ABCollege Student Portal

Teodoro D. Delson Jr.

May 20, 2020

Project Description

The project is aimed to be used by students and administrators of ABCollege. The admin can create account, view accounts, and create courses. While the student can view their transcript of records, view courses uploaded by the admin, and edit their personal information.

Technologies

**Node JS** – It is an open source runtime for JavaScript that runs on a V8 engine. This allows the program to communicate with back-end.

Node Packages:

-Express

- Express -sessions

- jest

- nodemon

-router

-super test

-fs for logs

**Express JS** – Is a framework for JavaScript mainly to develop web or mobile applications.

Jest – It is an open source Framework maintained by Facebook to test JavaScript.

**PostgreSQL –** Open source relational database. It is considered as the most advanced database system.

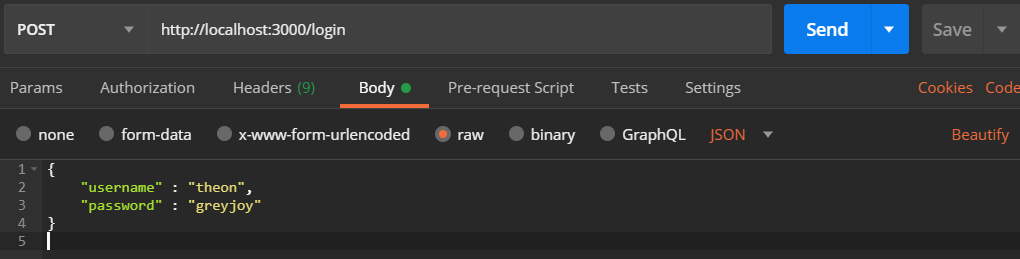
**JavaScript/TypeScript –** Java script as a front end more interactive while JavaScript as a backend enables the program to communicate with the server. TypeScript is a super set of JavaScript.

**Request Reference**

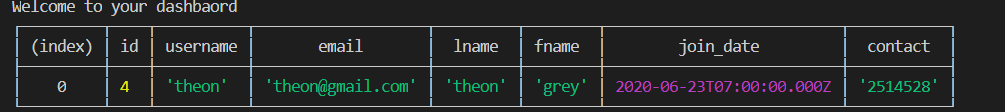
|  |  |  |
| --- | --- | --- |
| **POST /login** |  |  |
| **Parameters of the Request body** | username (string) | Required |
|  | Password (string) | Required |
| **HTTP status code when success** | 200 (OK) |  |
| **Request** **Format** | JSON |  |

Description: A user can log in by using the following format shown in the example. The login accepts json to get values.

**Example Request:**



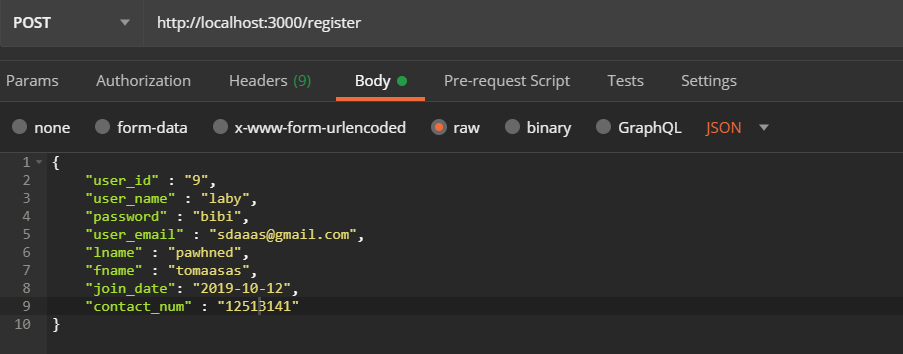
**Example Output:**



|  |  |  |
| --- | --- | --- |
| **POST /register** |  |  |
| **Parameters of the Request body** | User\_id | Required |
|  | user\_name | Required |
|  | password | Required |
|  | user\_email | Required |
|  | lname | Required |
|  | fname | Required |
|  | join\_date | Required |
|  | contact\_num | Required |
| **HTTP status code when success** | 200 (OK) |  |
| **Request** **Format** | JSON |  |
| **Requires Authentication** | Yes | Role must be admin |

Description: A end point to register users. Only admin can create users. All fields in the json are required. See example below.

