

DT TRACKER Service Layer

By: Caroline Newcom

Overview

The backend service layer of my web application (DT Tracker) will be created using the Django framework. This service layer is crucial in allowing the user to interact with the web application to input and retrieve data to and from the PostgreSQL database.

The backend service is composed of the following three layers:

1. URL routing: Django uses URL Dispatching to properly map URLs to the views that handle the matching HTTP requests.
2. Views: Django uses views instead of controllers. The views are Python functions that receive the HTTP requests and return the HTTP responses for the users.
3. Models: Django uses models to communicate information about the structure of the data to the PostgreSQL database. Each model corresponds to a table within the database. The fields within the model are mapped to the fields within the database tables. Models then use migrations to manipulate the data within the PostgreSQL database.

Clarification

For this document, I will be using placeholder URLs until the web application has been created. Once my web application is built, I plan to use Heroku as the cloud application platform.

Service Endpoints

1. User Login

Method: POST

URL: <https://dttracker.herokuapp.com/api/auth/login>

Purpose: When a user (Developmental Therapist) wants to gain access to their account to interact with the web application, they will go to the login page. At the login page, they will enter their username and password before clicking the ‘Submit’ button. If the information they enter is validated, they will be directed to the ‘Caseload’ page. If their information is incorrect, their access will be denied.

Example Request:

POST /api/auth/login HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

“username”: “cnewcom”,

“password”: “Maryville”

}

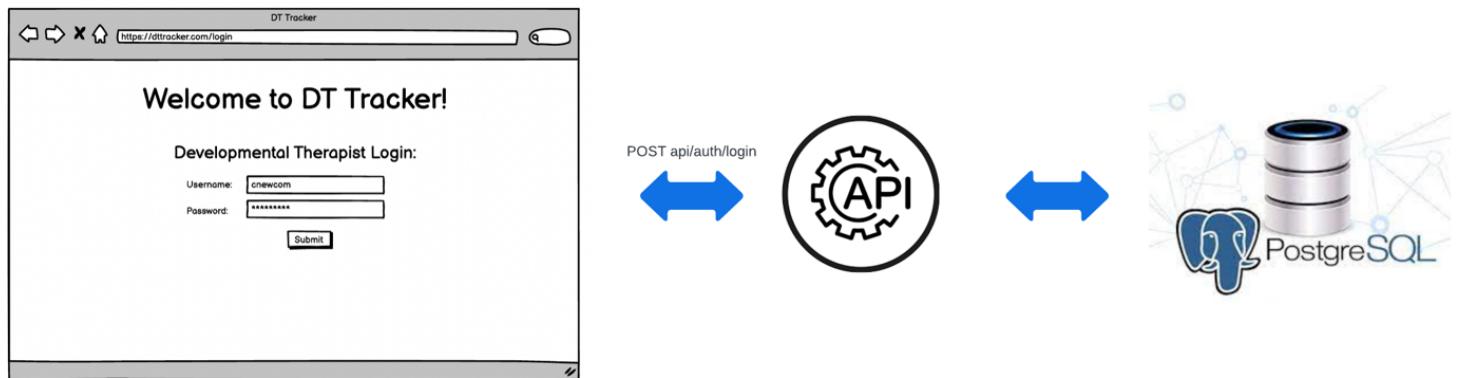
Success Response:

```
1  HTTP/1.1 200 OK
2  Content-Type: application/json
3
4  {
5      “message”: “User login was successful.”
6  }
7
```

Error Response:

```
1      HTTP/1.1 401 Unauthorized
2      Content-Type: application/json
3
4      {
5          "code": 401,
6          "message": "The username or password entered was invalid."
7      }
8
```

Diagram:



2.GET Caseload

Method: GET

URL: <https://dttracker.herokuapp.com/api/caseload/>

Purpose: After logging into the web application, the user will be directed to the “Caseload” page that displays the information of all the children on their caseload. The endpoint can also be called when the user clicks on the “Caseload” tab in the navigation bar.

