



# SYSTEM DESIGN

De Druppel

Team De druppel – O-S-OOS3-O-S31

Peter Askey  
Nick Kersten  
Inass Madhoun  
Derek Meuwissen  
Bas van Olphen  
Tygo van Ommen

## Inhoud

Versiebeheer .....	2
Inleiding.....	3
Project omschrijving .....	3
Klassendiagram .....	3
Database Design.....	4
Package diagram .....	4

## Versiebeheer

Versie	Auteur	Datum	Wijzigingen
<b>0.1</b>	Inass El Madhoun	05-09-2020	Eerste opzet document
<b>0.2</b>	Nick HW Kersten	4-11-2020	Toevoegen UML, uitschrijven specificaties, toevoegen klassediagram

## Inleiding

In dit opleverdocument worden diverse diagrammen voor de API getoond. Deze diagrammen geven weer hoe de gerealiseerde API in elkaar zit en hoe deze ingezet kan worden.

Tot slot wordt de kwaliteit van het gerealiseerde product aangetoond door een tabel die de testdekking weergeeft.

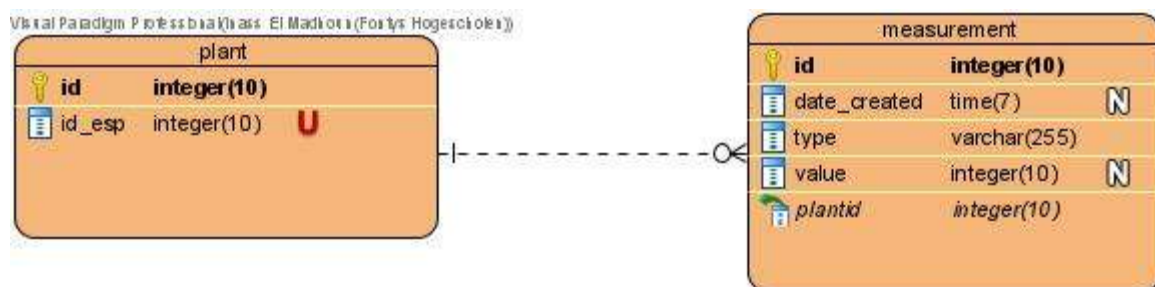
## Project omschrijving

Dit project is voor een druppel-irrigatie systeem. Dit systeem is gericht op klein thuisgebruik voor het gecontroleerd bewateren van planten in bijvoorbeeld een moestuin, bloemenperk of potplanten. Het systeem automatiseert de irrigatie door middel van een microcontroller. Deze meet het vochtgehalte in de grond middels een vochtsensor, en stuurt afhankelijk van minimale en maximale drempelwaardes een klepventiel of pomp aan.

De data over bewateringscycli en vochtgehaltes worden door de microcontroller over het WIFI-netwerk middels een API ge-upload naar een database. Deze data zijn met de API op te vragen.

## Database Design

De Java API maakt gebruik van een Microsoft MySQL Server database. Het data model is hieronder weergegeven.



