# OEFENCASUS PIZZA KOERIER

# Inleiding

De oefencasus De Pizza koerier wordt tijdens BIS Functioneel ontwerp gebruikt om de theorie te oefenen. De opdrachten voor de Pizza koerier maak je individueel. De opdrachten worden opgestuurd naar de docent en in de klas behandeld. Na de les krijg je de uitwerking van de opdracht.

Je gebruikt verschillende tools om processen, use case diagrammen en wireframes te tekenen. Neem in de opdrachten die je opstuurt naar de docent (en straks in het beroepsproduct) altijd een uitwerking in PDF op, zodat deze door iedereen kan worden ingezien, ook wanneer diegene niet het tool heeft dat jij hebt gebruikt.

Voor het beroepsproduct maak je vergelijkbare opdrachten in groepsverband voor een grotere casus.

# Beschrijving Pizza Koerier

De eigenaar van de Pizza koerier wil een nieuw systeem voor zijn pizzeria. In de huidige situatie wordt nog veel handmatig gedaan en kan geen goede service aan klanten worden geleverd. Zo heeft de eigenaar b.v. nog geen website waar klanten pizza’s kunnen bestellen.

**Gewenste situatie:**

De klant bestelt via de website, telefonisch of aan de balie een of meerdere pizza’s.

Via de websites krijgt de klant een overzicht van pizza’s en de dagaanbiedingen. Als de klant via de website besteld kan hij pizza’s kiezen, bestellen en betalen. Als de pizza is betaald krijgt de klant een bevestiging en een schatting van de levertijd.

Klanten kunnen pizza’s thuis laten afleveren of zelf ophalen bij de pizzeria. Als de pizza’s thuis worden bezorgd worden € 5,-- bezorgkosten gerekend.

Pizza’s die via de website besteld worden moeten betaald worden voordat de bestelling wordt omgezet naar een order (de klant krijgt een bevestiging van de succesvolle betaling in de vorm van een kassabon).

Als de klant telefonisch bestelt wordt de bestelling aangenomen door de balie-medewerker en wordt de bestelling geregistreerd. Voor een telefonische bestelling geldt dat de klant bij de pizzeria bekend moet zijn.

Controle vindt plaats door het uitvragen van postcode huisnummer in combinatie met de geboortedatum van de besteller. Als de klant niet bekend is wordt geen telefonische bestelling aangenomen.

Pizza’s die aan de balie worden besteld moeten direct betaald worden (de klant krijgt een kassabon). Telefonisch bestelde pizza’s worden bij aflevering betaald (de klant krijgt van de bezorger een kassabon).

Alle bestellingen worden door de orderwerker ingepland. De orderverwerker bepaalt de verwachte levertijd.

De pizzabakker bakt de bestellingen die zijn ingepland. Als de pizza’s gebakken zijn meldt de bakker de bestelling gereed en begint hij met de volgende bestelling. Er worden maximaal drie bestellingen verzameld (om in één keer bezorgd te worden). De orderverwerker geeft de pizza’s aan de pizzabezorger en de kassabon van de telefonisch bestelde pizza’s, die nog niet betaald zijn.

De pizzabezorger levert de bestellingen af bij de pizzaklant(en). De pizzabezorger bepaalt zelf de optimale route om de bestellingen bij klanten te bezorgen. Bij telefonische bestellingen wordt er bij aflevering betaald. De klant ontvangt een kassabon.

Als de klant een account heeft aangemaakt kan de klant de status van de pizza volgen.

De eigenaar wil vaste klanten met een account op termijn voordelen gaan bieden, zoals b.v. aanbiedingen en korting bij een volgende bestelling. Hij wil ook nieuwsbrieven en reclame gaan versturen.

De eigenaar heeft een bedrijf ingehuurd om het systeem te bouwen. Het verder ontwikkelen en het beheer van de site voor de Pizza koerier wordt uitbesteed. Het aanpassen van de content (b.v. aanbiedingen, nieuwe pizza’s) wordt door de eigenaar zelf gedaan.

