ELU 1.3 Functioneel Ontwerp



**Naam groep 1:** MountQua  
**Naam groep 2:** Student Services

**Docent:** Dhr. Rozendaal

**Naam auteurs:** Alexander Trappenberg, Levi den Ouden, Kaylee Rietveld, Joep Oomens, Qui Nguyen

**Plaats & datum:** Breda, Maart 2020

Table of Contents

[1. Inleiding](#_Toc20388_WPSOffice_Level1) [4](#_Toc20388_WPSOffice_Level1)

[2. Beschrijving nieuwe informatiesysteem](#_Toc30562_WPSOffice_Level1) [5](#_Toc30562_WPSOffice_Level1)

[2.1 Requirements](#_Toc30562_WPSOffice_Level2) [5](#_Toc30562_WPSOffice_Level2)

[3. Gegevensmodel](#_Toc15375_WPSOffice_Level1) [6](#_Toc15375_WPSOffice_Level1)

[Afbeelding 1: ERD](#_Toc15375_WPSOffice_Level2) [6](#_Toc15375_WPSOffice_Level2)

[3.1 Omschrijving entiteiten en attributen](#_Toc28870_WPSOffice_Level2) [7](#_Toc28870_WPSOffice_Level2)

[4. Gewenste uitvoer](#_Toc28870_WPSOffice_Level1) [11](#_Toc28870_WPSOffice_Level1)

[5. Benodigde invoer](#_Toc16985_WPSOffice_Level1) [14](#_Toc16985_WPSOffice_Level1)

[6. Menustructuur en autorisatie](#_Toc22183_WPSOffice_Level1) [16](#_Toc22183_WPSOffice_Level1)

[7. Organisatorische consequenties](#_Toc13848_WPSOffice_Level1) [17](#_Toc13848_WPSOffice_Level1)

[7.1 Testen](#_Toc16985_WPSOffice_Level2) [17](#_Toc16985_WPSOffice_Level2)

[7.2. Acceptatie](#_Toc22183_WPSOffice_Level2) [17](#_Toc22183_WPSOffice_Level2)

[7.3 Conversie](#_Toc13848_WPSOffice_Level2) [17](#_Toc13848_WPSOffice_Level2)

[8. Technische consequenties](#_Toc23655_WPSOffice_Level1) [18](#_Toc23655_WPSOffice_Level1)

[8.1 Werkplekken](#_Toc23655_WPSOffice_Level2) [18](#_Toc23655_WPSOffice_Level2)

[8.2 Apparatuur](#_Toc3722_WPSOffice_Level2) [18](#_Toc3722_WPSOffice_Level2)

[8.3 Installatie](#_Toc28705_WPSOffice_Level2) [18](#_Toc28705_WPSOffice_Level2)

[9. Usecases](#_Toc3722_WPSOffice_Level1) [19](#_Toc3722_WPSOffice_Level1)

[10. Usecasediagram](#_Toc28705_WPSOffice_Level1) [23](#_Toc28705_WPSOffice_Level1)

# Inleiding

In dit ontwerp zal de uitwerking van het probleem dat in het informatieplan is beschreven verder worden uitgewerkt. Zoals al naar voren is gekomen streeft MountQua naar duurzame en winstgevende groei. Vanwege de vele menselijke fouten en een gedateerd bestelsysteem is het genoodzaakt om een andere weg in te slaan. Om dit te kunnen realiseren is het van belang om een geautomatiseerd bestelsysteem in de organisatie te introduceren. Dit geautomatiseerde systeem zal zelf door MountQua gerealiseerd worden.

Verder zal het functionele ontwerp een overzicht bevatten van de verschillende klassen en objecten. Ook zijn de use-cases uitgewerkt zodat hiervan een POC gemaakt kan worden.

# Beschrijving nieuwe informatiesysteem

Het nieuwe informatiesysteem zal het voor klanten mogelijk maken om bestellingen via het internet te plaatsen. Tevens kunnen zij de bestelhistorie inzien. Het is noodzakelijk om eerst een geregistreerde gebruiker te zijn om gebruik te maken van deze functies. Op de producten pagina zal een bestelknop komen om het product aan de winkelwagen toe te voegen. Hier is ook de mogelijkheid om de hoeveelheid artikelen te bepalen. De klant kan de hoeveelheid handmatig invoeren of met de pijl-toetsen het correcte aantal aan te geven. Als de klant het aantal producten heeft ingevoerd zal dit in het winkelmandje worden aangepast. Vervolgens is het de keuze om verder te winkelen of om de bestelling af te ronden.