Example Request:

GET /api/caseload/ HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

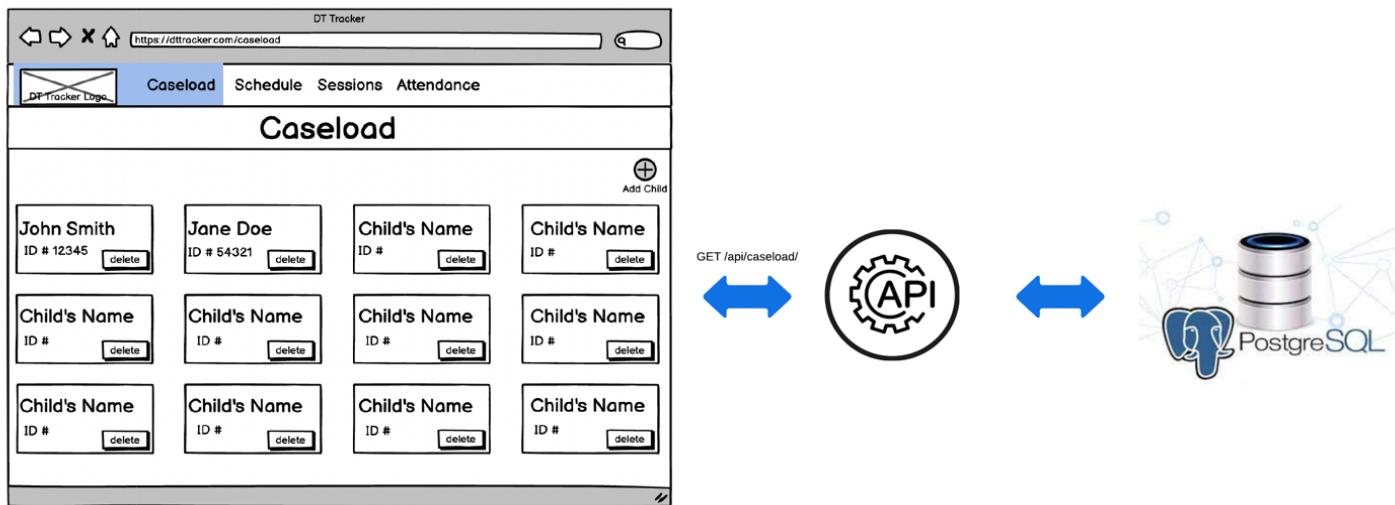
Success Response:

```
1  HTTP/1.1 200 OK
2  Content-Type: application/json
3
4  {
5      "caseload": [
6          {
7              "first_name": "John",
8              "last_name": "Smith",
9              "id_number": "12345"
10         },
11         {
12             "first_name": "Jane",
13             "last_name": "Doe",
14             "id_number": "54321"
15         },
16         {
17             "first_name": "George",
18             "last_name": "Lincoln",
19             "id_number": "51234"
20         },
21     ]
22 }
```

Error Response:

```
1  HTTP/1.1 403 Forbidden
2  Content-Type: application/json
3
4  {
5      "code": 403,
6      "message": "Caseload could not be retrieved."
7  }
8
```

Diagram:



3.DELETE Child

Method: DELETE

URL: <https://dttracker.herokuapp.com/api/caseload/:id>

Purpose: When a user (Developmental Therapist) wants to delete a child from their caseload, they will click on the 'delete' button within the child's box on the 'Caseload' page. Once this button is pressed, the endpoint will be called using the child's ID number as a parameter. The child will then be deleted from the 'Caseload' page.

Example Request:

DELETE /api/caseload/:id HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

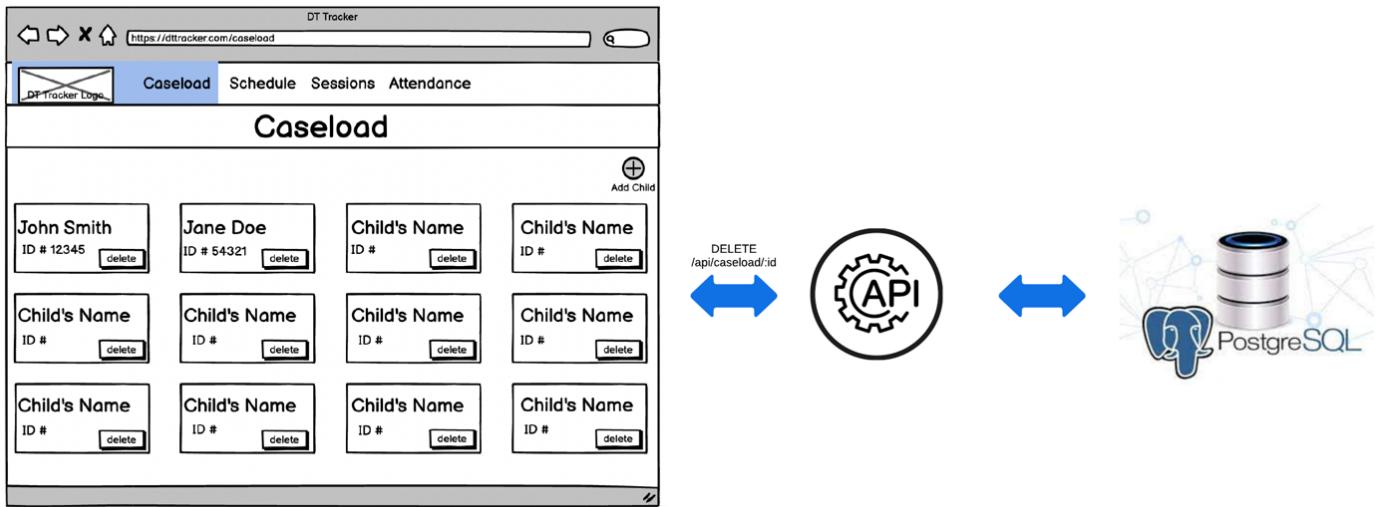
Success Response:

```
1  HTTP/1.1 200 OK
2
3  {
4      "message": "Child has been successfully removed."
5  }
6
```

