

The image features decorative curved lines in the top-left and bottom-right corners. These lines are composed of multiple concentric arcs in shades of light green and teal, creating a sense of movement and depth.

# strobba

## **Project Plan**

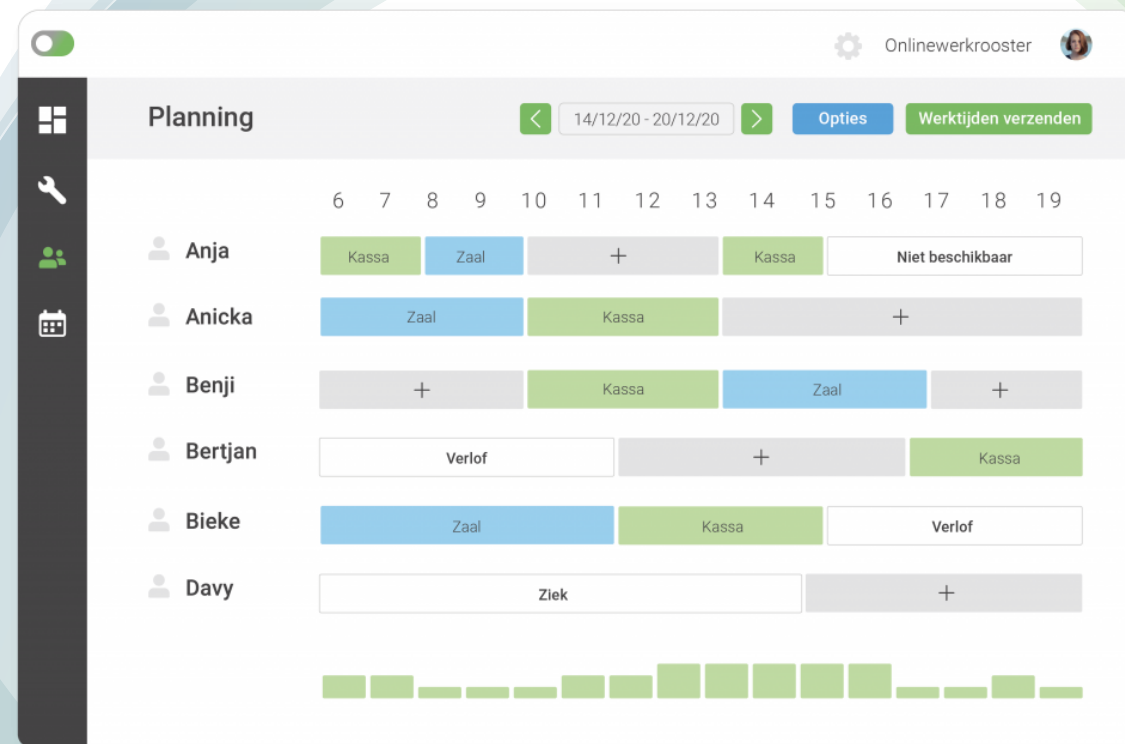


# Inhoud

1. Over Strobbo
2. Stageopdracht
3. Planning
4. Onderzoek

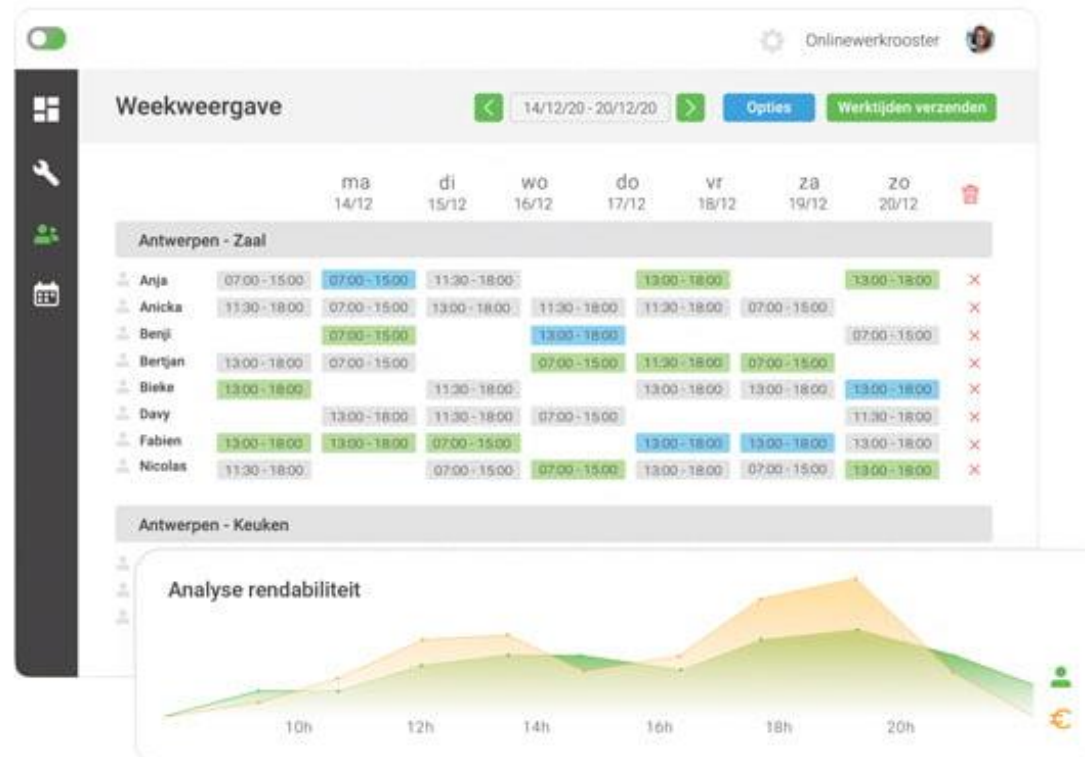
# Over Strobbo

- **Personeelsplanning:** Automatiseert werkroosters.
- **Tijdregistratie:** Nauwkeurige tijdregistratie.
- **Loonverwerking:** Vereenvoudigt de loonadministratie.
- **Dimona-aangiften:** Automatiseert verplichte meldingen.