Het nieuwe informatieysteem bevat dus 2 procedures. De registratieprocedure en de bestelprocedure.ten all

Om te kunnen registreren moet de gebruiker op de knop “Registreer” kunnen drukken. Hierna volgt een formulier om de registratie te kunnen voltooien.

## 2.1 Requirements

Het bestelsysteem moet aan bepaalde voorwaarden voldoen zodat het duidelijk is voor de ontwikkelaars en stakeholders wat er precies verwacht gaat worden. Deze requirements zijn smart opgesteld en hebben ieder een apart belang.

Requirements Belang

1: De gebruiker kan een account aanmaken. Essentieel

2: De applicatie verstuurd bevestigingsmail van registratie naar klant. Essentieel

3: De applicatie werkt actuele voorraad bij. Essentieel

4: De gebruiker kan een of meerdere artikelen bestellen. Essentieel

5: De applicatie verstuurd bevestigingsmail van bestelling naar klant. Essentieel

6: De applicatie verstuurd facturatie naar financiele afdeling. Essentieel

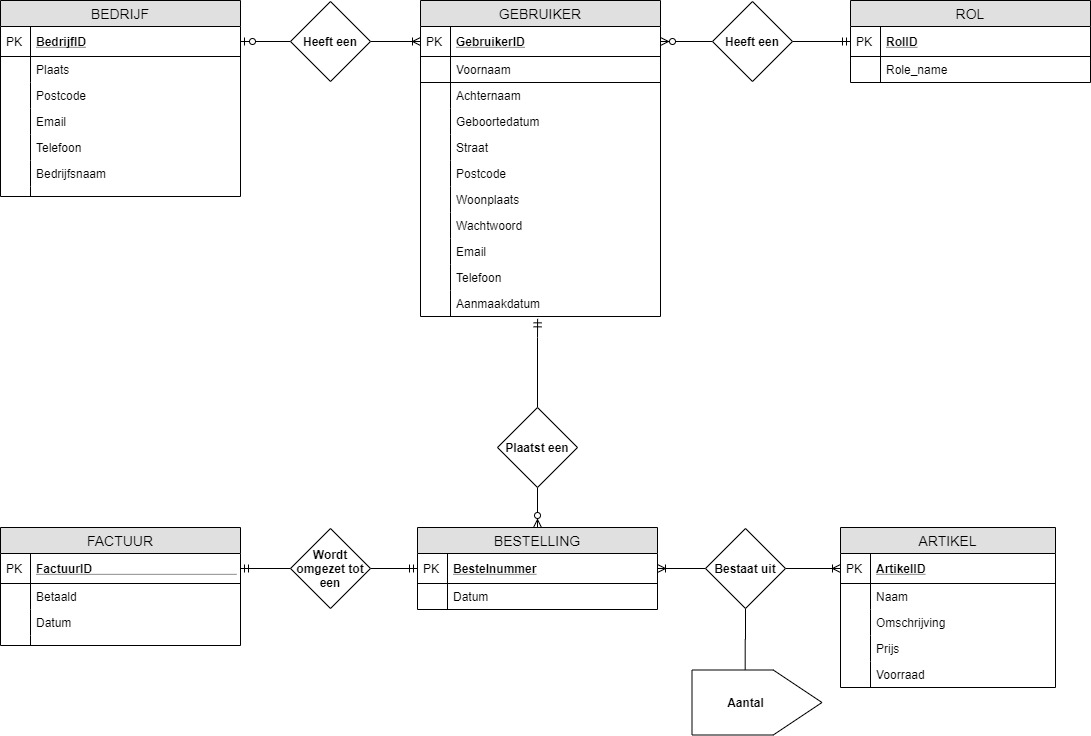
7: De applicatie verstuurd order naar magazijn. Essentieel

8: De gebruiker kan een overzicht van zijn accountgegevens zien. Essentieel

9: De gebruiker kan een overzicht van zijn bestelgeschiedenis zien. Essentieel

# Gegevensmodel

Dit gegevensmodel bestaat uit de benodigde entiteiten, attributen en een entiteit-relatiediagram. Hierin wordt bepaald welke gegevens van belang zijn en welke relaties er bestaan tussen de gegevens. Het ERD is in periode 2 gerealiseerd en zal in dit blok verder gehandhaafd worden.