Error Response:

```
1  HTTP/1.1 401 Unauthorized
2  Content-Type: application/json
3
4  {
5      "code": 401,
6      "message": "Authentication is required to delete child."
7  }
8
```

Diagram:



4.Add New Child

Method: POST

URL: <https://dttracker.herokuapp.com/api/caseload/addnewchild>

Purpose: When a user (Developmental Therapist) wants to add a new child to their caseload, they will click on the 'Add Child' button on the 'Caseload' page to be redirected to the 'Add New Child' page. The user will complete a form by entering the new child's information. Once the user hits the 'Submit' button, this endpoint will be called along with the data from the completed form for the submit request. If the information is processed successfully, the new child's information will be added to the database.

Example Request:

POST /api/caseload/addnewchild HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

```
        "first_name": "Betty",
        "last_name": "Johnson",
        "id_number": "11234"
    }
```

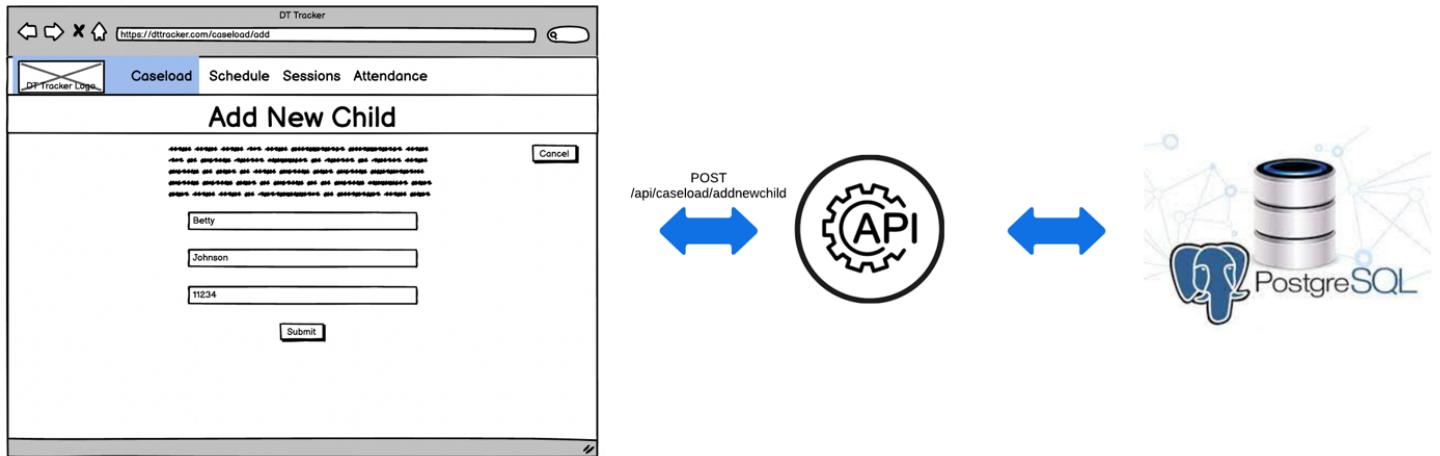
Success Response:

```
1  HTTP/1.1 201 Created
2  Content-Type: application/json
3
4  {
5      "message": "Child was successfully added."
6  }
```

Error Response:

```
1  HTTP/1.1 400 Bad Request
2  Content-Type: application/json
3
4  {
5      "code": 400,
6      "message": "All fields must be completed.",
7      "errors": [
8          {
9              "message": "First name is a required field.",
10             "field": "first_name"
11         }
12     ]
13 }
14
```

Design:



5.GET Schedule

Method: GET

URL: <https://dttracker.herokuapp.com/api/schedule/>

Purpose: When a user wants to view their schedule, they will click on the 'Schedule' tab in the navigation bar to call this endpoint. The 'Schedule' page will display the user's reoccurring weekly schedule.

