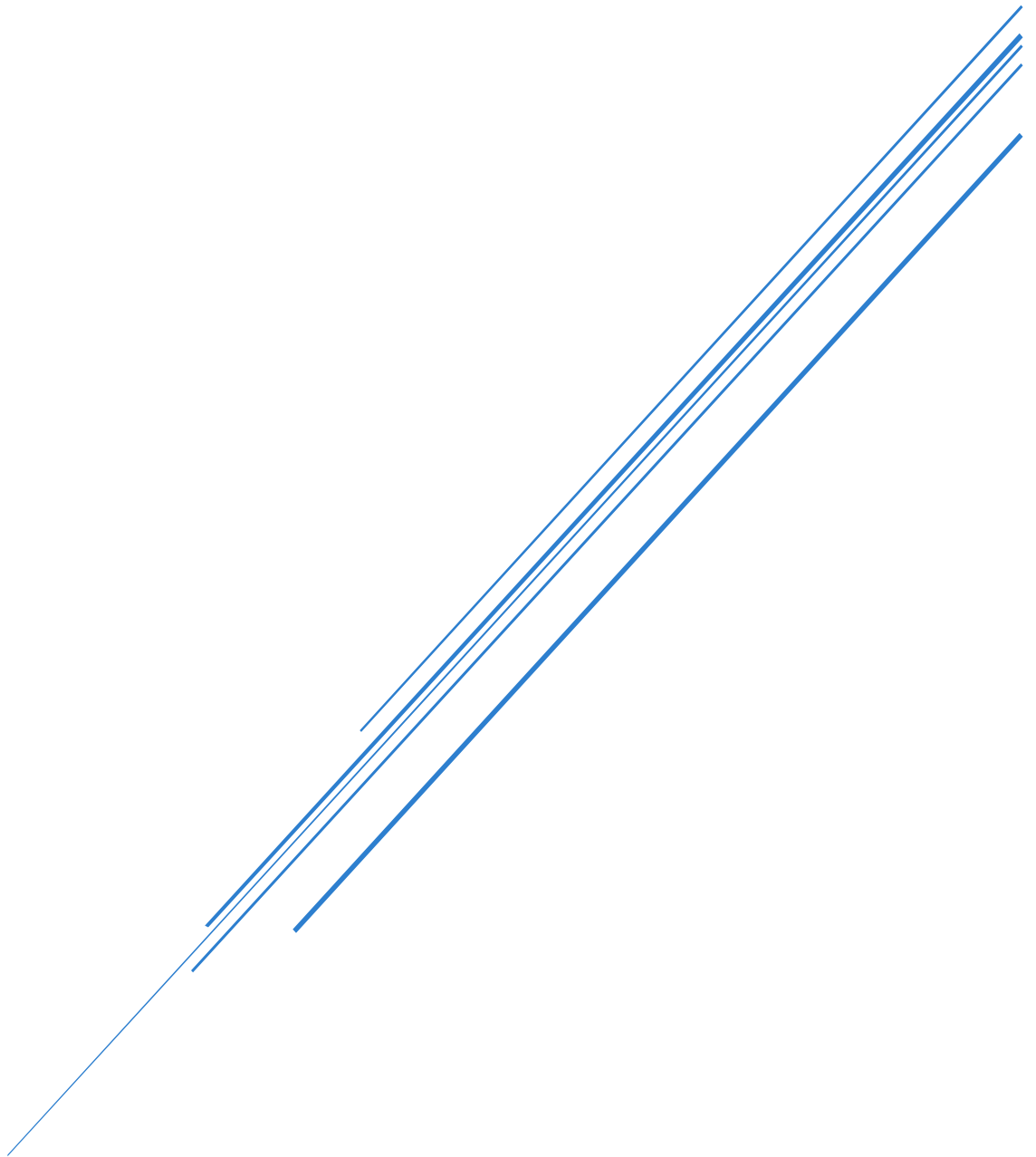


# HANDLEIDING ZAALINDELER EN BESTELWEBSITE

PWS door Jonathan Pierik en Seth Moerkerken



GSR

## Inhoudsopgave

Zaalindeler .....	3
Installatiedocument .....	3
Gebruiker .....	4
Importeren gasten .....	4
Informatie over de gasten en hun indeling .....	4
Instellingen .....	5
Zoeken gasten .....	5
Beheer .....	6
Opschonen database .....	6
Het tandwiel .....	6
Toevoegen gasten .....	8
BELANGRIJKE INFORMATIE .....	8
Developer .....	9
Algemene structuur .....	9
Main-gedeelte .....	9
Renderer-gedeelte .....	13
Redis database .....	14
Developer modus uitvoeren .....	18
Het programma inpakken .....	19
Mogelijke problemen bij het opstarten van Zaalindeler .....	22
Oneindig laadscherm .....	22
Andere problemen .....	23
Bestelwebsite en Supabase database .....	24
Informatie .....	24
Toegang tot, en updaten van het JavaScript-bestand, PHP-bestand en CSS-bestand (website bestanden) .....	24
Toegang tot, en updaten van de Supabase Edge Function .....	26
Wat te doen voor de kaartverkoop van een Grote Avond .....	28
Wat te doen tijdens de kaartverkoop van een Grote Avond .....	28
Wat te doen na de kaartverkoop van een Grote Avond .....	28
Supabase tabel legen .....	29
Supabase tabel exporteren .....	31
Optie één: De gehele tabel met alle gegevens erin exporteren .....	32
Optie twee: Zelf een aantal bestellingen selecteren .....	32
Dagen op de website aanpassen .....	34

Leden lo Vivat aanpassen .....	38
Meespelende leerlingen aanpassen.....	43
Bevestigingsmail aanpassen .....	46
Ereleden aanpassen .....	48
Tijdsduur meldingen aanpassen .....	49
Succesmelding website aanpassen .....	50
Kleuren meldingen website aanpassen .....	51
Accounts en belangrijke gegevens.....	52

# Zaalindeler

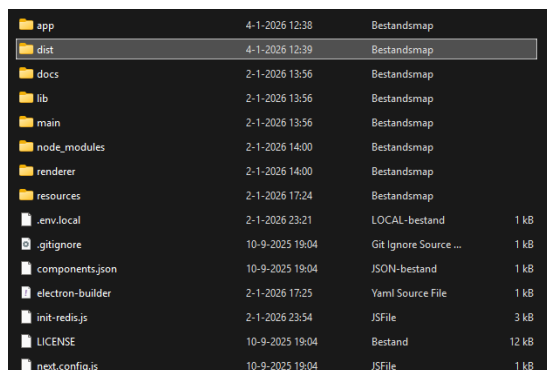
## Installatiedocument

### Stap 1

Na het downloaden van het Zaalindeler ZIP bestand zal het eerst uitgepakt moeten worden.

### Stap 2

Na het uitpakken, open de Zaalindeler folder. Zoek in deze folder voor de folder “dist” en open deze



app	4-1-2026 12:38	Bestandsmap	
dist	4-1-2026 12:39	Bestandsmap	
docs	2-1-2026 13:56	Bestandsmap	
lib	2-1-2026 13:56	Bestandsmap	
main	2-1-2026 13:56	Bestandsmap	
node_modules	2-1-2026 14:00	Bestandsmap	
renderer	2-1-2026 14:00	Bestandsmap	
resources	2-1-2026 17:24	Bestandsmap	
.env.local	2-1-2026 23:21	LOCAL-bestand	1 kB
.gitignore	10-9-2025 19:04	Git Ignore Source ...	1 kB
components.json	10-9-2025 19:04	JSON-bestand	1 kB
electron-builder	2-1-2026 17:25	Yaml Source File	1 kB
init-redis.js	2-1-2026 23:54	JSFile	3 kB
LICENSE	10-9-2025 19:04	Bestand	12 kB
next.config.js	10-9-2025 19:04	JSFile	1 kB

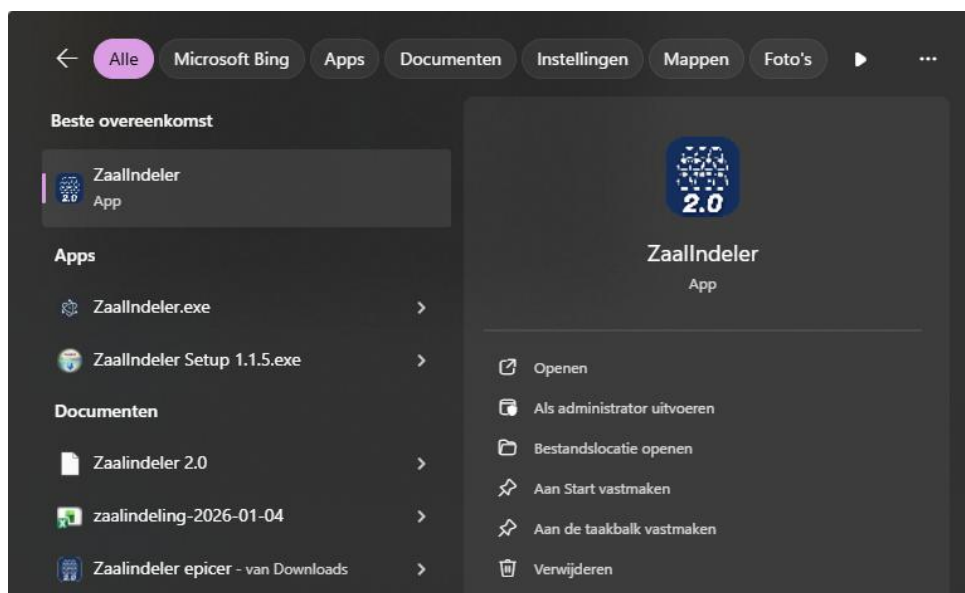
Figuur 1: Map dist

### Stap 3

Zoek in de dist folder voor het bestand “Zaalindeler setup 1.1.5” en voer deze uit. Het programma zal zichzelf gaan installeren, wacht totdat de installatie is afgerond.

### Stap 4

Als de installatie is voltooid is zaalindeler te vinden door te zoeken in de taakbalk



Figuur 2: Zaalindeler 2.0

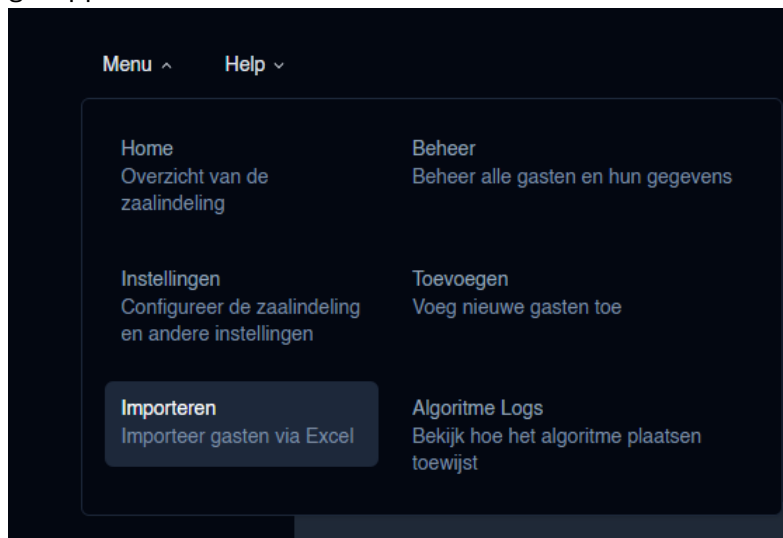
De installatie is nu voltooid.

## Gebruiker

Zaalindeler is een programma dat geschreven is om voor de grote avond van de GSR de grote zaal van het Zuidplein theater in te delen. Dit programma is bedoeld voor het bestuur van lo-Vivat, hierom is het sterk aangeraden om het niet zomaar rond te delen

## Importeren gasten

Om de zaal in te kunnen delen moeten er gasten geïmporteerd worden. Om te importeren is er een .csv spreadsheet nodig. Dit spreadsheet kan worden verkregen via de database die gekoppeld is aan de lo-vivat bestelwebsite.



Figuur 3: Knop importeren

Vervolgens kunt u de gasten importeren door naar de importeerpagina te gaan. Na een tijdje laden zal de zaalindeling gereed zijn!

## Informatie over de gasten en hun indeling

In het programma zijn er drie zaalindelingen over drie dagen te verdelen elke dag kan gevuld worden met 618 gasten, wat betekent dat er in totaal 1848 gasten ingedeeld kunnen worden.

Zaalindeler beschikt over vier soorten gasten, gesorteerd op rang: Gast (grijs), Docent (blauw), Speler (oranje), en Erelid (groen). De gast op de stoel bepaald waar in de zaal de stoel terecht komt. Hoe hoger de rang, hoe beter de plek.

Wanneer er een cursor over de bezette stoelen gaat is het mogelijk om informatie over de gasten te krijgen. De naam waarop het kaartje gereserveerd was, het email-adres, het stoel- en rijnummer, en of de persoon lid is van lo-vivat of niet.

## Instellingen

De instellingen bieden meerdere mogelijkheden om Zaalindeler aan te passen naar wens.

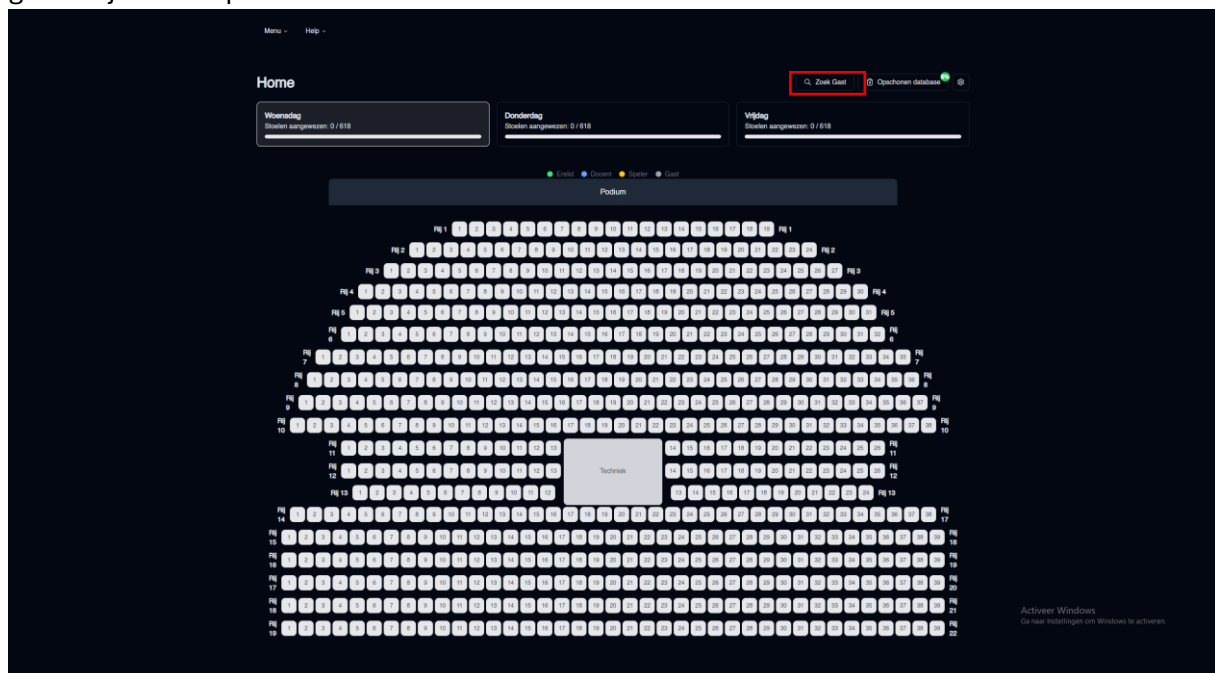


Figuur 4: Knop instellingen

Er is een mogelijkheid om niet beschikbare stoelen te blokkeren. Het algoritme zal geen gast indelen op deze stoelen. Ook kan je verschillende voorkeuren van het algoritme aanpassen in dit menu

## Zoeken gasten

Zaalindeler biedt ook een zoekfunctie. Hiermee kan er gezocht worden naar gasten in de gastenlijst door op hun naam te zoeken.



Figuur 5: Knop gasten zoeken

## Beheer

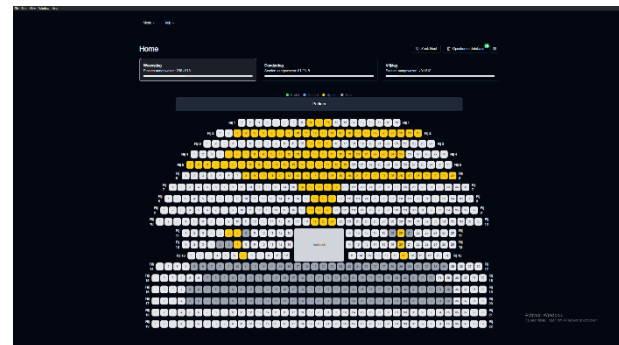
Het beheer toont de gasten die beschikbaar zijn in Zaalindeler. Het is ook mogelijk om meer info over de gasten te zien op deze pagina.



Figuur 6: Knop beheer

## Opschonen database

De database raakt door gebruik langzaam vol met onnuttige data. Hierom is er een “opschonen database” knop met daarbij een volheids-percentage indicator. Zorg ervoor dat dit percentage in de gaten gehouden wordt, en leeg de database wanneer deze te hoog wordt. De huidige zaalindeling wordt hierdoor wel afgebroken, daarnaast kan het algoritme even van slag zijn. Om dit zo snel mogelijk op te lossen. Klik op de “Herlaad Zitplaatsen” knop onder het tandwiel.



Figuur 7: Percentage opschonen database

## Het tandwiel

### Thema aanpassen

Zaalindeler beschikt over twee thema's, een donker- en een licht thema. Door op de “Thema Aanpassen”-knop te drukken wordt er gewisseld tussen de twee.



Figuur 8: Knop thema aanpassen

### Exporteer gasten

Wanneer de zaalindeling naar wens is kan de zaalindeling geëxporteerd worden met de aangegeven knop. Dit zal een Excel spreadsheet geven.

### Herlaad zitplaatsen

Door te drukken op de aangegeven knop zal de zaalindeling herladen worden. Dit kan de locaties van onbevoren stoelen wijzigen. Het zal soms nodig zijn om het algoritme een paar keer te herladen voor ideale resultaten. Daarnaast is het handig om de zaalindeling te herladen wanneer het algoritme verward is.

### Verwijder Alle Gasten

Door op deze knop te drukken zullen alle gasten van alle dagen verwijderd worden. Daarbij zal de applicatie herstart worden. Deze verwijdering kan niet ongedaan gemaakt worden.

### Bevriezen Geplaatste Gasten

Wanneer deze knop ingedrukt wordt zullen alle stoelen van de geselecteerde dag “bevoren” worden. Dat betekent dat de stoelen nooit meer verplaatst zullen worden door het algoritme, ook niet bij het herladen van de zitplaatsen.

Aan het eind van het bevriezen zal het programma een melding geven met het aantal juist bevoren stoelen

Een bevoren stoel wordt aangeduid met een blauw bolletje linksboven de stoel.

### Ontdooien Geplaatste Gasten

Met deze knop zullen alle bevoren stoelen van de geselecteerde dag ontdooien, wat betekent dat ze niet langer bevoren zullen zijn. De stoelen kunnen door het algoritme weer bewogen worden

Aan het eind van het ontdooien zal het programma een melding geven met het aantal ontdooide stoelen



Figuur 9: Knop exporteer gasten



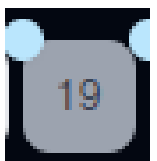
Figuur 11: Knop herlaad zitplekken



Figuur 10: Knop verwijder alle gasten



Figuur 12: Knop bevriezen geplaatste gasten



Figuur 13: Voorbeeld bevoren stoel



Figuur 14: Knop Ontdooien geplaatste gasten



## Toevoegen gasten

Zaalindeler biedt ook een optie om na het importeren van een spreadsheet gasten toe te voegen. Dat kan via de “toevoegen gasten” knop onder het menu. Hierdoor wordt er een nieuwe pagina geladen, waarin er nieuwe gasten kunnen worden toegevoegd. Het is hierbij belangrijk dat deze feature alleen gebruikt wordt voor het toevoegen van kleine hoeveelheden gasten.

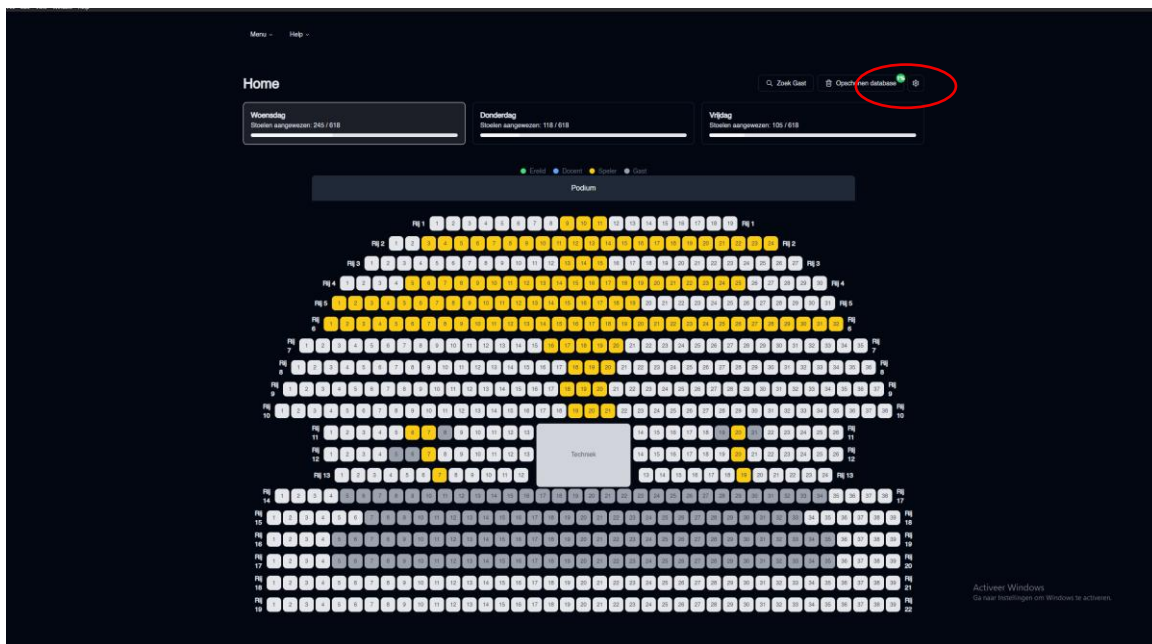


Figuur 15: Knop toevoegen gasten

## BELANGRIJKE INFORMATIE

Allereerst is het van uiterst belang dat de applicatie minimaal eens in de twee weken geopend wordt. Er wordt namelijk gebruik gemaakt van een gratis Redis cloud database. Dit is een database die zich opheft na een periode van inactiviteit. Er is online geen specifieke tijd te vinden voor wanneer dit gebeurt. Hierom is het belangrijk om regelmatig de [groteavond@gsr.nl](mailto:groteavond@gsr.nl) email te checken. Daarnaast is het belangrijk om zaalindeler eens in de twee weken een zaal in te laten delen. Dit houdt de database actief en online.

