|  |
| --- |
|  |
| PlantPal |
| Plant Management Systeem |

|  |
| --- |
| Sabawoon Enayat  2022-2023 |

# Het voorwoord

Ik ben Sabawoon Enayat en ik studeer Industriële ICT in Don Bosco Hoboken.

Mijn eindwerk is een plantmanagementsysteem. Ik heb dit onderwerp gekozen, omdat ik hou van planten en bloemen en als een programmeur wil ik altijd zo veel mogelijk automatiseren en op een makkelijke manier alles kunnen beheren.

Ik bedankt Dhr. K. van Reck en Dhr. L. Sienack om me te begeleiden met het eindwerk zowel op technische vlak, maar ook met het inhoud.

Inhoudsopgave

[Het voorwoord 1](#_Toc130210332)

[Inleiding 3](#_Toc130210333)

[Hardware 4](#_Toc130210334)

[Benodigdheden 4](#_Toc130210335)

[Werking 4](#_Toc130210336)

[Database 5](#_Toc130210337)

[PostgreSQL 5](#_Toc130210338)

[ORM 5](#_Toc130210339)

[Client 5](#_Toc130210340)

[WebAPI (Application Programming Interface) 6](#_Toc130210341)

[Python 6](#_Toc130210342)

[FastAPI 6](#_Toc130210343)

[Authenticatie 6](#_Toc130210344)

[Veiligheid 6](#_Toc130210345)

[Voorbeeld 6](#_Toc130210346)

[Web Interface 7](#_Toc130210347)

[TypeScript 7](#_Toc130210348)

[React.js 7](#_Toc130210349)

[Concepten 7](#_Toc130210350)

[Voorbeeld 7](#_Toc130210351)

[Bootstrap 8](#_Toc130210352)

[Voorbeeld 8](#_Toc130210353)

[Data fetching 9](#_Toc130210354)

[React Hook 9](#_Toc130210355)

[Besluit 10](#_Toc130210356)

[Bibliografie 11](#_Toc130210357)

# Inleiding

Mijn eindwerk is een plantmanagementsysteem. Dit wil zeggen dat een gebruiker zich kan registreren op de centrale website. Hij kan planten linken aan zijn profiel. Zo kan hij ook instellen op welke tijd de planten bewaterd worden en of dat het automatisch moet gebeuren. Hij kan in de dashboard ook zien hoe droog dat de grond van de plant is.

Er kan ook een grafiek bekeken worden van hoeveel water een plant heeft gekregen. Zo kunnen ook planten vergeleken worden.

Voor mijn eindwerk maak ik gebruik van heel wat technologieën om alles voor de “end-user”, zo makkelijk en beveiligd mogelijk te maken. Dit gebeurt door de “userinterface” en de rest van de applicatie te scheiden.

Voor de UI (userinterface) heb ik React.js met TypeScript gebruikt. React.js heb ik gebruikt omdat het alles versimpelt voor de ontwikkelaar en omdat er heel veel documentatie errond is. TypeScript omdat het strikt is en omdat het veel problemen elimineert.

Voor de API (Application Programming Interface, wat ik bedoelde met de “rest van de applicatie”), heb ik FastAPI gebruikt. De authenticatie gebeurt door JWT-tokens.

Ik heb PostgreSQL als database gebruikt, omdat ik ermee vertrouwd was en omdat er veel documentatie is.

Voor de ORM (Object Relation Mapping, de middleware tussen de API en de database), heb ik Prisma gebruikt, omdat het alles versimpelt.

Voor de bewatering van de planten, heb ik een ESP32 gebruikt. De ESP32 heeft ook een ingebouwde Wifi module, die in AP (Access Point) mode, STA (Station) mode, of beide modes gecombineerd kan zijn.

# Hardware

## Benodigdheden

* Waterpomp
* Debietmeter
* Vochtigheidsmeter voor grond
* Relais
* LEDs
* Knoppen