**Example Request**



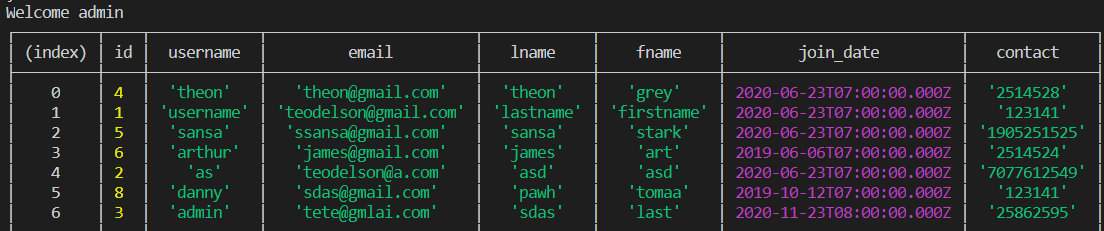
|  |  |  |
| --- | --- | --- |
| **GET /getallAccounts** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must be admin |

Description: This endpoint returns all the user accounts as long as authenticated as admin

**Example Request:**

GET <http://localhost:3000/getallAccounts>

**Example Response**



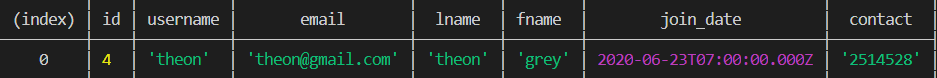
|  |  |  |
| --- | --- | --- |
| **GET /getAccountsById:id** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must be admin |
| **URI parameters** | Id = number |  |

Description: This endpoint returns specific user accounts by id as long as authenticated as admin

**Example Request:**

**GET** //localhost:3000/getAccountsById/4

**Example Response:**



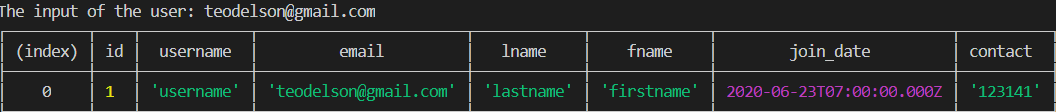
|  |  |  |
| --- | --- | --- |
| **GET /getAccountByEmail:email** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must be admin |
| **URI parameters** | Id = number |  |

Description: This endpoint returns specific user accounts by email as long as authenticated as admin

**Example Request:**

**GET** localhost:3000/getAccountByEmail/teodelson@gmail.com

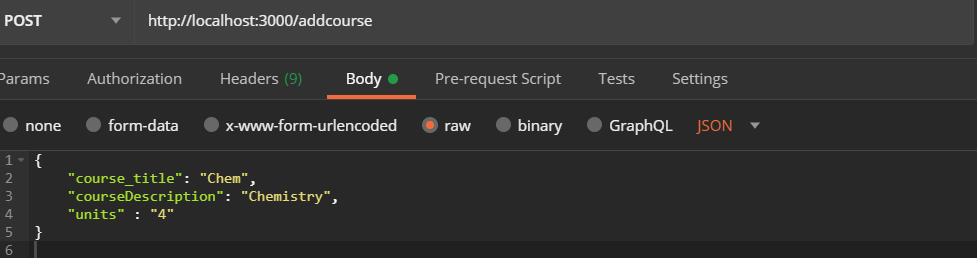
**Example Response**



|  |  |  |
| --- | --- | --- |
| **POST /addcourse** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must admin |
| **Request body parameter** | course\_title | Required |
|  | courseDescription | Required |
|  | units | Required |

Description: This endpoint allows the admin to add a new course. All fields in the body parameter is required. See example below.