Figuur 1: ERD

Bovenstaande ERD geeft een overzicht van de verschillende entiteiten en de relaties onderling. De intersectie tussen bestelling en artikel wordt in het ERD als een ruit weergegeven maar zal in de database een aparte tussen tabel zijn. Vandaar dat deze ook is uitgewerkt in de beschrijving van de entiteiten. De reden hiervan is dat er in een ERD geen tussen entiteiten worden weergegeven.

## 3.1 Omschrijving entiteiten en attributen

Entiteit-Bedrijf

|  |  |
| --- | --- |
| **Naam** | **Bedrijf** |
| **Omschrijving** | Elke gebruiker is gebonden aan het bedrijf. |
| **Aantal records** | 1 |
| **Attributen** | BedrijfsID type: INT  Het bedrijfsID waaraan de vestiging is te herkennen. Indien er een vestiging bijkomt zal deze een nieuw bedrijfsID krijgen. |
|  | Plaats type: VARCHAR  De plaats waar het bedrijf is gevestigd. |
|  | Postcode type:VARCHAR  Postcode vestiging. |
|  | Email type:VARCHAR  Emailadres MountQua. |
|  | Telefoon type:VARCHAR  Telefoonnummer van de vestiging. |
|  | Bedrijfsnaam type:VARCHAR  Naam van het bedrijf. |

Entiteit-Gebruiker

|  |  |
| --- | --- |
| **Naam** | **Gebruiker** |
| **Omschrijving** | Elke gebruiker bij MountQua is geregistreerd. Zonder registratie zal men geen bestelling kunnen plaatsen. |
| **Aantal records** | 5, maar dit zal toenemen naarmate er meer gebruikers zich registreren. |
| **Attributen** | GebruikerID type: INT  De gebruikerID zal de primary key zijn om gebruikers te vinden en identificeren. |
|  | Voornaam type:VARCHAR |
|  | Achternaam type:VARCHAR |
|  | Geboortedatum type:VARCHAR |
|  | Straat type:VARCHAR  Woonadres van de gebruiker. |
|  | Postcode type:VARCHAR  Postcode van de gebruiker. |
|  | Woonplaats type:VARCHAR  Woonplaats van de gebruiker |
|  | Wachtwoord type:VARCHAR  Wachtwoord van de gebruiker. |
|  | Email type:VARCHAR  Emailadres van de gebruiker. |
|  | Telefoon type:VARCHAR  Telefoonnummer gebruiker. |
|  | Aanmaakdatum type:VARCHAR  Datum waarop account is aangemaakt. |

Entiteit - Rol

|  |  |
| --- | --- |
| **Naam** | **Rol** |
| **Omschrijving** | De entiteit rol zal bepalen welke rechten de verschillende gebruikers hebben. |
| **Aantal records** | 1 |
| **Attributen** | RolID type:INT  Dit RolID zal bepalen welke permissies een gebruiker heeft. |
|  | Rol\_Name type:VARCHAR  Beschrijft de naam van de rol. Bv: admin, read-only of edit. |

Entiteit-Factuur

|  |  |
| --- | --- |
| **Naam** | **Factuur** |
| **Omschrijving** | De factuur entiteit wordt automatisch aangemaakt nadat een bestelling is gemaakt. |
| **Aantal records** | 1, maar dit zal toenemen naarmate er meer bestellingen binnenkomen. |
| **Attributen** | FactuurID type:INT  Het factuur wordt gekenmerkt door de factuurID. |
|  | Betaald type:BOOL  Betaalstatus van de factuur. |
|  | Datum type:DATE  Datum waarop factuur is verstuurd. |