Example Request:

GET /api/schedule/ HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

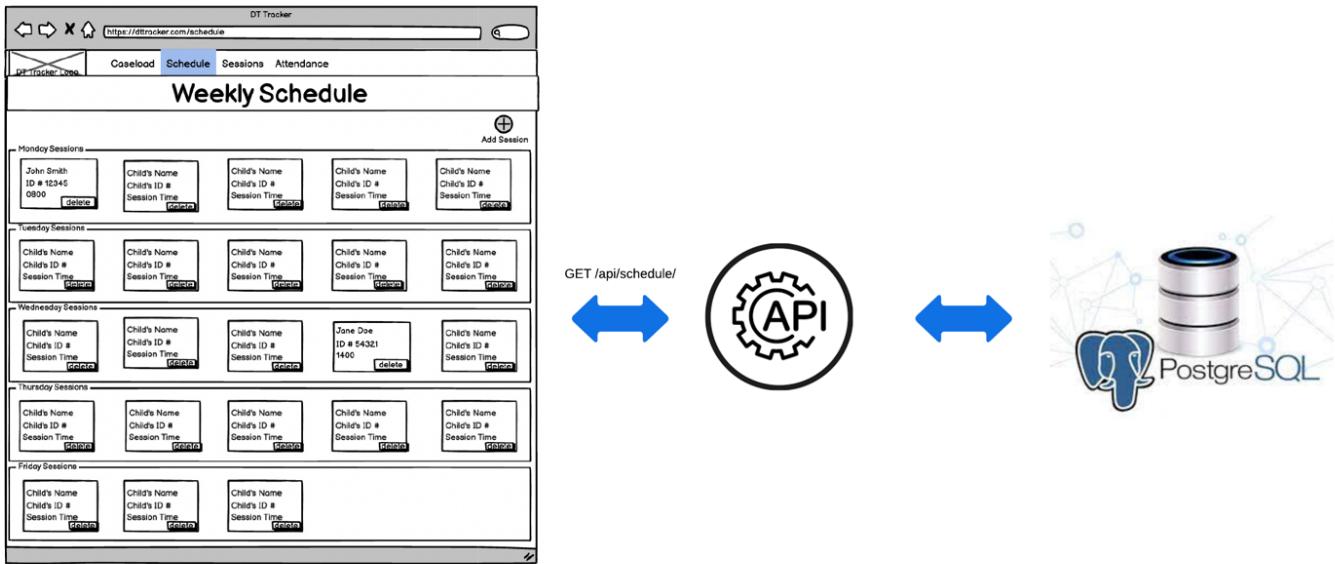
Success Response:

```
1  HTTP/1.1 200 OK
2  Content-Type: application/json
3
4  {
5      "session_schedule": [
6          {
7              "first_name": "John",
8              "last_name": "Smith",
9              "id_number": "12345",
10             "scheduled_week_day": "Monday",
11             "session_time": "0800"
12         },
13         {
14             "first_name": "Jane",
15             "last_name": "Doe",
16             "id_number": "54321",
17             "scheduled_week_day": "Wednesday",
18             "session_time": "1400"
19         },
20     ]
21 }
```

Error Response:

```
1  HTTP/1.1 403 Forbidden
2  Content-Type: application/json
3
4  {
5      "code": 403,
6      "message": "Schedule could not be retrieved."
7  }
8
```

Diagram:



6. DELETE Scheduled Session

Method: DELETE

URL: <https://dttracker.herokuapp.com/api/schedule/:id>

Purpose: When a user (Developmental Therapist) wants to delete a session from their schedule, they will click on the 'delete' button within the child's schedule box on the 'Schedule' page. Once this button is pressed, the endpoint will be called using the child's ID number as a parameter. The scheduled session will then be deleted from the 'Schedule' page.