**Example of Request**



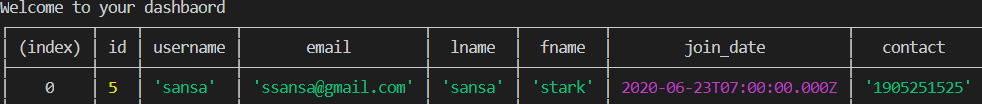
**Example of Response**: Course has been added successfully.

|  |  |  |
| --- | --- | --- |
| **GET /studDashboard** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must be student |

Description: Upon logging in as a student, the user will be redirected to the dashboard for students which contains their personal information.

**Example Request: GET localhost:3000/studDashboard**

**Example of Response:**



|  |  |  |
| --- | --- | --- |
| **PATCH /changeContact** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must be student |
| **Request body parameter** | new\_contact | Required |

Description: This endpoint allows the user to update his or her contact information. The request body only needs one parameter (new\_contact).

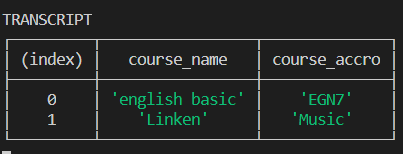
**Example Request: PATCH localhost:3000/changeContact**

|  |  |  |
| --- | --- | --- |
| **GET /records** |  |  |
| **HTTP status code when success** | 200 (OK) | 403 (Forbidden) – If not authenticated |
| **Requires authentication** | Yes | Role must be student |

Description: This endpoint returns all the classes and courses the user has been assigned to. This serves as a record or transcript.

**Example Request: GET localhost:3000/records**

**Example Response:**

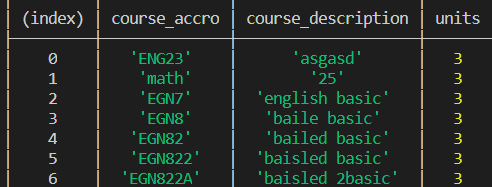


|  |  |  |
| --- | --- | --- |
| **GET /getallCourses** |  |  |
| **HTTP status code when success** | 200 (OK) |  |
| **Requires authentication** | No |  |

Description: Any user can access this endpoint. This displays all courses available.

**Example Request: GET** [**http://localhost:3000/getallCourses**](http://localhost:3000/getallCourses)

**Example Response**

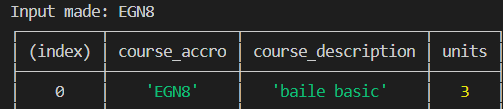


|  |  |  |
| --- | --- | --- |
| **GET /getallCourses:Id** |  |  |
| **HTTP status code when success** | 200 (OK) |  |
| **Requires authentication** | No |  |
| **URI parameter** | Id | String |

Description: This endpoint needs an id parameter for searching specific course by Id.

**Example Request GET** [**http://localhost:3000/getCourses/EGN8**](http://localhost:3000/getCourses/EGN8)

**Example Response**

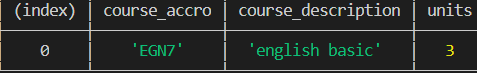


|  |  |  |
| --- | --- | --- |
| **GET /http://localhost:3000/getCoursesByDescription:coursename** |  |  |
| **HTTP status code when success** | 200 (OK) |  |
| **Requires authentication** | No |  |
| **URI parameter** | coursename | String |

Description: This endpoint needs an id parameter for searching specific course by coursename.

**Example Request GET http://localhost:3000/getCoursesByDescription/english basic**

**Example Response**



|  |  |  |
| --- | --- | --- |
| **PATCH**  **/http://localhost:3000/changepassword** |  |  |
| **HTTP status code when success** | 200 (OK) |  |
| **Requires authentication** | YES |  |
| **Applies only to Student role** | YES |  |
| **Applies to Admin role** | NO |  |
| **Request body parameter : new\_password** | Required |  |

Description: This endpoint changes the password of a student.