# Stageopdracht

## Internal tooling





# Probleemstelling

- Logs van consultants zijn niet zichtbaar in de interne tooling.
- Handmatig zoeken in de database bij fouten.
- Consultants kunnen niet beheerd worden
- Er kan niet gezien worden welke consultant aanpassingen heeft gedaan
- Er moet worden uitgelogd als je van klant wilt wisselen

# Oplossing

- Logs zichtbaar maken in de interne tooling.
- Zoekfilters implementeren (klant, consultant, datum).
- Consultants beheren (toevoegen, toegang geven en intrekken).
- Een aanpassing dat gedaan wordt linken aan een consultant
- Functionaliteit voorzien om te wisselen van klant zonder uit te loggen



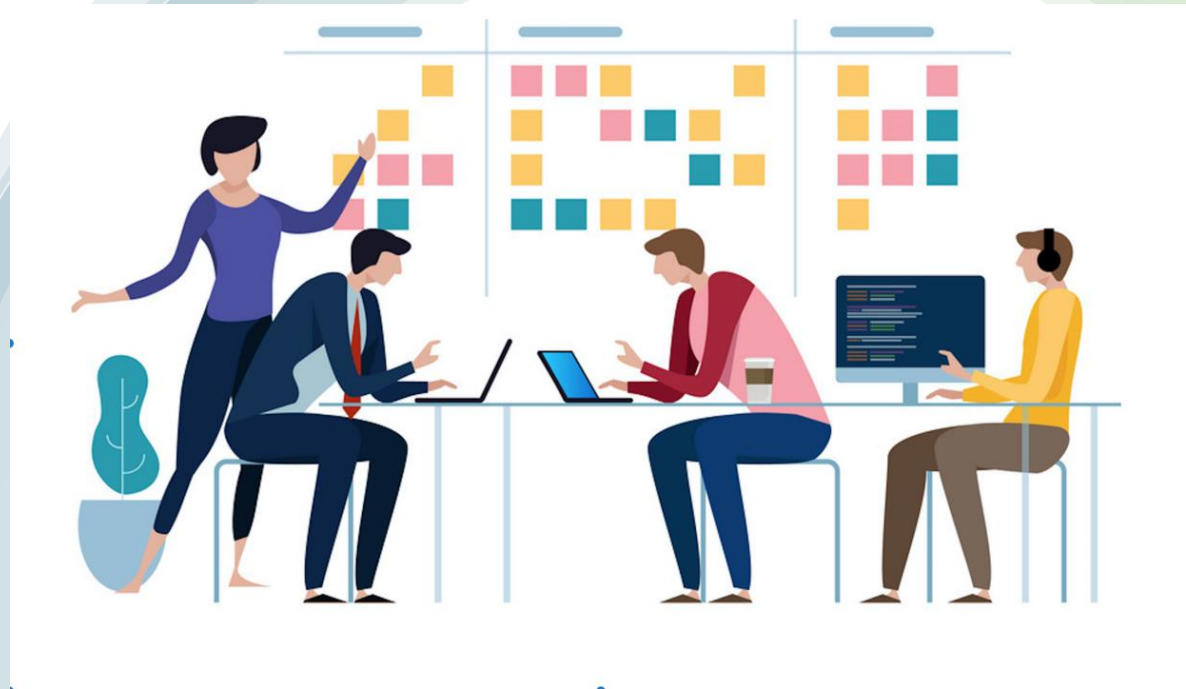
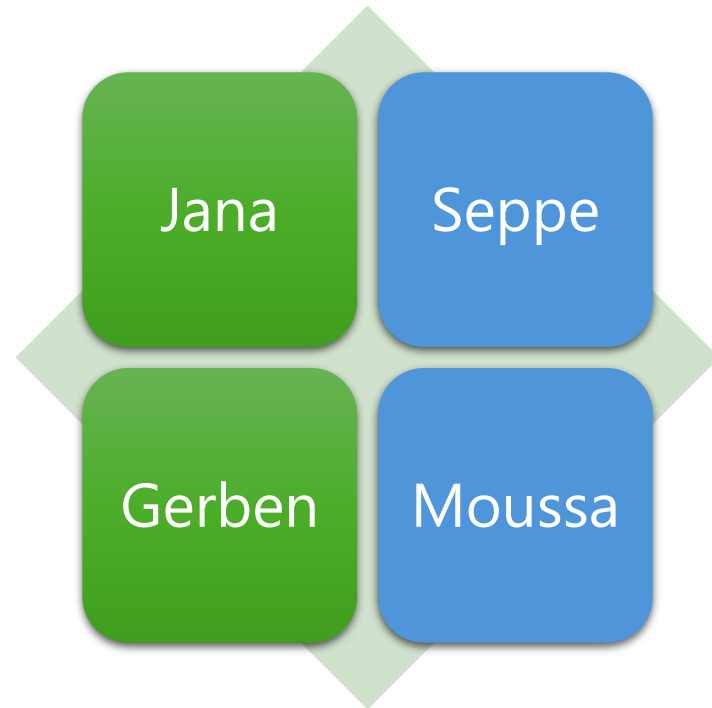
# Technologieën

