G

Knaepkens Senne

Schooljaar 2020 - 2021

GIP – ontwerpen van een databank

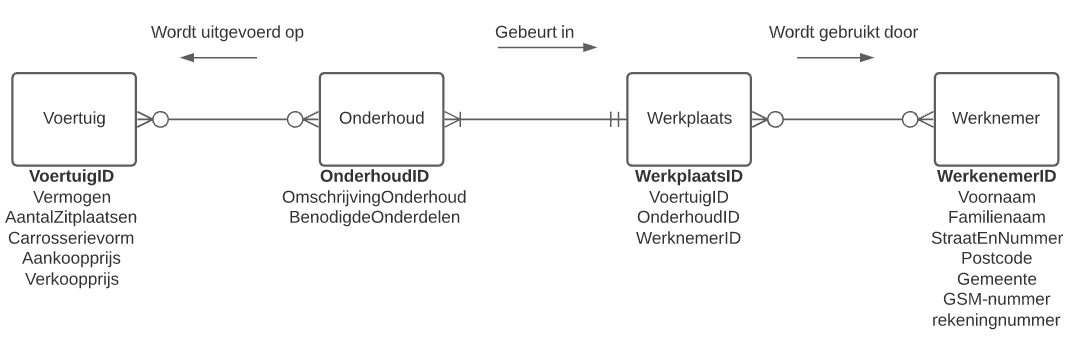
# Analyse informatiebehoefte

Onderneming: garage Peeters te Oostmalle

Bij garage Peeters zullen ze een database nodig hebben om bij te houden welke auto’s ze in stock hebben. Hier horen uiteraard bijbehorende gegevens (zoals aankoopprijs, aantal pk’s, …) ook bij. Ze zullen ook rapporten moeten genereren met hierin bijvoorbeeld het aantal pk en het aantal zitplaatsen in een wagen of filteren op het aantal zitplaatsen enzovoort. In de database zullen ze ook bijhouden wanneer een wagen op onderhoud is geweest of wanneer een wagen is aangekocht of verkocht. Nieuwe verkochte voertuigen moeten niet altijd een onderhoud krijgen. Een ander aspect is de werkplaats hiervan zal opgeslagen worden wat voertuig op welke werkplaats is geweest. Maar bijvoorbeeld ook wat voor onderhoud er gebeurd is en welke werknemer dit uitgevoerd heeft. Van de werknemers bij garage Peeters zullen we een aantal persoonlijke gegevens bijhouden. Ook zullen we hun rekeningnummer bijhouden zodat we hen kunnen betalen.

# Opstellen datamodel

Dit is het ERD-model voor de databank van garage Peeters.



Een voertuig moet geen onderhoud krijgen maar mag er meerdere krijgen.

Een onderhoud wordt uitgevoerd op een of meerdere of op geen voertuig.

Een onderhoud gebeurt in maximaal één werkplaats.

Een werkplaats kan meerdere onderhoudsbeurten doen maar hoeft er geen te doen.

Een werkplaats(en) moet(en) niet gebruikt worden maar kan door meerdere werknemers gebruikt worden.

Een werknemer moet niet maar kan gebruik maken van een of meerdere werkplaats(en).

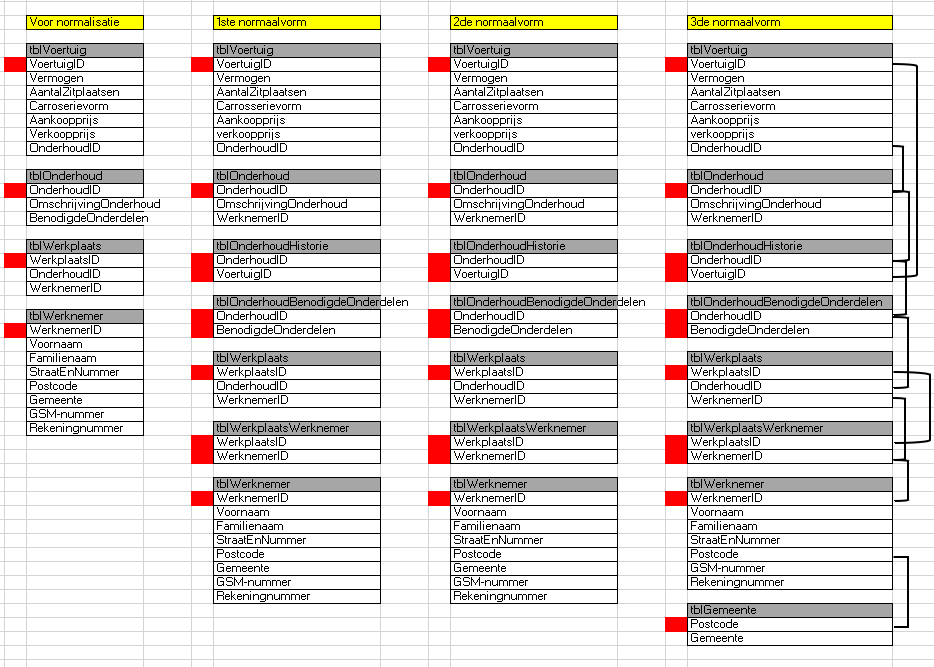
# Normalisatie

In deze stap gaan we het ERD-model omzetten in een genormaliseerde weergave waarvan we later een echte database kunnen maken. Er zijn 3 stappen tijdens het normalisatieproces, deze worden hieronder uitgebeeld.

Om naar de 1ste normaalvorm te gaan moeten we de repeterende groepen apart zetten. Repeterende groepen zijn gegevens die vaker voorkomen in 1 cel. In deze databank is dit tblOnderhoudHistorie, tblOnderhoudBenodigdeOnderdelen en tblWerkplaatsWerknemer.

Om van de 1ste naar 2de normaalvorm te gaan moeten we velden die partieel functioneel afhankelijk zijn apart zetten, in deze tabel is dit al in orde.

De laatste stap is om dit om te zetten tot de 3de normaalvorm. Dit doen we door velden die transitief functioneel afhankelijk zijn van een ander veld apart te zetten. In deze tabel krijgen we zo de tabel tblGemeente omdat we de gemeente weten als we de postcode kennen en omgekeerd.



# Ontwerp van de databank in MySQL Workbench

In deze stap gaan we de genormaliseerde databank verwerken in MySQL Workbench. Dit EER-diagram geeft de database weer.

In MySQL moeten we eerst de tabel voertuig, onderhoud, werkplaats en werknemer zetten. Tussen deze tabellen kunnen we dan de juiste relaties leggen. De tabellen met een veel-op-veel relatie krijgen een tussentabel zoals je hieronder ook kan terugvinden. Dan moeten we de tabel OnderhoudBenodigdeOnderdelen maken en hier ook weer de juiste relatie leggen namelijk een een-op-veel relatie. Als laatste maken we de tabel gemeente nog aan met ook de juiste relatie. In al deze tabellen moeten we uiteraard de juiste velden nog een primaire sleutel geven en de juiste gegevenstypes instellen.

