ReadMe:-

\*Run the code from app.js on Port 3000

Functionality: Log In as an already registered staff

Route: /

Request type: post

Request body: {“email” : “bake@” , “password”: “bake”}

(After logging in a code will appear in the response area, Copy the code and paste it in the “Value” area in Headers in postman to save the current user’s information.)

Functionality: Change current password, or change the password if it’s the first time for the user to login with a password of “123456”

Route: /changepassword

Request type: Put

Request body: {“email” : “bake@” , “newpassword”: “newpassword”}

Response : “Password updated successfully!”

Functionality: Update the current user’s profile. If the user is in department ‘hr’ then he/she can update everything except his/her id & name, however is the User is in any other department, she/ he can’t update their id, name, salary, department

Route: /updateprofile

Request type: Put

Request body: {“salary”: “700”, “daysOff”: “[Monday, Friday]”, “officeLocation”: “C7-119”}

Response : “Profile updated successfully!”

Functionality: View the User’s profile

Route: /viewprofile

Request type: Get

Example of how to call the route: http://localhost:5000/viewprofile

Response: A detailed String of all the current user’s information.

Functionality: View all of the current user attendance Records

Route: /records

Request type: Get

Response: An array filled with all the days the User attended

Functionality: View all of the current user attendance Records in a certain month starting from the 11th of the chosen month till the 10th of the next month

Route: /records

Request type: post

Request body: {“month” : “May”}

(The month specified should start with an upper case letter)

Response: An array filled with all the days the User attended the specified month

Functionality: Sign in for the day, adds a new record to the attendance collection with a signout set to null

Route: /signin

Request type: Get

Response: “Signed in successfully”

Functionality: Sign out for the day, updates the last attendance entry for a user

Route: /signout

Request type: Put

Response: “Signed in successfully”

HR FUNCTIONALITIES:

1)-Functionality: add location to the database in module location with its location

,type and capacity

-Route:/locationAffairs

-Request type : POST

Example of how to call the route: /locationAffairs

Example for using request body:

Request type: POST

Request body: {"location":"c5","capacity":2,"type":"office"}

Example for response:

Functionality: add the location to the system

Route: /locationAffairs

Request type: POST

Response: the added location

ex:{

"\_id": "5fe518030423ef2ba8203247",

"location": "c5",

"remainingPlaces": 2,

"capacity": 2,

"type": "office",

"\_\_v": 0

}

2)Functionality:update a location already in the database

-Route:/locationAffairs

-Request type : PUT

-parameters : 1)unique location string

2)the update .ex: "type":"room"

Example of how to call the route :/locationAffairs

Example for using request body:

Request type: PUT

Request body: {"location":"c5","capacity":7,"type":"room"}

Example for response:

Response: the UPDATED location

ex:{

"\_id": "5fe518030423ef2ba8203247",

"location": "c5",

"remainingPlaces": 7,

"capacity": 7,

"type": "room",

"\_\_v": 0

}

3)Functionality:delete a location with a given location string

-Route:/locationAffairs

-Request type : DELETE

-parameters : 1)unique location string

Example for using request body:

Request type: DELETE

Request body: {"location":"c5"}

Example for response: the location deleted successfuly

4)Functionality:ADD a new faculty to the database in module faculty

-Route:/facultyAffairs

-Request type : POST

-parameters : 1)unique facultyName string

2)optionaly : array of departmentName

Example for using request body:

Request type: POST

Request body: it is optionaly to add array of departments

{"facultyName":"engineering","departmentName":["cs","networks"]}

or {"facultyName":"engineering"}

Example for response:the INSERTED FACULTY

{

"departmentName": [],

"\_id": "5fe5214cda0d9521a0c57a11",

"facultyName": "engineering",

"departmentName":["cs","networks"]//optionaly

"\_\_v": 0

}

5)Functionality:update a already inserted faculty

-Route:/facultyAffairs

-Request type : PUT

-parameters : 1)unique facultyName string

2)the update

Example of how to call the route :/facultyAffairs

Example for using request body:

Request type: PUT

Request body: {"facultyName":"engineering","departmentName":[mecha]}

Example for response:

Response: the UPDATED faculty

{

"departmentName": [

"mecha"

],

"\_id": "5fe5214cda0d9521a0c57a11",

"facultyName": "engineering",

"\_\_v": 0

}

6)Functionality: delete already existing faculty and the departments under it

-Route:/facultyAffairs

-Request type :DELETE

-parameters : 1)unique facultyName string

Example for using request body:

Request type: DELETE

Request body: {"facultyName":"engineering"}

Example for response:

Response: the faculty deleted successfuly

7) Functionality: ADD department under a faculty

-Route:/departmentAffairs

-Request type :POST

-parameters : 1)unique facultyName string

2)unique departmentName string

Example for using request body:

Request type: POST

Request body: {"facultyName":"engineering","departmentName":"cs"}

Example for response:it return the faculty with the added department

response:{

"departmentName": [

"cs"

