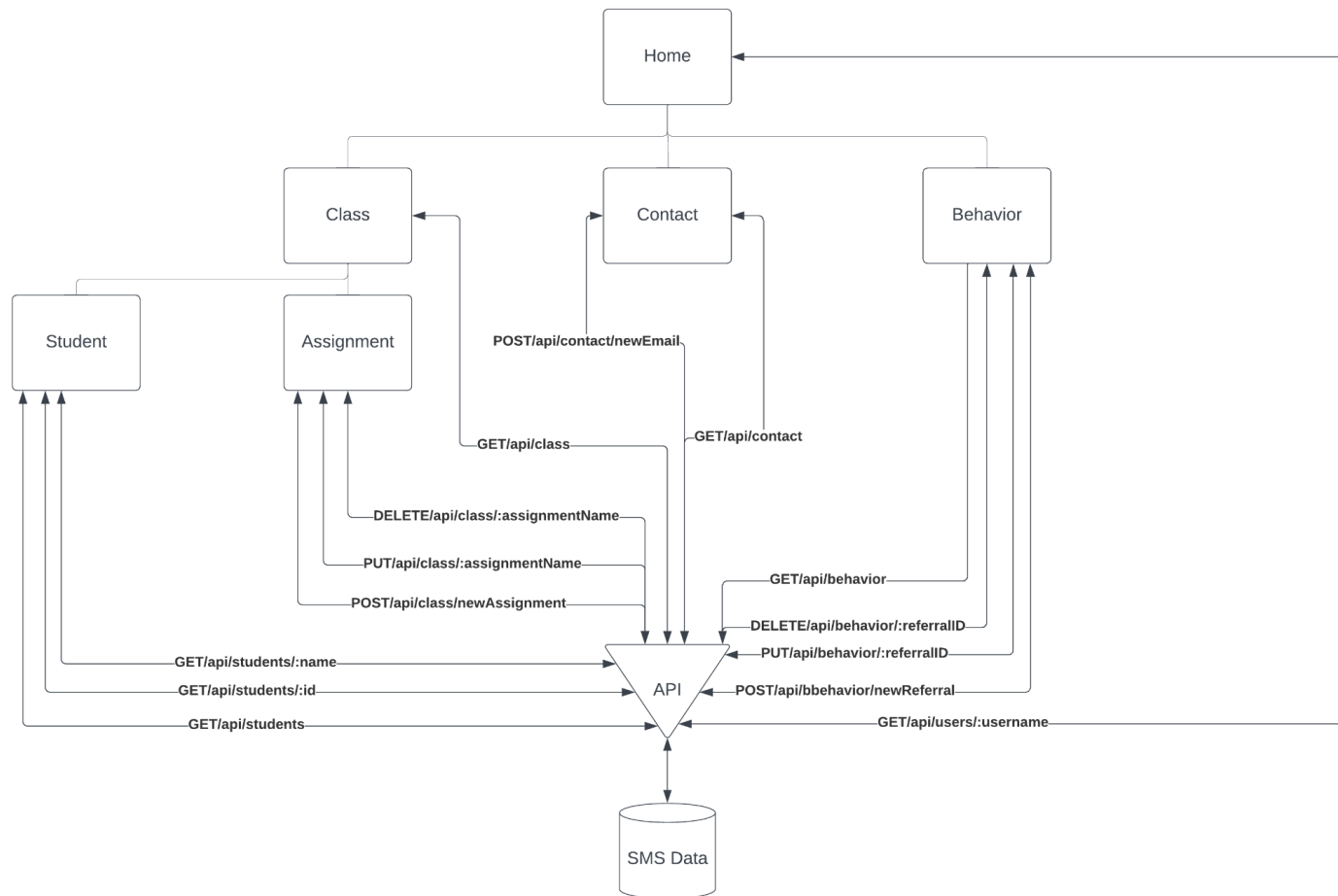


Student Management System Service Layer

Designed by: Zachary Rhodes



<u>Get Students</u>	
Method:	GET
URL:	https://swdv691.herokuapp.com/api/students
Purpose:	
This endpoint will be activated when a user travels to a student page. It will select that the correct students are accessible to each user (i.e. themselves for a student user or a whole class for a teacher user).	
Example Request:	
curl --request GET --url https://swdv691.herokuapp.com/api/students	
Success Response:	
<pre>{ "data": [{ "studentID": "100101", "firstName": "Zachary", "lastName": "Rhodes", "grade": "senior", "parentID": "101010" }] }</pre>	
Error Response:	
<pre>{ "status": 400, "message": "User does not have student access." }</pre>	