- Node.js (back-end)
- Vue (front-end)
- Miro (story map)
- Github (version control)
- Discord (communicatie)



# Ons Team

Byte-Me



- Interne Tooling
- Mobile Applicatie





# Planning



# Werkmethode

- **Agile:** nieuwe planning en nieuw onderzoek elke sprint
- **Sprints:** 2 weken.
- **Sprint review:** Resultaten presenteren voor alle developers
- **Retrospective:** Evaluatie aan het einde van elke sprint.
- **Daily Standup:** Dagelijkse meetings met ons team.

# Ons plan

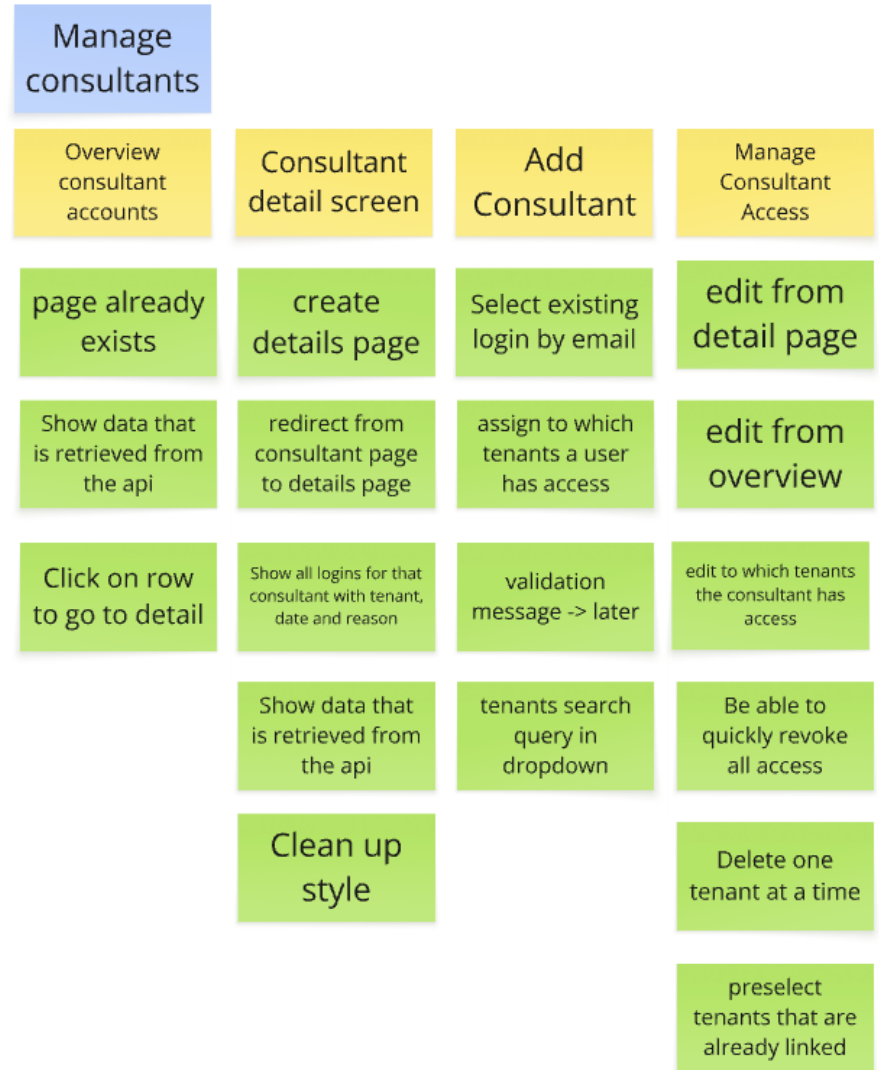
5 delen:

- Consultants beheren
- Login logs
- Klanten informatie
- Audit logging
- Wisselen van klant



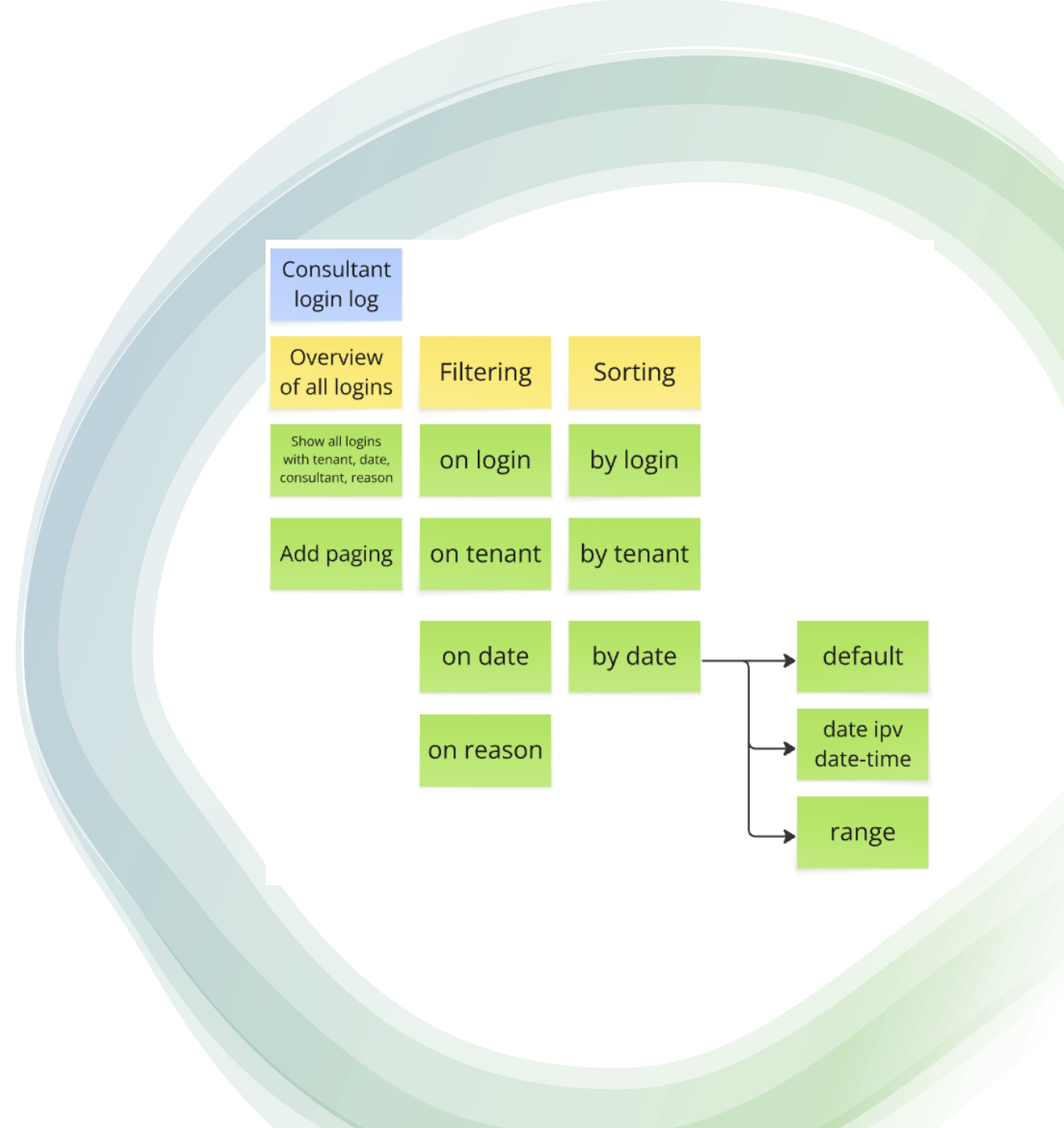
# Consultants beheren

- Overzicht van consultants:
  - Consultant toevoegen
  - Toegang kunnen geven tot klanten
  - Filter consultants
- Detail pagina:
  - Toegang tot klanten beheren
  - Login logs weergeven
  - Filters klanten en login logs



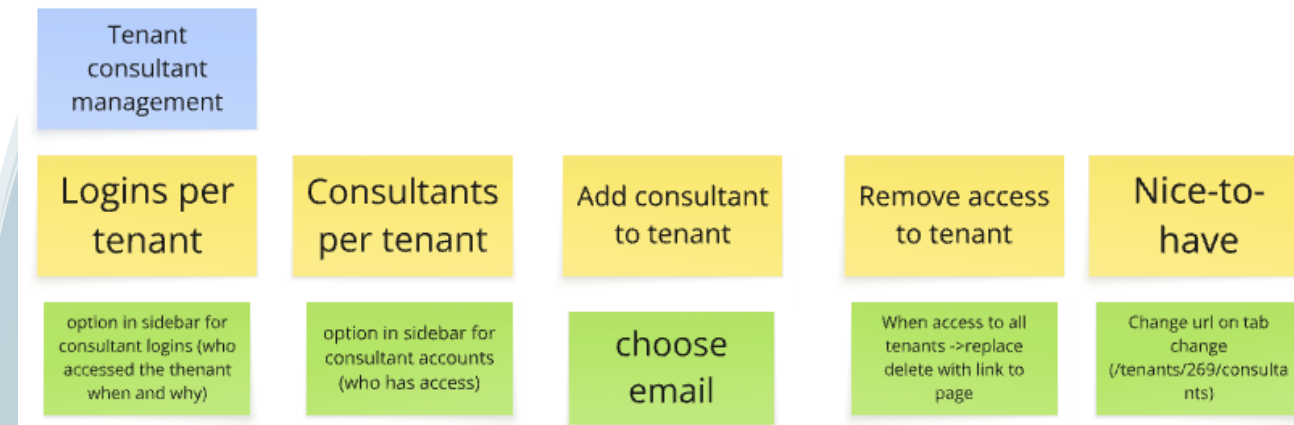
# Login logs

- Overzicht van login logs:
  - Welke consultant heeft ingelogd bij welke klant
  - Datum en tijd van de login
  - Ingegeven reden van de login
- Filters:
  - Op klant
  - Op consultant
  - Datum range



# Klant informatie beheren

- Overzicht van klanten (bestaat al)
- Consultants tab
  - Alle consultants die toegang hebben tot de klant weergeven
  - Consultants toegang kunnen geven en wegnemen
  - Filters
- Logs tab
  - Overzicht van logins van consultants bij die klant
  - Filters



# Audit logging

- Bijhouden welke consultant aanpassingen heeft gedaan
- Als een consultant een klant impersonate en aanpassingen doet dat dit correct wordt opgeslagen

Audit logging in  
Strobbo for  
consultant accounts

Mark  
consultant login

Add consultant  
email in token  
(auth api)