Example Request:

DELETE /api/schedule/:id HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

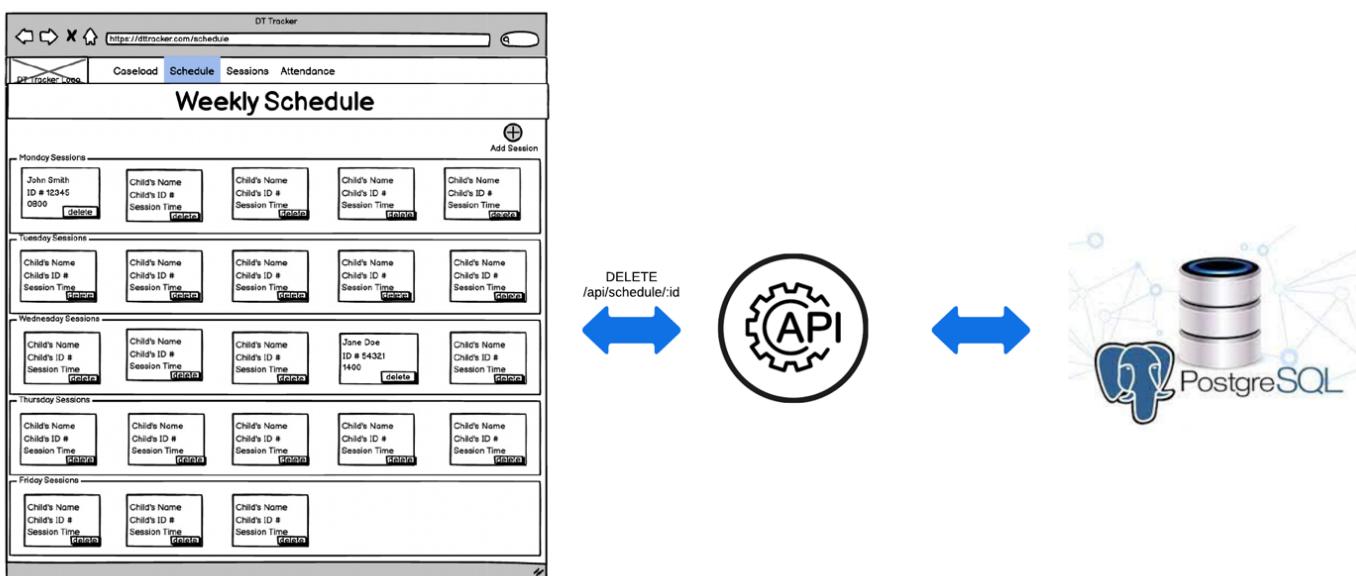
Success Response:

```
1  HTTP/1.1 200 OK
2  Content-Type: application/json
3
4  {
5      "message": "Session has been successfully removed."
6  }
7
```

Error Response:

```
1  HTTP/1.1 401 Unauthorized
2  Content-Type: application/json
3
4  {
5      "code": 401,
6      "message": "Authentication is required."
7  }
8
```

Diagram:



7.Add New Session

Method: POST

URL: <https://dttracker.herokuapp.com/api/caseload/addnewsession>

Purpose: When a user (Developmental Therapist) wants to add a new session to their schedule, they will click on the ‘Add Session’ button on the ‘Schedule’ page to be redirected to the ‘Add New Session’ page. The user will complete a form by entering the new session’s information. Once the user hits the ‘Submit’ button, this endpoint will be called along with the data from the completed form for the submit request. If the information is processed successfully, the new session’s information will be added to the database.

Example Request:

POST /api/caseload/addnewsession HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

```
“first_name”: “Betty”,  
“last_name”: “Johnson”,  
“id_number”: “11234”,  
“scheduled_week_day”: “Tuesday”,  
“session_time”: “1100”
```