<u>Get Student by ID</u>	
Method:	GET
URL:	https://swdv691.herokuapp.com/api/students/:id
Purpose:	
When a user clicks on a student name, this endpoint will be activated to take the user to the	

specific student's page. The student information will be displayed on the page.

Example Request:

```
curl --request GET --url https://swdv691.herokuapp.com/api/students/100101
```

Success Response:

```
{
  "data": [
    {
      "studentID": "100101",
      "firstName": "Zachary",
      "lastName": "Rhodes",
      "grade": "senior",
      "parentID": "101010"
    }
  ]
}
```

Error Response:

```
{
  "code": 404,
  "message": "Student not found."
}
```

Get Student by Name

Method: GET

URL: https://swdv691.herokuapp.com/api/students/:name

Purpose:

A teacher or administrator user can search for students by name from the home page. When searching for a student, this endpoint is activated and the user is taken to the correct student page to view the relevant data.

Example Request:

```
curl --request GET --url https://swdv691.herokuapp.com/api/students/zacharyrhodes
```

Success Response:

```
{
  "data": [
    {
```

```
{
  "studentID": "100101",
  "firstName": "Zachary",
  "lastName": "Rhodes",
  "grade": "senior",
  "parentID": "101010"
}
```

Error Response:

```
{
  "code": 404,
  "message": "Student not found."
}
```

User Login**Method:** GET**URL:** https://swdv691.herokuapp.com/api/users/:username**Purpose:**

When a user logs in this endpoint will be activated. The endpoint directs the user to the correct home page- teacher, student, parent, etc. This ensures the correct permissions are maintained within the system.

Example Request:

```
curl --request GET --url https://swdv691.herokuapp.com/api/users/zrhodes
```

Success Response:

```
{
  "data": [
    {
      "teacherID": "123456",
      "firstName": "Zachary",
      "lastName": "Rhodes",
      "email": "zrhodes@school.edu",
      "permissions": "teacher"
    }
  ]
}
```

Error Response:

```
{
  "code": 404,
  "message": "User not found."
}
```

Get Class

Method: GET

URL: <https://swdv691.herokuapp.com/api/class>

Purpose:

The endpoint will be active when a teacher clicks on a class page link. The endpoint will pull in a list of students and assignments relevant to that particular class to be viewed by the teacher user.

Example Request:

```
curl --request GET --url https://swdv691.herokuapp.com/api/class
```

Success Response:

```
{
  "data": [
    {
      "classID": "6655",
      "teacherName": "Rhodes",
      "className": "Biology",
      "students": {
        {
          "name": "Zachary Rhodes",
          "grade": "B"
        },
        {
          "name": "Nina Flynn",
          "grade": "A"
        }
      }
    },
    "assignments": {
      {
        "assignmentsName": "Photosynthesis Project",
        "pointsPossible": "100"
      },
      {
        "assignmentsName": "Cellular Respiration Project",
        "pointsPossible": "100"
      }
    }
  ]
}
```

```
}  
]  
}
```

Error Response:

```
{  
  "status": 400,  
  "message": "User does not have student access."  
}
```

Create an Assignment**Method:** POST**URL:** <https://swdv691.herokuapp.com/api/class/newAssignment>**Purpose:**

When a teacher user enters the class page, they can choose to create a new assignment for the class. The teacher must enter the name, description, due date, and points to be stored in the database.