Daarnaast is het ontzettend belangrijk dat de opslag van de database in de gaten wordt gehouden. Dit kan via zaalindeler, naast de knop “opschonen database” staat een bolletje dat het opslag-percentage aangeeft.



Figuur 16: Afbeelding opschonen database

Wanneer dit bolletje richting de 100% gaat zal hij roder worden. Zorg ervoor dat de database af en toe geleegd wordt. Als dit niet gebeurt zal er tegen een van de gedocumenteerde problemen aangelopen worden (zie [problemen](#)).

Tot slot is het belangrijk dat het programma niet gebruikt kan worden op de gewone GSR schoolwifi. Het programma zal een melding geven dat het niet kan verbinden met de database. Hierom is het belangrijk dat tijdens gebruik op school er een beschikbare hotspot of andere wifi-bron beschikbaar is.

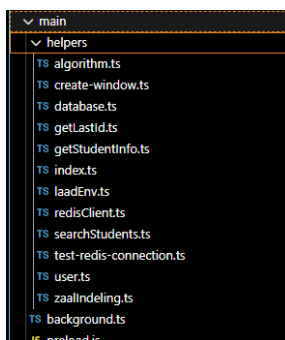
# Developer

## Algemene structuur

Zaalindeler is gebouwd op een Electron-structuur, dat betekent dat het programma verschillende lagen heeft, een renderer, een main, en een preload stuk. Hiermee wordt het meeste van het harde processingwerk in de main component gedaan, de renderer is voornamelijk voor alles wat direct in contact staat met de gebruiker. Voor communicatie tussen de renderer en de main moet er gebruik gemaakt worden van 2 bestanden: background.ts en preload.js.

## Main-gedeelte

Het main gedeelte (zie figuur 17) is geschreven in Typescript. Deze verzameling aan bestanden is de centrale plaats voor bijna al het rekenwerk van Zaalindeler. De volgende kopjes zullen een aantal van de functies van de main bestanden beschrijven.



Figuur 17: Mappenstructuur main

### Algorithm.ts

Dit is het grootste bestand binnen zaalindeler, dit is namelijk waar het stoelindelingsalgoritme zich bevindt. Het is belangrijk om hierbij te weten dat het algoritme werkt met een strafpunten systeem. Allereerst wordt er een seatingMatrix opgebouwd, dus rijen van rijen stoelen (de zaalindeling). Deze stoelen hebben bepaalde eigenschappen (een Typescript interface) genaamd seatAssignment, hierin zijn eigenschappen zoals stoelnummer en of de stoel bevroren is gedefinieerd.

Vervolgens maakt generatePriorityMatrix een nieuwe matrix waarin elke stoel een waarde heeft. Hoe beter de plek hoe groter de kans dat een docent, spelende, of erelid de plek krijgt. Een andere functie die zorgt voor deze grens tussen rollen is getAdjustedPriority. Deze functie zorgt ervoor dat de docenten/spelende/ ereleden eigenlijk alleen maar goede stoelen kunnen krijgen door een multiplier.

Autoassignseating is de belangrijkste functie. Deze functies roept verschillende functies aan om te bepalen wie waar geplaatst wordt. Hierbij wordt rekening gehouden met het soort gast, hun voorkeuren en de dagcapaciteit. Daarnaast roept hij tryAssignSeatsToDay op. Deze functie kijkt of de voorkeursdag van de gast past in de zaalindeling.

Vervolgens is het tijd voor de daadwerkelijke plaatsing, tryNormalSeating. Hier wordt voor elke stoel bekeken of het een goede optie is.

Elke optie krijgt een aantal strafpunten, hoe hoger het aantal strafpunten hoe minder snel het algoritme op een plek zal indelen, op het balkon of gaten achterlaten geeft bijvoorbeeld heel veel strafpunten. Een stoeltoewijzing ziet er hier als volgt uit `seat.guest = guest` de `guest` eigenschap van deze stoel heeft nu de naam van de gast en de stoel is ingedeeld.

Verder heeft het algoritme ook de mogelijkheid om bevroren stoelen te bevatten via de `freezerFunctie` die stoelen een bevroren eigenschap kan geven. Wanneer een stoel bevroren is zal het algoritme de stoel niet verplaatsen. Het checken of een stoel bevroren is en het vervolgens blokkeren van het algoritme wordt gedaan met helper functies zoals `isStoelBeschikbaar`. Zo'n blokkade is hier te zien:

```
// Assign seats (safe)
for (const seat of seatsToCheck) {
  if (!isStoelBeschikbaar(seat)) {
    // blokkeren wanneer bevroren
    console.log(`stoel niet beschikbaar bij toewijzen ${guest.voornaam}`);
    return false;
  }
}
```

Figuur 18: Voorbeeldcode blokkade

### Background.ts

Dit is het bestand dat ervoor zorgt dat het programma wordt opgestart. Het start de app op en het maakt een venster aan op je computer. Daarnaast brengt het heel veel main process functies naar de renderer door ze eerst te importeren uit verschillende main bestanden en ze vervolgens via ipc-handlers doorgeeft

Eerst importeren

```
import './helpers/laadEnv';
import fs from 'node:fs';
import path from 'node:path';
import dotenv from 'dotenv';
import { app } from 'electron';

const { redis: db } = require('./helpers/redisClient');
import { BrowserWindow, ipcMain, nativeTheme } from 'electron'; // Remove dialog
import serve from 'electron-serve';
import {
  getGuests,
  addGuest,
  addGuestsBulk,
  addSeat,
  getSeats,
  connect,
  deleteGuest,
  deleteAllUsers,
  getSeatsForDay,
  setSeatsForDay,
  getDayAssignments,
  updateDayAssignments,
  resetSeatingAndAssignments,
  TOTAL_CAPACITY,
  setSeatingStatus,
  getSeatingStatus,
  getSettings,
  getBlockedSeats,
  blockSeat,
  unblockSeat,
  unblockAllSeats,
  leegDatabase,
  redisGevuldPercentage,
  getAlgorithmLogs
} from './helpers/database';
import { autoAssignSeating, freezerFunctie, ontDooierFunctie } from './helpers/algorithm';
import { Guest } from '../renderer/types/guest';
```

Figuur 19: Voorbeeldcode importeren

Dan doorgeven

```
ipcMain.handle("herstart-app", async () => {
  app.relaunch();
  app.exit(0);
});
```

Figuur 20: Voorbeeldcode doorgeven

### Zaalindeling.ts

Dit bestand toont een gecodeerde weergave van de zaalindeling met arrays. Het bestand wordt overal door Zaalindeler gebruikt om te kunnen werken met de juiste zaalindeling

```
export interface ZaalIndeling {
  rijen: {
    [key: number]: {
      stoelen: any[];
      maxStoelen: number;
      rolstoelPlek?: string[];
      balkon?: boolean;
      ereLidStoelen?: any[];
    };
  };
}

export const zaalIndeling: ZaalIndeling = {
  rijen: {
    0: {
      stoelen: [],
      maxStoelen: 19,
      ereLidStoelen: [],
    },
    1: {
      stoelen: [],
      maxStoelen: 24,
      ereLidStoelen: [],
    },
    2: {
      stoelen: [],
      maxStoelen: 27,
      ereLidStoelen: [],
    },
    3: {
      stoelen: [],
      maxStoelen: 30,
      ereLidStoelen: [],
    },
    4: { stoelen: [], maxStoelen: 31, ereLidStoelen: [] },
    5: { stoelen: [], maxStoelen: 32, ereLidStoelen: [] },
    6: { stoelen: [], maxStoelen: 35, ereLidStoelen: [] },
    7: { stoelen: [], maxStoelen: 36, ereLidStoelen: [] },
    8: { stoelen: [], maxStoelen: 37, ereLidStoelen: [] },
    9: { stoelen: [], maxStoelen: 38, ereLidStoelen: [] },
    10: { stoelen: [], maxStoelen: 13, ereLidStoelen: [] },
    11: { stoelen: [], maxStoelen: 13, ereLidStoelen: [] },
    12: { stoelen: [], maxStoelen: 13, ereLidStoelen: [] },
    13: { stoelen: [], maxStoelen: 13, ereLidStoelen: [] },
    14: {
      stoelen: [],
      maxStoelen: 12,
      rolstoelPlek: ["2 stoelen voor 1 rolstoel"],
      ereLidStoelen: [],
    },
  },
}
```

Figuur 21: Voorbeeldcode zaalindeling.ts

## Preload.js

In dit bestand worden de functies voorgoed doorgegeven aan de renderer.

```
const { contextBridge, ipcRenderer } = require('electron');

contextBridge.exposeInMainWorld('electronAPI', {
  addGuest: (guest) => ipcRenderer.invoke('add-guest', guest),
  sortUsers: () => ipcRenderer.invoke('sortUsers'),
  updateSeatPriority: (rowNumber, seatNumber, newPriority) => ipcRenderer.invoke('update-seat-priority', rowNumber, seatNumber, newPriority),
  getSeatPriority: (seatPriority, ) => ipcRenderer.invoke('get-seat-priority', seatPriority),
  deleteSeatPriority: () => ipcRenderer.invoke('delete-seat-priority'),
  deleteGuest: (guest) => ipcRenderer.invoke('delete-guest', guest),
  deleteAllUsers: () => ipcRenderer.invoke('delete-all-users'),
  addGuestsBulk: (guests) => ipcRenderer.invoke('add-guests-bulk', guests),
  exportUsers: () => ipcRenderer.invoke('export-users'),
  resetSeating: () => ipcRenderer.invoke('reset-seating'),
  getSeatsForDay: (day) => ipcRenderer.invoke('get-seats-for-day', day),
  getDayAssignments: () => ipcRenderer.invoke('get-day-assignments'),
  getGuests: () => ipcRenderer.invoke('get-guests'),
  updateGuest: (guest) => ipcRenderer.invoke('update-guest', guest),
  updateSettings: (settings) => ipcRenderer.invoke('update-settings', settings),
  getSettings: () => ipcRenderer.invoke('get-settings'),
  freezeSeating: (day) => ipcRenderer.invoke('freeze-seating', day),
  ontdooiSeating: (day) => ipcRenderer.invoke('ontdooi-seating', day),
  herstartApp: () => ipcRenderer.invoke('herstart-app'),
  leegDatabase: () => ipcRenderer.invoke('leeg-database'),
  redisGevuldPercentage: () => ipcRenderer.invoke('redis-gevuld-percentage'),
  onSeatingReady: (callback) => {
    const removeListener = () => ipcRenderer.removeListener('seating-ready', callback);
    ipcRenderer.on('seating-ready', callback);
    return removeListener;
  },

  onInitProgress: (callback) => {
    const removeListener = () => ipcRenderer.removeListener('initialization-progress', callback);
    ipcRenderer.on('initialization-progress', (_, progress) => callback(progress));
    return removeListener;
  },

  getSeatingStatus: () => ipcRenderer.invoke('get-seating-status'),

  onDatabaseError: (callback) => {
    const removeListener = () => ipcRenderer.removeListener('database-connection-error', callback);
    ipcRenderer.on('database-connection-error', (_, error) => callback(error));
    return removeListener;
  },

  // Add new methods for blocked seats
  getBlockedSeats: (day) => ipcRenderer.invoke('get-blocked-seats', day),
  blockSeat: (blockedSeat) => ipcRenderer.invoke('block-seat', blockedSeat),
  unblockSeat: (day, row, seat) => ipcRenderer.invoke('unblock-seat', day, row, seat),
  unblockAllSeats: () => ipcRenderer.invoke('unblock-all-seats'),
```

Na het declareren in preload.js zijn de functies aan te roepen in de renderer met `window.electronAPI.functienaam`

Het doorgeven van functies via background.ts en preload.js naar de renderer is essentieel voor het werken van Zaalindeler

## Renderer-gedeelte

### Algemeen

De renderer maakt gebruik van een aantal belangrijke dingen. Allereerst wordt er gebruik gemaakt van React, een javascript library dat ervoor zorgt dat er gemakkelijk HTML geïntegreerd kan worden in javascript. Daarnaast maakt de renderer op de pages (de home pagina, de import pagina etc) gebruik van Tailwind, een efficiëntere CSS dat meteen de CSS integreert in HTML met afgekorte commando's. Een voorbeeld hieronder:

```
<span className="text-sm text-gray-600">Techniek</span>
```

Figuur 22: Voorbeeldcode tailgate

Text-sm betekent dat de tekstgrootte klein moet zijn

Text-gray-600 betekent dat de tekst grijs moet zijn met een intensiteit van 600 (1000 is maximaal).

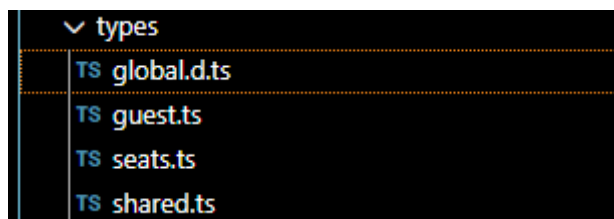
### Pages

De renderer bevat alle pagina's van zaalindeler. Elk van de pagina's is geschreven in Tsx (een soort React-Typescript)

### Types

Typescript verwacht van elk van de variabelen en objecten een soort blauwdruk van wat voor type elk variabele is en welke interface elk object heeft. Dat wordt voor een groot deel afgehandeld in de bestanden in het "types" mapje.

De belangrijkste hiervan is global.d.ts. Hier worden grote functies die in het hele programma gebruikt worden gedefinieerd voor Typescript. Als dit niet juist gebeurt zullen er errors komen.



Figuur 23: Mappenstructuur types

Voorbeeld van redisGevuldPercentage global.d.ts het product van deze functie mag nu alleen een nummer zijn.

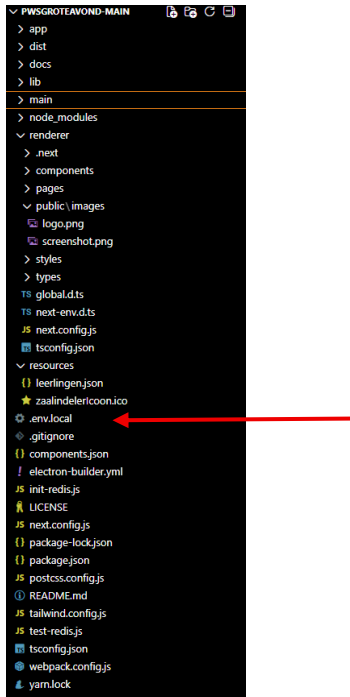
```
redisGevuldPercentage: () => Promise<number>;
```

Figuur 24: Voorbeeldcode redisGevuldPercentage

## Redis database

### Algemene informatie

Zaalindeler maakt ook gebruik van een Redis cloud database. Deze is aangemaakt op het [groteavond@gsr.nl](mailto:groteavond@gsr.nl) account, toegang tot dit email-account is te verkrijgen via de ICT en mevrouw Groot-Nibbelink. Het programma zelf moet natuurlijk ook verbinding maken met Redis. Dat doet het programma doormiddel van een .env bestand, .env.local. Deze los te vinden in de hoofdmap



Figuur 25: Locatie .env.local

In dit bestand moet de volgende informatie staan:

- De Redis username
- De Redis Port
- Het Redis wachtwoord (niet van het account, maar van de database zelf)
- De Redis Host
- En Redis TLS op false

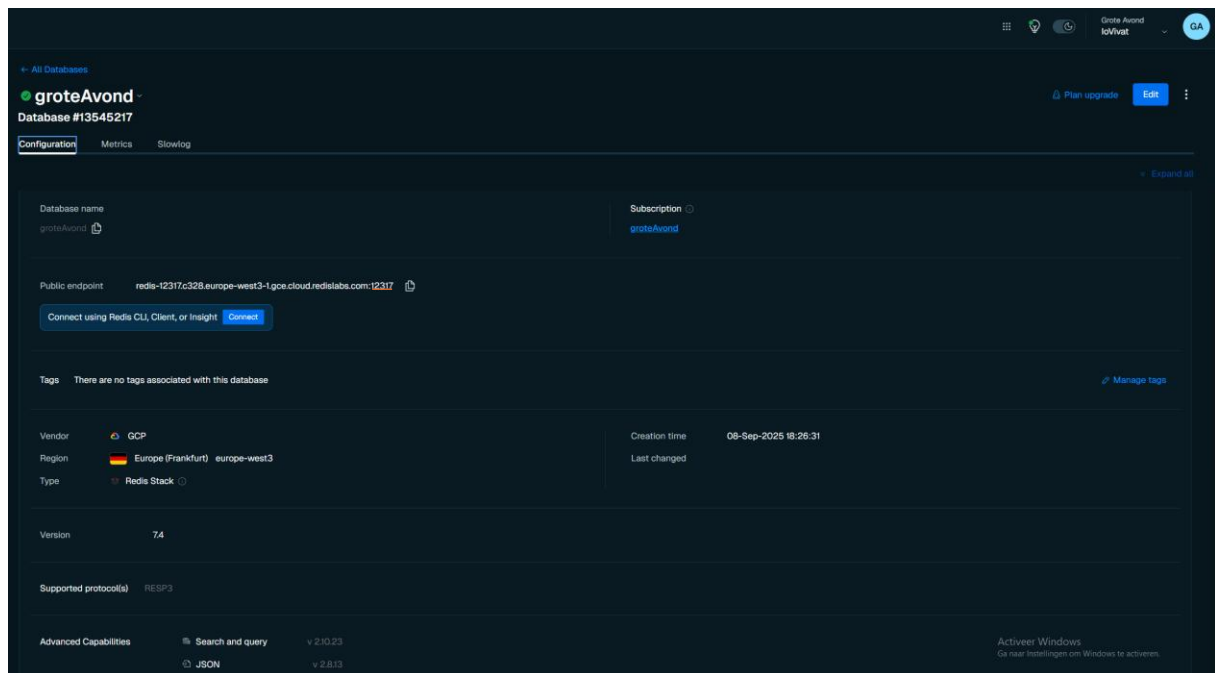
```
2 REDIS_USERNAME=default
3 REDIS_PASSWORD=4rxjIYeGT63iwRUn1bAS79abigdyb2YI
4 REDIS_HOST=redis-12317.c328.europe-west3-1.gce.redns.redis-cloud.com
5 REDIS_PORT=12317
6 REDIS_TLS=false
```

Figuur 26: Voorbeeldcode .env.local

Elk van deze dingen zijn te vinden op een ingelogde Redis website.

Redis username: Default, als er geen andere users zijn (ons geval)

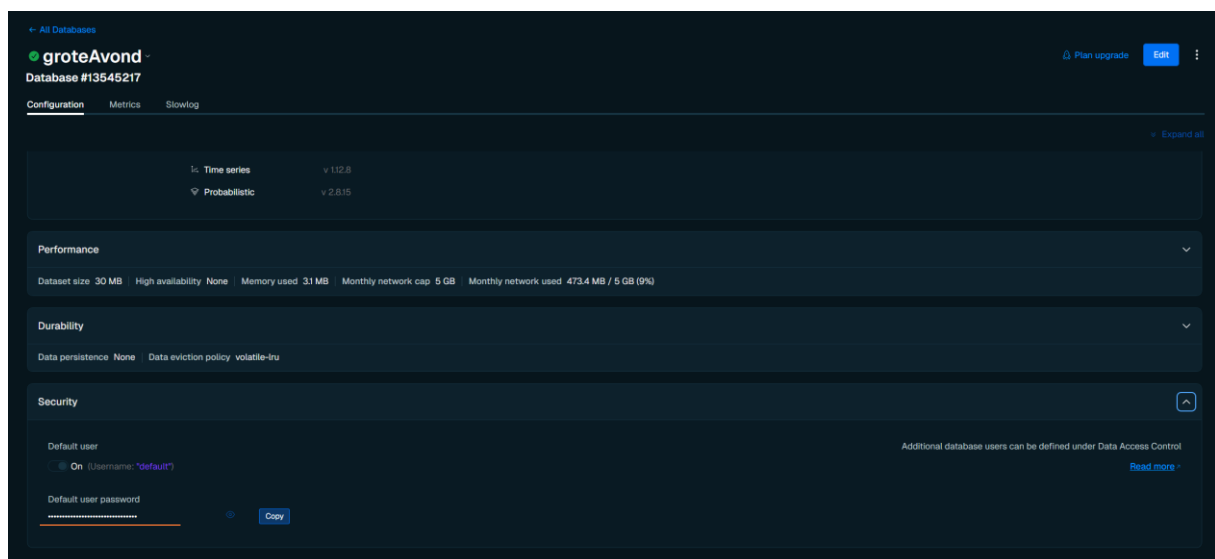
Redis port:



Figuur 27: Locatie redis port

Zorg dat alleen het getal aan het eind van de link gekopieerd wordt voor de port

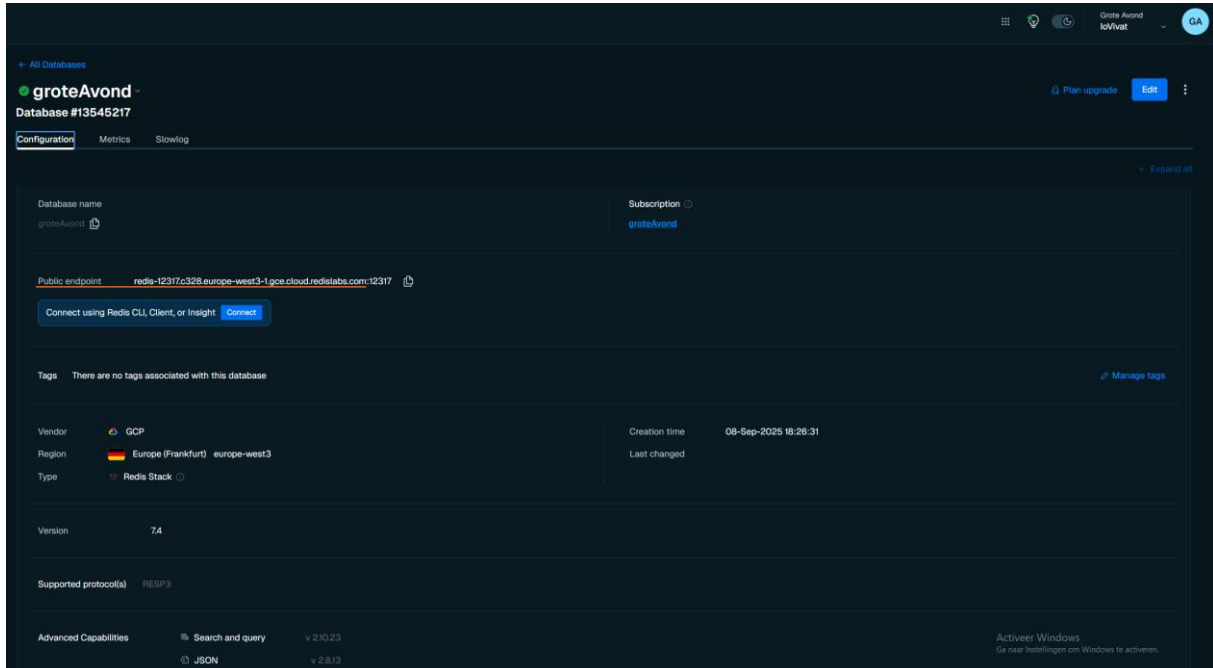
Redis wachtwoord:



Figuur 28: Locatie redis wachtwoord



Redis host:



Figuur 29: Locatie redis host

Zorg ervoor dat er zonder het getal (in dit geval 12317) gekopieerd wordt.

