15-5-2018

Leon Frielinck, Rens Koster, Theepa Satheeskumar en Mathijn Smits

15-05-2018

Technisch ontwerp

Een Technisch ontwerp over het Project Oriëntatie.

Inhoudsopgave

[1. Inleiding 2](#_Toc515355379)

[2. Versiebeheer 3](#_Toc515355380)

[3. De applicatie 4](#_Toc515355381)

[4. Technische specificaties 5](#_Toc515355382)

[4.1. Database 5](#_Toc515355383)

[4.2. MCV (Visual studio) 5](#_Toc515355384)

[5. De opbouw van de applicatie 6](#_Toc515355385)

[5.1. Het Inlogscherm: 6](#_Toc515355386)

[5.2. De mutatie: 7](#_Toc515355387)

[5.3. Welkom pagina: 8](#_Toc515355388)

[5.4. Onderhoud pagina: 9](#_Toc515355389)

[5.5. Overzicht pagina: 10](#_Toc515355390)

[6. Relationeel datamodel 11](#_Toc515355391)

[6.1. Nulde normaalvorm (0NV) 11](#_Toc515355392)

[6.2. Eerste normaalvorm (1NV) 12](#_Toc515355393)

[6.3. Tweede normaalvorm (2NV) 12](#_Toc515355394)

[6.4. Derde normaalvorm (3NV) 12](#_Toc515355395)

[7. Stroomdiagram 13](#_Toc515355396)

# Inleiding

In verband met de les project oriëntatie maken wij dit technisch ontwerp voor semester 4. De bedoeling van het project is om te laten zien wat je allemaal kunt en om ervaring op te doen met projectmatig werken door een database-applicatie te bouwen. We hebben een opdracht gekregen van de heer van den Boogaerd van het bedrijf ROCit om een database-applicatie te bouwen voor zijn Wagenpark, zodat het bedrijf de gegevens van de 50 auto’s bij kunt houden.

# Versiebeheer

Hieronder staan de versies beschreven die tijden de technisch ontwerp worden bijgehouden. Er wordt ook doorgevoerde wijzigingen beschreven.

|  |  |  |  |
| --- | --- | --- | --- |
| **Versienr.** | **Datum** | **Beschrijving** | **Wie** |
| 0.1 | 15-05-2018 | Eerst opzet van Technisch ontwerp. Voorblad en inleiding maken. | S. Theepa |
| 0.2 | 16-05-2018 | Hoofdstukken toevoegen. | S. Theepa |
| 0.3 | 17-05-2018 | Aanvullingen gedaan aan de hoofdstukken. | S. Theepa |
| 1.0 | 24-05-2018 | Technisch ontwerp bewerken. | S. Theepa |

# De applicatie

Bij dit project gaan wij een database-applicatie bouwen voor het bedrijf Wagenpark. Door het web moet de onderhoudsbeurt kunnen invoeren en het overzicht kunnen zien van de 50 auto’s, daarvoor moet er een datamodel gemaakt moet worden.

**Wat komt in de Wagenpark applicatie te staan:**

* Een inlog scherm (met sessions) voor de dealer.
* Een inlog scherm voor de applicatiebeheerder die rechten heeft om wijzigingen te doen.
* Een Homepagina
* Een optie voor de dealer via online de onderhoudsbeurt te kunnen invoeren/ registeren.
* Een optie voor de dealer via online een overzicht van een specifieke auto kunnen zien/ bewerken.
* Een mogelijkheid in de homepagina voor out te loggen.

Deze applicatie wordt gemaakt zoals hierboven vermeld. In het web zal er informatie voor zien te krijgen en de gebruiker kan navigeren tussen de verschillende pagina’s/richtingen die hierboven vermeld zijn met behulp van het menu. Uiteindelijk zal deze applicatie gebruikt worden om de gegevens van de Wagenpark te onderhouden.

# Technische specificaties

# Database

**MS SQL Server**

We gaan gebruik maken van de MS SQL Server om een ERD te maken van de genormaliseerde tabellen.