# Opdracht 1. Beschrijf de processen

Dit functioneel ontwerp start met het beschrijven van het proces. Met het beschrijven van het proces wordt in beeld gebracht welke activiteiten er binnen dat proces plaatsvinden en welke activiteiten mogelijk door een informatiesysteem kunnen worden ondersteund (dat is een keuze die je samen met de opdrachtgever maakt).

Beschrijf het proces van de Pizza Koerier in BPMN level 1. Hou bij welke vragen je over de casus hebt. Die worden in de klas aan de opdrachtgever, de eigenaar van de Pizza koerier, gesteld.

Voor het maken van de BPMN modellen wordt bij voorkeur Aris-express of Bizagi modeler gebruikt. (tenzij je zelf bekend bent met een ander geschikt tool voor het maken van een BPMN model; deze twee tools zorgen ervoor dat je exclusief de beschikking hebt over de BPMN symbolenset, en niet ook nog allerlei andere symbolen)

Je gebruikt hiervoor al de symbolen van BPMN level 1, m.u.v. de data store en het data object. De data store en het data object beschrijven of data wordt vastgelegd. Dat hoeft nu nog niet. Het activiteittype voeg je pas toe nadat je de activiteiten hebt gedocumenteerd (zie opdracht 3).

*Tip:*

Het is bij het beschrijven van de taken soms moeilijk om te bepalen wat het juiste detailniveau is. Op pagina 39 t/m 46 van het boek ‘Use cases opstellen en testen’ van Sander Hoogendoorn wordt dit uitgelegd:

*Bedrijfsprocessen zijn in een diagram op te splitsen tot elementaire bedrijfsprocessen. Een elementair bedrijfsproces kenmerkt zich doordat het door één persoon op eén tijdstip op één plaats in de tijd wordt uitgevoerd.*

In BPMN terminologie betekent dit dat een bedrijfsproces op taakniveau beschreven moet worden.

# Opdracht 2. Documenteer de processen

Om te zorgen dat het BPMN model wordt begrepen door opdrachtgever en zijn medewerkers, de gebruikers, moet het BMPN model gedocumenteerd worden.

Documenteer de taken van de Pizza Koerier volgens de IPO methode (BIS reader hoofdstuk 1).

*Tips:*

* Zorg voor een duidelijke omschrijving van de input en de output. De namen van input en output zijn zelfstandige naamwoorden.
* De namen van taken zijn werkwoorden.
* De output van de taak is vaak input voor de volgende taak.

# Opdracht 3. Bepaal de te automatiseren taken

Beoordeel welke taken geautomatiseerd uitgevoerd kunnen worden. Geef in het BPMN schema uit opdracht 1 wat voor type taak (user task of service task) het is.

# Opdracht 4. Stel requirements op

De processen zijn beschreven en er is bekend welke taken geautomatiseerd uitgevoerd kunnen worden. De requirements (eisen) die gesteld worden aan het systeem worden in deze opdracht beschreven.

Requirements worden opgesteld op verschillende niveaus. Er zijn ook verschillende soorten requirements. Stel de volgende requirements op:

1. Business needs. Business needs worden in natuurlijke taal SMART beschreven.   
   Het aantal business needs is niet groot (meestal tussen de 2-5), omdat hiermee op strategisch niveau het doel wordt vastgelegd.
2. Functionele (gebruikers) requirements. Gebruik hiervoor de user stories techniek.

Functionele requirements worden opgesteld op elementair procesniveau (dat is het taakniveau in BPMN).

1. Niet-functionele requirements. Niet-functionele requirements worden in natuurlijke taal SMART beschreven.

In de case van de Pizza Koerier zijn geen niet-functionele requirements opgenomen. Het is vaak moeilijk voor de opdrachtgever en gebruikers om de niet-functionele requirements te definiëren, omdat ze hier geen ervaring mee hebben. Meestal stelt de analist deze requirements op, op basis van interviews die gehouden zijn met o.a. opdrachtgever, gebruikers en architecten e.d.