### *Wat als de database uit de lucht gaat?*

Bij het ongelukkige geval dat de database uit de lucht gaat moeten er een aantal stappen uitgevoerd worden. Om sommige van deze stappen uit te voeren is het aangeraden om visual studio code te installeren (<https://code.visualstudio.com/>).

### Stap 1: Maak een nieuwe Redis database aan

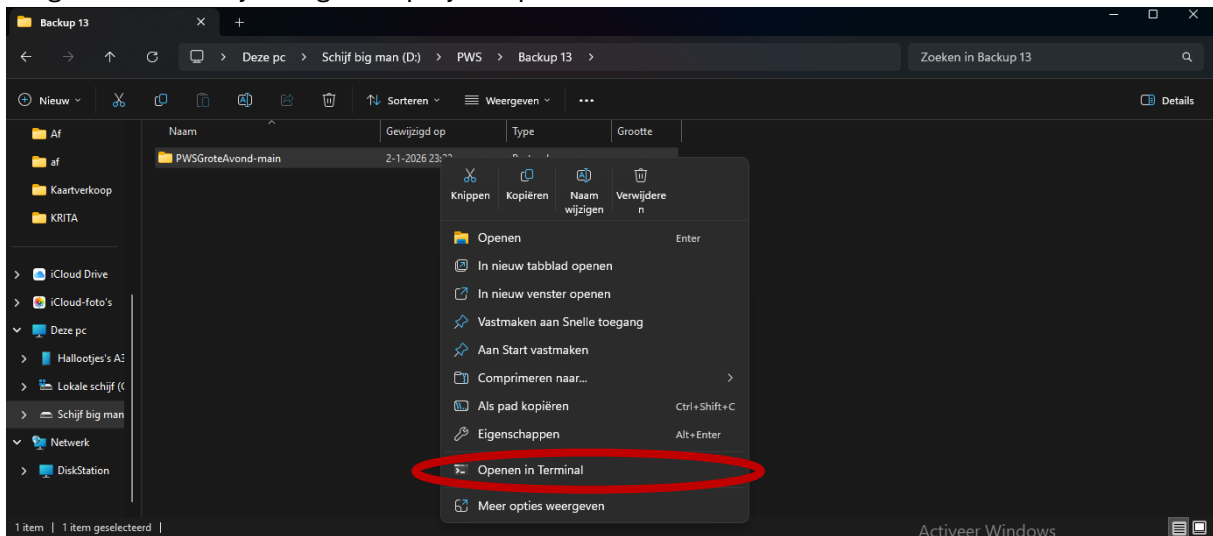
De allereerste stap vereist dat er een nieuwe database gemaakt moet worden op de Redis website: <https://redis.io/>. Maak hierop een nieuwe gratis database op het [groteavond@gsr.nl](mailto:groteavond@gsr.nl) email-account.

### Stap 2: Vul .env.local

Vind het .env.local bestand en pas het aan met de gegevens van de nieuwe database.

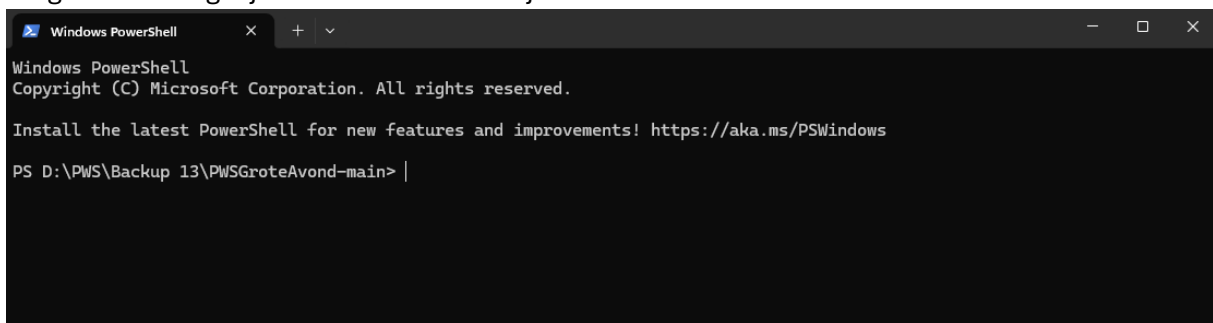
### Stap 3: Voer het init-redis.js script uit

Zorg allereerst dat je het gehele project opent in Terminal:



Figuur 30: Openen in Terminal

Zorg dat een vergelijkbaar scherm verschijnt:



Figuur 31: Goed scherm powershell

Voer de volgende commando's hier uit:

- node -v
- npm install ioredis dotenv
- node init-redis.js

Als de cmd window dit teruggeeft is het gelukt: Verbinding succesvol

Stoelindeling geladen. Totale capaciteit =

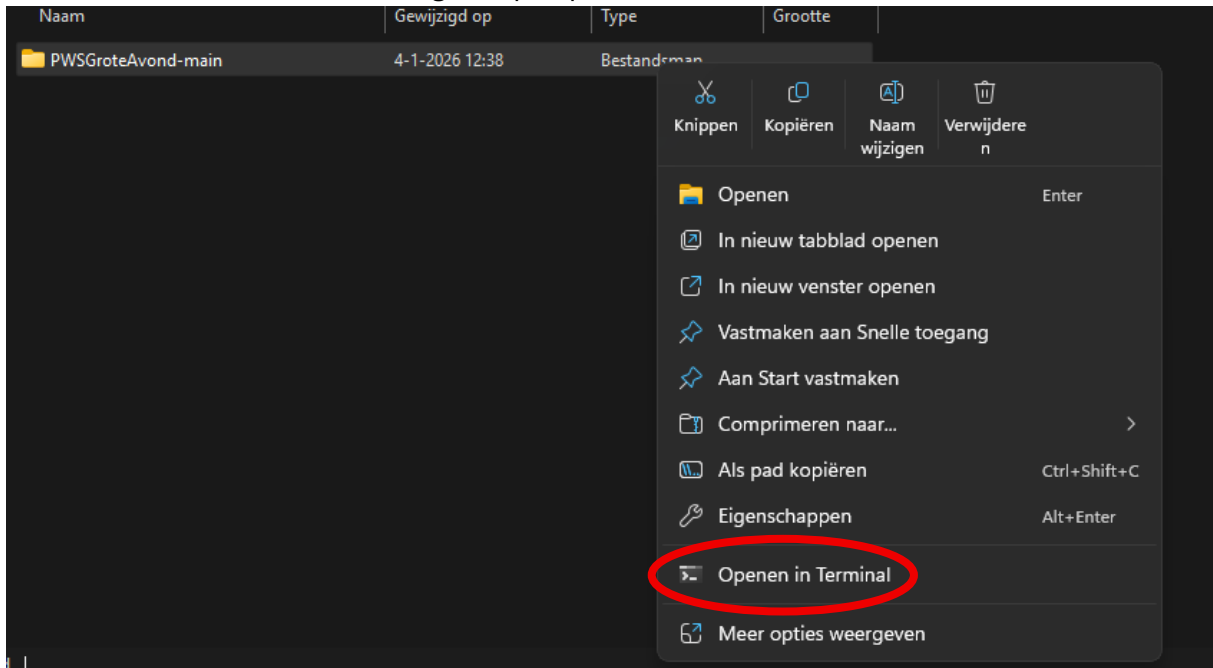
Het is gelukt, de stoelen zijn ingeladen, WooO!!.

Als de vorige stappen goed uitgevoerd waren is er nu een nieuwe geladen database

## Developer modus uitvoeren

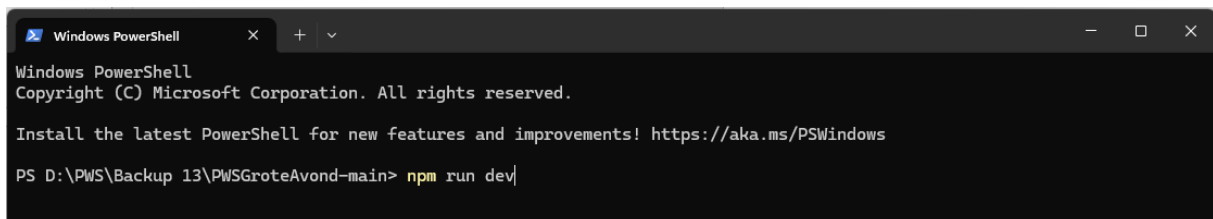
Wanneer er als developer gesleuteld wordt moet het programma vaak worden herstart om veranderingen in de code in actie te zien. Hier is het niet handig dat het programma elke keer opnieuw ingepakt moet worden. Hierom is er een developer modus, waarin Zaalindeler uitgevoerd kan worden. Om dit te laten werken moet Nodejs van de volgende website gedownload worden: <https://nodejs.org/en/download>. Download hier de Windows installer.

Pak vervolgens Zaalindeler uit, wanneer dit gedaan is klik met rechtermuisknop op het mapje PWSGroteAvond-main. Klik vervolgens op “Openen in Terminal”



Figuur 32: Openen in Terminal

In het scherm dat nu verschijnt moet het volgende geschreven worden: `npm run dev`

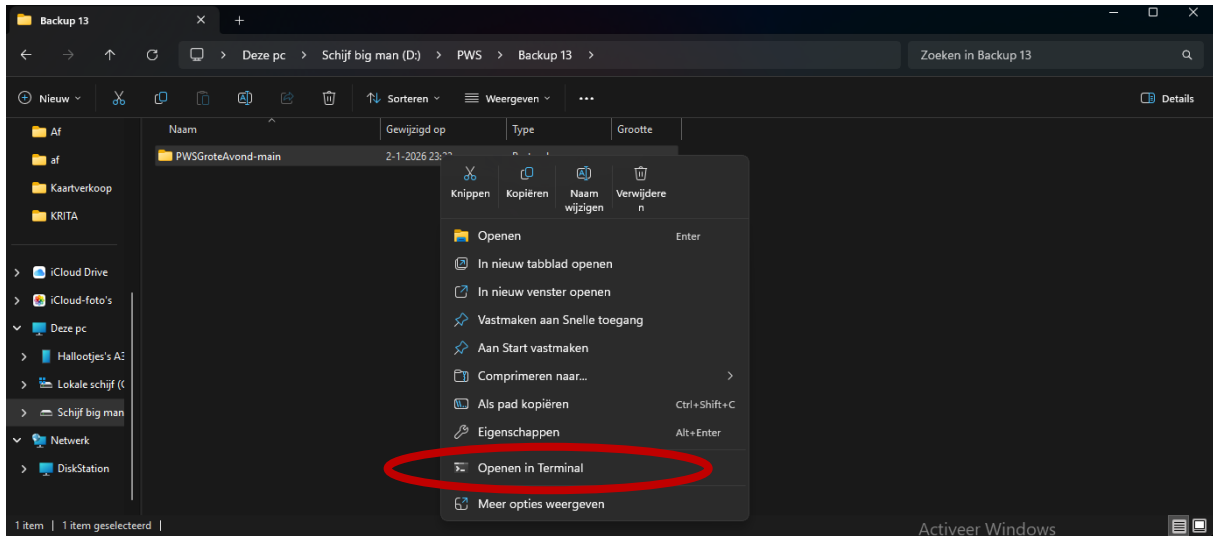


Figuur 33: `npm run dev`

Zaalindeler zal nu in development modus uitgevoerd worden. Dit is een stuk sneller dan Zaalindeler bij elke nieuwe test in te moeten pakken.

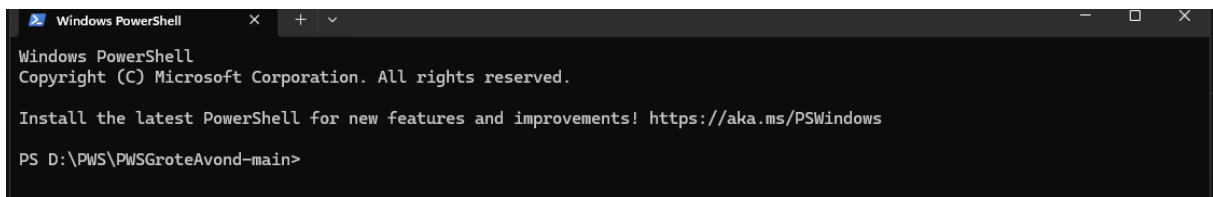
## Het programma inpakken

Om het programma na aanpassingen weer in te pakken tot een app moet het volgende gedaan worden: Open het bestand opnieuw in cmd.



Figuur 34: Openen in Terminal

Zorg dat er een vergelijkbaar scherm komt:



Figuur 35: Openingsscherm powershell

Voer het volgende commando uit in dit scherm: `npm run build`

Het programma zal vervolgens voor enkele minuten aan het laden zijn. Als het afgerond is zal het de volgende lines laten zien:



Figuur 36: Lines bij geslaagd

Ga nu naar het PWSGroteAvond-main bestand, en zoek de folder genaamd “dist”.

Naam	Gewijzigd op	Type	Grootte
app	3-1-2026 00:56	Bestandsmap	
dist	3-1-2026 00:57	Bestandsmap	
docs	2-1-2026 13:56	Bestandsmap	
lib	2-1-2026 13:56	Bestandsmap	
main	2-1-2026 13:56	Bestandsmap	
node_modules	2-1-2026 14:00	Bestandsmap	
renderer	2-1-2026 14:00	Bestandsmap	
resources	2-1-2026 17:24	Bestandsmap	
.env.local	2-1-2026 23:21	LOCAL-bestand	1 kB
.gitignore	10-9-2025 19:04	Git Ignore Source ...	1 kB
components.json	10-9-2025 19:04	JSON-bestand	1 kB
electron-builder	2-1-2026 17:25	Yaml Source File	1 kB
init-redis.js	2-1-2026 23:54	JSFile	3 kB
LICENSE	10-9-2025 19:04	Bestand	12 kB
next.config.js	10-9-2025 19:04	JSFile	1 kB

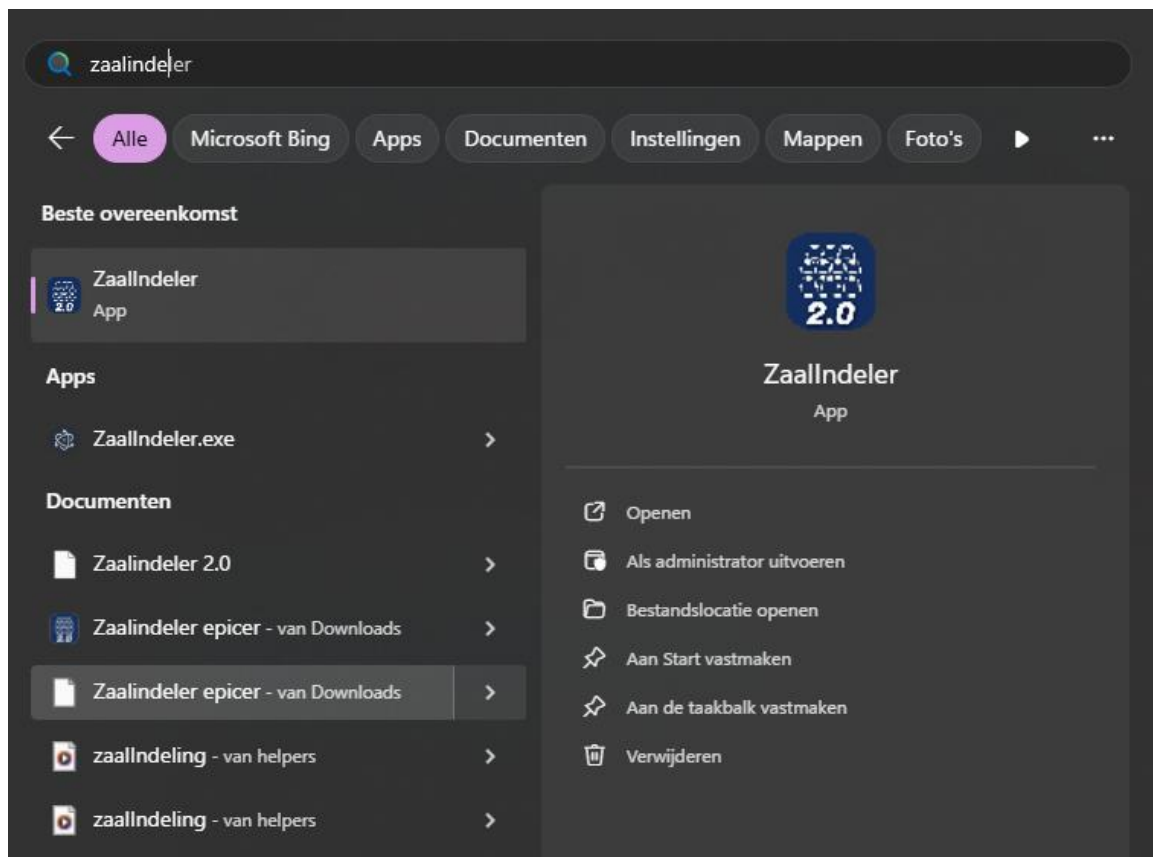
Figuur 37: Map dist

Voer hier het bestand “ZaalIndeler Setup 1.1.5” uit

Naam	Gewijzigd op	Type	Grootte
win-unpacked	3-1-2026 00:56	Bestandsmap	
builder-debug	3-1-2026 00:57	Yaml Source File	7 kB
builder-effective-config	3-1-2026 00:56	Yaml Source File	1 kB
ZaalIndeler Setup 1.1.5	3-1-2026 00:57	Toepassing	140.325 kB
ZaalIndeler Setup 1.1.5.exe.blockmap	3-1-2026 00:57	BLOCKMAP-besta...	148 kB

Figuur 38: ZaalIndeler Setup 1.1.5

Het programma zal even installeren, wanneer het klaar is zal het programma te vinden zijn in de taakbalk:

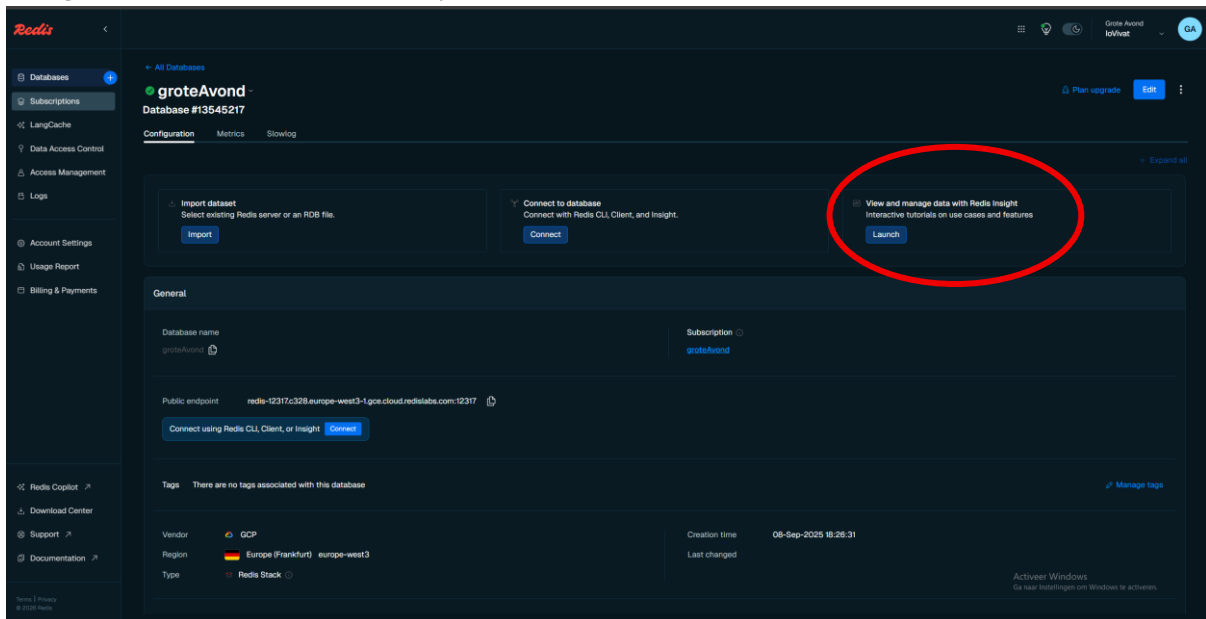


Figuur 39: ZaalIndeler in taakbalk

# Mogelijke problemen bij het opstarten van Zaalindeler

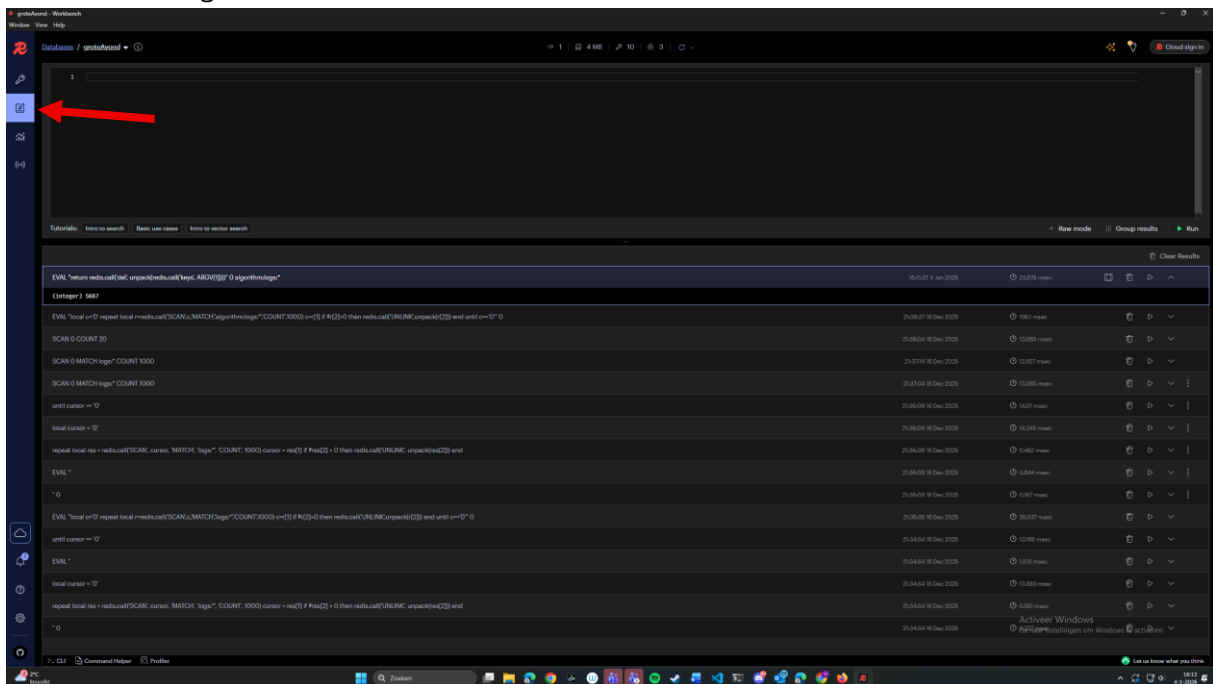
## Oneindig laadscherm

Wanneer de database vol is zal het programma oneindig blijven laden. Dit is op te lossen door allereerst naar de Redis cloud database website te gaan en daar in te loggen <https://redis.io/>. Ga vervolgens naar het dashboard van de groteAvond database en launch Redis insight (als het nog niet geïnstalleerd is, installeer het.)