Add email in  
token when  
impersonating

brand audit logs  
with the email

save consultant  
email in db

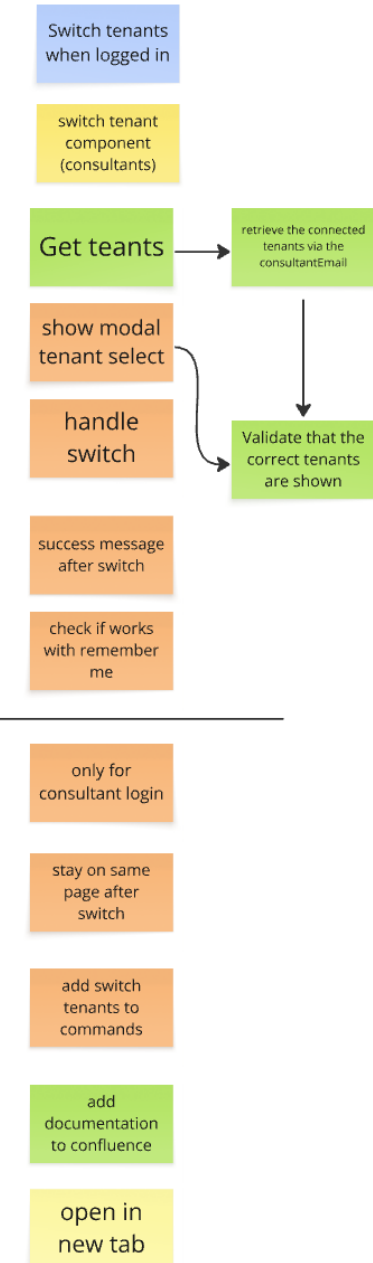
show  
consultant login

root admin ->  
name + email  
(purple)

admin ->  
name

# Wisselen van klant

- Wisselen van klant zonder uit te loggen
- Een reden in kunnen geven voor in te loggen
- Er voor zorgen dat je op dezelfde pagina blijft na de switch