Beoordeel welke niet-functionele requirements voor deze case gebruikt kunnen worden. Stel tussen de 1 - 5 niet functionele requirements op.

*Tips:*

* Voor iedere taak die geautomatiseerd wordt moeten één of meer gebruikersrequirements opgesteld zijn. Controleer of je volledig bent.
* Nummer de requirements. Bij het opstellen van de use cases wordt naar de requirements verwezen om te controleren of al de requirements zijn afgedekt door use cases.
* Ieder user taak en server taak heeft een of meerdere gebruikersrequirements.

# Opdracht 5. Prioriteer de functionele (gebruikers) requirements

De eigenaar van de Pizzazaak heeft een beperkt budget voor de bouw van het informatiesysteem. Hij moet heel kritisch zijn op welke requirements hij echt nodig heeft en welke niet. In deze opdracht worden de gebruikersrequirements geprioriteerd.

*Werkwijze: (tijdens de face to face les)*

* Vorm groepjes van vijf personen:

Kies iemand die de voorzittersrol heeft en de tijd in de gaten houdt.

Kies iemand die de resultaten presenteert.

Kies een opdrachtgever (de eigenaar van de Pizza zaak) en gebruikers (werknemers van de Pizza zaak).

* Selecteer van één of meerdere studenten een overzicht met opgestelde gebruikersrequirements.

Gebruik voor het prioriteren van de requirements de Moscow techniek. Bepaal eerst individueel welke requirements Must have, Should have, Could Have en Would not have zijn.

* Vergelijk de verschillende prioriteiten van de individuele studenten. Kijk wat de verschillen zijn. Beargumenteer de individuele keuzes en probeer tot overeenstemming te komen. Als dat niet lukt bepaalt de opdrachtgever de prioriteit. Leg vast welke argumenten gebruikt zijn voor de gekozen prioriteit.
* Presenteer het overzicht met gebruikersrequirements en de prioriteiten. Licht de gemaakte keuzes toe.

*Duur 45 minuten*

30 minuten Opstellen prioriteiten van de requirements

15 minuten presenteren van het resultaat

# Opdracht 6. Stel een use case diagram op

In deze opdracht wordt een use case diagram opgesteld en gedocumenteerd. Een use case diagram geeft de functies weer van het te bouwen informatiesysteem.

Stel het use case diagram op door volgende acties uit te voeren:

1. Identificeer de use cases en geef de use cases een naam.
2. Identificeer de actoren en geef de actoren een naam.
3. Geef aan wat de associaties zijn tussen de actoren en de use cases.
4. Teken de use cases, de actoren en de associaties in een use case diagram.
5. Documenteer de actoren door bij iedere actor: de naam, de omschrijving en de indicatie primaire/secundair aan te geven. Maak hiervoor een tabel.
6. Documenteer de use cases m.b.v. brief use cases (BIS reader, hoofdstuk 4).
7. Geef bij iedere use case de nummers van de gebruikersrequirements aan die bij de use case horen. Zorg dat al de gebruikersrequirements zijn opgenomen en dat deze niet dubbel voorkomen.
8. Het kan zijn dat bij het documenteren van een use case (of straks bij het uitwerken van een brief use case in een fully dressed use case) blijkt dat een use case te groot of te klein is gekozen. Splits de use case of voeg deze samen. Pas het use case diagram vervolgens aan.
9. Het opstellen van een use case diagram is iteratief. Je doorloopt de acties waarschijnlijk een paar maal.
10. Teken de systeemgrens als je klaar bent.