Figuur 40: Redis Insight

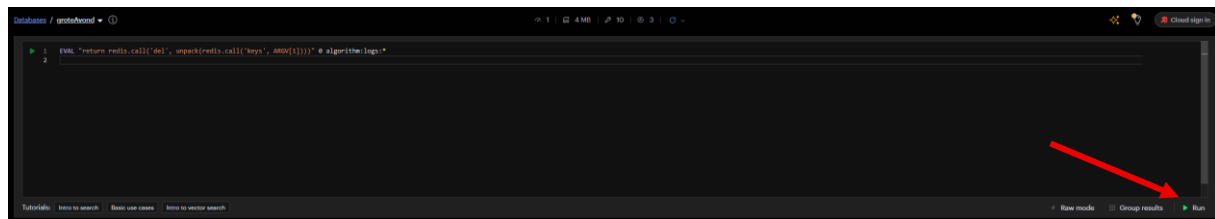
Ga in Redis insight naar de workbench:



Figuur 41: Workbench

En typ boven in de volgende command: EVAL "return redis.call('del', unpack(redis.call('keys', ARGV[1])))" 0 algorithm:logs:\*

Klik vervolgens op “Run”



*Figuur 42: Knop run*

De database is nu geleegd en Zaalindeler zal weer werken.

## Andere problemen

Bij andere problemen met Zaalindeler kunt u altijd terecht bij de ICT.



# Bestelwebsite en Supabase database

## Informatie

De bestelwebsite werkt door een plugin genaamd “bestellingGA”, welke drie verschillende bestanden bevat. Het JavaScript-bestand “bestellingGA.js” voor de functionaliteit, het PHP-bestand “bestellingGA.php” voor onder andere de foutmeldingen en het CSS-bestand “bestellingGA.css” voor het uiterlijk. In deze handleiding wordt op eenvoudige wijze beschreven hoe de belangrijkste punten kunnen worden aangepast, zonder de functionaliteit te breken.

De Supabase database kan worden gezien als de tussenpersoon tussen de bestelwebsite en de Zaalindeler. Voor de functionaliteit van de database wordt het TypeScript-bestand “index.ts” gebruikt.

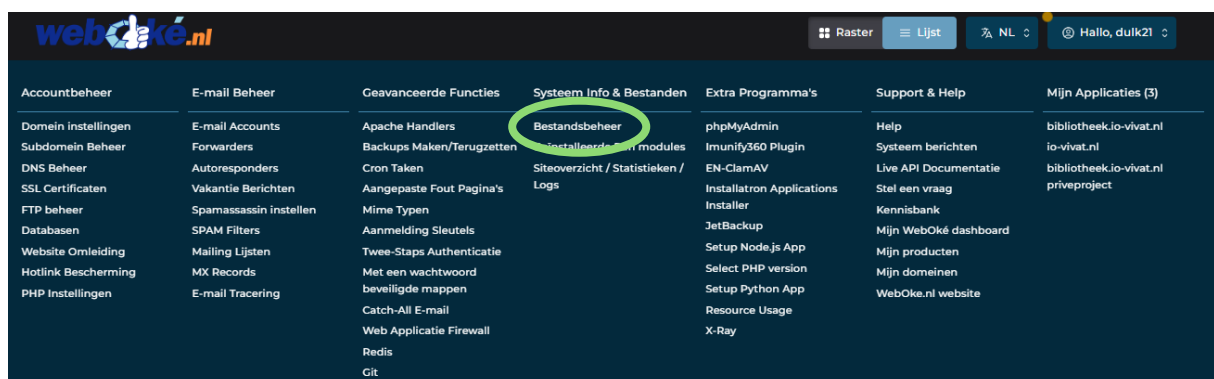
Bij Supabase is er nu een gratis account, waardoor de database na 1 week van inactiviteit wordt gepauzeerd. Daarna heb je 90 dagen om de database weer te activeren, voordat deze definitief wordt verwijderd. Het opnieuw activeren van de Supabase database kan op de website van Supabase, op het dashboard. Hiervan worden ook mails gestuurd naar [groteavond@gsr.nl](mailto:groteavond@gsr.nl). Daarom is het belangrijk om dit e-mailadres in de gaten te houden.

## Toegang tot, en updaten van het JavaScript-bestand, PHP-bestand en CSS-bestand (website bestanden)

De toegang tot, en het updaten van het JavaScript-bestand, PHP-bestand en CSS-bestand is belangrijk, zodat de webpagina kan worden bijgewerkt. Het JavaScript-bestand heet: *bestellingGA.js*. Het PHP-bestand heet: *bestellingGA.php*. Het CSS-bestand heet: *bestellingGA.css*.

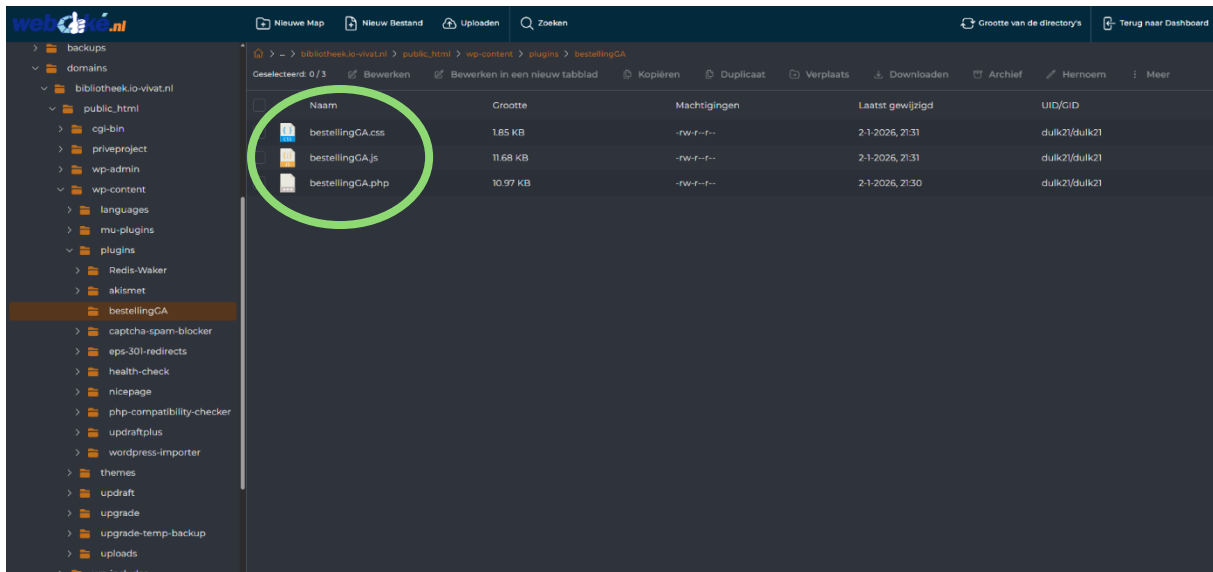
Stap 1: Om toegang te krijgen tot de bestanden, moet je inloggen op het WebOke admin dashboard, waarvan helemaal onderaan dit document de inloggegevens en link naar de website staan beschreven.

Stap 2: Klik in het menu bovenaan de pagina op de knop: *Bestandsbeheer*. (afbeelding x)



Figuur 43: Toegang en updaten website bestanden; Stap 2 Bestandsbeheer

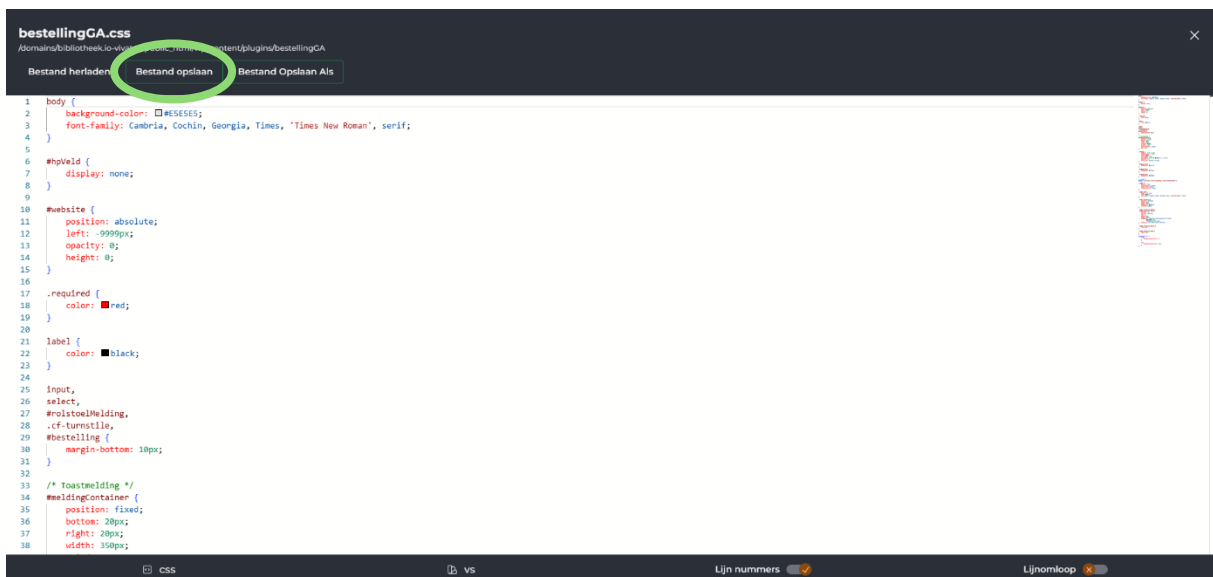
Stap 3: Dubbelklik vervolgens op de volgende 5 mappen, om naar de juiste bestanden te gaan: *bibliotheek.io-vivat.nl* -> *publik\_html* -> *wp\_content* -> *plugins* -> *bestellingGA*. Nu zie je het scherm dat te zien is op de afbeelding bovenaan de volgende pagina. De 3 bestanden zijn op de afbeelding omcirkeld. (afbeelding x)



Figuur 44: Toegang en updaten website bestanden; Stap 2 toegang gevonden

Stap 3: Om een bestand te bewerken en updaten, moet je dubbel klikken op het bestand en het wordt geopend. Bewerk nu de code die je wilt bewerken.

Stap 4: Om het bestand te updaten na het bewerken van de code, moet je op de knop: Bestand opslaan klikken. Deze staat links bovenaan de pagina. (afbeelding x)



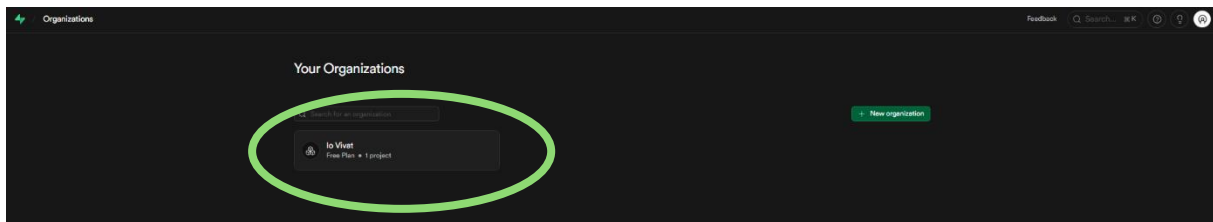
Figuur 45: Toegang en updaten website bestanden; Stap 4 Bestand opslaan

## Toegang tot, en updaten van de Supabase Edge Function

Ook is het belangrijk om de code van de database te kunnen bewerken en updaten, zodat deze goed samen blijft werken met de website. Deze code van de database heet de Edge Function.

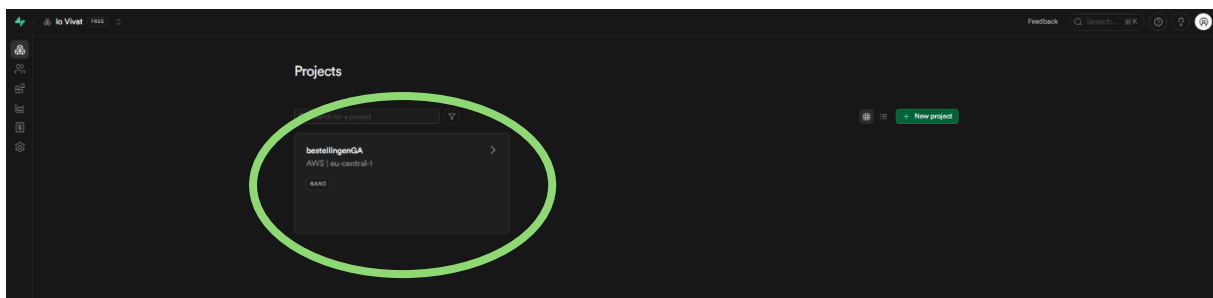
Stap 1: Om toegang te krijgen tot de Edge Function, moet je inloggen op de Supabase website, waarvan helemaal onderaan dit document de inloggegevens en link naar de website staan beschreven.

Stap 2: Klik op organisatie: *Io Vivat*. (afbeelding x)



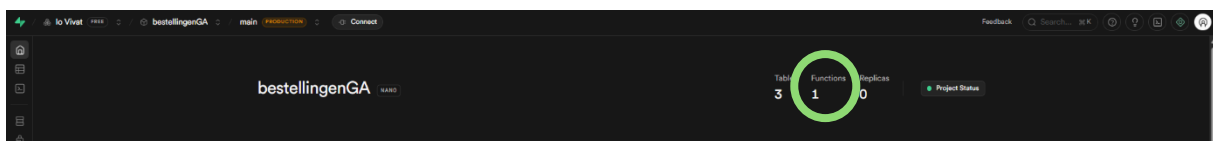
Figuur 46: Toegang en updaten Edge Function; Stap 2 Organisations

Stap 3: Klik daarna op het project: *bestellingGA*. (afbeelding x)



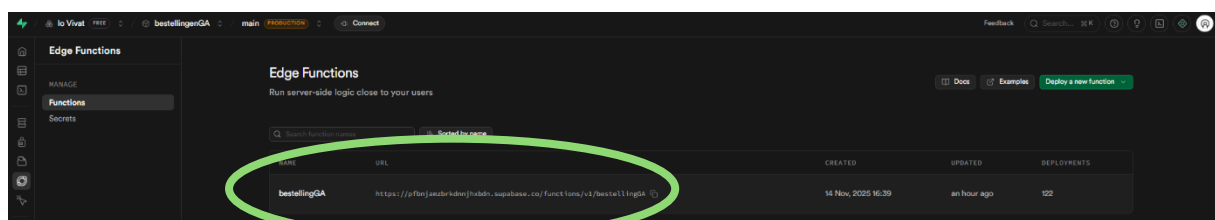
Figuur 47: Toegang en updaten Edge Function; Stap 3 Projects

Stap 4: Klik daarna rechts van het midden op *Functions*. (afbeelding x)



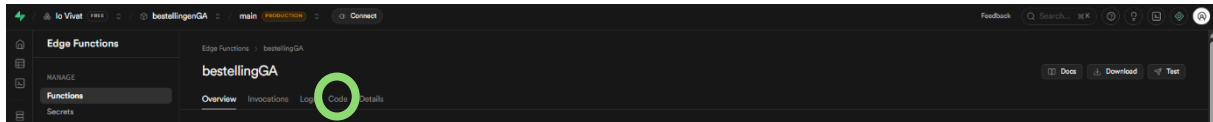
Figuur 48: Toegang en updaten Edge Function; Stap 4 Functions

Stap 5: Klik op de Edge Function: *bestellingGA*. (afbeelding x)



Figuur 49: Toegang en updaten Edge Function; Stap 5 Edge Function bestellingGA

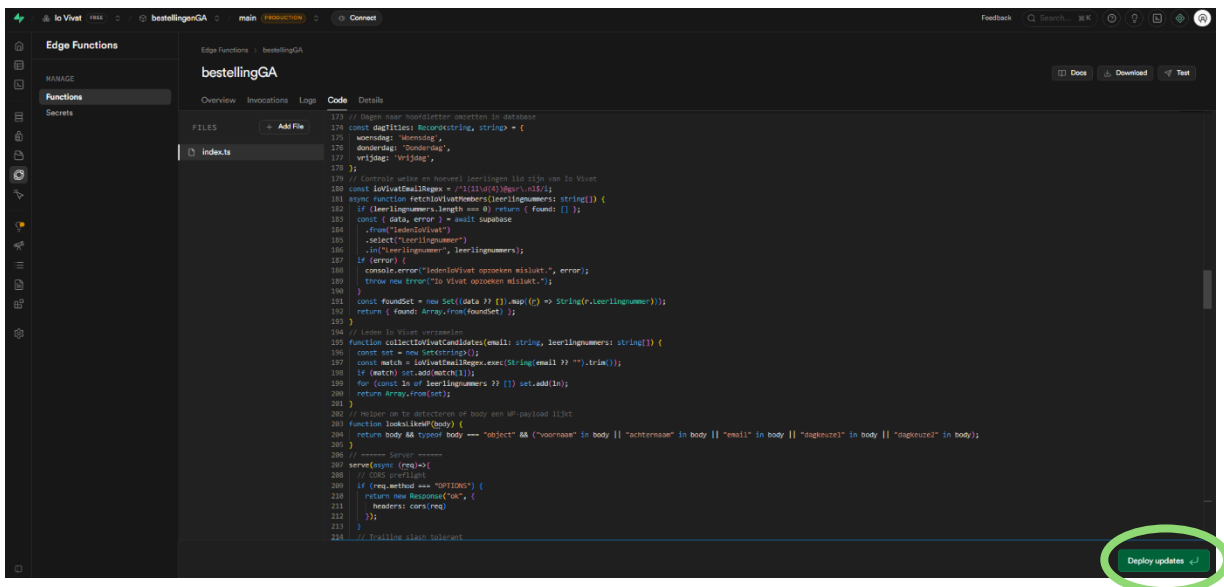
Stap 6: Klik daarna op: **Code** (afbeelding x)



Figuur 50: Toegang en updaten Edge Function; Stap 6 Code

Stap 7: Je ziet de Edge Function code van de database. Deze kan je nu bewerken.

Stap 8: Om het bestand te updaten na het bewerken van de code, moet je op de groene knop: **Deploy updates** drukken, helemaal rechtsonder in de pagina. (afbeelding x)



Figuur 51: Toegang en updaten Edge Function; Stap 8 Deploy updates

## Wat te doen voor de kaartverkoop van een Grote Avond

Voordat de Grote Avond weer kan starten, zijn er 5 dingen die gedaan moeten worden. Is dit belangrijk, zodat alles vlekkeloos kan verlopen.

Stap 1: Controleer of de Supabase tabel met bestellingen: *bestellingenGA* leeg is. Hoe dit moet staat beschreven onder het kopje: [Supabase tabel legen](#).

Stap 2: Update de Supabase tabel met leden van lo Vivat: *ledenloVivat*. Hoe dit moet staat beschreven onder het kopje: [Leden lo Vivat aanpassen](#). Volg stap 8 t/m 14 om de leden in de tabel te zetten.

Stap 3: Update de Supabase tabel met meespelende leerlingen: *spelendeLeerlingen*. Hoe dit moet staat beschreven onder het kopje: [Meespelende leerlingen aanpassen](#). Volg stap 7 t/m 8 om de leerlingen in de tabel te zetten.

Stap 4: Update de data op de website. Hoe dit moet staat beschreven onder het kopje: [Dagen op de website aanpassen](#).