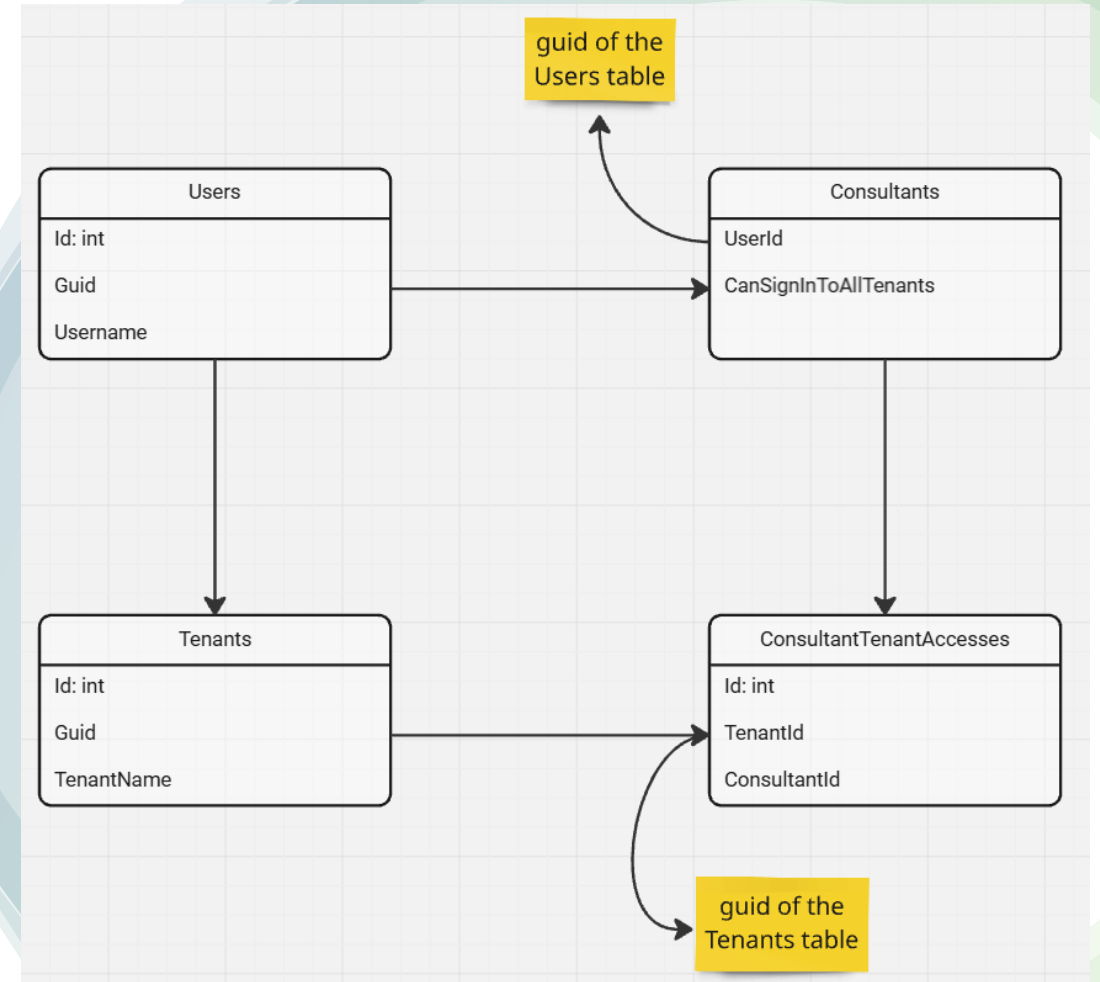


# Onderzoek



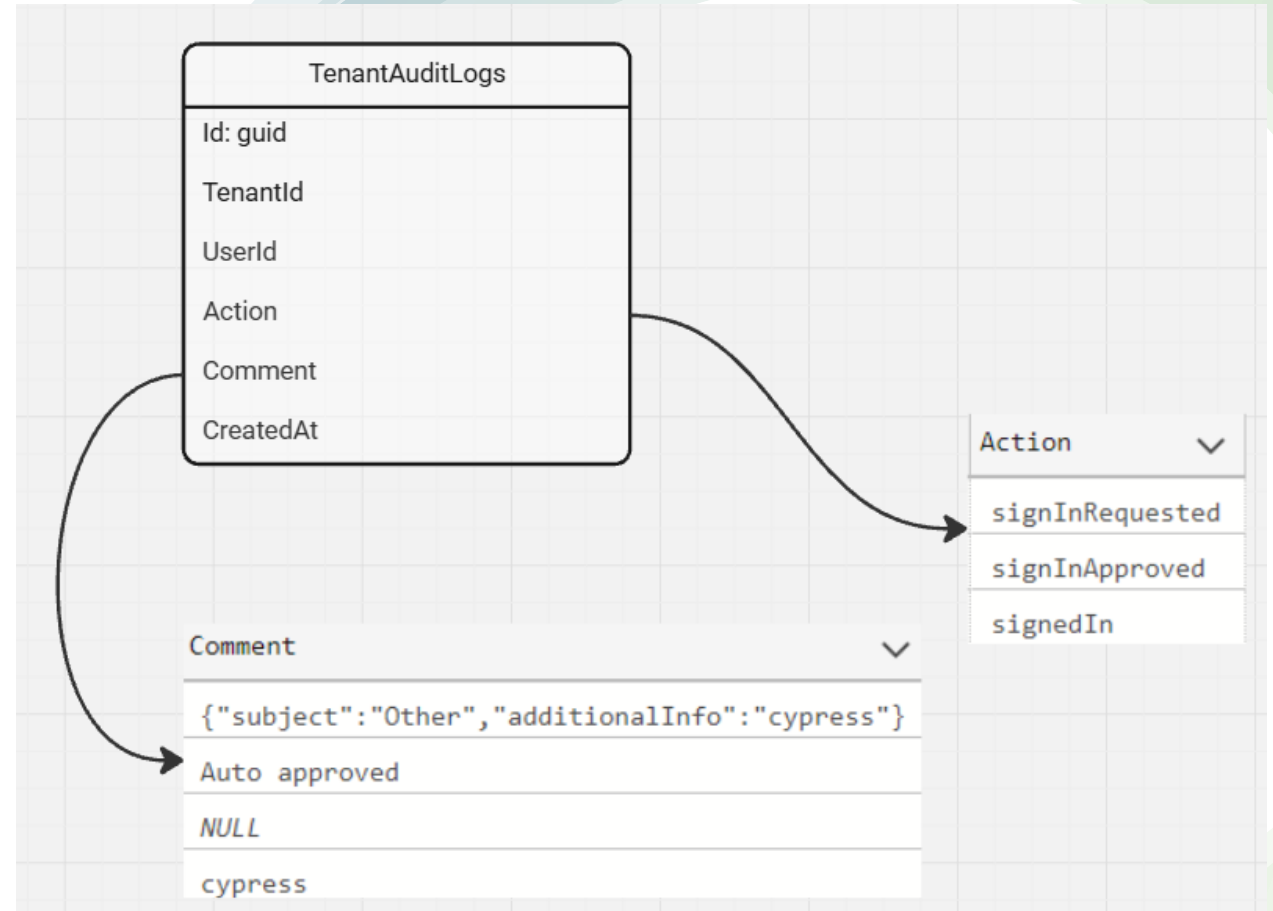
# Database

- Informatie over:
  - klanten
  - Consultants
  - Consultant toegang



# Database

- Informatie over:
  - Logins van consultants
- 3 acties
- Inconsistente comment data