],

"\_id": "5fe529e5853fdb1f20b3b50c",

"facultyName": "engineering",

"\_\_v": 2

}

8)Functionality:update department under a faculty

-Route:/departmentAffairs

-Request type :PUT

-parameters : 1)unique facultyName string

2)unique departmentName string

3)unique departmentName2 string

Example for using request body:

Request type: PUT

Request body: {"facultyName":"engineering","departmentName":"cs","departmentName2":"cs1"}

Example for response:it return the faculty with the UPDATED department

{

"departmentName": [

"cs1"

],

"\_id": "5fe529e5853fdb1f20b3b50c",

"facultyName": "engineering",

"\_\_v": 3

}

9)Functionality:delete department under a faculty

-Route:/departmentAffairs

-Request type :DELETE

-parameters : 1)unique facultyName string

2)unique departmentName string

Example for using request body:

Request type: DELETE

Request body: {"facultyName":"engineering","departmentName":"cs"}

Example for response:it return the faculty with the DELETED department

{

"departmentName": [

],

"\_id": "5fe529e5853fdb1f20b3b50c",

"facultyName": "engineering",

"\_\_v": 4

}

10)Functionality:ADD a course under department

-Route:/courseAffairs

-Request type : POST

-parameters : 1)unique departmentName string

2)unique courseName string

Example for using request body:

Request type: POST

Request body: {"departmentName":"math","courseName":"math7"}

Example for response:

{

"courseName": [

"math7"

],

"\_id": "5fe52b35b4bd5a28e025eee9",

"departmentName": "math",

"\_\_v": 3

}

11)Functionality:UPDATE a course under a department

-Route:/courseAffairs

-Request type : PUT

-parameters : 1)unique departmentName string

2)unique courseName string

Example for using request body:

Request type: PUT

Request body: {"departmentName":"math","courseName":"math7","courseName2":"math10"}

Example for response:

{

"courseName": [

"math10"

],

"\_id": "5fe52b35b4bd5a28e025eee9",

"departmentName": "math",

"\_\_v": 4

}

12)Functionality:DELETE a course under a department

-Route:/courseAffairs

-Request type : DELETE

-parameters : 1)unique departmentName string

2)unique courseName string

Example for using request body:

Request type: DELETE

Request body: {"departmentName":"math","courseName":"math10"}

Example for response:the department with the course deleted

{

"courseName": [

],

"\_id": "5fe52b35b4bd5a28e025eee9",

"departmentName": "math",

"\_\_v": 6

}

13)Functionality: ADD a new staff member to the system

-Route:/staffAffairs

-Request type : POST

-parameters :{office, unique email,department,salary,name}

Example for using request body:

Request type:POST

Request body: {"name":"sara","office":"c3","email":"sara@guc.edu","salary":"20000","department":"csen"}

Example for response:the added staff member with the default password

{

"password": "123456",

"daysOff": [

"friday"

],

"hours": 0,

"missingDays": 30,

"\_id": "5fe5ca0e93be4e0a50c2f0a8",

"name": "sara",

"email": "sara@guc.edu",

"id": "ac-2",

"officeLocation": "c3",

"salary": 20000,

"department": "csen",

"role": "academicMember",

"\_\_v": 0

}

14)Functionality: UPDATE existing staff member

-Route:/staffAffairs

-Request type : PUT

-parameters :the staff member id and the update you want to make

Example for using request body:

Request type:PUT

Request body:{"id":"ac-2","department":"mecha"}

Example for the response : the staff member with the update

{

"password": "123456",

"daysOff": [

"friday"

],

"hours": 0,

"missingDays": 30,

"\_id": "5fe5ca0e93be4e0a50c2f0a8",

"name": "sara",

"email": "sara@guc.edu",

"id": "ac-2",

"officeLocation": "c3",

"salary": 20000,

"department": "mecha",

"role": "academicMember",

"\_\_v": 0

}

15)Functionality:DELETE already existing staff member by his/her id

-Route:/staffAffairs

-Request type : DELETE

-parameters :the staff member id

Example for using request body:

Request type:DELETE

Request body:{"id":"ac-2"}

Example for response: the staff member deleted successfully !

16)functionalty : Manually add a missing sign in/sign out record of a staff member except for himself/herself.

-Route:/addMissingSign

-Request type : PUT

-parameters :1)the staff member id

2) the date of the attendance

Example for using request body:

Request type:PUt

Request body: "id":"ac-2",

"date":"2020-12-24T23:01:42.110+00:00",

"checkOut":"2020-12-24"

Example for response: checkout added!

17)Functionality : View any staff member attendance record by his/her id

-Route:/viewAtttendanceRecord

-Request type : GET

-parameters : the staff member id

Example for using request body:

Request type:Get

Request body: {"id":"ac-2" }

Example for response:

[

{

"\_id": "5fe51dd6b73f0d20b0d845a4",

"id": "ac-2",

"date": "2020-12-24T23:01:42.110Z",

"checkIn": "2020-12-24T23:01:42.110Z",

"checkOut": "2020-12-24T00:00:00.000Z",

"\_\_v": 0

}

]

18)Functionality : View staff members with missing hours/days by his/her id.