# MCV (Visual studio)

We gaan de applicatie in MVC bouwen.

MVC wordt door programmeurs regelmaat gebruikt bij het programmeren van uiteenlopende computerprogramma's. MVC staat voor Model View Controller.

Volgens MVC zijn softwareapplicaties verdeelt in verschillende modules. Er bestaan drie elementen: een model, een view en een controller.

* **Een model**

Een model geeft aan hoe data voorstelt wordt op database niveau. Welke entiteiten zijn verbonden aan welke databasetabellen.

* **Een view**

Hoe wordt data voorgesteld op het scherm of op de website? Hoe worden de gegevens gewerkt en overzichtelijk weergegeven als output van de controller? Een locale gebruiker wil zijn totale uitgaven in een webshop zien in dollars in plaats van de euro. De view gaat over dit soort veranderen.

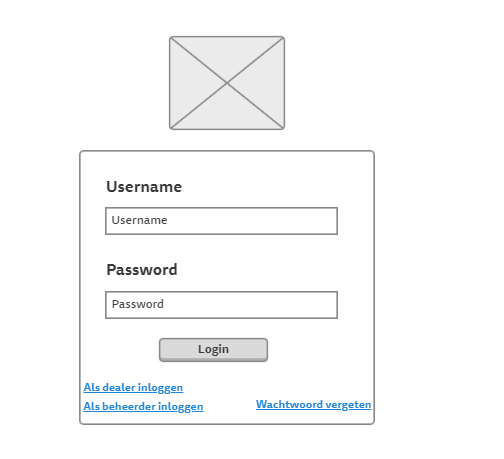
* **Een controller**

De controller is de lijm tussen de model en view en het zorgt ervoor dat MVC samenkomen. De controller bestaat uit zogenaamde actions, die gelden als handelaars (uitvoerders) van de aanvraag van de website bezoeker. Aan de hand van de action worden het juiste model en de juiste view opgespoord en gebruikt.

# De opbouw van de applicatie

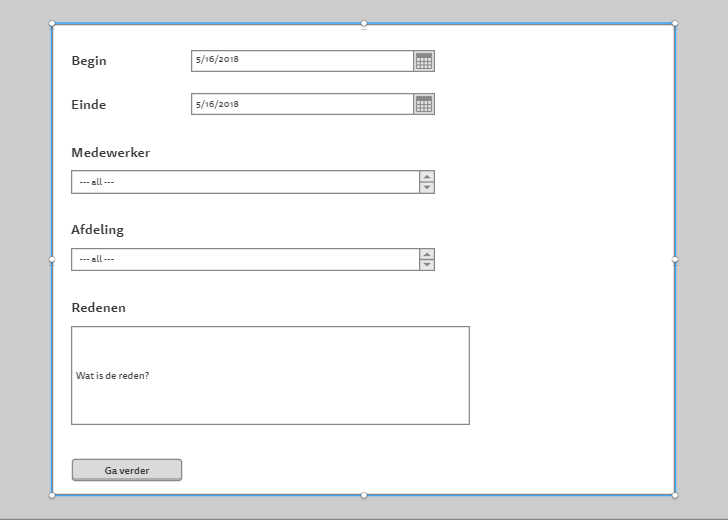
# Het Inlogscherm:

Het inlogscherm is voor de dealer en voor de applicatiebeheerder. De beheerder moeten kunnen inloggen om aanpassingen toe te voegen.



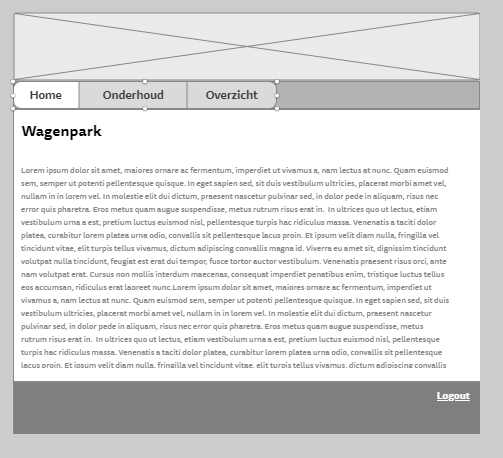
# De mutatie:

De applicatiebeheerder heeft rechten om een mutaties door te voeren.



