deze repository.

Link naar de repository: <https://github.com/BasVanSch1/>