}

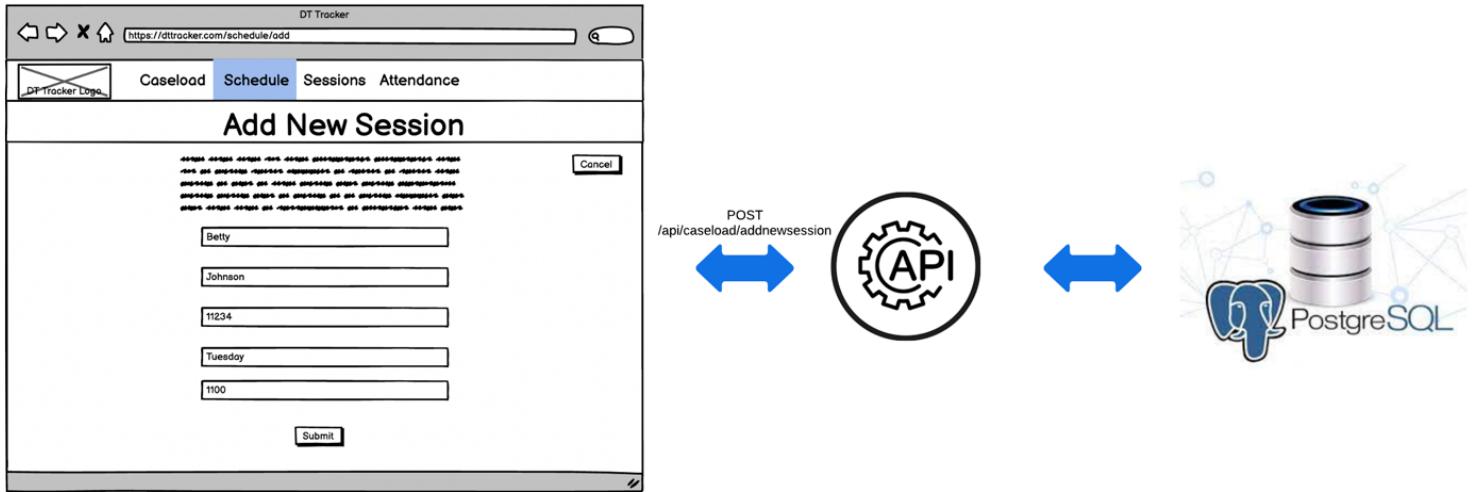
Success Response:

```
1  HTTP/1.1 201 Created
2  Content-Type: application/json
3
4  {
5      "message": "Session was successfully added."
6  }
7
```

Error Response:

```
1  HTTP/1.1 400 Bad Request
2  Content-Type: application/json
3
4  {
5      "code": 400,
6      "message": "All fields must be completed.",
7      "errors": [
8          {
9              "message": "Session time is a required field.",
10             "field": "session_time"
11         }
12     ]
13 }
14
```

Design:



8. Search by Child Name or ID to be Redirected to Child's Session Log

Method: POST

URL: <https://dttracker.herokuapp.com/api/sessions/search/:id>

Purpose: When a user (Developmental Therapist) wants to add a new session log for a child in their caseload, they will go to the 'Sessions' page. In the search bar, the user will enter with the child's name or ID number. Once the user hits the 'Submit' button, this endpoint will be called and produce a URL that will redirect the user to the 'Sessions Log' page for that child. If the child information entered is incorrect, the user will receive an error message.

Example Request:

POST /api/sessions/search/:id HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

```
        "id_number": "11234"  
    }  

```

Success Response:

```
1  HTTP/1.1 204 No Content  
2  Content-Type: application/json  
3  
4  {  
5      Location: https://dttracker.herokuapp.com/api/sessions/bettyjohnson/log  
6  }  
7
```

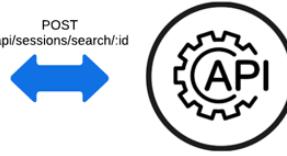
Error Response:

```
1  HTTP/1.1 404 Not Found  
2  Content-Type: application/json  
3  
4  {  
5      "code": 404,  
6      "message": "Child information not found."  
7  }  
8
```

Design:

The screenshot shows a web browser window titled 'DT Tracker' with the URL 'https://dttracker.com/sessions'. The page has a header with tabs: 'Caseload', 'Schedule', 'Sessions' (which is highlighted in blue), and 'Attendance'. Below the header is a section titled 'Sessions'. On the left side of this section is a search form with a text input field containing '11234', a placeholder 'Search for child by name or ID number', and a 'Submit' button.

POST
/api/sessions/search/:id



9.Add Session Log

Method: POST

URL: <https://dttracker.herokuapp.com/api/sessions/childname/log>

Purpose: When a user wants to add session information to their session log, the user will go to the ‘Sessions’ page and enter the child’s name into the search bar. The user will then see a form to complete regarding the child’s most recent session. The user will enter the session information for the session log before hitting the ‘Submit’ button. Once this button is pressed, this endpoint will be called and the data, if processed successfully, will be transferred to the database.