# Welkom pagina:

De homepagina bevat met informatie over het bedrijf en in de homepagina is het mogelijk om uit te loggen.



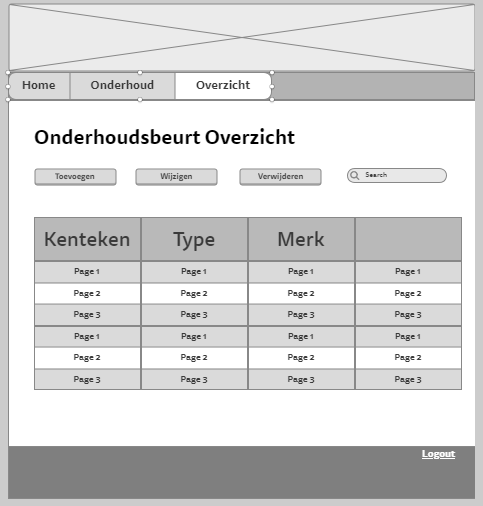
# Onderhoud pagina:

In de onderhoud pagina kunt de dealer onderhoudsbeurt online invoeren/ registeren.



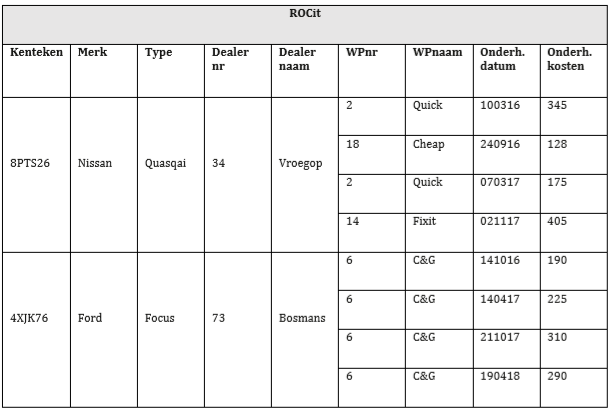
# Overzicht pagina:

In de overzicht pagina kunt de dealer een overzicht inzien en nieuwe onderhoudsbeurt overzicht toevoegen, wijzigen, verwijderen en/of zoeken.



# Relationeel datamodel

Een database is een verzameling gegevens. Zonder structuur is het niet meer dan een opeenhoping van data. Om een database goed te ontwerpen maken we een normalisatieproces.

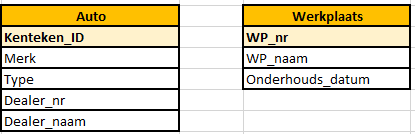


Dit formulier geeft de informatie die we moeten onderzoeken.

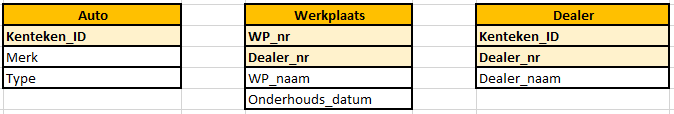
# Nulde normaalvorm (0NV)



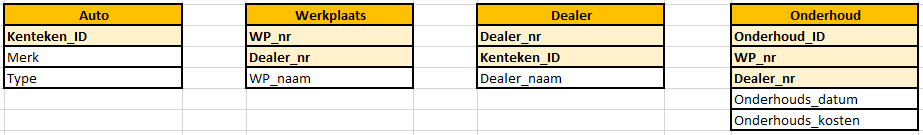
# Eerste normaalvorm (1NV)



# Tweede normaalvorm (2NV)



# Derde normaalvorm (3NV)



# Stroomdiagram