Stap 5: Update de bevestigingsmail qua thema naar het thema van de nieuwe Grote Avond. Hoe dit moet staat beschreven onder het kopje: [Bevestigingsmail aanpassen](#).

## Wat te doen tijdens de kaartverkoop van een Grote Avond

Om een goede zaalindeling te kunnen maken met de Zaalindeler, is het belangrijk dat de Supabase tabel met bestellingen: *bestellingenGA* goed wordt geëxporteerd. Hoe dit moet staat beschreven onder het kopje: [Supabase tabel exporteren](#).

## Wat te doen na de kaartverkoop van een Grote Avond

Als de Grote Avond is afgelopen, zijn er x dingen die moeten worden gedaan, zodat alles weer klaar is voor het volgende jaar.

Stap 1: Verwijder alle bestellingen uit de Supabase tabel met bestellingen: *bestellingenGA*. Hoe dit moet staat beschreven onder het kopje: [Supabase tabel legen](#).

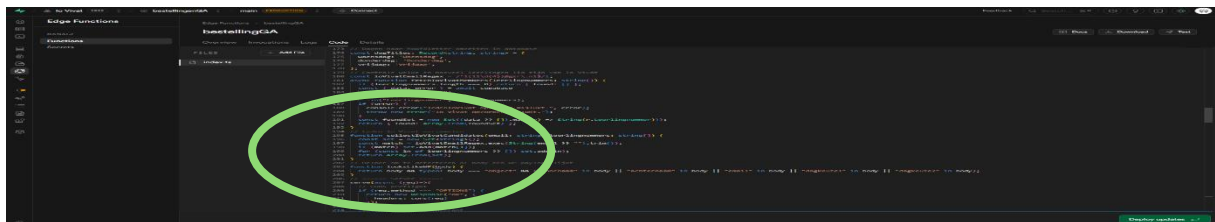
Stap 2: Verwijder alle leden van lo Vivat uit de Supabase tabel met leden van lo Vivat: *ledenloVivat*. Hoe dit moet staat beschreven onder het kopje: [Leden lo Vivat aanpassen](#). Volg stap 1 t/m 7 om de leden te verwijderen.

Stap 3: Verwijder alle meespelende leerlingen uit de Supabase tabel met meespelende leerlingen: *spelendeLeerlingen*. Hoe dit moet staat beschreven onder het kopje: [Meespelende leerlingen aanpassen](#). Volg stap 1 t/m 6 om de leerlingen te verwijderen.

## Supabase tabel legen

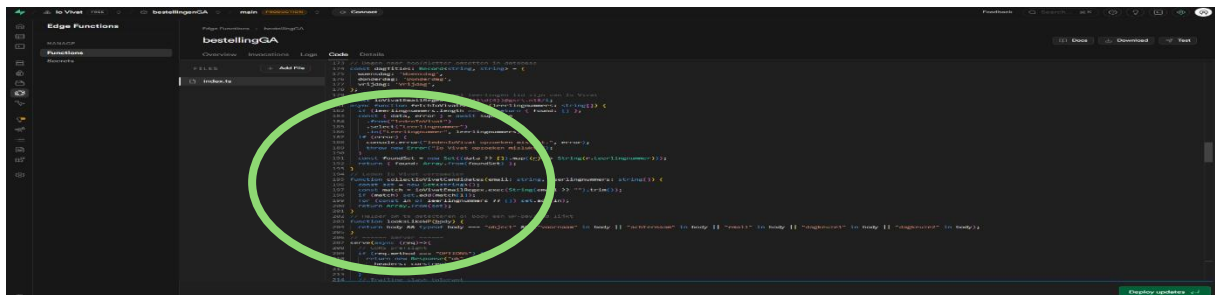
Het legen van de Supabase tabel met bestellingen: *bestellingenGA* is belangrijk, omdat met een lege tabel de bestellingen opnieuw beginnen. Let op, met de code uit stap 5 worden alle gegevens uit de tabel: *bestellingenGA* verwijderd, deze kan daarna niet meer terugkrijgen.

Stap 1: Log in op de Supabase website, en klik op de organisatie: Io Vivat. (afbeelding x)



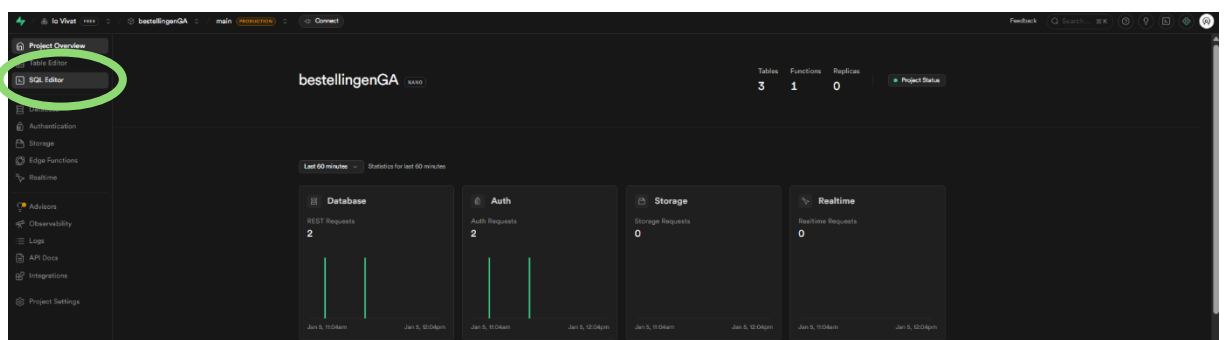
Figuur 52: Supabase tabel legen; Stap 1 Organisation

Stap 2: Klik daarna op het project: *bestellingGA*. (afbeelding x)



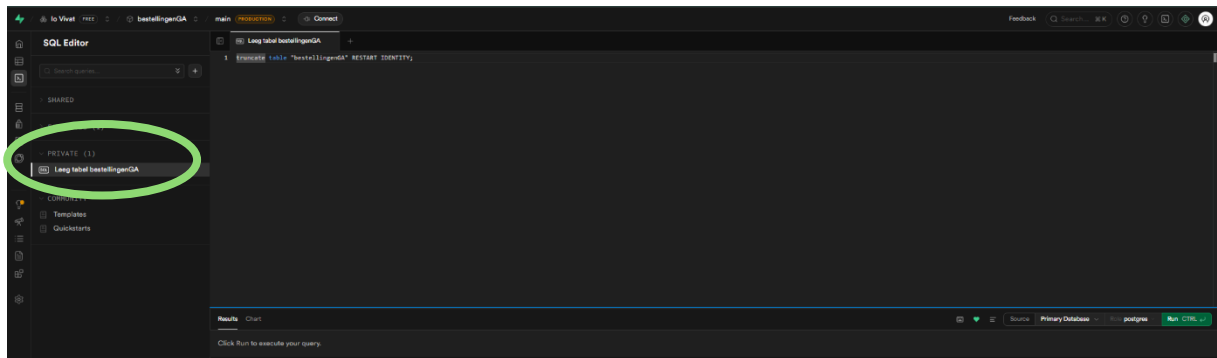
Figuur 53: Supabase tabel legen; Stap 2 Projects

Stap 3: Ga met je muis helemaal links op één van de icoontjes staan, en wacht tot het scherm open gaat zoals te zien is op onderstaande afbeelding. Klik op de derde knop van boven: *SQL Editor*. (afbeelding x)



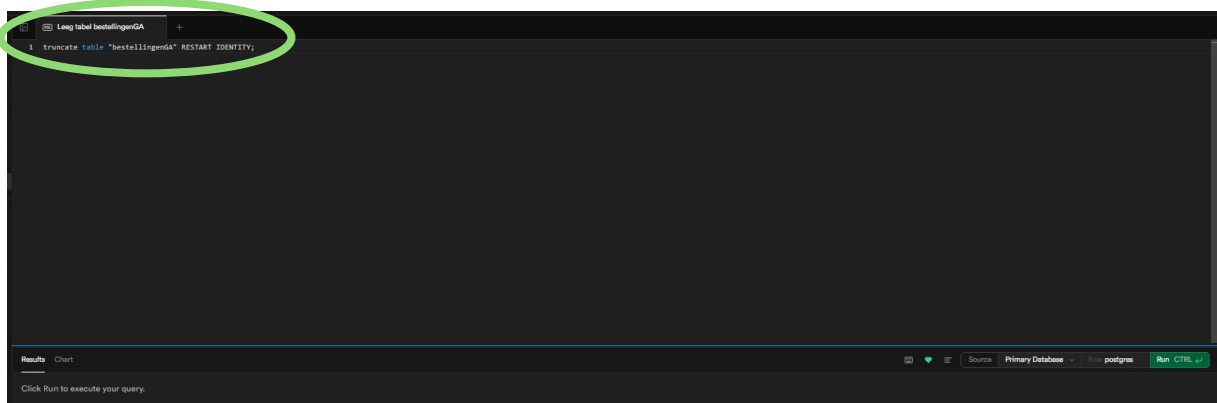
Figuur 54: Supabase tabel legen; Stap 3 SQL Editor

Stap 4: Klik links op het pijltje links van *PRIVATE (1)*, zodat deze open klapt. Klik daarna op *Leeg tabel bestellingenGA*. Dan zie je het scherm zoals in onderstaande afbeelding. (afbeelding x)



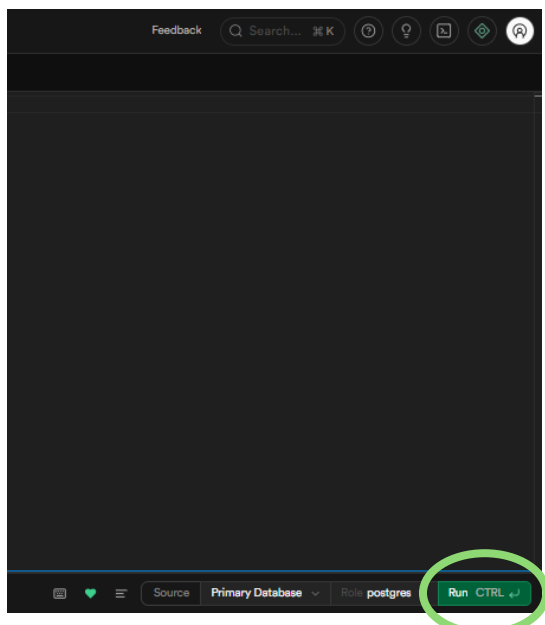
Figuur 55: Supabase tabel legen; Stap 4 Leeg tabel bestellingenGA

Stap 5: Controleer of er de volgende regel code staat: *truncate table "bestellingenGA" RESTART IDENTITY;*. Als deze code er niet staat, verwijder dan alle code in het scherm, en plak bovenstaande code in het scherm. (afbeelding x)



Figuur 56: Supabase tabel legen; Stap 5 controle code

Stap 6: Nu de code er staat, kunnen we de code laten uitvoeren. Doe dit door op de groene knop rechts halverwege het scherm te klikken: *Run*.

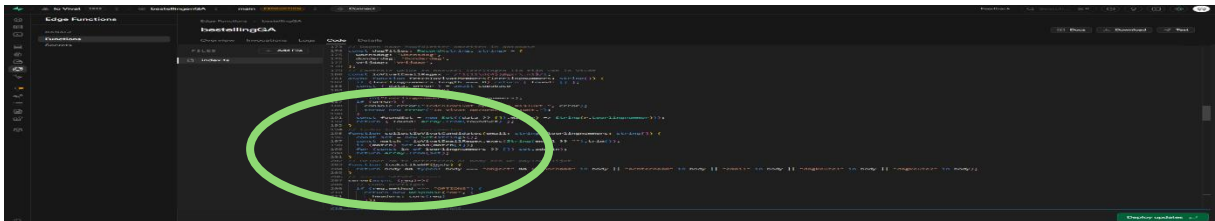


Figuur 57: Supabase tabel legen; Stap 6 Run

## Supabase tabel exporteren

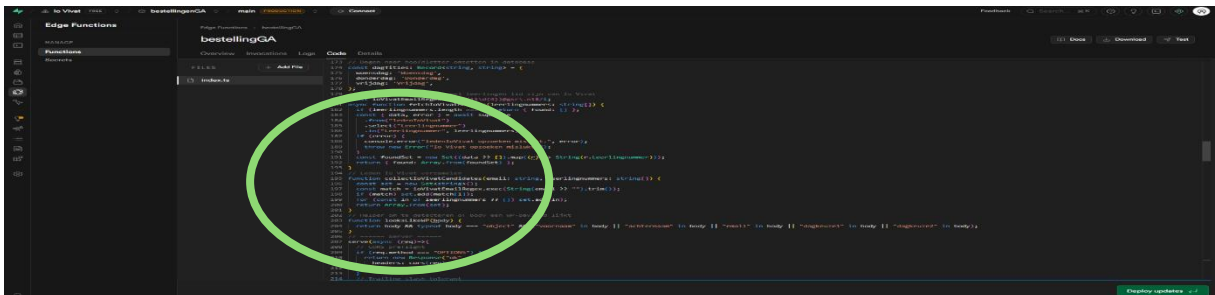
Om in Supabase de gegevens uit een tabel te exporteren, heb je twee opties. Optie één is om de gehele tabel met alle gegevens die erin staan te exporteren. Optie twee is om zelf een aantal bestellingen te kiezen, en deze te exporteren. Beide opties worden behandeld in deze handleiding.

Stap 1: Log in op de Supabase website, en klik op de organisatie: lo Vivat. (afbeelding x)



Figuur 58: Supabase tabel exporteren; Stap 1 Organisation

Stap 2: Klik daarna op het project: *bestellingGA*. (afbeelding x)



Figuur 59: Supabase tabel exporteren; Stap 2 Projects

Stap 3 Klik daarna rechts van het midden op *Tables*. (afbeelding x)



Figuur 60: Supabase tabel exporteren; Stap 3 Tables

Stap 4: Klik daarna in het linker navigatie-menu op de bovenste tabel, genaamd *bestellingenGA*. (afbeelding x)

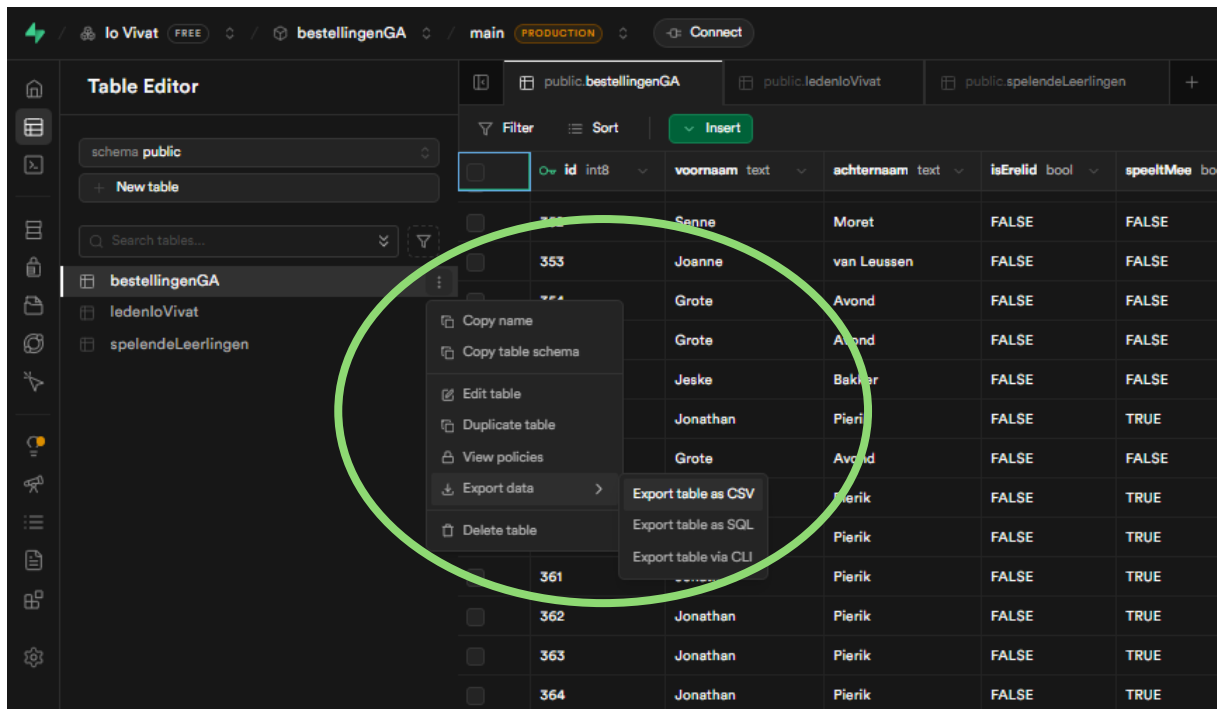
Id	voornaam	achternaam	leErvind	speeltMoe
314	Jonathan	Pierik	FALSE	TRUE
315	Jonathan	Pierik	FALSE	TRUE
319	Jonathan	Pierik	FALSE	TRUE
320	Jonathan	Pierik	FALSE	TRUE
321	Jonathan	Pierik	FALSE	TRUE
322	Jonathan	Pierik	FALSE	TRUE
323	Jonathan	Pierik	FALSE	TRUE
324	Jonathan	Pierik	FALSE	TRUE
325	Seth	Moerkerken	FALSE	TRUE
326	Jonathan	Pierik	FALSE	TRUE
327	Simon	Moerkerken	FALSE	TRUE
328	Arne	van Zijl	FALSE	FALSE
329	Ellanne	Hennip	FALSE	FALSE

Figuur 61: Supabase tabel exporteren; Stap 4 tabel bestellingenGA



## Optie één: De gehele tabel met alle gegevens erin exporteren

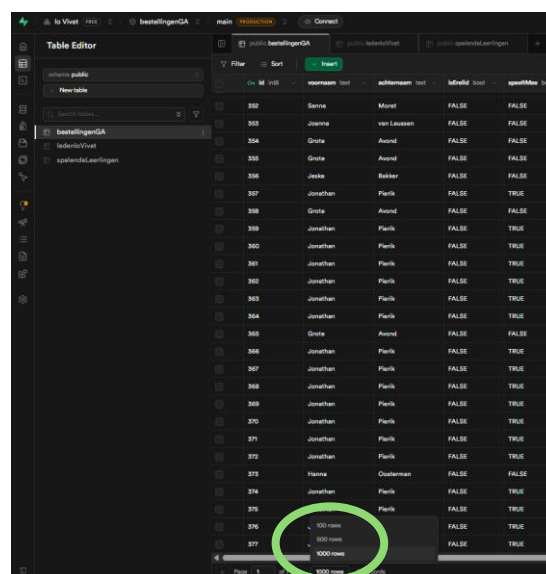
Stap 5: Klik op de drie puntjes links van de tabel: *bestellingenGA*. Klik vervolgens op de knop: *Export data*, en daarna op: *Export table as CSV*. Nu worden alle bestellingen geëxporteerd als een CSV-bestand. Dit bestand staat nu in de Downloads map van je computer, en dit bestand kan je in de Zaalindeler importeren om een zaalindeling te laten maken.



Figuur 62: Supabase tabel exporteren; Stap 5 Export table as CSV

## Optie twee: Zelf een aantal bestellingen selecteren

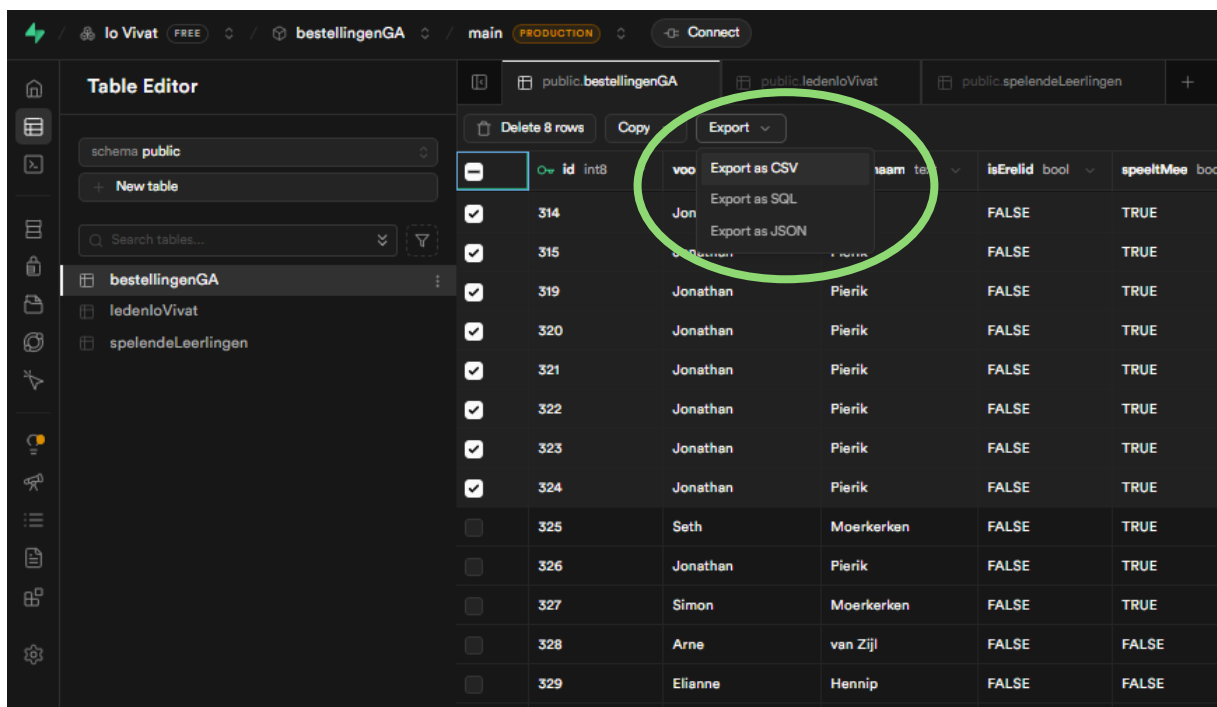
Stap 6: Klik helemaal onderaan, links van het midden op de knop: *100 rows*, en kies klik vervolgens op de optie: *1000 rows*. Zie onderstaande afbeelding (afbeelding x). Nu ziet u alle geplaatste bestellingen.