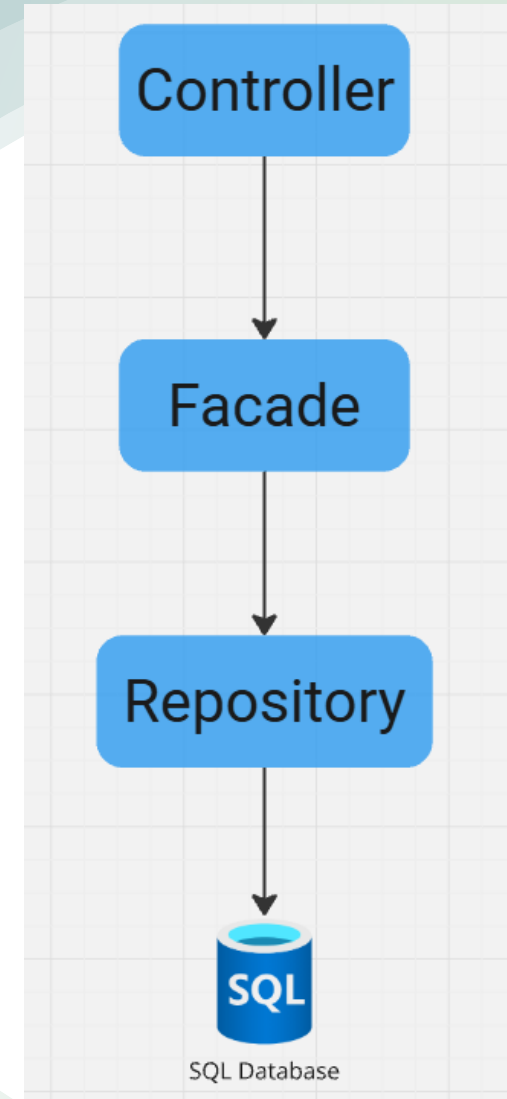


# Backend



# Werking

- **Controller:**
  - Verwerkt gebruikersinvoer en werkt de weergave bij.
  - Roept de facade aan voor bedrijfsoperaties.
- **Facade:**
  - Biedt een vereenvoudigde interface aan de controller.
  - Delegeert taken aan de repository.
- **Repository:**
  - Beheert gegevensopslag en -toegang.
  - Voert CRUD-operaties uit met de database.
- **Database:**
  - Slaat de daadwerkelijke gegevens op en beheert deze.
  - Zorgt voor gegevensintegriteit en consistentie.



# Repositories

- Opgesplitst in 2:
  - Domain/command
  - Query
- Domain/Command repository:
  - commando's die de toestand van de applicatie wijzigen
- Query repository:
  - Behandelt queries die gegevens ophalen zonder de toestand te wijzigen

Domain/command:

POST

PUT

DELETE

Query:

GET

# Knex js

- Query Builder voor SQL Databases:
  - Ondersteunt meerdere SQL-databases.
  - Intuïtieve JavaScript API voor complexe query's.
- Betere Code:
  - Verbeterd leesbaarheid en onderhoudbaarheid.
  - Vermijdt direct werken met ruwe SQL.
- Geen Ruwe SQL Nodig:
  - Vermindert risico op SQL-injecties.
  - Bespaart tijd en moeite.



KNEX.JS

# Unit test

- In-Memory Unit Tests:
  - Simulatie: Bootst een database na zonder een echte database te gebruiken.
  - Tijdelijke Data: Data bestaat alleen tijdens de test en wordt daarna verwijderd.
  - Efficiëntie: Snellere tests omdat er geen echte databaseverbinding nodig is.
  - Isolatie: Tests zijn geïsoleerd van externe factoren, wat zorgt voor consistente resultaten.





# In-Memory Repositories

- State als Mimic van Database Tabel:
  - Een state bootst de structuur van een database tabel na.
  - Data wordt in het geheugen opgeslagen voor testdoeleinden.
- In-Memory Repositories:
  - Query Repository: Bevat functies om gegevens op te halen.
  - Command Repository: Bevat functies om gegevens te wijzigen.

Op de volgende dia een diagram ter illustratie van de werking.



## ConsultantState

```
1 export type ConsultantState {
2   consultantId: string;
3   canSignInToAllTenants: boolean;
4 }
```

## ConsultantTenantAssignmentState

```
1 export type ConsultantTenantAssignmentState {
2   consultantId: string;
3   tenantId: string;
4 }
```

db.consultants

db.consultantTenantAssignments

InMemoryConsultantQuery

InMemoryConsultantCommand

this.consultants: ConsultantState[]

this.consultantTenantAssignments:  
ConsultantTenantAssignmentState[]

```
getAssignedTenantIds:
> this.consultantTenantAssignments.filter(...)
```

```
findOneById
> this.consultants.find()
```

```
find
> this.consultants.slice()
```

```
getTotalCount
> this.consultants.length
```

this.consultants: ConsultantState[]

this.consultantTenantAssignments:  
ConsultantTenantAssignmentState[]

```
assignTenants:
> update this.consultantTenantAssignments
```

```
createConsultant:
> this.consultants.push()
```



# strobba

## **Project Plan**