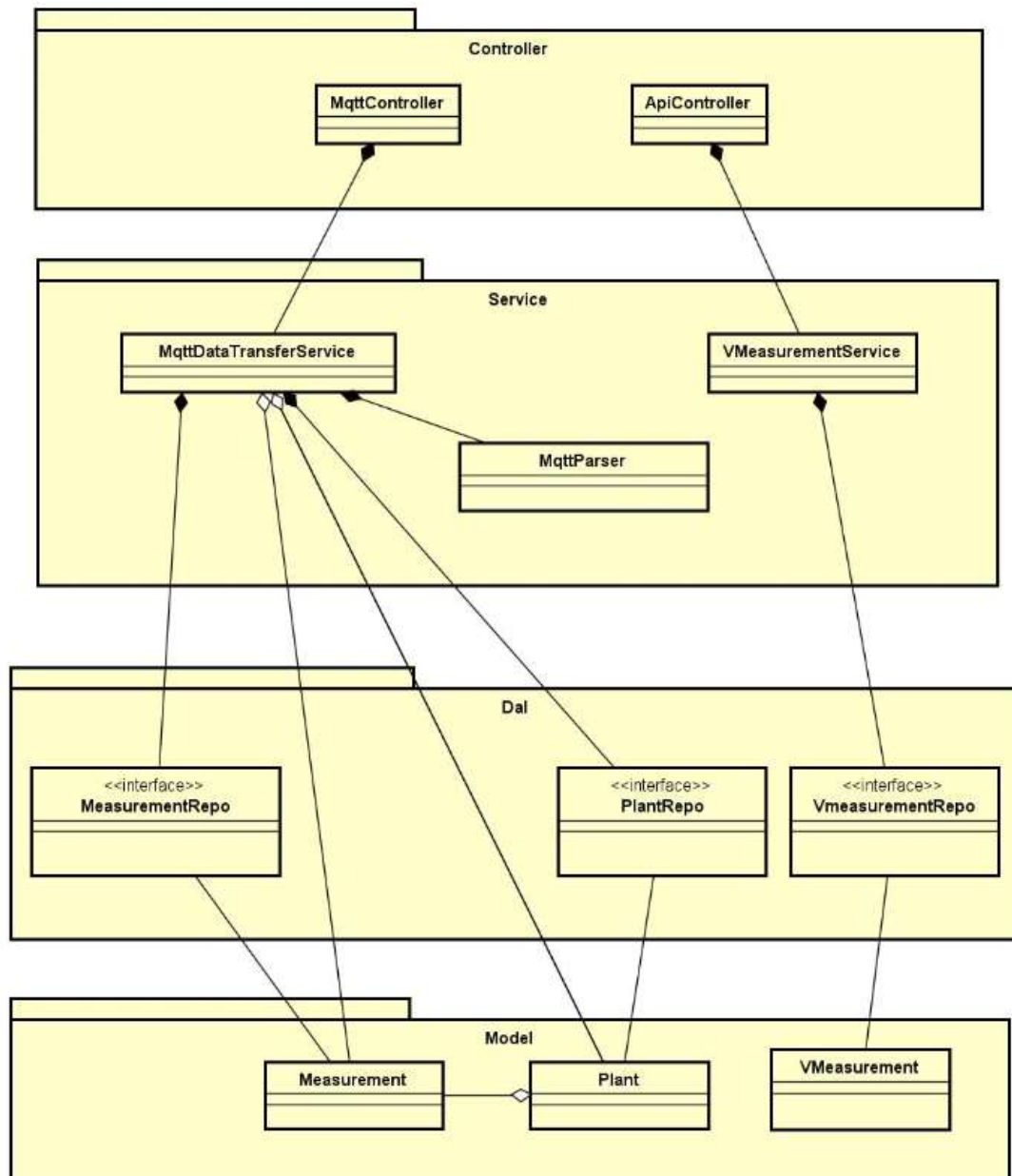
Alle klassen en interfaces in de Druppel API worden weergegeven in het onderstaande klassendiagram



De API bevat vier lagen: Controller, Service, Model, Datasource. Elke laag speelt zijn unieke rol in de API.

De service laag bevat alle business logica die vereist is om aan de requirements te voldoen. In de model laag bevinden zich alle entiteit objecten. Deze objecten worden gebruikt voor het transport van data en bevatten geen gedrag.

De datasource laag Dal zorgt dat er functies beschikbaar zijn die entiteitobjecten op de juiste manier in en uit de MySQL database kunnen halen.



## API Specificaties

- Voor CRUD-operaties wordt de Spring Data JPA gebruikt
- Om de modelobjecten naar de MySQL database te mappen. Wordt ORM Hibernate gebruikt
- De Restcontroller en MQTT-cliënt controller zijn de entry-points van de API. Deze worden middels de Spring Boot IoC container geïnstantieerd.
- Om loose coupling te bewerkstelligen wordt het Spring Dependency Injection framework toegepast.