Figuur 63: Supabase tabel exporteren; Stap 6 rows aanpassen

Stap 7: Selecteer alle bestellingen die je wilt exporteren, door links van de kolom: *id* (met het groene sleuteltje) de vakjes aan te klikken. (afbeelding x)

Stap 8: Exporteer de geselecteerde bestellingen door boven de kolom: *voornaam* op de knop: *Export* te drukken, en klik daarna op: *Export as CSV*. Nu worden de geselecteerde bestellingen geëxporteerd als een CSV-bestand. Dit bestand staat nu in de Downloads map van je computer, en dit bestand kan je in de Zaalindeler importeren om een zaalindeling te laten maken.



Figuur 64: Supabase tabel exporteren; Stap 8 Export as CSV

## Dagen op de website aanpassen

Om de mogelijke voorkeursdagen op de website aan te passen, moeten er in drie bestanden gegevens worden aangepast. Dit gaat worden uitgelegd aan de hand van voorbeeld. Dit voorbeeld houdt in dat dat we de dagen van woensdag, donder en vrijdag willen vervangen door de dagen dinsdag, woensdag en donderdag.

Stap 1: Zoek de website bestanden op, zoals staat beschreven in het kopje: [Toegang tot, en updaten van het JavaScript-bestand, PHP-bestand en CSS-bestand \(website bestanden\)](#).

Stap 2: In het JavaScript-bestand moet rond regels 45 t/m 52 worden gezocht naar code die eruit ziet als in onderstaande afbeelding (afbeelding x).

```
45 // ===== Dropdown 2 vullen op basis van keuze 1 =====
46 function initDagkeuzeListener() { // Dagen en dagkeuzes definiëren in dropdown
47     const dagen = ["woensdag", "donderdag", "vrijdag"];
48     const dagTekst = {
49         woensdag: "Woensdag 8 april",
50         donderdag: "Donderdag 9 april",
51         vrijdag: "Vrijdag 10 april",
52     };
53 }
```

Figuur 65: Data aanpassen; Voorbeeld stap 2 en 3 niet aangepast

Hierin veranderen we `const dagen = []`; van “woensdag”, “donderdag”, “vrijdag” naar “dinsdag”, “woensdag”, “donderdag”. Let op dat de dagen hier in kleine letters zijn, en tussen dubbele aanhalingstekens “” staan, gescheiden door een komma en spatie.

Stap 3: Ook moeten we `const dagTekst` veranderen. Hierbij moet er worden opgelet dat de dagen zonder dubbele aanhalingstekens in kleine letters staan, en aan het eind van het tweede dubbele aanhalingsteken een komma staat. (afbeelding x)

```
45 // ===== Dropdown 2 vullen op basis van keuze 1 =====
46 function initDagkeuzeListener() { // Dagen en dagkeuzes definiëren in dropdown
47     const dagen = ["dinsdag", "woensdag", "donderdag"];
48     const dagTekst = {
49         dinsdag: "Dinsdag 7 april",
50         woensdag: "Woensdag 8 april",
51         donderdag: "Donderdag 9 april",
52     };
53 }
```

Figuur 66: Data aanpassen; Voorbeeld stap 2 en 3 wel aangepast

Stap 4: In het PHP-bestand moeten de dagen worden aangepast. Rond regels 116 t/m 117 staat onderstaande code. (afbeelding x)

```
116 // validatie
117 $days = ['woensdag','donderdag','vrijdag'];
```

Figuur 67: Data aanpassen; Voorbeeld stap 4 niet aangepast

In onderstaande code zijn de dagen aangepast naar 'dinsdag','woensdag','donderdag'. Let hier goed op dat de dagen tussen enkele aanhalingstekens staan, gescheiden door een komma zonder spatie. (afbeelding x)

```
116 // validatie
117 $days = ['dinsdag','woensdag','donderdag'];
```

Figuur 68: Data aanpassen; Voorbeeld stap 4 wel aangepast

Stap 5: Verder moet in het PHP-bestand ook de foutmelding wordt aangepast. Nu staat er nog: *kies uit woensdag, donderdag of vrijdag*. (afbeelding x)

```
129 if ($dag1 === '' || !in_array($dag1, $days, true)) $errors[] = 'Een eerste voorkeursdag kiezen is verplicht, kies uit woensdag, donderdag of vrijdag.';
130 if ($dag2 === '' || !in_array($dag2, $days, true)) $errors[] = 'Een tweede voorkeursdag kiezen is verplicht, kies uit woensdag, donderdag of vrijdag.';
```

Figuur 69: Data aanpassen; Voorbeeld stap 5 niet aangepast

Maar dit willen we veranderen in: *kies uit dinsdag, woensdag of donderdag*. (afbeelding x)

```
129 if ($dag1 === '' || !in_array($dag1, $days, true)) $errors[] = 'Een eerste voorkeursdag kiezen is verplicht, kies uit dinsdag, woensdag of donderdag.';
130 if ($dag2 === '' || !in_array($dag2, $days, true)) $errors[] = 'Een tweede voorkeursdag kiezen is verplicht, kies uit dinsdag, woensdag of donderdag.';
```

Figuur 70: Data aanpassen; Voorbeeld stap 5 wel aangepast

Stap 6: In het PHP-bestand moeten we rond regels 221 t/m 226 ook nog HTML code aanpassen. Nu staan de dagen woensdag, donder en vrijdag er nog, dit willen we veranderen naar dinsdag, woensdag en donderdag. (afbeelding x)

```
221 <select id="dagkeuze1" required>
222 <option value="" disabled selected>-- Kies een dag --</option>
223 <option value="woensdag">woensdag 8 april</option>
224 <option value="donderdag">donderdag 9 april</option>
225 <option value="vrijdag">vrijdag 10 april</option>
226 </select>
```

Figuur 71: Data aanpassen; Voorbeeld stap 6 niet aangepast

Tussen > en < kan de tekst worden geschreven die daadwerkelijk zichtbaar is op de website. Let er hierbij op dat er na *value=* dubbele aanhalingstekens komen, en de dag met kleine letters wordt geschreven. (afbeelding x)

```
221 <select id="dagkeuze1" required>
222 <option value="" disabled selected>-- Kies een dag --</option>
223 <option value="dinsdag">Dinsdag 7 april</option>
224 <option value="woensdag">woensdag 8 april</option>
225 <option value="donderdag">donderdag 9 april</option>
```

Figuur 72: Data aanpassen; Voorbeeld stap 6 wel aangepast

Stap 7: Helemaal onder in het PHP-bestand staat er rond regel 260 een versienummer van het JavaScript bestand, deze moet je altijd één hoger maken. In dit geval staat er 1.0.1. (afbeelding x)

```
256 wp_enqueue_script(  
257     handle: 'bestellingGA-js',  
258     src: plugin_dir_url(file: __FILE__) . 'bestellingGA.js',  
259     deps: [],  
260     ver: '1.0.1',  
261     args: true  
262 );
```

Figuur 73: Data aanpassen; Voorbeeld stap 7 niet aangepast

Het versienummer één hoger maken wordt dan 1.0.2. (afbeelding x)

```
256 wp_enqueue_script(  
257     handle: 'bestellingGA-js',  
258     src: plugin_dir_url(file: __FILE__) . 'bestellingGA.js',  
259     deps: [],  
260     ver: '1.0.2',  
261     args: true  
262 );
```

Figuur 74: Data aanpassen; Voorbeeld stap 7 wel aangepast

Stap 8: Zoek de database Edge Function code op, zoals staat beschreven in het kopje: *Toegang tot, en updaten van de Supabase Edge Function.*

Stap 9: Hierin staat rond regels 75 t/m 79 de onderstaande code. Ook nu weer moeten we de dagen veranderen van woensdag, donderdag, vrijdag naar dinsdag, woensdag, donderdag. (afbeelding x)

```
75 const allowedDays = [  
76     "woensdag",  
77     "donderdag",  
78     "vrijdag",  
79 ];
```

Figuur 75: Data aanpassen;  
Voorbeeld stap 9 niet aangepast

Let erop dat de dagen met kleine letters worden geschreven, en tussen twee dubbele aanhalingstekens staan. Ook moet aan het eind een komma staan. (afbeelding x)

```
75 const allowedDays = [  
76     "dinsdag",  
77     "woensdag",  
78     "donderdag",  
79 ];
```

Figuur 76: Data aanpassen;  
Voorbeeld stap 9 wel aangepast

Stap 10: Net als in het PHP-bestand moet ook hier in het TypeScript-bestand een foutmelding worden aangepast. (afbeelding x)

```
97 if (voorkeurDag1 === "" || !allowedDays.includes(voorkeurDag1)) errors.push("Een eerste voorkeursdag kiezen is verplicht, kies uit woensdag, donderdag of vrijdag.");
98 if (voorkeurDag2 === "" || !allowedDays.includes(voorkeurDag2)) errors.push("Een tweede voorkeursdag kiezen is verplicht, kies uit woensdag, donderdag of vrijdag.");
```

Figuur 77: Data aanpassen; Voorbeeld stap 10 niet aangepast

Weer van woensdag, donderdag of vrijdag naar dinsdag, woensdag of donderdag. (afbeelding x)

```
97 if (voorkeurDag1 === "" || !allowedDays.includes(voorkeurDag1)) errors.push("Een eerste voorkeursdag kiezen is verplicht, kies uit dinsdag, woensdag of donderdag.");
98 if (voorkeurDag2 === "" || !allowedDays.includes(voorkeurDag2)) errors.push("Een tweede voorkeursdag kiezen is verplicht, kies uit dinsdag, woensdag of donderdag.");
```

Figuur 78: Data aanpassen; Voorbeeld stap 10 wel aangepast

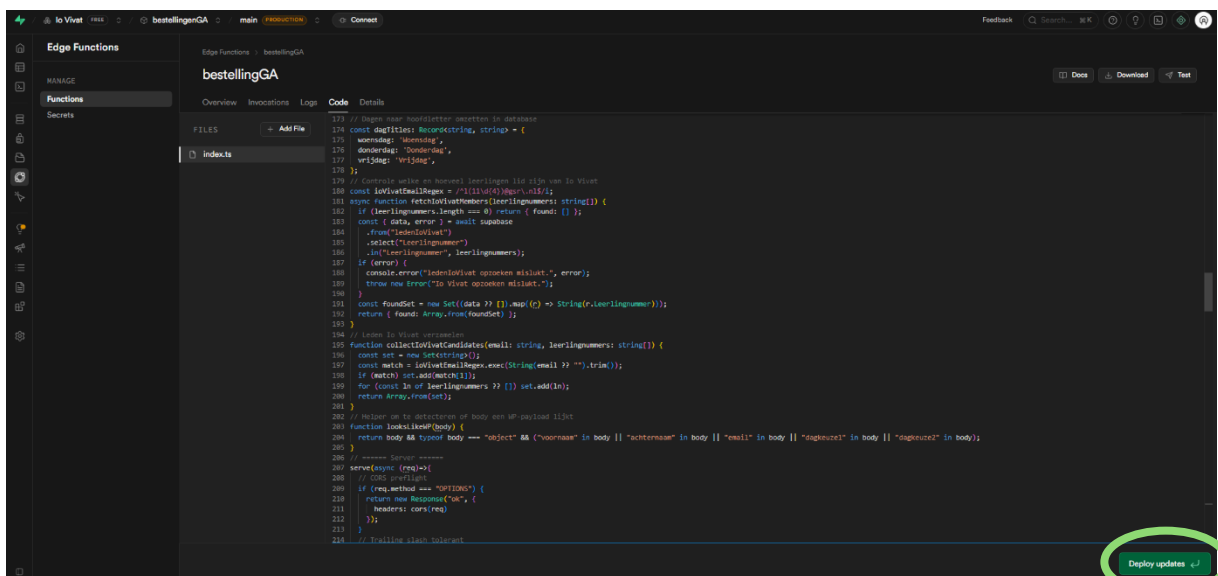
Stap 11: Nu de laatste aanpassing, nog steeds in het TypeScript-bestand. Rond regels 174 t/m 178 staat woensdag, donderdag, vrijdag. Dit gaan we aanpassen naar dinsdag, woensdag, donderdag. (afbeelding x)

```
173 // Dagen naar hoofdletter omzetten in database
174 const dagTitles: Record<string, string> = {
175   woensdag: 'Woensdag',
176   donderdag: 'Donderdag',
177   vrijdag: 'Vrijdag',
178 };
```

Figuur 79: Data aanpassen; Voorbeeld stap 11 niet aangepast

Let er weer op dat de eerste dag op een regel in kleine letters staat zonder aanhalingstekens, en dat de tweede dag op een regel begint met een hoofdletter met tussen dubbele aanhalingstekens staat. Aan het eind van een regel komt een komma.

Stap 12: De laatste stap is om de Supabase Edge Function op te slaan, dit doe je door helemaal rechts onderin op *Deploy updates* te drukken. Ververs de webpagina, en de dagen zijn aangepast. (afbeelding x)

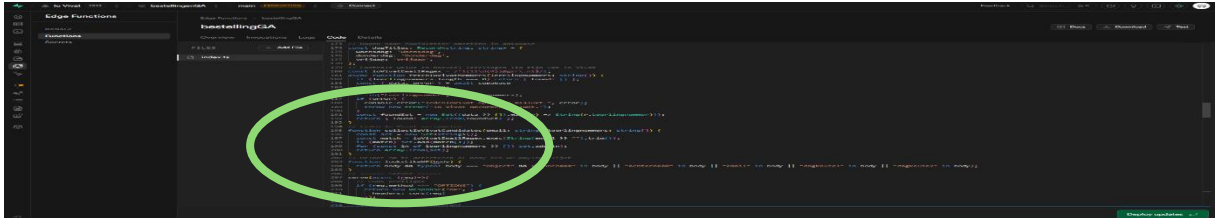


Figuur 80: Data aanpassen; Voorbeeld stap 12 Supabase scherm Deploy updates

## Leden lo Vivat aanpassen

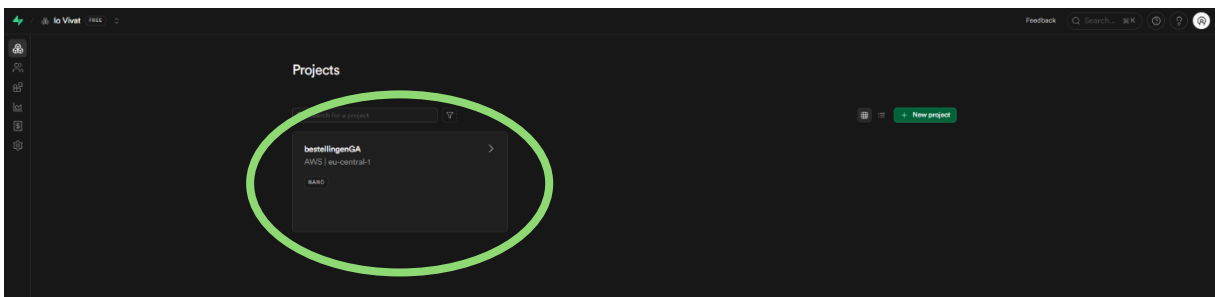
Elk jaar verandert de lijst met leden van lo Vivat, daarom moet deze lijst ook elk jaar in de Supabase database worden vervangen. Dit doen we als volgt.

Stap 1: Log in op de Supabase website, en klik op de organisatie: lo Vivat. (afbeelding x)



Figuur 81: Leden lo Vivat aanpassen; Stap 1 Organisation

Stap 2: Klik daarna op het project: *bestellingenGA*. (afbeelding x)



Figuur 82: Leden lo Vivat aanpassen; Stap 2 Projects

Stap 3 Klik daarna rechts van het midden op *Tables*. (afbeelding x)

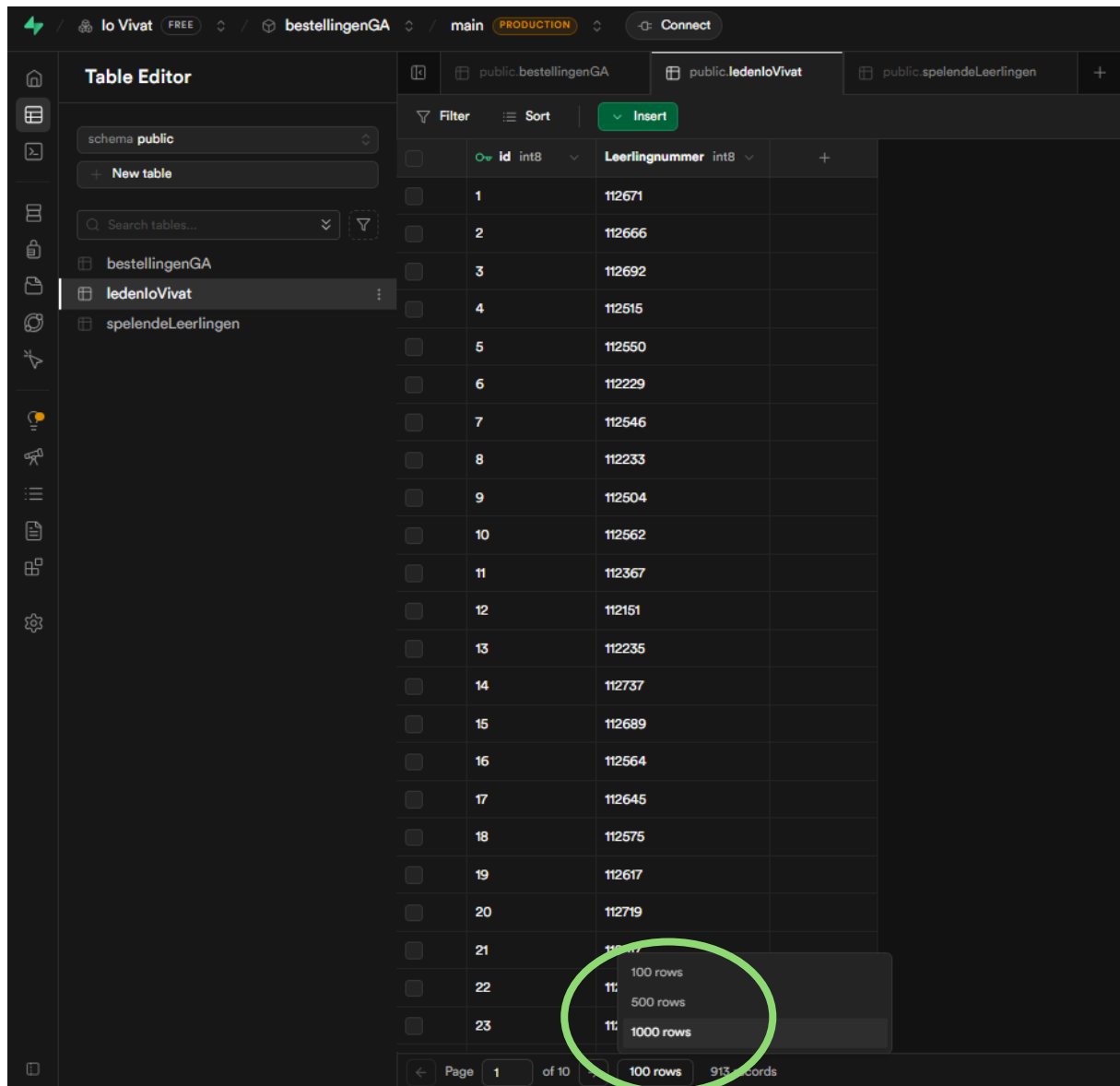


Figuur 83: Leden lo Vivat aanpassen; Stap 3 Tables

Stap 4: Klik daarna in het linker navigatie-menu op de middelste tabel, genaamd *ledenloVivat*. (afbeelding x)

id	Leeringnummer
1	112671
2	112666
3	112692
4	112515
5	112550
6	112229
7	112546
8	112233
9	112504
10	112562
11	112367
12	112151
13	112235

Stap 5: Klik helemaal onderaan, links van het midden op de knop: *100 rows*, en kies klik vervolgens op de optie: *1000 rows*. Zie onderstaande afbeelding (afbeelding x).

