Droeftoeters-ICT – Back end

Inlever datum: 04/04/2025

# 1 – Informatie

## 1.1 - Studenten

Christian Scott Horler - 2227807

Lilah Sanchez -

Martijn van Houwelingen - 2225486

Sophie Iersel - 2232301

Thomas Koman – 2226974

## 2.2 – Project

Groepsnaam: Droeftoeters

Projectnaam: Droeftoeters-ICT

# 2 - Inhoud

Contents

[1 – Informatie 1](#_Toc194659822)

[1.1 - Studenten 1](#_Toc194659823)

[2.2 – Project 1](#_Toc194659824)

[2 - Inhoud 1](#_Toc194659825)

[3 – Inleiding 2](#_Toc194659826)

[4 – Algemene informatie 2](#_Toc194659827)

[4.1 – Beveiliging 2](#_Toc194659828)

[4.1.1 – Foutmeldingen 2](#_Toc194659829)

[4.2 - Data checks 2](#_Toc194659830)

[5 – Endpoints 3](#_Toc194659831)

[5.1- Procedures (traject) 3](#_Toc194659832)

[5.1.1 - Beschrijving 3](#_Toc194659833)

[5.1.2 - (Database)model 3](#_Toc194659834)

[5.1.3 - Endpoints 3](#_Toc194659835)

# 3 – Inleiding

Dit document zal uitleg geven over de gehele backend van droeftoeters-ict. Het zal gaan over de beschikbare endpoints, beveiliging, datachecks, welke responses je kan verwachten en welke modellen je moet gebruiken.

# 4 – Algemene informatie

## 4.1 – Beveiliging

### 4.1.1 – Foutmeldingen

Als er iets fout gaat in de api, zal de requester nooit weten wat er fout is gegaan. Iedere fout word afgevangen, gelogt en er word een **badrequest(400)** response gestuurd zonder body.  
De 2 andere responses die je kan krijgen zijn een **ok(200)** met een body als alles goed is gegaan, of een **unauthorized(401)** als je niet ingelogt bent.

## 4.2 - Data checks

Iedere controller endpoint zal een aantal dingen checken voordat het naar de data laag word verstuurd.

Alle controllers hebben een check om te kijken of alle uuid/guid strings geparsed kunnen worden naar een Guid datatype.

Alle **Read** (get) endpoints checken of het resultaat van de data laag request niet null is.   
Als dit null is betekent het dat er niks gevonden is met de gegeven conditie/parameter.   
Alle **ReadAll**(get) endpoints kunnen daarnaast wel een lege lijst terug geven, in het geval dat er geen entries in die table staan.

Alle **Write(**post/put/delete) endpoints checken of het resultaat true of false is. Een true betekent dat er 1 of meer rij in de database is aangepast.   
Daarnaast word er ook gecheckt of een entiteit met dezelfde id wel of niet bestaat om duplicates te voorkomen/proberen te schrijven naar een niet bestaande entiteit

# 5 – Endpoints

## 5.1- Procedures (traject)

### 5.1.1 - Beschrijving

De procedure controller bestuurt de data van alle trajecten. Het word gebruikt als je de trajecten wilt ophalen, aanpassen of verwijderen.  
Zie [data checks](#_4.2_-_Data) voor wat er allemaal word gedaan voordat de data naar de datalaag worden gestuurd.

### 5.1.2 - (Database)model

Het procedure model bestaat uit de volgende properties:

* Id : string [Primary key, required]
  + De string is een uuid/guid. De reden dat het niet als guid word opgeslagen is omdat met een string werken in de meeste instanties voor minder conversion issues zorgt op een database level
* Title : string [Required]
  + De titel van het traject
* Description : string [Optional]
  + De beschrijving van het traject, kan leeg zijn
* ProcedureItems : List<ProcedureItem> [Optional]
  + De lijst van onderliggende traject stappen word hierin terug gegeven bij een read request. Hierin data zetten zal niks doen op een write request

### 5.1.3 - Endpoints

Alle endpoints waar een traject word terug gestuurd zal ook de onderliggende trajectstappen hebben in de ProcedureItems lijst.

* ReadAll : IActionresult(List<Procedure>)
  + **Request** 
    - GET: /api/Procedure/all
  + **Body** 
    - Leeg
  + **Returnwaarden**
    - **OkObjectResult** (200): Als alles goed is verlopen zal het een OK terug sturen met de lijst van alle trajecten.
    - **BadRequestResult** (400): Als er iets maar fout gaat zal er een error gethrowed worden. zoals besproken in [beveiliging](#_4.1_–_Beveiliging), zal hier geen informatie over worden terug gestuurd in de response, maar deze word wel gelogd.
    - **Unauthorized** (401): De gebruiker is niet ingelogd (of geen bearer token in autorization header)
  + **Beschrijving**
    - ReadAll stuurt een lijst terug met alle trajecten in de database. Er word over iedere traject heen gelooped om de onderliggende stappen toe te voegen.
* Read(string id) : IActionresult([Procedure](#_5.1.2_-_(Database)model))
  + **Request** 
    - GET: /api/Procedure/all
  + **Body**
    - Leeg
  + Returnwaarden
    - **OkObjectResult (200):** Als alles goed is verlopen zal het een OK terug sturen met het gevonden traject
    - **BadRequestResult** (400): Als er iets maar fout gaat zal er een error gethrowed worden. zoals besproken in [beveiliging](#_4.1_–_Beveiliging), zal hier geen informatie over worden terug gestuurd in de response, maar deze word wel gelogd.
    - **Unauthorized** (401): De gebruiker is niet ingelogd (of geen bearer token in autorization header)
  + Beschrijvin
    - Read zoekt een specifieke traject op gebaseerd op welke id je hem meegeeft. Als er niks gevonden word, zal er een
* Write(Procedure procedure) : IActionresult(bool)
  + **Request**
    - POST: /api/Procedure/
  + **Body:**
    - [Procedure Model](#_5.1.2_-_(Database)model)
  + **Returnwaarden** 
    - **OkObjectResult** (200): Als alles goed is verlopen zal het een OK terug sturen met een boolean
    - **BadRequestResult** (400): Als er iets maar fout gaat zal er een error gethrowed worden. zoals besproken in [beveiliging](#_4.1_–_Beveiliging), zal hier geen informatie over worden terug gestuurd in de response, maar deze word wel gelogd.
    - **Unauthorized** (401): De gebruiker is niet ingelogd (of geen bearer token in autorization header)
  + **Beschrijving**
    - Write schrijft een procedure naar de database. Er word verwacht dat de id guid in de request al word meegegeven. Deze moet valide en uniek zijn.