Entiteit-Bestelling:

|  |  |
| --- | --- |
| **Naam** | **Bestelling** |
| **Omschrijving** | De entiteit bestelling wordt aangemaakt zodra een gebruiker een bestelling plaatst. |
| **Aantal records** | 1, maar dit zal toenemen naarmate er meer bestellingen binnenkomen. |
| **Attributen** | Bestelnummer type:INT  De bestelling wordt gekenmerkt met een bestelnummer. Elke bestelling heeft een uniek bestelnummer. |
|  | Datum type:DATE  Datum waarop de bestelling is geplaatst. |

Entiteit-Artikel:

|  |  |
| --- | --- |
| **Naam** | **Artikel** |
| **Omschrijving** | Elk artikel heeft een aparte tabel in de database. |
| **Aantal records** | 3, maar dit wordt meer als MountQua er voor kiest om meerdere producten te gaan verkopen. |
| **Attributen** | ArtikelID type:INT  Elk artikel heeft een eigen artikelID. |
|  | Naam type:VARCHAR  Naam van het artikel. |
|  | Omschrijving type:VARCHAR  Korte omschrijving van het product zodat de klant weet wat hij/zij koopt. |
|  | Prijs type:DECIMAL  Prijs van het artikel. |
|  | Voorraad type:INT  Toont de hoeveelheid voorraad. |

Entiteit-Bestelling\_Artikel

|  |  |
| --- | --- |
| **Naam** | **Bestelling\_Artikel** |
| **Omschrijving** | De bestelling\_artikel entiteit is gemaakt om de hoeveelheid per product te kunnen bepalen. |
| **Aantal records** | 2, maar dit zal toenemen naarmate meer bestellingen binnenkomen. |
| **Attributen** | ArtikelID type:INT  ArtikelID dat bij het artikel hoort. |
|  | Bestelnummer type:INT  Bestelnummer dat bij de bestelling hoort. |
|  | Aantal type:INT  Aantal exemplaren van het artikel in een bestelling. |

# Gewenste uitvoer

De gewenste uitvoer zal altijd op het beeldscherm van de gebruiker te zien zijn. Dit betekent dat de gebruiker zijn bestelling kan inzien en tevens ook een overzicht van zijn bestelgeschiedenis kan inzien.

Een uitzondering op deze regel is het uitprinten van de factuur. De financiele afdeling zal de factuur handmatig uitprinten en deze versturen naar de klant.

Verder zal de gebruiker een bevestigings-pagina te zien krijgen wanneer hij/zij zich succesvol registreerd.

Om de gewenste uitvoer te kunnen tonen is er dus directe invoer van de gebruiker noodzakelijk.

Het opvragen van informatie via het beeldscherm wordt beschouwd als een uitvoertaak. Het resultaat van de uitvoertaak is een uitvoerproduct.

Onderstaande tabel wordt gebruikt om de uitvoerproducten weer te geven.

|  |  |
| --- | --- |
| **Code uitvoerproduct** | **Overzicht 001** |
| Naam | Overzicht accountgegevens |
| Gebruiker | Klant |
| Doel | Duidelijke weergave van de persoonlijke gegevens van de klant. |
| Af te drukken gegevens (plus berekeningen) | A Naam  B Achternaam  C Geboortedatum  D Adres  E Contactgegevens |
| Opmerkingen |  |

|  |  |
| --- | --- |
| **Code uitvoerproduct** | **Overzicht 002** |
| Naam | Overzicht bestelgeschiedenis |
| Gebruiker | Klant |
| Doel | Duidelijke weergave van de bestellingen van de klant. |
| Af te drukken gegevens (plus berekeningen) | A Bestelnummer  B FactuurID  C Totaal bedrag bestelling  C Datum bestelling |
| Opmerkingen | Per bestelling worden de af te drukken gegevens opgevraagd. |

|  |  |
| --- | --- |
| **Code uitvoerproduct** | **Overzicht 003** |
| Naam | Overzicht bestelling |
| Gebruiker | Klant |
| Doel | Duidelijke weergave van de bestelling. |
| Af te drukken gegevens (plus berekeningen) | A Bestelnummer  B Naam artikelen  C Aantal artikelen  D Totaal bedrag bestelling  E Datum bestelling |

