Task Management App

The **Task Management App API** allows users to:

1. Create a new task
2. Retrieve tasks in a particular state
3. Approve a task from "Doing to Done" state

using GET and POST requests.

1. Create a new task

**Endpoint**: http://localhost:5000/task/a3/create-task/<username>/<password>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Method** | **Parameters** | **Behaviour** | **Data type** |
| CreateTask | POST | username | Mandatory | string |
| password | Mandatory | string |
| **Body** | | |
| acronym | Mandatory | string  VARCHAR(3) |
| planName | Optional | string  VARCHAR(45) |
| taskName | Mandatory | string VARCHAR(45) |
| description | Mandatory | string VARCHAR(255) |
| Example | | | | |
| 1. Using Javascript:   **Request**:  var myHeaders = new Headers();  myHeaders.append("Content-Type", "application/json");  var raw = JSON.stringify({  "acronym": "APP",  "planName": "New Plan 2",  "taskName": "Task 1",  "description": "Task description"  });  var requestOptions = {  method: 'POST',  headers: myHeaders,  body: raw,  redirect: 'follow'  };  fetch("http://localhost:5000/task/a3/create-task/User13/St3ng12!", requestOptions)  .then(response => response.text())  .then(result => console.log(result))  .catch(error => console.log('error', error));  *Note: Order in which the fields are entered does not matter*   1. Using cURL:   **Request**:    curl --location --request POST http://localhost:5000/task/a3/create-task/User13/St3ng12!' \  --data-raw '{  "acronym": "APP",  "planName": "New Plan 2",  "taskName": "Task 1",  "description": "Task description"  }'  *Note: Order in which the fields are entered does not matter*   1. Output:   {    "message": "Task created"  } | | | | |
| Error Codes | | | | |
| | **Error Code** | **Description** | | --- | --- | | 102 | Invalid user | | 103 | Invalid password | | 104 | Inactive user | | 200 | Unauthorized | | 302 | Invalid plan | | 303 | Invalid app | | 305 | Missing parameters | | 402 | Invalid route | | | | | |

1. Retrieve tasks in a particular state

**Endpoint**: http://localhost:5000/task/a3/get-tasks-by-state/<username>/<password>/<acronym>/<state>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Method** | **Parameters** | **Behaviour** | **Data type** |
| GetTaskbyState | GET | username | Mandatory | string |
| password | Mandatory | string |
| acronym | Mandatory | string  VARCHAR(3) |
| state | Mandatory | String  *Available states 1. Open 2. Todo 3. Doing 4. Done 5. Closed*  *(Case sensitive)* |
| Example | | | | |
| 1. Using Javascript:   **Request**:  var requestOptions = {  method: 'GET',  redirect: 'follow'  };  fetch("http://localhost:5000/task/a3/get-tasks-by-state/User13/St3ng12!/APP/Open", requestOptions)  .then(response => response.text())  .then(result => console.log(result))  .catch(error => console.log('error', error));   1. Using cURL:   **Request**:    curl --location --request GET http://localhost:5000/task/a3/get-tasks-by-state/User13/St3ng12!/APP/Open'   1. Output**:**   [  {  "task\_id": "APP\_17",  "task\_name": "fdsa",  "description": "fdsa",  "plan\_name": null,  "acronym": "APP",  "state": "Open",  "creator": "User13",  "owner": "User13",  "create\_date": "2022-05-26T08:29:08.000Z",  "notes": "[{\"details\":\"fdsa\",\"timestamp\":\"2022-05-26 4:46:59 pm\",\"creator\":\"User13\",\"state\":\"Open\"},{\"details\":\"notes\",\"timestamp\":\"2022-05-26 4:32:38 pm\",\"creator\":\"User13\",\"state\":\"Open\"},{\"details\":\"Task created\",\"timestamp\":\"2022-05-26 4:29:08 pm\",\"creator\":\"User13\",\"state\":\"Open\"}]"  },  {  "task\_id": "APP\_18",  "task\_name": "A3 Task",  "description": "desc",  "plan\_name": null,  "acronym": "APP",  "state": "Open",  "creator": "User13",  "owner": "User13",  "create\_date": "2022-05-26T09:14:40.000Z",  "notes": "[{\"details\":\"Task created\",\"timestamp\":\"2022-05-26 5:14:40 pm\",\"creator\":\"User13\",\"state\":\"Open\"}]"  },  ] | | | | |
| Error Codes | | | | |
| | **Error Code** | **Description** | | --- | --- | | 102 | Invalid user | | 103 | Invalid password | | 104 | Inactive user | | 303 | Invalid app | | 304 | Invalid state | | 402 | Invalid route | | | | | |

1. Approve a task from "Doing to Done" state

**Endpoint**: http://localhost:5000/task/a3/promote-task-2-done/<username>/<password>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Method** | **Parameters** | **Behaviour** | **Data type** |
| PromoteTask2Done | POST | username | Mandatory | string |
| password | Mandatory | string |
| **Body** | |  |
| acronym | Mandatory | string  VARCHAR(3) |
| taskID | Mandatory | string  VARCHAR(45) |
| Example | | | | |
| 1. Using Javascript:   **Request**:  var myHeaders = new Headers();  myHeaders.append("Content-Type", "application/json");  var raw = JSON.stringify({  "acronym": "APP",  "taskID": "APP\_13"  });  var requestOptions = {  method: 'POST',  headers: myHeaders,  body: raw,  redirect: 'follow'  };  fetch("http://localhost:5000/task/a3/promote-task-2-done/User13/St3ng12!", requestOptions)  .then(response => response.text())  .then(result => console.log(result))  .catch(error => console.log('error', error));  *Note: Order in which the fields are entered does not matter*   1. Using cURL:   **Request**:    curl --location --request POST http://localhost:5000/task/a3/promote-task-2-done/User13/St3ng12!' \  --data-raw '{  "acronym": "APP",  "taskID": "APP\_13"  }'  *Note: Order in which the fields are entered does not matter*   1. Output:   {  "message": "Task updated from Doing to Done"  } | | | | |
| Error Codes | | | | |
| | **Error Code** | **Description** | | --- | --- | | 102 | Invalid user | | 103 | Invalid password | | 104 | Inactive user | | 200 | Unauthorized | | 301 | Invalid task ID | | 303 | Invalid app | | 305 | Missing parameters | | 401 | Task state is not "Doing" | | 402 | Invalid route | | | | | |