## 5.2 – ProcedureItem (Traject item)

### 5.2.1 - Beschrijving

De procedure item controller bestuurt de data van alle traject stappen. Het word gebruikt als je de traject stappen wilt ophalen, aanpassen, verwijderen of als je het bovenliggende traject wilt ophalen.  
Zie [data checks](#_4.2_-_Data) voor wat er allemaal word gedaan voordat de data naar de datalaag worden gestuurd.

### 5.2.2 - (Database)model

Het procedure item model bestaat uit de volgende properties:

* Id : string [Primary key, required]
  + De string is een uuid/guid. De reden dat het niet als guid word opgeslagen is omdat met een string werken in de meeste instanties voor minder conversion issues zorgt op een database level
* Title : string [Required]
  + De titel van de trajectstap
* Description : string [Optional]
  + De beschrijving van de trajectstap, kan leeg zijn
* PreviousItemId : string [Optional]
  + Het ID van de vorige trajectstap, kan leeg zijn
* NextItemId : string [Optional]
  + Het ID van de volgende trajectstap, kan leeg zijn

### 5.1.3 – Endpoints

* ReadAll : IActionresult(List<Procedure>)
  + **Request** 
    - GET: /api/Procedure/all
  + **Body** 
    - Leeg
  + **Returnwaarden**
    - **OkObjectResult** (200): Als alles goed is verlopen zal het een OK terug sturen met de lijst van alle trajecten.
    - **BadRequestResult** (400): Als er iets maar fout gaat zal er een error gethrowed worden. zoals besproken in [beveiliging](#_4.1_–_Beveiliging), zal hier geen informatie over worden terug gestuurd in de response, maar deze word wel gelogd.
    - **Unauthorized** (401): De gebruiker is niet ingelogd (of geen bearer token in autorization header)
  + **Beschrijving**
    - ReadAll stuurt een lijst terug met alle trajecten in de database. Er word over iedere traject heen gelooped om de onderliggende stappen toe te voegen.

## 5.2 – ParentChild (Ouder-kind koppeling)

### 5.2.1 - Beschrijving

De parent-chid controller bestuurt de data van alle koppelingen tussen de ouder- en kinderaccounts. Het word gebruikt als je de koppelingen wilt ophalen, aanpassen of verwijderen.  
Zie [data checks](#_4.2_-_Data) voor wat er allemaal word gedaan voordat de data naar de datalaag worden gestuurd.

### 5.2.2 - (Database)model

Het parent child model bestaat uit de volgende properties:

* Id : string [Primary key, required]
  + De string is een uuid/guid. De reden dat het niet als guid word opgeslagen is omdat met een string werken in de meeste instanties voor minder conversion issues zorgt op een database level
* ParentId : string [Required]
  + De string is een uuid/guid. De reden dat het niet als guid word opgeslagen is omdat met een string werken in de meeste instanties voor minder conversion issues zorgt op een database level
* ChildId : string [Required]
  + De string is een uuid/guid. De reden dat het niet als guid word opgeslagen is omdat met een string werken in de meeste instanties voor minder conversion issues zorgt op een database level

### 5.1.3 - Endpoints

Alle endpoints waar een traject word terug gestuurd zal ook de onderliggende trajectstappen hebben in de ProcedureItems lijst.

* ReadAll : IActionresult(List<Procedure>)
  + **Request** 
    - GET: /api/Procedure/all
  + **Body** 
    - Leeg
  + **Returnwaarden**
    - **OkObjectResult** (200): Als alles goed is verlopen zal het een OK terug sturen met de lijst van alle trajecten.
    - **BadRequestResult** (400): Als er iets maar fout gaat zal er een error gethrowed worden. zoals besproken in [beveiliging](#_4.1_–_Beveiliging), zal hier geen informatie over worden terug gestuurd in de response, maar deze word wel gelogd.
    - **Unauthorized** (401): De gebruiker is niet ingelogd (of geen bearer token in autorization header)
  + **Beschrijving**
    - ReadAll stuurt een lijst terug met alle trajecten in de database. Er word over iedere traject heen gelooped om de onderliggende stappen toe te voegen.