# Benodigde invoer

|  |  |
| --- | --- |
| **Code invoertaak** | **Invoertaak 001** |
| Naam van de taak | Registratie account |
| Autorisatie | Gebruiker |
| Doel | Registreren van klant |
| Beschrijving werkwijze | Via de knop registreer kan de klant een account registreren. Via het registratie formulier worden de onderstaande gegevens ingevoerd.   * Naam en achternaam * Adresgegevens * Contactgegevens |
| Bestanden | Gebruiker record wordt toegevoegd |
| Gebruikte schermen | Registratiescherm gebruiker |

|  |  |
| --- | --- |
| **Code invoertaak** | **Invoertaak 002** |
| Naam van de taak | Plaatsen bestelling |
| Autorisatie | Gebruiker |
| Doel | Gebruiker plaatst bestelling |
| Beschrijving werkwijze | Op de artikelen pagina kan de gebruiker zijn artikel kiezen en toevoegen aan de winkelwagen met een button. Om de bestelling te plaatsen is ook een bestelbutton toegevoegd.   * Artikelnaam * Aantal * Plaats bestelling |
| Bestanden | Bestelling record wordt toegevoegd  Bestelling\_Artikel record wordt toegevoegd. |
| Gebruikte schermen | Artikelen pagina gebruiker  Bestel pagina gebruiker |

# Menustructuur en autorisatie

Niet iedereen mag elke gebruikerstaak uitvoeren. Zoals al naar voren is gekomen zijn er 3 soorten autorisaties. Met een x wordt aangegeven wie geautoriseerd is voor een bepaalde functie

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Menustructuur | | | Autorisaties | | | |
| Hoofdmenu | Submenu |  | Admin | Read-only | Edit |  |
|  |  |  |  |  |  |  |
| Beheer | Basistabellen | Gebruikertabel | x |  | x |  |
|  |  | Bestellingtabel | x | x |  |  |
|  |  | Artikeltabel | x | x |  |  |
|  |  |  |  |  |  |  |
|  | Authorisatie |  | x |  |  |  |
|  | Systeeminstellingen |  | x |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| Invoeren | Klant |  | x |  |  |  |
|  | Artikel |  | x |  |  |  |
|  | BestelID |  | x |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| Rapporten | Financieel | Factuurtabel | x |  | x |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# Organisatorische consequenties

Binnen de organisatie MountQua zal er veel veranderen als het nieuwe bestelsysteem eenmaal in werking treedt. Dit betekend dus veel organisatorische consequenties voor iedereen binnen het bedrijf.

## 7.1 Testen

De applicatie zal getest moeten worden en dit zal gedaan worden door het ontwikkelteam zelf op een lokale server.

## 7.2. Acceptatie

Omdat het nieuwe bestelsysteem een geheel nieuw systeem op zich zelf is zal de acceptatie door gebruikers makkelijker zijn. Dit omdat ze gewend zijn bestellingen telefonisch of via de mail te doen en nu de mogelijkheid hebben om het via de website zelf te bestellen door middel van een handige knop.

## 7.3 Conversie

Om over te gaan van het oude naar het nieuwe systeem hoeft er niet veel te veranderen. Er zal meer geautomatiseerd worden en dit betekend minder werkzaamheden voor het personeel. Wel zal er natuurlijk onderling tussen afdelingen goed besproken moeten worden wie wat mag/moet uitvoeren.

# Technische consequenties

Zowel op organisatorisch vlak als technisch vlak zijn er consquenties. De technische consequenties voor MountQua zullen daarom hier ook besproken worden.

## 8.1 Werkplekken

Er zullen werkplekken moeten komen voor het ontwikkelaars team die de applicatie zal bouwen.

## 8.2 Apparatuur

Indien noodzakelijk zullen er extra beeldschermen, muizen en toetsenborden moeten worden aangeschafd. MountQua zal een server gebruiken van een derde partij omdat dit rendabeler is.

## 8.3 Installatie

Het ontwikkelaars team van MountQua zal de installatie van de software op het netwerk installeren.