*Tips voor het identificeren en beschrijven van use cases:*

* Use cases zijn de taken/elementaire processen van BPMN model. Gebruik het BPMN model dat in opdracht 1 en 3 is gemaakt.
* Gebruik de tips voor de granulariteit op pagina 54 uit het boek ‘Use cases opstellen en testen’ van Sander Hoogendoorn voor het identificeren van use cases.
* De naam van een use case bevat een werkwoord

*Tips voor identificeren en beschrijven van actoren:*

* Actoren zijn geen personen/functionarissen. Actoren zijn rollen.
* De namen van actoren zijn zelfstandige naamwoorden.
* Primaire actoren staan links in het use case diagram. Secundaire actoren staan rechts in het use case diagram.

*Tips voor het use case model:*

* Nummer de use cases
* Zorg dat de (communicatie) associatie naar de actoren juist is
* Wees je bewust van het gebruik van includes (wordt meerdere keren gebruikt) en extends (extra optioneel gedrag).

**Toelichting op Opdracht 7 en Opdracht 8.**

Let op: je kunt niet afzonderlijk de wireframes en de fully dressed use cases apart van elkaar uitwerken. Deze onderdelen zijn namelijk aan elkaar gerelateerd. Als je een aantal wireframes hebt opgesteld is het sowieso gewenst ook de bijbehorende fully dressed use cases uit te werken.

*Tip: Begin met het uitwerken van de wireframe(s).*

# Opdracht 7. Maak wireframes

Maak low of high fidelity wireframes voor Pizza Koerier. Gebruik hiervoor het stappenplan: Stappenplan Wireframes: <https://webdesign.tutsplus.com/articles/a-beginners-guide-to-wireframing--webdesign-7399>

Kies zelf een tool of gebruik het tool dat in de weekbrief staat (Axure).

Zorg voor de documentatie van de wireframes (zei BIS Reader, bijlage Wireframes)

# Opdracht 8. Stel fully dressed use cases op

Werk voor deze opdracht de brief use cases uit in fully dressed use cases voor de use case bestellen pizza. Gebruik het format voor de fully dressed use cases uit de BIS reader, hoofdstuk 4.

Tips voor het beschrijven van fully dressed use cases:

* Modelleer een nieuwe use case als er een onevenredig aantal stappen onder eenzelfde conditie worden uitgevoerd. Verplaats deze stappen naar een nieuwe use case.
* Wanneer je op zoek bent naar vocabulaire gebruik dan tabel 3 op pagina 56 uit het boek ‘Use cases opstellen en testen’ van Sander Hoogendoorn.

# Toelichting bij opdrachten 9 en 10

Een informatiebeveiligingsplan is een document waarin beschreven staat welke beveiligingsmaatregelen gekozen zijn en waarom de betreffende maatregelen gekozen zijn.

In de aangereikte literatuur wordt een aantal stappen beschreven om te komen tot een set van basismaatregelen. De opdrachten 9 en 10 helpen je om deze stappen uit te voeren.

*Opdracht 9 Pizzakoerier - Beschrijf risico’s en maatregelen*

Gebruik de reader informatiebeveiliging of een andere bron om te bepalen hoe je risico’s kunt beschrijven / scoren.

Inventariseer de risico’s voor het informatiesysteem en werk voor de 3 belangrijkste risico’s een aantal noodzakelijke maatregelen uit. Let op dat jullie voor één risico meerdere maatregelen kunnen uitwerken.

*Opdracht 10. Pizzakoerier - Geef een overzicht van maatregelen op basis van wet- en regelgeving*

Kies één van de op <https://nl.wikipedia.org/wiki/Informatiebeveiliging> genoemde wetten, en bedenk voor deze wet 3 maatregelen die een bedrijf zou kunnen toepassen om aan de wettelijke verplichtingen te voldoen. Gebruik de reader over het informatiebeveilingsplan als inspiratiebron, maar zorg wel dat je kunt uitleggen waarom deze maatregel juist voor de Pizzakoerier nodig is, informatiebeveiliging doe je altijd in een bepaalde context.