Example Request:

POST /api/sessions/childname/log HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

“session_date”: “2024-03-19”,

“session_attendance”: “excused”

}

Success Response:

```
1  HTTP/1.1 201 Created
2  Content-Type: application/json
3
4  {
5      “message”: “Session log was successfully added.”
6  }
7
```

Error Response:

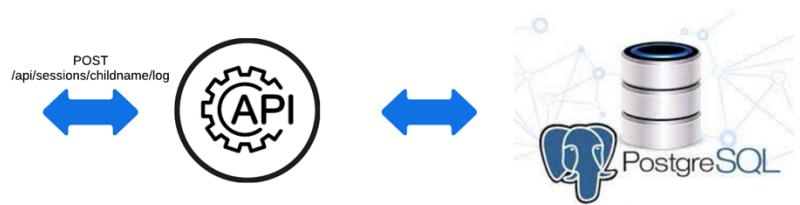
```

1   HTTP/1.1 400 Bad Request
2   Content-Type: application/json
3
4   {
5       "code": 400,
6       "message": "All fields must be completed.",
7       "errors": [
8           {
9               "message": "Date is a required field.",
10              "field": "session_date"
11           }
12       ]
13   }
14

```

Design:

The screenshot shows a web browser window for 'DT Tracker' at the URL <https://dttracker.com/sessions/log>. The page title is 'Session Log'. The navigation bar includes links for 'Caseload', 'Schedule', 'Sessions' (which is highlighted in blue), and 'Attendance'. Below the navigation bar, the child's name 'Betty Johnson' is displayed. A date input field shows '3/19/2024'. Below the date is a label 'Select Developmental Therapy session attendance for this week'. Underneath is a group of three checkboxes: 'session completed' (unchecked), 'excused absence' (checked), and 'unexcused absence' (unchecked). At the bottom is a 'Submit' button.



10. Search by Child Name or ID or by Total Caseload to be Redirected to Attendance Report

Method: POST

URL: <https://dttracker.herokuapp.com/api/attendance/search/:id>

Purpose: When a user wants to view attendance percentages, they will go to the ‘Attendance’ page. In the search bar, the user will either enter the child’s name or ID number or select total caseload. The user will then enter the month in which they would like to view the attendance report for. Once the user hits the ‘Submit’ button, this endpoint will be called and produce a URL that will redirect the user to the ‘Attendance Report’ page for either that child or the user’s total caseload. If the child information entered is incorrect, the user will receive an error message.

Example Request:

POST /api/attendance/search/:id HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

“id_number”: “12345”,

“month”: “03-2024”

}

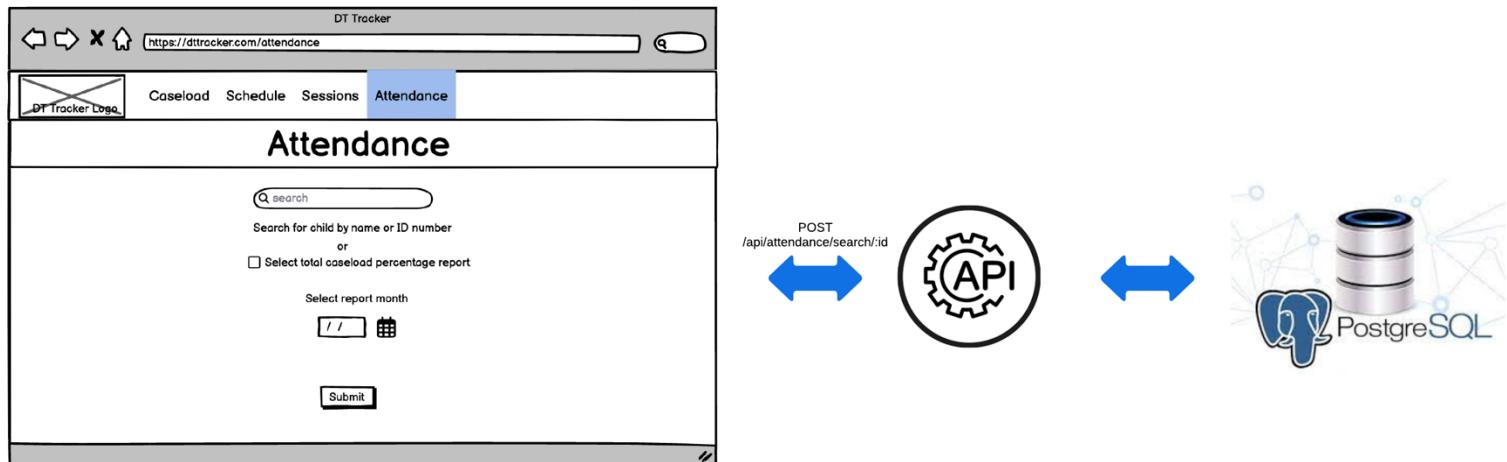
Success Response:

```
1  HTTP/1.1 204 No Content
2  Content-Type: application/json
3
4  {
5      Location: https://dttracker.herokuapp.com/api/attendance/johnsmith/percentagereport
6  }
7
```

