**Ontwerpdocument Formule 1 webshop**

**Afbeelding met persoon, kleding, person, shirt

Automatisch gegenereerde beschrijving**

**Studentnaam: Matthijs Dolmans**  
**Studentnummer: 487249**  
**Klas: S2-DB02**  
**Vakdocenten: Dennis Cools, Timo Hermans**  
  
**Versie: 1**  
**Datum: 05/09/2022**

# Inleiding

Ik heb een ontwerpdocument maakt voor verschillende redenen een hiervan is zodat je voordat je begint met coderen al een idee hebt wat je gaat coderen. Je kunt namelijk makkelijk zien wat voor classes je van tevoren in je hoofd hat en wat voor attributen/methodes je verzonnen had. Daarnaast is het ook makkelijk om eerst na te denken over wat voor data je wilt opslaan (database ontwerp) zodat je hier rekening mee kan houden. En uiteraard heb ik dit document ook geschreven om **LU3**: Je documenteert **gevalideerde gebruikersspecificaties**voor applicaties en vertaalt deze in**correcte softwareontwerpen aan te tonen.**

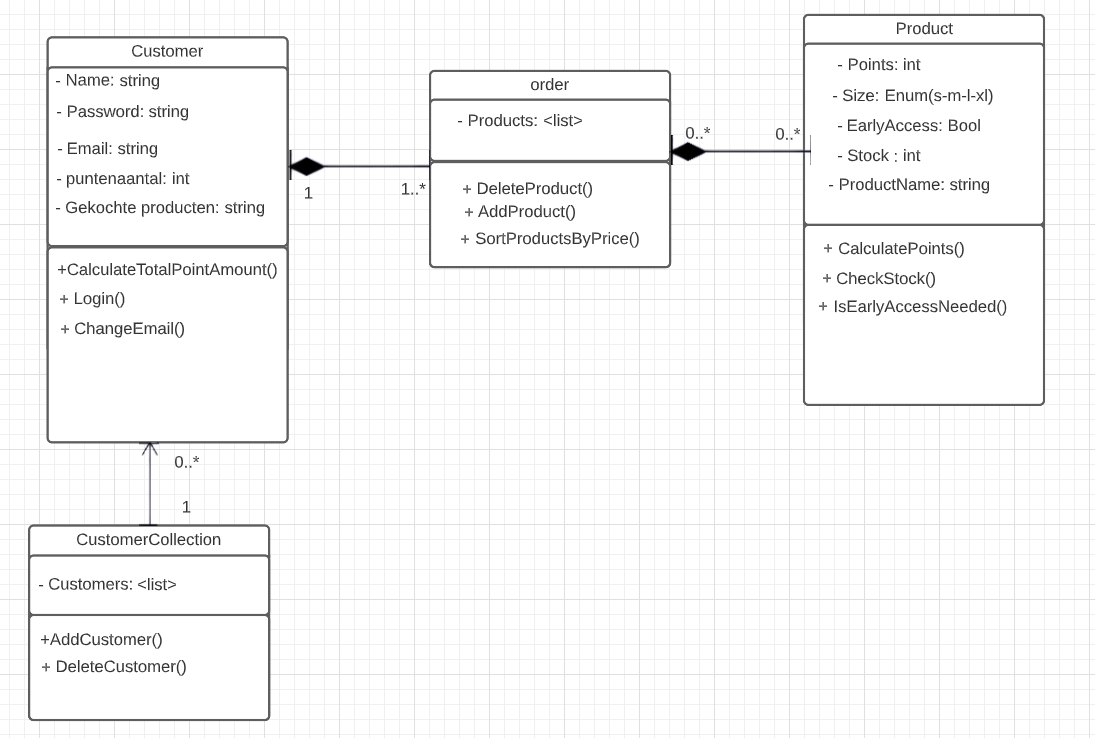
# Inhoudsopgave

[Inleiding 2](#_Toc115866567)

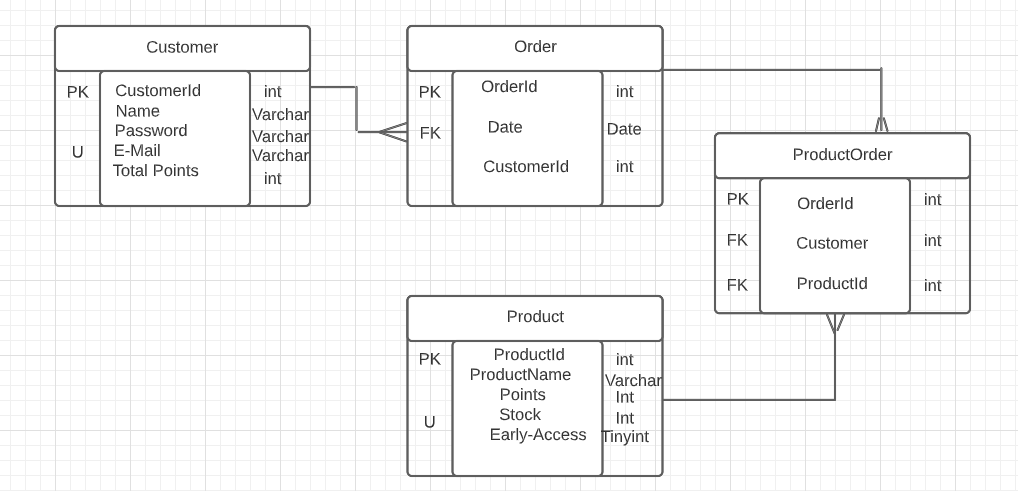
[Inhoudsopgave 2](#_Toc115866568)

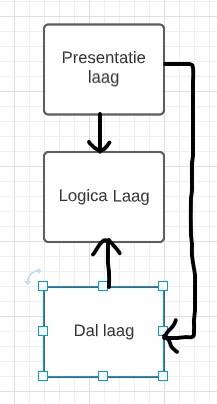
[Klassen diagram 2](#_Toc115866569)

# Klassen diagram

Ik heb een Klassendiagram gemaakt om een idee te krijgen hoe mijn applicatie eruit komt te zien en wat voor classes ik heb. Dit zorgt ervoor dat ik een idee heb hoe en wat in mijn applicatie komt, zodat je niet begint met coderen zonder enig idee hoe het eruit komt te zien. Want dan kom je tijdens het coderen erachter dat je iets mist en dit kan ervoor zorgen dat je veel code anders moet gaan schrijven.

# Database Ontwerp



**Lagenstructuur**

Mijn programma bestaat uit drie lagen, deze lagen zijn de presentatie laag, logica laag en de dal laag. De belangrijkste reden dat ik gekozen

heb voor deze meerdere-lagen structuur is omdat hierdoor mijn programma

duidelijk en overzichtelijk blijft, hierdoor kan ik het programma makkelijk uitbreiden en onderhouden zonder dat ik op 200 verschillende plekken iets moet toevoegen of aanpassen zodra ik een nieuwe feature wil toevoegen. Verder heb ik voor deze structuur gekozen omdat ik hiermee goed kan bepalen welke laag waarvoor verantwoordelijk is, daardoor kan ik ook makkelijk complete lagen vervangen zonder dat ik heel mijn programma hoef aan te passen. Stel ik wil mijn database vervangen met een Oracledatabase, dan hoef ik alleen maar de Dal laag te vervangen en kan de gehele back-end hetzelfde blijven als elke laag afhankelijk is van elke andere laag dan is het heel moeilijk om een laag te vervangen of aan te passen. Dit heb ik gevisualiseerd in dit diagram. De pijlen geven aan welke laag afhankelijk is van een andere laag.