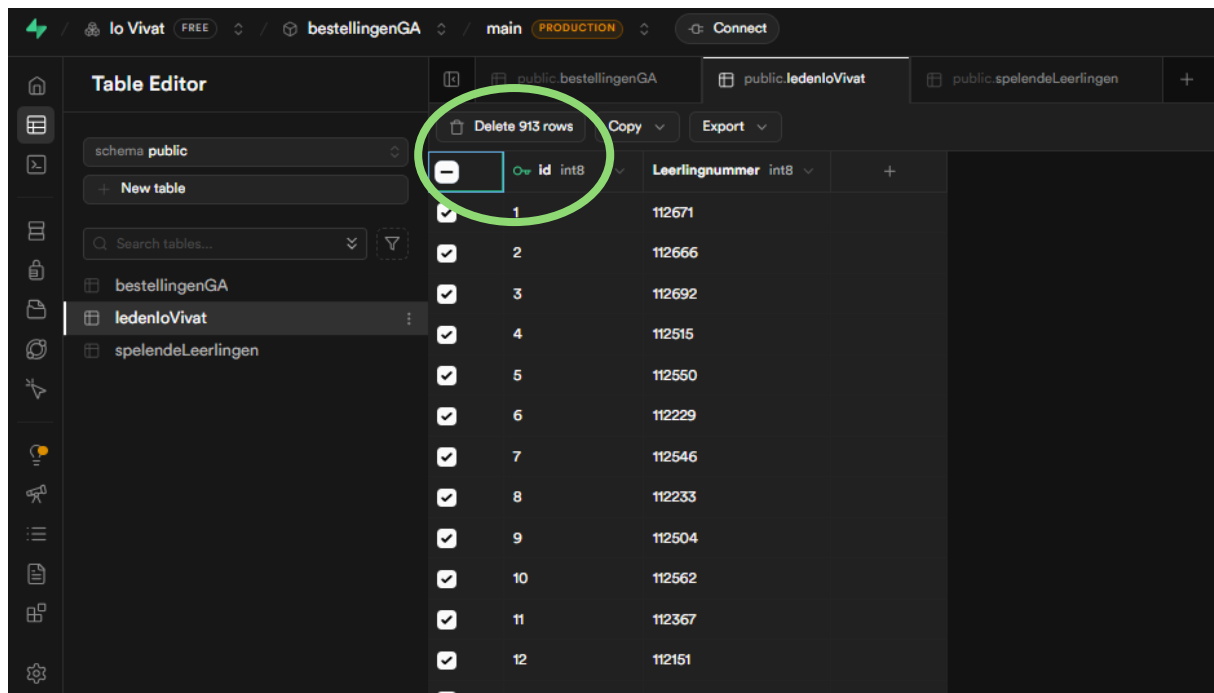


Figuur 85: Leden lo Vivat aanpassen; Stap 5 rows aanpassen

Stap 6: Selecteer alle leerlingnummers door op het vierkantje te klikken rechts van het groene sleuteltje. Deze staat links bovenaan de tabel. (afbeelding x)

Stap 7: Verwijder alle leerlingnummers door op de knop: *Delete x rows* te klikken. X betekent hier het aantal leerlingnummers dat al in de tabel staan. In het geval van de afbeelding bovenaan de volgende pagina staan er 913 leerlingnummers in de tabel, dus staat er op de knop: *Delete 913 rows*. (afbeelding x)

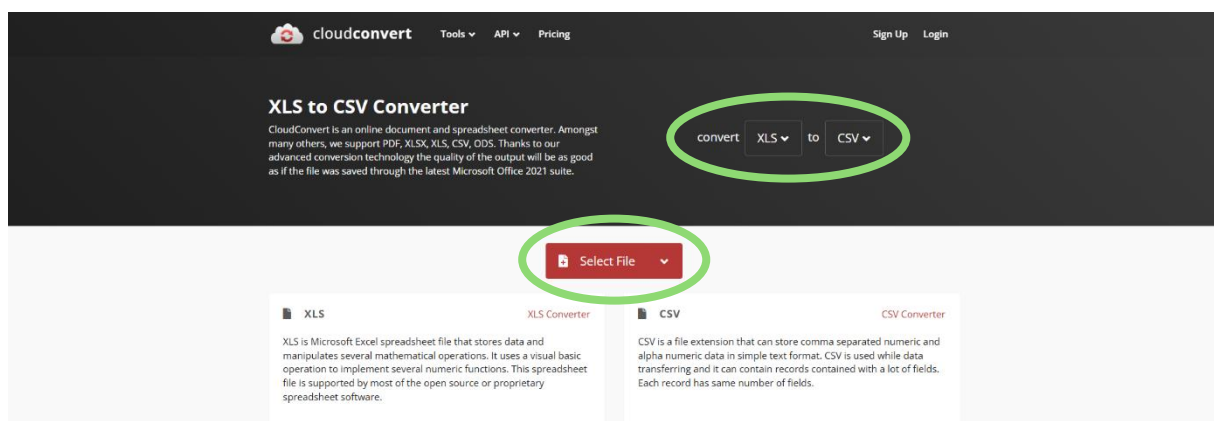




Figuur 86: Leden lo Vivat aanpassen; Stap 6 en 7 leden selecteren en verwijderen

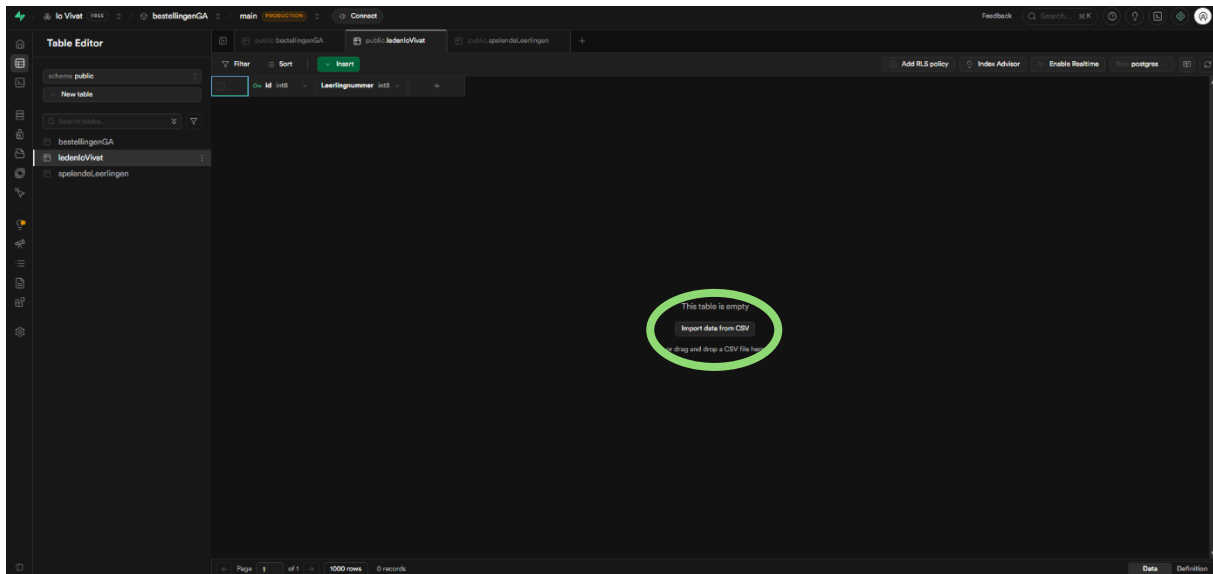
Stap 8: Ga naar de administratie van de GSR, en vraag het Excelbestand met leerlingnummers op waarvan de ouders de ouderbijdrage voor lo Vivat hebben betaald. Je hebt alleen de leerlingnummers nodig, geen andere gegevens zoals klas, voornaam, achternaam, vestiging, enzovoort.

Stap 9: Zet dit Excel bestand om in een CSV bestand, dit kan bijvoorbeeld door gebruik van een website zoals cloudconvert.com (<https://cloudconvert.com/xls-to-csv>). Controleer op deze website of je convert van XLS (Excel) naar CSV. Klik daarna op de grote rode knop in het midden: *Select File* om het Excel bestand van de administratie te importeren. (afbeelding x)



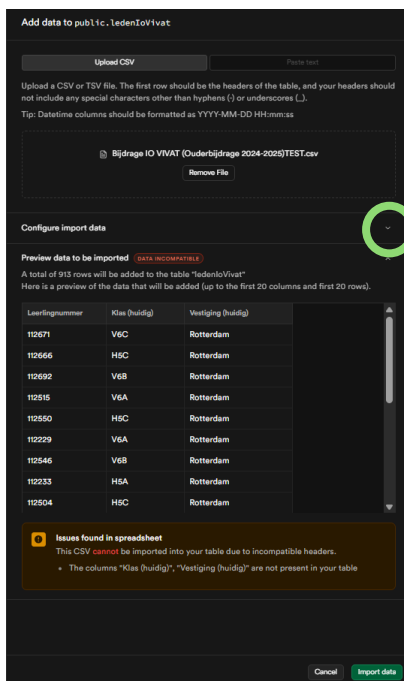
Figuur 87: Leden lo Vivat aanpassen; Stap 9 XLS naar CSV converter

Stap 10: De tabel is nu leeg, en in het midden staat een grote knop: *Insert data from CSV*. Klik hierop. (afbeelding x)

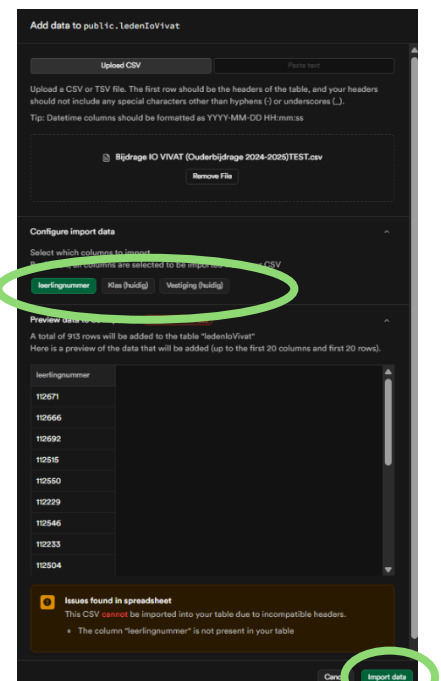


Figuur 88: Leden Io Vivat aanpassen; Stap 10 knop Insert data from CSV

Stap 11: Upload het CSV-bestand. Zeer waarschijnlijk geeft Supabase een error dat sommige kolommen in het CSV-bestand niet in de tabel staan. Dit is geen probleem en is gemakkelijk op te lossen. Klik hiervoor op de neerwaardse pijl helemaal rechts van de tekst *Configure import data*. (afbeelding x)



Figuur 90: Leden Io Vivat aanpassen; Stap 11 pijl Configure import data



Figuur 89: Leden Io Vivat aanpassen; Stap 12 Leerlingnummer en Import data

Stap 12: Klik alle overige kolommen uit tot ze grijs zijn, en alleen de kolom *Leerlingnummer* groen is. Klik vervolgens helemaal rechts onderin op *Import data*. (afbeelding x)

Stap 13: Als je alle kolommen hebt uitgeschakeld en alleen de kolom *Leerlingnummer* nog groen is, en je nog steeds een error krijgt, dan komt dit zeer waarschijnlijk doordat de naam van de kolom in CSV niet goed is. Open hiervoor het CSV-bestand in Excel, en pas de naam van de kolom goed aan. In onderstaand voorbeeld heet kolom A: *leerlingnummer*. Dit is met een kleine letter, en niet met de hoofdletter die Supabase verwacht: *Leerlingnummer*. (afbeelding x en y)

	A	B	C
1	leerlingnummer	Klas (huidig)	Vestiging (huidig)
2	112671	V6C	Rotterdam
3	112666	H5C	Rotterdam
4	112692	V6B	Rotterdam
5	112515	V6A	Rotterdam
6	112550	H5C	Rotterdam
7	112229	V6A	Rotterdam

Figuur 91: Leden lo Vivat aanpassen; Stap 13 CSV-bestand foute kolomnaam

	A	B	C
1	Leerlingnummer	Klas (huidig)	Vestiging (huidig)
2	112671	V6C	Rotterdam
3	112666	H5C	Rotterdam
4	112692	V6B	Rotterdam
5	112515	V6A	Rotterdam
6	112550	H5C	Rotterdam
7	112229	V6A	Rotterdam

Figuur 92: Leden lo Vivat aanpassen; Stap 13 CSV-bestand goede kolomnaam

Stap 14: Nu zou Supabase geen errors meer moeten geven. Klik helemaal rechts onderin op *Import data* om de leerlingnummers in de tabel te zetten. (afbeelding x)

Add data to public: ledenloVivat

Upload CSV | Paste text

Upload a CSV or TSV file. The first row should be the headers of the table, and your headers should not include any special characters other than hyphens (-) or underscores (\_).  
Tip: Datetime columns should be formatted as YYYY-MM-DD HH:mm:ss

Bijdrage IO VIVAT (Ouderbijdrage 2024-2025)TEST.csv | Remove file

Configure import data

Select which columns to import  
By default, all columns are selected to be imported from your CSV

Leerlingnummer | Klas (huidig) | Vestiging (huidig)

Preview data to be imported

A total of 913 rows will be added to the table "ledenloVivat"  
Here is a preview of the data that will be added (up to the first 20 columns and first 20 rows).

Leerlingnummer

112671  
112666  
112692  
112515  
112550  
112229  
112546  
112233  
112804

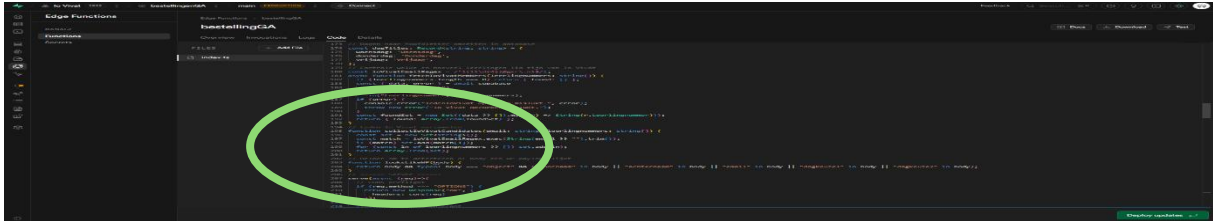
Cancel | Import data

Figuur 93: Leden lo Vivat aanpassen; Stap 14 Import data

## Meespelende leerlingen aanpassen

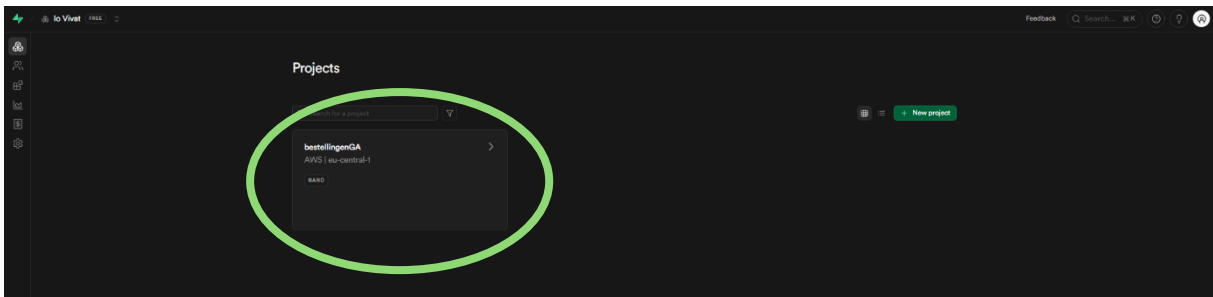
Afhankelijk van welke rang je hebt, krijg je een plek aangewezen door de Zaalindeler. Zo krijg een meespelend lid een betere plek als een normale leerling. Om deze reden is het belangrijk dat elk jaar de tabel met meespelende leden wordt vervangen.

Stap 1: Log in op de Supabase website, en klik op de organisatie: Io Vivat. (afbeelding x)



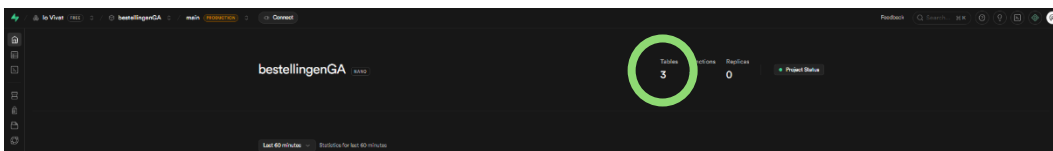
Figuur 94: Meespelende leden aanpassen; Stap 1 Organisation

Stap 2: Klik daarna op het project: *bestellingGA*. (afbeelding x)



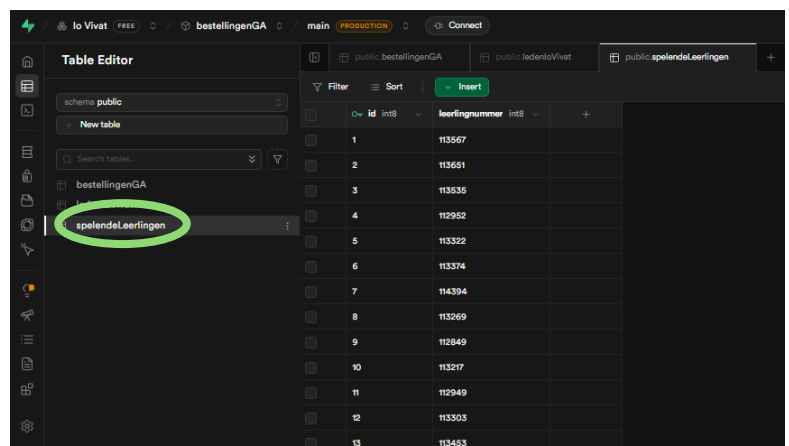
Figuur 95: Meespelende leden aanpassen; Stap 2 Projects

Stap 3 Klik daarna rechts van het midden op *Tables*. (afbeelding x)



Figuur 96: Meespelende leden aanpassen; Stap 3 Tables

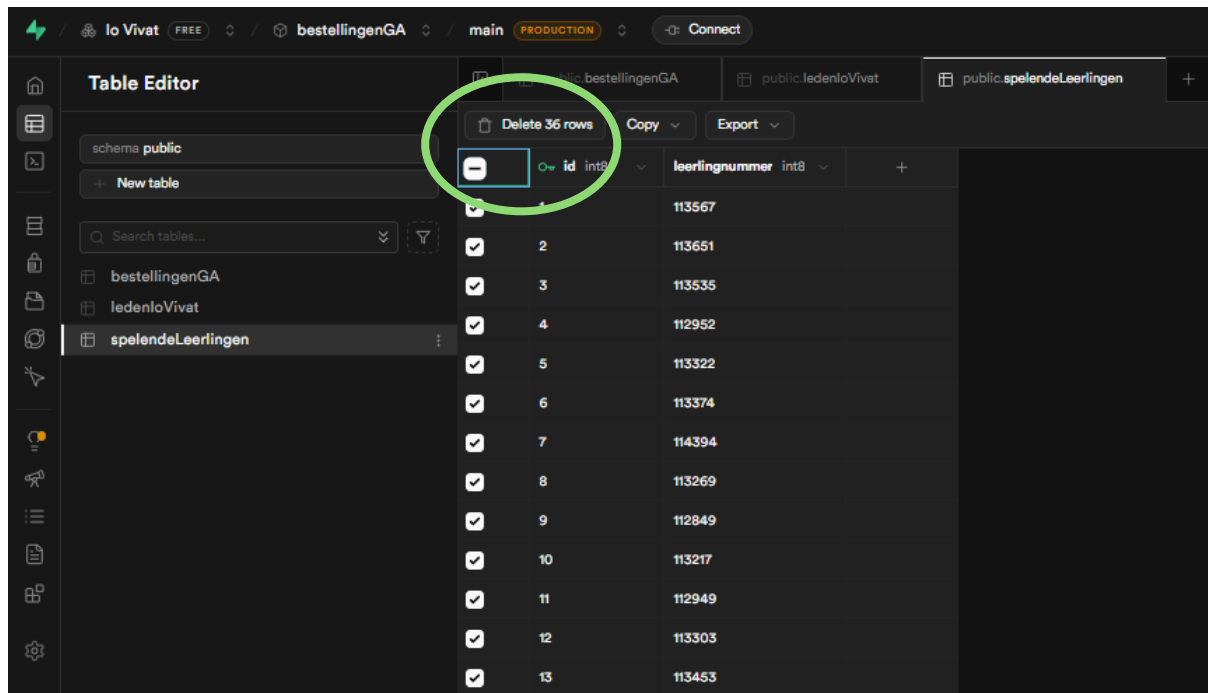
Stap 4: Klik daarna in het linker navigatie-menu op de middelste tabel, genaamd *spelendeLeerlingen*. (afbeelding x)



Figuur 97: Meespelende leden aanpassen; Stap 4 tabel spelendeLeerlingen

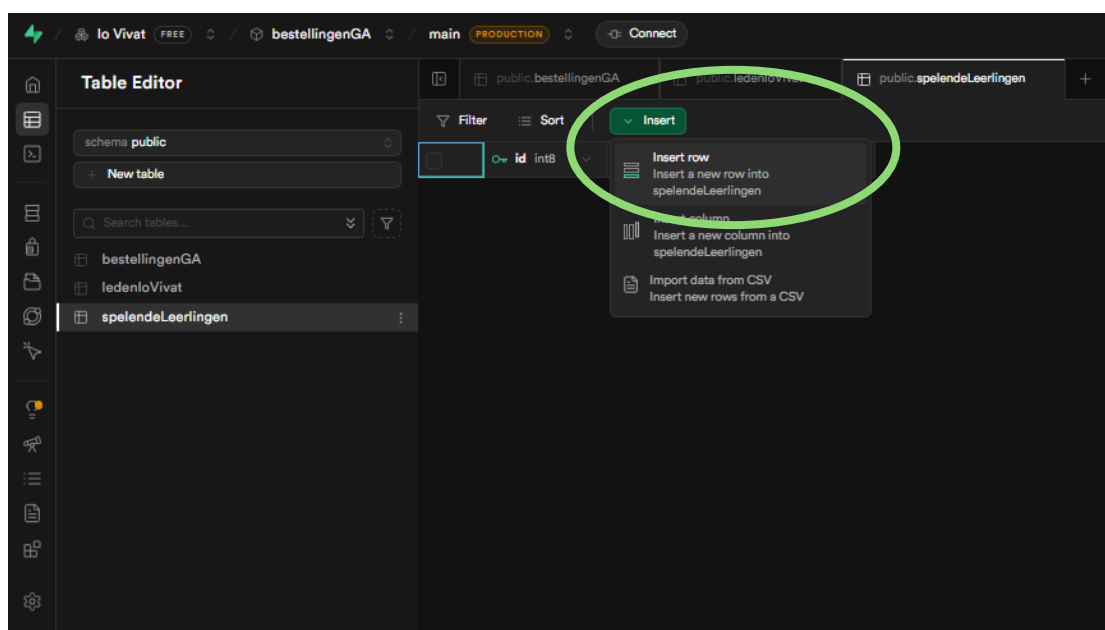
Stap 5: Selecteer alle leerlingnummers door op het vierkantje te klikken rechts van het groene sleuteltje. Deze staat links bovenaan de tabel. (afbeelding x)

Stap 6: Verwijder alle leerlingnummers door op de knop: *Delete x rows* te klikken. X betekent hier het aantal leerlingnummers dat al in de tabel staan. In het geval van de onderstaande afbeelding staan er 36 leerlingnummers in de tabel, dus staat er op de knop: *Delete 36 rows*. (afbeelding x)



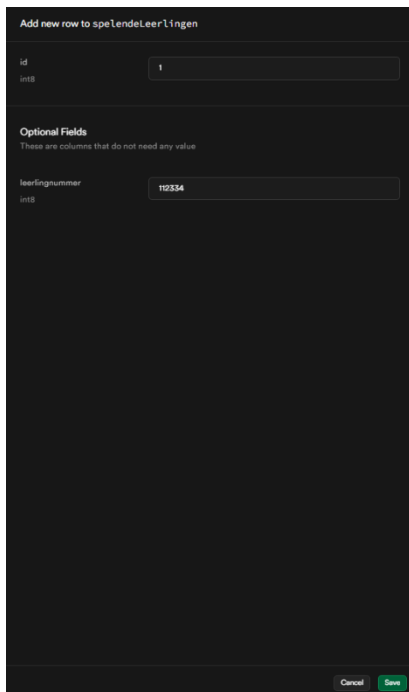
Figuur 98: Meespelende leden aanpassen; Stap 5 en 6 leden selecteren en verwijderen

Stap 7: De tabel is nu leeg, en links bovenaan staat een groene knop: *Insert*. Klik hierop, en klik vervolgens op de knop: *Insert row*. (afbeelding x)



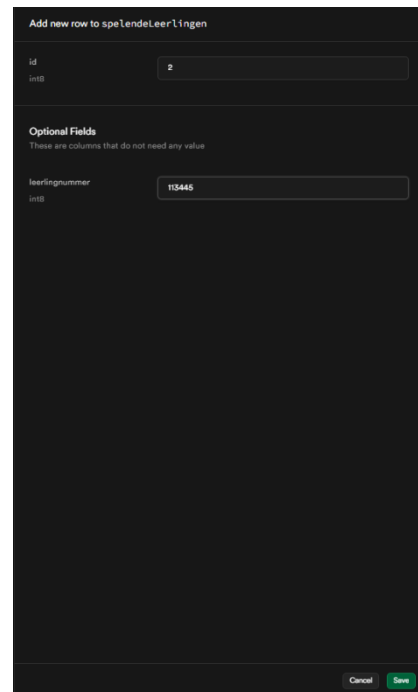
Figuur 99: Meespelende leden aanpassen; Stap 7 knop Insert

Stap 8: Elk leerlingnummer moet handmatig worden ingevuld. id is het unieke nummer van elk leerlingnummer, dus deze moet elke keer één hoger worden. Voor het eerste leerlingnummer is het id 1, bij het tweede leerlingnummer 2, enzovoort. Bij het vakje leerlingnummer vul je het leerlingnummer van de meespelende leerling in. Let op dat dit alleen de zes cijfers van het leerlingnummer zijn, en niet letters of andere tekens. Zie afbeeldingen x en y als voorbeeld. (afbeelding x en y)



The screenshot shows a dark-themed form titled "Add new row to spelendeLeerlingen". It has two input fields. The first field is labeled "id" and "int8" and contains the value "1". Below it is a section titled "Optional Fields" with the subtitle "These are columns that do not need any value". The second field is labeled "leerlingnummer" and "int8" and contains the value "112354". At the bottom right, there are "Cancel" and "Save" buttons.