# Usecases

|  |  |
| --- | --- |
| **Usecase Name** | Activatie mail versturen naar klant |
| **Unique Usecase ID** | 1 |
| **Description** | Wanneer een klant geen account heeft maakt de klant een account aan. Het systeem moet een mail ter activatie versturen naar de klant. Als klant activatie bevestig moeten de gegevens van de klant opgeslagen worden en moet de klant inloggegevens ontvangen. |
| **Actors** | PHP mail function, DB |
| **Pre-conditions** | Klant maakt account aan; Klant heeft een valid email adres; Database is beschikbaar |
| **Flow events** | 1. Klant vraagt account aan 2. Script ontvangt data van aanmeld form 3. Script stuurt mail naar opgegeven mail adres met activatie link 4. Klant drukt op activatie link 5. Script stuurt aanmeld gegevens naar database 6. Script stuurt gebruikersnaam en tijdelijk wachtwoord naar klant |
| **Exceptions** | Klant bevestigd activatie niet |
| **Postcondities** | Het opvangen en versturen van data als deze wordt aangeleverd; Succesvol mails versturen naar klant |

Figuur 2: Usecase 1

|  |  |
| --- | --- |
| **Usecase Name** | Product voorraad check |
| **Unique Usecase ID** | 2 |
| **Description** | Wanneer een klant een product wil toevoegen aan de winkelwagen moet het systeem checken of dit product op voorraad is. Als dit niet het geval is moet het systeem een melding geven. Als dit wel zo is moet het systeem de te verwachtte leverdatum tonen. |
| **Actors** | DB, Webshop |
| **Pre-conditions** | Database is beschikbaar; Webshop is online |
| **Flow events** | 1. Klant wil product toevoegen aan winkelwagen 2. Systeem voert database query uit om inzicht in product voorraad te krijgen 3. Als product niet op voorraad is toont systeem melding in de webshop 4. Als product wel op voorraad is toont systeem leverdatum, huidige datum + 3 dagen muz van weekenden. |
| **Exceptions** |  |
| **Postcondities** | Systeem leest succesvol product voorraad uit en voert hier actie op uit |

Figuur 3: Usecase 2

|  |  |
| --- | --- |
| **Usecase Name** | Verwerkt geplaatste order |
| **Unique Usecase ID** | 3 |
| **Description** | Wanneer een klant een bestelling plaatst moet het systeem de bestelgegevens versturen. Een bevestigingsmail naar de klant, facturatie gegevens naar de financiële afdeling, ordergegevens naar het magazijn en werkt actuele voorraad bij in database. |
| **Actors** | PHP mail function, DB, Webshop |
| **Pre-conditions** | Database is beschikbaar; Webshop is online |
| **Flow events** | 1. Klant plaatst bestelling 2. Systeem leest accountgegevens van de huidige klant uit de database uit 3. Systeem ontvangt geplaatste order informatie 4. Systeem stuurt bevestigingsmail naar klant met een overzicht van geplaatste order 5. Systeem stuurt email met overzicht geplaatste order en accountgegevens naar financiële afdeling 6. Systeem stuurt email met overzicht geplaatste order en verzendgegevens naar het magazijn 7. Systeem doet update van het aantal producten in de database die besteld zijn |
| **Exceptions** |  |
| **Postcondities** | Het opvangen en versturen van data als deze wordt aangeleverd; Succesvol mails versturen naar klant, financiële afdeling en het magazijn; Succesvol database bijwerken naar huidige product aantallen |

Figuur 4: Usecase 3

|  |  |
| --- | --- |
| **Usecase Name** | Verstuurd transport bevestiging door |
| **Unique Usecase ID** | 4 |
| **Description** | Wanneer het magazijn een order aanmeld voor transport moet het systeem deze order doorsturen naar de financiële afdeling |
| **Actors** | PHP mail function, DB |
| **Pre-conditions** | Database is beschikbaar |
| **Flow events** | 1. Magazijnmedewerker meldt order aan voor transport 2. Systeem verstuurd ordergegevens door naar financiële afdeling 3. Systeem werkt orderstatus bij naar op transport |
| **Exceptions** |  |
| **Postcondities** | Het opvangen en versturen van data als deze wordt aangeleverd; Succesvol mails versturen naar financiële afdeling; Succesvol orderstatus in database bijwerken |

Figuur 5: Usecase 4

## Usecasediagram

# 

Figuur 6: Usecasediagram