## Werking

# Database

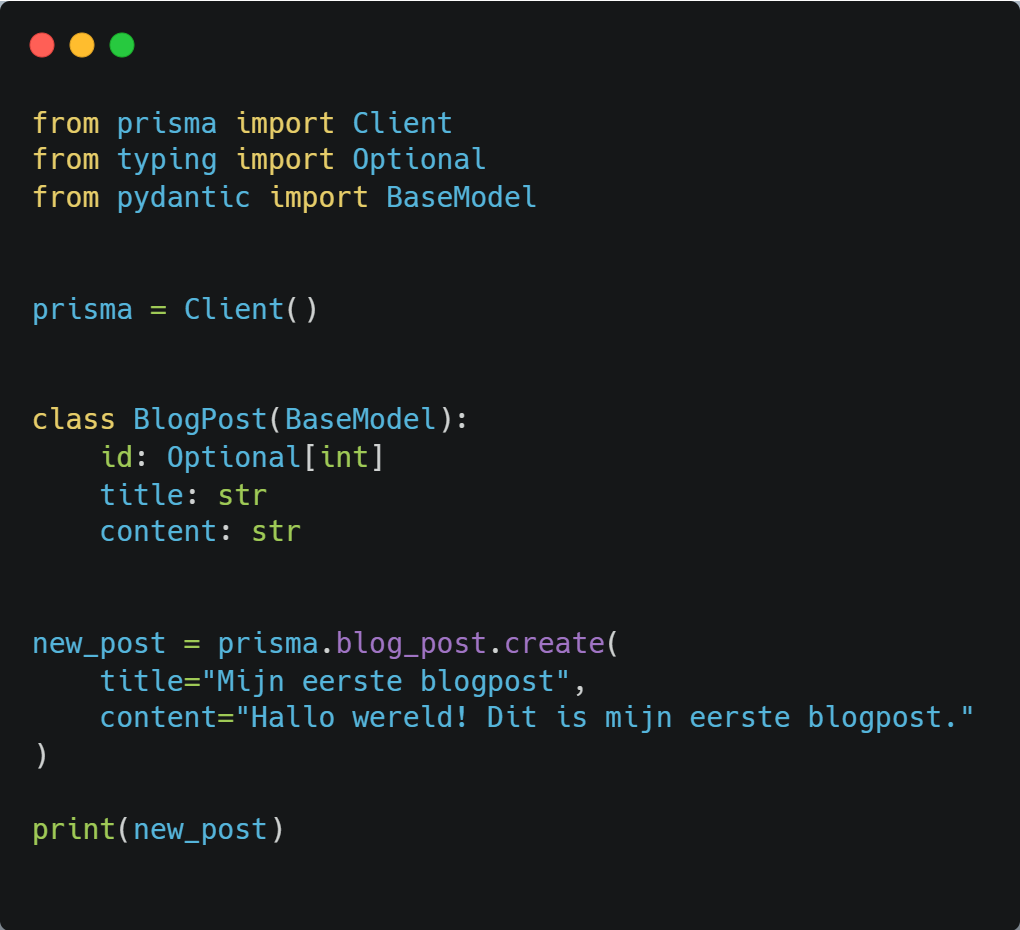
## PostgreSQL

PostgreSQL is een krachtige en veelzijdige database die ontwikkelaars kunnen gebruiken om hun applicaties te ondersteunen. Het is betrouwbaar en heeft veel handige functies, zoals ondersteuning voor complexe zoekopdrachten, ruimtelijke gegevens en tekst doorzoeken. PostgreSQL werkt op verschillende systemen en kan gebruikt worden met verschillende programmeertalen. Bovendien is het gratis en open-source, wat betekent dat iedereen het kan gebruiken en aanpassen. Dit maakt PostgreSQL een aantrekkelijke optie voor ontwikkelaars die op zoek zijn naar een krachtige, betrouwbare en flexibele database-oplossing.

## ORM

Een ORM (Object-Relational Mapping) is een tool die ontwikkelaars kunnen gebruiken om te communiceren met hun database zonder SQL te schrijven. Prisma is een populair ORM dat werkt met Python en verschillende databases, waaronder PostgreSQL en MySQL. Prisma zorgt ervoor dat ontwikkelaars gemakkelijk objecten kunnen maken en ophalen vanuit hun database.

### Voorbeeld



## Client

# WebAPI (Application Programming Interface)

Dit is de gedeelte van de applicatie die voor de authenticatie en beheer van data zorgt. Het is in essentie een interface tussen de database en de toepassing. De programmeertaal die ik gebruikt heb voor de WebAPI, is Python. De web framework die ik gebruikt heb is FastAPI.

## Python

Python heb ik voor de volgende redenen gebruikt:

* Eenvoudig gebruik
* Veel documentatie

## FastAPI

### Authenticatie

### Veiligheid

### Voorbeeld

# Web Interface

Voor de web-interface heb ik, TypeScript, React.js en bootstrap gebruikt. HET platform waarop dit gemaakt wordt is Node.js.

## TypeScript

Ik heb TypeScript gebruikt, omdat het strikter is dan JavaScript, de taal van de webbrowsers. De striktheid zorgt ervoor dat er minder bugs worden gemaakt en dus minder frustratie voor de ontwikkelaar. Het is een superset van en transpileert (vertaalt) naar JavaScript. Het heeft bepaalde programmeerfuncties (features) die JavaScript niet heeft, zoals types, interfaces (structs in C), generics.

## File:React-icon.svg - Wikimedia CommonsReact.js

React.js is een bibliotheek die veelgebruikt is en heel ontwikkelaar-vriendelijk is. Het maakt gebruikt van componenten, die oftewel “classes” oftewel “functions” kunnen zijn. Ik heb “functions” gebruikt, omdat het makkelijker is om mee te werken en omdat mijn project minder complex is. De syntax die wordt gebruikt noemt “JSX” en het is een variant van JavaScript. Dit is om bijvoorbeeld componenten makkelijker te kunnen representeren, maar deze worden ook naar “gewone” JavaScript gecompileerd.

### Concepten

### Voorbeeld



## File:Bootstrap logo.svg - Wikimedia CommonsBootstrap

Bootstrap is een styling bibliotheek, die vaak gebruikt wordt. Het zorgt voor de basis CSS classes, zodat we de styling niet moeten schrijven, maar gewoon gebruiken. Er bestaan meer geavanceerde styling bibliotheken, zoals, mui, maar ik wou de styling simpel houden.

### Voorbeeld

## Data fetching

De website krijgt (en stuurt) de data via de ingebouwde “fetch” functie in JavaScript. De authenticatie token wordt via de request headers gestuurd.

### React Hook

Dit is ook gebruikt als een react hook. De hook is, “useApi”, genoemd. En ziet er als volgt uit:

# Besluit

# Bibliografie