*Figuur 101: Meespelende leden aanpassen; Stap 8 eerste invoer leerlingnummer*



The screenshot shows the same form as Figure 101, but with the second input field now containing the value "115445". The "id" field still contains "2". The "Optional Fields" section and the "Cancel" and "Save" buttons are also present.

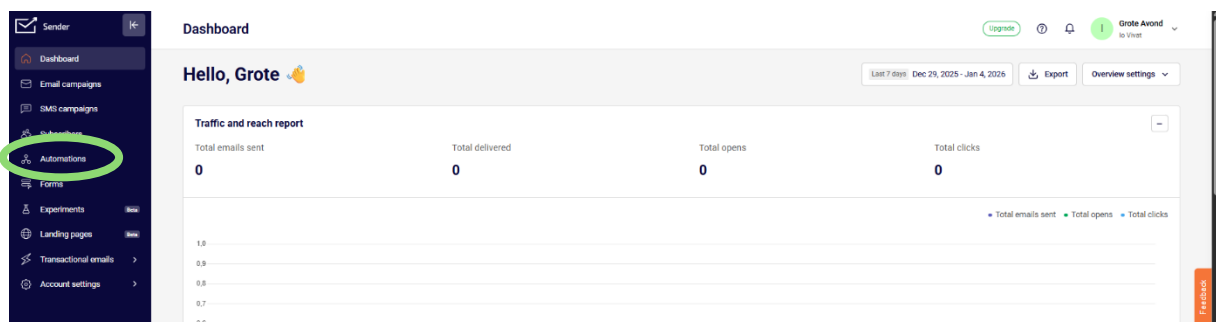
*Figuur 100: Meespelende leden aanpassen; Stap 8 tweede invoer leerlingnummer*

## Bevestigingsmail aanpassen

Het is leuk om elk jaar een nieuwe bevestigingsmail te maken, waarvan het thema van de mail aansluit bij het thema van de Grote Avond. Om de mail aan te passen zijn er x stappen die moeten worden doorlopen. De bevestigingsmail wordt gestuurd door Sender

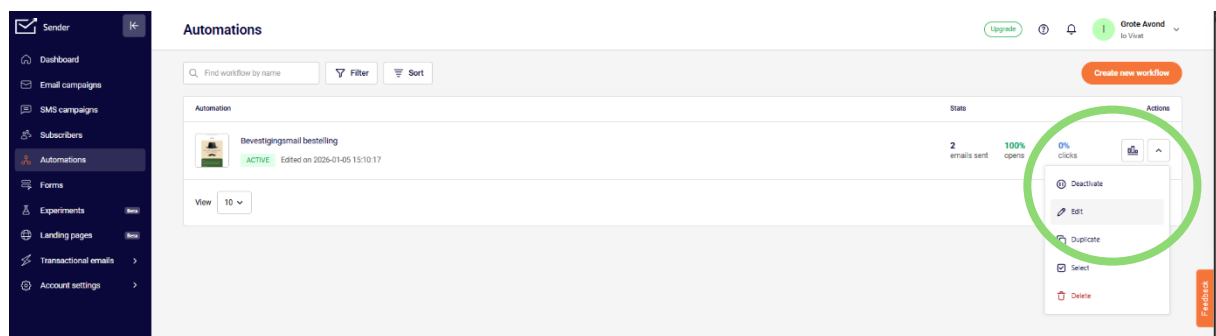
Stap 1: Om toegang te krijgen tot de bevestigingsmail, moet je inloggen op de Sender website, waarvan helemaal onderaan dit document de inloggegevens en link naar de website staan beschreven.

Stap 2: Klik links in het menu op de knop: *Automations*. Dit heeft het icoontje van drie vierkantjes in een driehoekvormige verhouding. (afbeelding x)



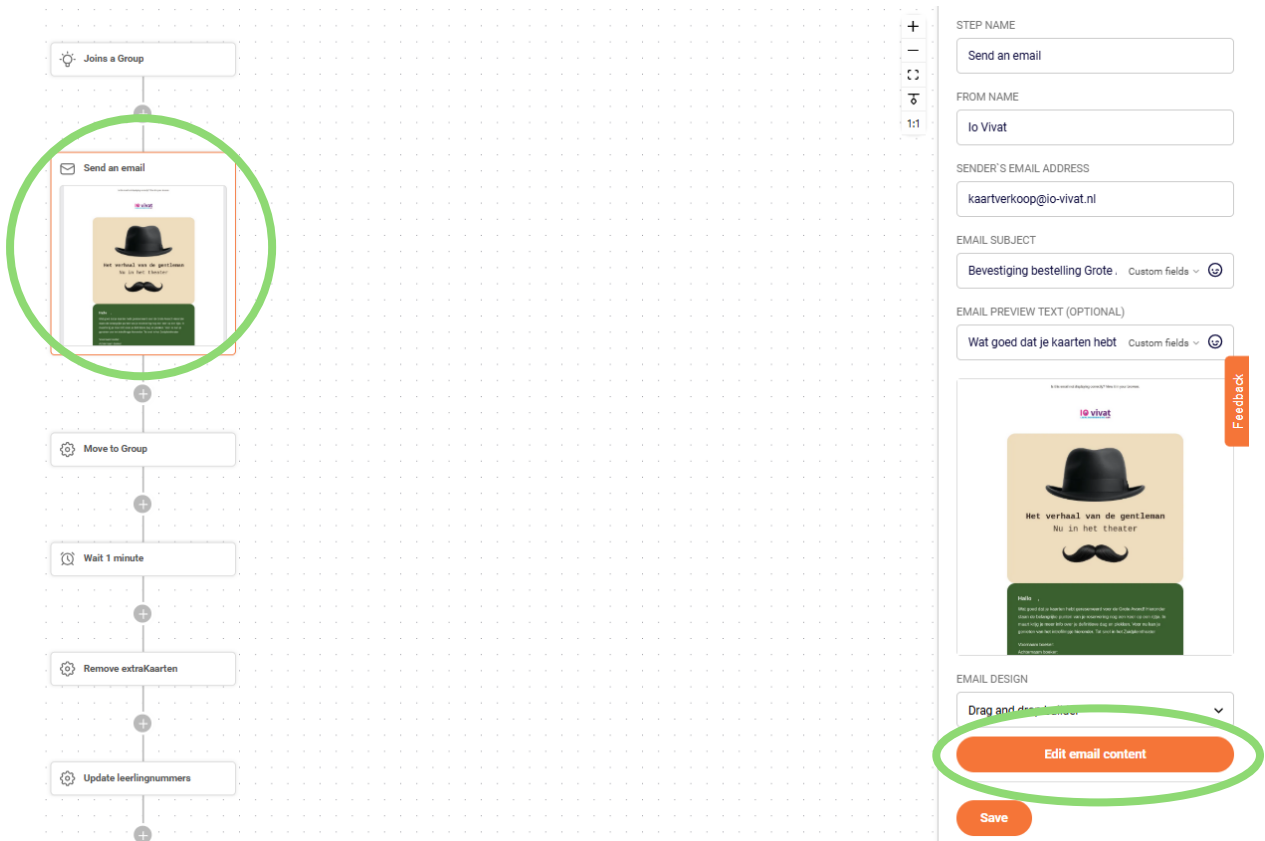
Figuur 102: Bevestigingsmail aanpassen; Stap 2 Automations

Stap 3: Je ziet de Automation: *Bevestigingsmail bestelling*. Helemaal rechts hiervan is een pijltje dat naar beneden wijst, klik hierop. Er opent zich een menu, klik op de knop: *Edit*. (afbeelding x)



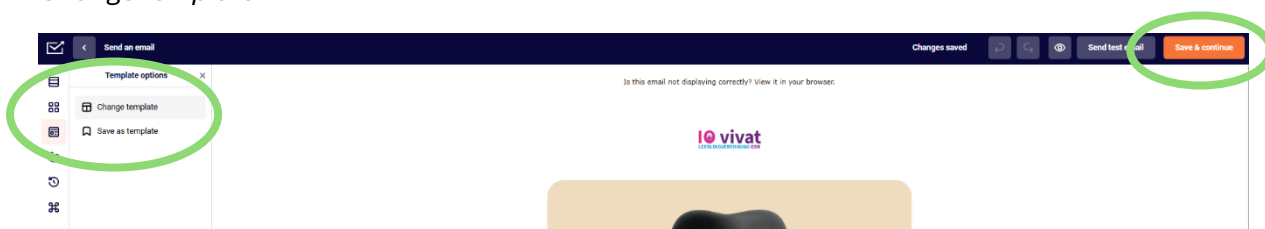
Figuur 103: Bevestigingsmail aanpassen; Stap 3 Edit

Stap 4: Klik op het tweede vakje van boven: *Send an email*. Klik vervolgens op de grote oranje knop rechts onderin: *Edit email content*.



Figuur 104: Bevestigingsmail aanpassen; Stap 4 Edit email content

Stap 5: Je kan nu als een soort Canva met drag and drop een email ontwerpen. Ook kan je een ander template kiezen als je iets minder creatief wilt zijn. Doe dit laatste door helemaal links op het derde icoontje van boven te klikken: *Templates*. Klik vervolgens op de bovenste knop: *Change template*.



Figuur 105: Bevestigingsmail aanpassen; Stap 5 en 6 Templates en Save & continue

Stap 6: Sla na het bewerken van de email altijd de email weer goed op. Doe dit door op de oranje knop rechts bovenin het scherm te klikken: *Save & continue*. (afbeelding x)



Stap 7: Activeer altijd de Automation weer nah het bewerken ervan. Doe dit door op de oranje knop rechts bovenin te klikken: Activate. Let op, zonder het opnieuw activeren van de Automation werkt het gehele automatische e-mailsysteem niet meer. (afbeelding x)



Figuur 106: Bevestigingsmail aanpassen; Stap 7 Activate

## Ereleden aanpassen

In de Edge Function van Supabase is rond regels 128 t/m 135 de onderstaande code te zien (afbeelding x). Om een erelid toe te voegen is het persoonlijke e-mailadres van het erelid nodig.

```
128 // Controle of hoofdboeker erelid is
129 const ereleden = new Set(
130   [
131     "bri@gsr.nl",
132     "gdendulk@ziggo.nl",
133     "h.de.lange@kpnmail.nl",
134   ].map((e) => e.toLowerCase().replace(/\s+/g, ""))
135 );
```

Figuur 107: Ereleden aanpassen; Stap 1 voorbeeldcode

Stap 1: Voeg dit e-mailadres toe aan de huidige lijst e-mailadressen van ereleden. Doe dit door het e-mailadres tussen dubbele aanhalingstekens te zetten, en na het tweede dubbele aanhalingsteken een komma te plaatsen.

Stap 2: Sla de Edge Function op door op *Deploy updates* te klikken. Zie hiervoor het stappenplan onder het kopje: [Toegang tot, en updaten van de Supabase Edge Function](#).

## Tijdsduur meldingen aanpassen

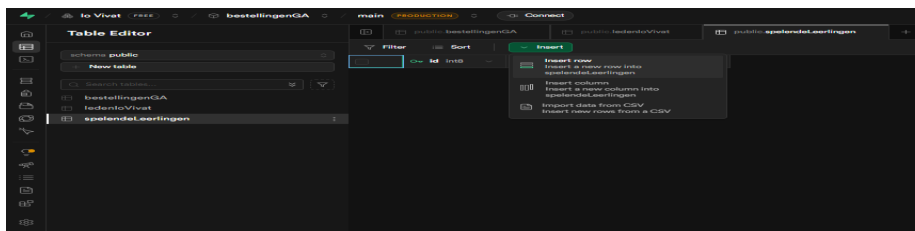
In het JavaScript-bestand kan ook de tijdsduur van de meldingen rechtsonder op de bestelpagina worden aangepast. Rond regels 154 t/m 158 is onderstaande code (afbeelding x) te zien. Hierin is te zien dat de succesmelding 20000 milliseconden blijft staan (20 seconden), een foutmelding door een error 10000 milliseconden blijft staan (10 seconden), evenals een infomelding welke ook 10000 milliseconden (10 seconden) blijft staan.

Stap 1: De tijdsduur van de meldingen is eenvoudig aan te passen, door het huidige getal te vervangen door de nieuwe gewenste tijdsduur.

```
154 const DUUR_PER_TYPE = { // Tijdsduur melding in ms
155     succes: 20000,
156     fout: 10000,
157     info: 10000,
158 };
```

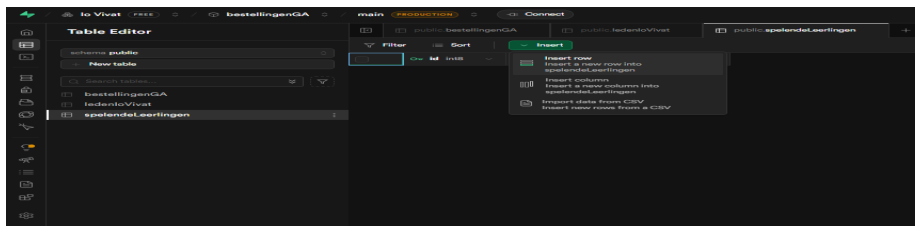
Figuur 108: Tijdsduur meldingen aanpassen; Stap 1 voorbeeldcode

Stap 2: Helemaal onder in het PHP-bestand staat er rond regel 260 een versienummer van het JavaScript bestand, deze moet je altijd één hoger maken. In dit geval staat er *1.0.1*. (afbeelding x)



Figuur 109: Data aanpassen; Voorbeeld stap 2 niet aangepast

Het versienummer één hoger maken wordt dan *1.0.2*. (afbeelding x)



Figuur 110: Data aanpassen; Voorbeeld stap 2 wel aangepast

## Succesmelding website aanpassen

Nadat een bestelling geslaagd is, krijgt de klant een succesmelding te zien. Deze melding staat ongeveer op regel 327 van het JavaScript-bestand.

Stap 1: Vervang de tekst van het eerste oranje deel (voor de komma) dat tussen de dubbele aanhalingstekens staat. Dit is dus niet het tweede oranje deel waar “succes” staat. Let erop dat

```

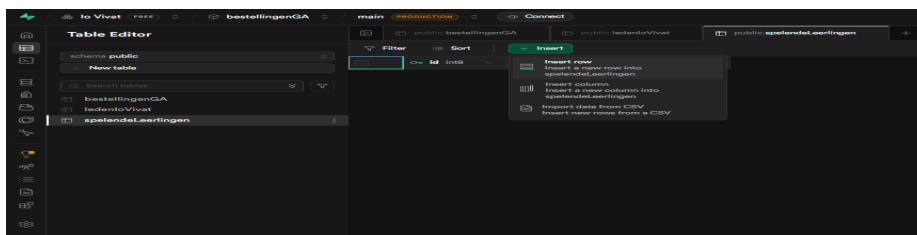
325     if (res.ok) {
326         verstuurenMelding.remove()
327         toonMelding("Bestelling gelukt! 🎉 U ontvangt een bevestigingsmail, controleer ook uw spam. Bij deze melding is de bestelling altijd gelukt, ook als u niet direct een bevestigingsmail ontvangt.", "succes")
328         form.reset();
329         resetDagkeuze2();
330         updateTotaalKaarten();
331         document.getElementById("leerlingnummersContainer").innerHTML = "";
332         mapping["leerlingnummers[]"] = [];
333         initMedeleerlingenInputs(mapping);

```

*Figuur 111: Succesmelding website aanpassen; Stap 1 voorbeeldcode*

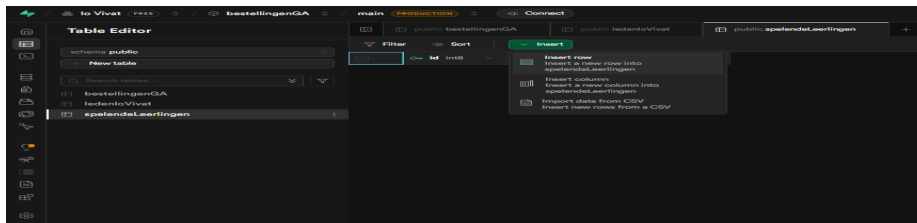
de dubbele aanhalingstekens aan het begin en einde van de tekst blijven staan.

Stap 2: Helemaal onder in het PHP-bestand staat er rond regel 260 een versienummer van het JavaScript bestand, deze moet je altijd één hoger maken. In dit geval staat er *1.0.1*. (afbeelding x)



*Figuur 112: Succesmelding website aanpassen; Stap 2 niet aangepast*

Het versienummer één hoger maken wordt dan 1.0.2. (afbeelding x)



*Figuur 113: Succesmelding website aanpassen; Stap 2 wel aangepast*

## Kleuren meldingen website aanpassen

Om de achtergrondkleuren van de meldingen op de website aan te passen, moet het CSS-bestand worden aangepast. Tussen ongeveer regels 53 t/m 63 staat de code die hiernaast is afgebeeld (afbeelding x). De gek uitzijnde combinatie van cijfers en letters is een hexadecimale kleurcode.

Stap 1: Om een andere kleur te gebruiken voor de meldingen, moet je een kleur kiezen en hier de hexadecimale kleurcode van opzoeken op internet. Deze kleurcode, beginnend met een hashtag, plaats je op de plek van de oude kleurcode.

```
53  .melding-succes {
54    background: #2ecc71;
55  }
56
57  .melding-fout {
58    background: #e74c3c;
59  }
60
61  .melding-info {
62    background: #34495e;
63  }
```

Figuur 114: Kleuren meldingen website aanpassen; Stap 1 voorbeeldcode

Stap 2: Helemaal onder in het PHP-bestand staat er rond regel 260 een versienummer van het JavaScript bestand, deze moet je altijd één hoger maken. In dit geval staat er 1.0.1. (afbeelding x)

```
266  // css inladen
267  wp_enqueue_style(
268    handle: 'bestellingGA-css',
269    src: plugin_dir_url(file: __FILE__) . 'bestellingGA.css',
270    deps: [],
271    ver: '1.0.1'
272  );
273  }
274  }
275  };
```

Figuur 115: Kleuren meldingen website aanpassen; Stap 2 niet aangepast

Het versienummer één hoger maken wordt dan 1.0.2. (afbeelding x)

```
266  // css inladen
267  wp_enqueue_style(
268    handle: 'bestellingGA-css',
269    src: plugin_dir_url(file: __FILE__) . 'bestellingGA.css',
270    deps: [],
271    ver: '1.0.2'
272  );
273  }
274  }
275  };
```

Figuur 116: Kleuren meldingen website aanpassen; Stap 2 wel aangepast

# Accounts en belangrijke gegevens

Github, verzamelplaats waar alle code staat

- Website Zaalindeler: <https://github.com/groteAvond/zaalindeler>
- Website bestelpagina: <https://github.com/groteAvond/bestelpagina>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

GSR Outlook account van de Grote Avond

- Website: <https://outlook.office365.com/mail/>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

Redis database voor de Zaalindeler

- Website: <https://redis.io/>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

Supabase database voor de website en Edge Function

- Website: <https://supabase.com/dashboard/organizations>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

Automatische emailsysteem

- Website: <https://app.sender.net/>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

Cloudflare, voor website records en bot prevention

- Website: <https://dash.cloudflare.com/bafd019bd2e7f04c91d42f368fa28036>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

Wordpress website beheer van [bibliotheek.io-vivat.nl](https://bibliotheek.io-vivat.nl)

- Website: <https://bibliotheek.io-vivat.nl/wp-admin/>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

webOke client dashboard

- Website: <https://www.weboke.nl/clientarea.php>
- Gebruikersnaam: [iovivat@gsr.nl](mailto:iovivat@gsr.nl)
- Wachtwoord: @lovivat23!

WebOke admin dashboard, website bestanden beheer

- Website: <https://linux2023.webawere.nl:2222/evo/>
- Gebruikersnaam: dulk21
- Wachtwoord: eeFAc2t5ff2X0RP2pYey

WebOke email account, inbox automatische

- Website: <https://webmail.weboke.nl/>
- Gebruikersnaam: [groteavond@gsr.nl](mailto:groteavond@gsr.nl)
- Wachtwoord: Zuidplein2025!

Secret key voor koppeling Wordpress en Supabase

- Naam: WP\_SHARED\_SECRET
- Wachtwoord: whriFPNrA571M77y7dXlrI6C^jXy2NhQBtLxgWOc8ljNo7eUkonEaCPT2daF