Example Request:

```
curl --request POST --url https://swdv691.herokuapp.com/api/class/newAssignment  
-data {  
  "assignmentName": "Photosynthesis Project",  
  "assignmentDescription": "Students will create a poster explaining photosynthesis.",  
  "dueDate": "8/27/2023",  
  "possiblePoints": "100",  
  "pointsEarned": "--"  
}
```

Success Response:

```
{  
  "assignmentName": "Photosynthesis Project",  
  "assignmentDescription": "Students will create a poster explaining photosynthesis.",  
  "dueDate": "8/27/2023",  
  "possiblePoints": "100",  
  "pointsEarned": "--"  
}
```

Error Response:

```
[
```

```

{
  "message": "Due date is required.",
  "path": [
    "dueDate"
  ],
  "type": "any.required",
  "context": {
    "key": "dueDate",
    "label": "dueDate"
  }
}
}
]

```

Grade an Assignment

Method: PUT

URL: <https://swdv691.herokuapp.com/api/class/:assignmentName>

Purpose:

When an existing assignment needs to be graded, the teacher user will enter a grade in the points earned column and select save. This will activate this endpoint. The endpoint will update the assignment to enter the number of points earned. This data will then be displayed and a percentage grade will be calculated in turn.

Example Request:

```

curl --request PUT --url https://swdv691.herokuapp.com/api/class/Photosynthesis%20Project
--data {
  "assignmentName": "Photosynthesis Project",
  "assignmentDescription": "Students will create a poster explaining photosynthesis.",
  "dueDate": "8/27/2023",
  "possiblePoints": "100",
  "pointsEarned": "80"
}

```

Success Response:

```

{
  "assignmentName": "Photosynthesis Project",
  "assignmentDescription": "Students will create a poster explaining photosynthesis.",
  "dueDate": "8/27/2023",
  "possiblePoints": "100",
  "pointsEarned": "80"
}

```

Error Response:

```
[
  {
    "message": "Points earned is required.",
    "path": [
      "pointsEarned"
    ],
    "type": "any.required",
    "context": {
      "key": "pointsEarned",
      "label": "pointsEarned"
    }
  }
]
```

Delete an Assignment

Method: DELETE

URL: <https://swdv691.herokuapp.com/api/students/class/:assignmentName>

Purpose:

When a teacher user enters the class page, they can choose to create a new assignment for the class. The teacher must enter the name, description, due date, and points to be stored in the database.

Example Request:

```
curl --request DELETE --url
https://swdv691.herokuapp.com/api/class/Photosynthesis%20Project
--data {
  "assignmentName": "Photosynthesis Project",
  "assignmentDescription": "Students will create a poster explaining photosynthesis.",
  "dueDate": "8/27/2023",
  "possiblePoints": "100",
  "pointsEarned": "--"
}
```

Success Response:

```
{
  "status": "200",
  "message": "Photosynthesis Project was deleted."
}
```

Error Response:

```
{
```



```
{
  "status": "404",
  "message": "Assignment does not exist."
}
```

Create a Referral

Method: POST

URL: <https://swdv691.herokuapp.com/api/behavior/newReferral>

Purpose:

A teacher or administrator will enter a new behavioral referral by entering the necessary information into a form and selecting save. This endpoint will be activated and save the data to the database.

Example Request:

```
curl --request POST --url https://swdv691.herokuapp.com/api/class/:newReferral
  -data {
    "referralID": "10025",
    "studentName": "Nina Flynn",
    "incidentDate": "8/27/2023",
    "teacherName": "Rhodes",
    "description": "Student was caught drinking alcohol behind the bleachers"
  }
```

Success Response:

```
{
  "referralID": "10025",
  "studentName": "Nina Flynn",
  "incidentDate": "8/27/2023",
  "teacherName": "Rhodes",
  "description": "Student was caught drinking alcohol behind the bleachers"
}
```

Error Response:

```
[
  {
    "message": "Incident date is required.",
    "path": [
      "incidentDate"
    ],
    "type": "any.required",
    "context": {
      "key": "incidentDate",

```

```
    "label": "incidentDate"
  }
}
]
```

Update a Referral

Method: PUT

URL: <https://swdv691.herokuapp.com/api/behavior/:referralID>

Purpose:

When an existing referral needs to be updated, the teacher or administrator will click the update button- activating this endpoint. The user can change the data and then save the edited data to the database.