-Route:/viewMissingHoursOrDays

-Request type : GET

-parameters : 1) the staff member id

2) the type either days or hours

Example for using request body:

Request type:Get

Example for response:

Request body : {"id":"ac-2","type":"days" }

Example for response:

[

"2020-12-11T12:01:30.027Z",

"2020-12-13T12:01:30.444Z",

"2020-12-14T12:01:30.643Z",

"2020-12-15T12:01:30.848Z",

"2020-12-16T12:01:31.060Z",

"2020-12-17T12:01:31.257Z",

"2020-12-18T12:01:31.466Z",

"2020-12-20T12:01:31.881Z",

"2020-12-21T12:01:32.097Z",

"2020-12-22T12:01:32.297Z",

"2020-12-23T12:01:32.501Z",

"2020-12-24T12:01:32.711Z",

"2020-12-25T12:01:32.925Z"

]

19) Functionality : update salary of a staff member by his/her id and the updated salary

-Route:/updateSalary

-Request type : PUT

-parameters : 1) the staff member id

2) the newSalary update

Example for using request body:

Request type:PUT

Request body: {"id":"ac-2","newSalary":50000 }

Example for response : the staff member profile along with the newSalary

{

"password": "123456",

"daysOff": [

"friday"

],

"hours": 0,

"missingDays": 30,

"\_id": "5fe5d47219fc7a20784f0292",

"name": "sara",

"email": "sara@guc.edu",

"id": "ac-2",

"officeLocation": "c3",

"salary": 50000,

"department": "csen",

"role": "academicMember",

"\_\_v": 0

Functionality: View the status of all submitted requests. They can also view only the accepted requests,

only the pending requests or only the rejected requests.

Route: /viewReqState

Request type: GET

Request body: example: {"state":"accepted"}

Request response: Array of requests. Example of a single request :{"state": "accepted","\_id": "5fe34d4de727ad25349f2b13","reciverID": "10","senderID": "1","type": "replacement","reqId": 2,

"\_\_v": 0}

Functionality: Cancel a still pending request or a request whose day is yet to come.

Route: /cancelReq

Request type: DELETE

Request body: example: {"rId":2}

Request response: Message stating that its canceled or cant be canceled.

Functionality: send Replacement request.

Route: /sendReplacementReq

Request type: POST

Request body: {"replID":"","cname":"","sTime":"","sDay":"","date":""}

Request response: Request that is sent to either the replacement or hod. Example of the request :{"state": "accepted","\_id": "5fe34d4de727ad25349f2b13","reciverID": "10","senderID": "1","type": "replacement","reqId": 2,

"\_\_v": 0}

Functionality: view Replacement request.

Route: /viewReplacmentReq

Request type: GET

Request body: {"type":"replacement"}

Request response: Array of requests. Example of a single request :{"state": "accepted","\_id": "5fe34d4de727ad25349f2b13","reciverID": "10","senderID": "1","type": "replacement","reqId": 2,

"\_\_v": 0}

Functionality: View their schedule. Schedule should show teaching activities and replacements if present.

Route: /viewSchedule

Request type: GET

Request response: Array of teachingSlots. example of a single teachingslot: {"\_id": "5fe32a9c0f743e446c8b62ff","staffID": "1","slotTime": "2nd", "slotLoc": "c3", "slotDay": "saturday","courseName": "cs1","slotDate": "2020-12-19T22:00:00.000Z"}

Functionality: Change their day off by sending a “change day off” request (automatically sent to HOD),

and optionally leave a reason.

Route: /changeDayReq

Request type: POST

Request body: { "reason":""}

Request response: Request that is sent to the hod({"recieverID":})

Functionality: Send a “slot linking” request (automatically sent to course coordinator). A “slot linking”

request is a request done by the academic member to indicate their desire to teach a slot.

Route: /sendSlotLinkingRequest

Request type: POST

Request body: Example {"cname":"csen01","stime":"1st","sDay":"sunday","sLoc":"c2101","sdate":"2020-12-28"}

Request response: request that is sent to the course coordinator

Functionality: Notified whenever their requests are accepted or rejected.

Route: /getNotified

Request type: GET

Request response: Array of messages. example of a single message: {"your request number 1 has been accepted"}

Functionality: Submit any type of “leave” request (automatically sent to HOD). “Compensation” leave

must have a reason. For any other leave, academic members can optionally write a brief

reason behind it.

Route: /sendLeaveRequest

Request type: POST

Request body: example: {"id":"","leavetype":"","date":"","reason":""}

Request response: Request that has been sent to the hod.

//co-ordinator

Functionalty : view the slotlinking request sent to the coordinator

Route:'/SLreq'

Type : GET

Example to request body :

Type :GET

Response: docs of all slotlinking requests sent to the coordinator