Error Response:

```
1 HTTP/1.1 404 Not Found
2 Content-Type: application/json
3
4 {
5     "code": 404,
6     "message": "Child information not found."
7 }
8
```

Design:



Stretch Feature

Add Session Log

Method: POST

URL: <https://dttracker.herokuapp.com/api/sessions/childname/log>

Purpose: When a user wants to add session information to their session log, the user will go to the ‘Sessions’ page and enter the child’s name into the search bar. The user will then see a form to complete regarding the child’s most recent session. The user will enter the session/attendance information for the session log before hitting the ‘Submit’ button. Once this button is pressed, this endpoint will be called and the data, if processed successfully, will be transferred to the database.

Example Request:

POST /api/sessions/childname/log HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

“session_date”: “2024-03-19”,

“session_attendance”: “excused”,

“absence_reason”: “child and/or family member illness”

}

Success Response:

```

1   HTTP/1.1 201 Created
2   Content-Type: application/json
3
4   {
5     "message": "Session log was successfully added."
6   }
7

```

Error Response:

```

1   HTTP/1.1 400 Bad Request
2   Content-Type: application/json
3
4   {
5     "code": 400,
6     "message": "All fields must be completed.",
7     "errors": [
8       {
9         "message": "Reason for missed session is required.",
10        "field": "absence_reason"
11      }
12    ]
13  }
14

```

Design:

DT Tracker

<https://dttracker.com/sessions/log>

Caseload Schedule Sessions Attendance

Session Log

Betty Johnson

2024-03-19

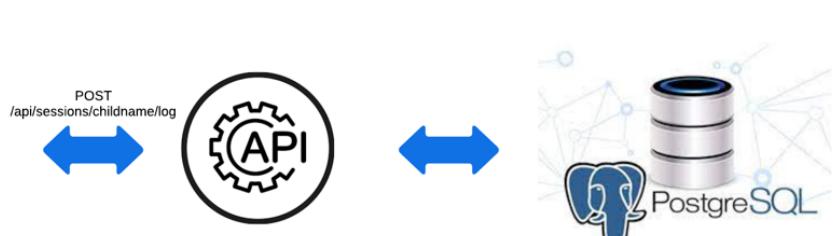
Select Developmental Therapy session attendance for this week

session completed
 excused absence
 unexcused absence

If absent, select from reasons below:

child and/or family member illness
 child and/or family out of town
 conflicting appointment
 family no show
 other

Submit



Search by Child Name or ID or by Total Caseload to be Redirected to Absence Pattern Report

Method: POST

URL: <https://dttracker.herokuapp.com/api/attendance/search/:id>

Purpose: When a user wants to view attendance percentages, they will go to the ‘Attendance’ page. In the search bar, the user will either enter the child’s name or ID number or select total caseload. The user will then enter the month in which they would like to view the attendance report for. Lastly, the user will select the box beside “Select for absence pattern report”. Once the user hits the ‘Submit’ button, this endpoint will be called and produce a URL that will redirect the user to the ‘Absence Pattern Report’ page for either that child or the user’s total caseload. If the child information entered is incorrect, the user will receive an error message.

Example Request:

POST /api/attendance/search/:id HTTP/1.1

Host: dttracker.herokuapp.com

Accept: application/json

Content-Type: application/json

{

“id_number”: “12345”,

“month”: “03-2024”,

“absence_reason”: true

}

Success Response:

```
1 HTTP/1.1 204 No Content
2 Content-Type: application/json
3
4 {
5     Location: https://dttracker.herokuapp.com/api/attendance/johnsmith/absencereport
6 }
7
```

Error Response:

```
1 HTTP/1.1 404 Not Found
2 Content-Type: application/json
3
4 {
5     "code": 404,
6     "message": "Child information not found."
7 }
8
```

Design:

The screenshot shows the DT Tracker web interface. At the top, there's a navigation bar with links for Caseload, Schedule, Sessions, and Attendance. The Attendance tab is active. Below the navigation, there's a section titled "Attendance". It contains a search bar with placeholder text "Search" and a dropdown menu. Below the search bar, there are two options: "Search for child by name or ID number" and "Select total caseload percentage report". Underneath these, there's a "Select report month" section with a date picker and two checkboxes: "Select for percentage report" and "Select for absence pattern report". At the bottom of the form is a "Submit" button.