Example Request:

```
curl --request PUT --url https://swdv691.herokuapp.com/api/behavior/10025
  -data {
    "referralID": "10025",
    "studentName": "Nina Flynn",
    "incidentDate": "8/27/2023",
    "teacherName": "Stewart",
    "description": "Student was caught drinking alcohol behind the bleachers"
  }
```

Success Response:

```
{
  "referralID": "10025",
  "studentName": "Nina Flynn",
  "incidentDate": "8/27/2023",
  "teacherName": "Rhodes",
  "description": "Student was caught drinking alcohol behind the bleachers"
}
```

Error Response:

```
[
  {
    "message": "Incident date is required.",
    "path": [
      "incidentDate"
    ],
    "type": "any.required",
    "context": {
```

```
{
  "key": "incidentDate",
  "label": "incidentDate"
}
]
```

Delete a Referral

Method:	DELETE
URL:	https://swdv691.herokuapp.com/api/behavior/:referralID
Purpose:	
If a behavior referral is entered mistakenly, the teacher or administrator can select the delete button to remove the record from the database. This endpoint will be activated and remove the data from the database.	
Example Request:	
<pre>curl --request DELETE --url https://swdv691.herokuapp.com/api/behavior/10025 -data { "referralID": "10025", "studentName": "Nina Flynn", "incidentDate": "8/27/2023", "teacherName": "Stewart", "description": "Student was caught drinking alcohol behind the bleachers" }</pre>	
Success Response:	
<pre>{ "status": "200", "message": "Referral: 10025 was deleted." }</pre>	
Error Response:	
<pre>{ "status": "404", "message": "Referral does not exist." }</pre>	

Get Referrals

Method:	GET
----------------	-----

URL:	https://swdv691.herokuapp.com/api/behavior
Purpose:	
This endpoint will be activated when a user travels to the behavior page. When a user navigates to the behavior page- this endpoint will pull all active behavior referral from the database to display on the behavior page.	
Example Request:	
curl --request GET --url https://swdv691.herokuapp.com/api/behavior	
Success Response:	
<pre>{ "data": [{ "referralID": "10025", "studentName": "Nina Flynn", "incidentDate": "8/27/2023", "teacherName": "Stewart", "description": "Student was caught drinking alcohol behind the bleachers" }] }</pre>	
Error Response:	
<pre>{ "status": 400, "message": "User does not have behavior access." }</pre>	

<u>Send an Email</u>	
Method:	POST
URL:	https://swdv691.herokuapp.com/api/contact/newEmailt
Purpose:	
When a user wants to send a contact email, this endpoint is active. The contact email information is recorded in a form. The data is then stored and can be sent via an email server to the recipient.	
Example Request:	
curl --request POST --url https://swdv691.herokuapp.com/api/class/newAssignment	

```

-data {
  "senderName": "Zachary Rhodes",
  "recipientName": "Nina Flynn",
  "senderEmail": "zrhodes@school.edu",
  "recipientEmail": "nflynn@parents.com",
  "subject": "Grades",
  "body": "Your student's grades are failing in my class.",
}

```

Success Response:

```

{
  "senderName": "Zachary Rhodes",
  "recipientName": "Nina Flynn",
  "senderEmail": "zrhodes@school.edu",
  "recipientEmail": "nflynn@parents.com",
  "subject": "Grades",
  "body": "Your student's grades are failing in my class.",
}

```

Error Response:

```

[
  {
    "message": "Recipient name is required.",
    "path": [
      "recipientName"
    ],
    "type": "any.required",
    "context": {
      "key": "recipientName",
      "label": "recipientName"
    }
  }
]

```

Get Contacts

Method: GET

URL: <https://swdv691.herokuapp.com/api/contact>

Purpose:

When a user navigates to the contact page, this endpoint is activated to display all current contacts that the user has sent.

Example Request:

```
curl --request GET --url https://swdv691.herokuapp.com/api/contact
```

Success Response:

```
{
  "data": [
    {
      "senderName": "Zachary Rhodes",
      "recipientName": "Nina Flynn",
      "senderEmail": "zrhodes@school.edu",
      "recipientEmail": "nflynn@parents.com",
      "subject": "Grades",
      "body": "Your student's grades are failing in my class."
    }
  ]
}
```

Error Response:

```
{
  "status": 400,
  "message": "User does not have contact access."
}
```