Run index.js to launch the server.

The server is listening to port 3000.

**Note**: We are using a scheduler which runs at time 23:59 every day. This scheduler is responsible for calculating the missing hours and if this day is a missing day for each staff in the system. So missing hours and missing days are calculated for each staff at midnight. We used “node-schedule” for this.

1 –

**Request**:

Functionality: login the staff member to the system

Route: /login

Request type: POST

Request body: { “username”:”Mohamed”, “password”:”password1234”}

**Response**:

Functionality: get an authentication token if the login credentials are valid/correct.

Route: /login

Request type: POST

Response ex : eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjVmZDRkZjNhMTIxNDAxMWYwMDIwODdmMCIsImlhdCI6MTYwNzc4NjM1NSwiZXhwIjoxNjA3Nzg5OTU1fQ.wafQf42FDx9mmoaOPzjjK9Yps5xh0Ve-kdZf1kNxpZM

2 –

Functionality: logout the staff member from the system

Route: /logout

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request ex : post http://localhost:3000/logout

auth-token:eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjVmZDVmYmQ0NmE1OGI2MmYzMGJmMmRjYSIsImlhdCI6MTYwNzg2MDk2MCwiZXhwIjoxNjA3ODY0NTYwfQ.qDLqEcIinox70mGhv-KuQc3JPiznCSZFhgFiwbthaLU

Content-Type: application/json

3 –

**Request**:

Functionality: Update profile of a staff member

Route: /updateProfile

Request type: PUT

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “gender”:”male”, “displayName”:”Midoo”, “imgPath”:” <https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg>”}

4 –

**Request**:

Functionality: View profile of a staff member

Route: /viewProfile

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality: view profile of logged in user.

Route: /viewProfile

Request type: GET

Response ex: { "gender": "male", "displayName": "mido", "imgPath": "<https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg>" }

5 –

**Request**:

Functionality: Reset password of a staff member

Route: /resetPassword

Request type: PUT

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “oldPassword”:”password1234”, “newPassword”:”newPass123”}

6 –

**Request**:

Functionality: Staff Member Sign in to Campus

Route: /signInToCampus

Request type: PUT

Request header : {“auth-token” : “the token we got from login route”}

7 –

**Request**:

Functionality: Staff Member Sign out from Campus

Route: /signOutFromCampus

Request type: PUT

Request header : {“auth-token” : “the token we got from login route”}

8 –

**Request**:

Functionality: Logged in staff member view attendance

Route: /viewAttendance

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality: View attendance records.

Route: /viewAttendance

Request type: GET

Response ex: [{

"\_id": "5fe5c00491cca6238c062708",

"signIn": "2020-12-25T10:31:24.198Z",

"signOut": "2020-12-25T10:33:40.013Z"

},

{

"\_id": "5fe5d3eef465a42bb0f7a689",

"signIn": "2020-12-28T04:00:00.000Z",

"signOut": "2020-12-28T20:00:00.000Z"

}]

9 –

**Request**:

Functionality: Logged in staff member view attendance by month

Route: /viewAttendanceByMonth

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request Body : {“month” : “December”}

**Response**:

Functionality: View attendance records by month.

Route: / viewAttendanceByMonth

Request type: POST

Response ex: [{

"\_id": "5fe5c00491cca6238c062708",

"signIn": "2020-12-25T10:31:24.198Z",

"signOut": "2020-12-25T10:33:40.013Z"

},

{

"\_id": "5fe5d3eef465a42bb0f7a689",

"signIn": "2020-12-28T04:00:00.000Z",

"signOut": "2020-12-28T20:00:00.000Z"

}]

10 –

**Request**:

Functionality: Logged in staff member view his Missing Days

Route: /viewMissingDays

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality: View Missed Days records.

Route: /viewMissingDays

Request type: GET

Response ex: ["December 19th 2020”]

11 –

**Request**:

Functionality: View if they are having missing hours or extra hours.

Route: /viewMissingExtraHours

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality: View missed and extra hours in the current attendance month.

Route: /viewMissingExtraHours

Request type: GET

Response ex: {

"missingHours": "0 hours",

"missingMinutes": "0 minutes",

"extraHours": "3 hours",

"extraMinutes": "36 minutes"

}

----------------------------------------------

= HR Functionalities

* Make sure to enter with a member with “HR" as his/her role as in the following example otherwise any route will send a message saying:(“you are not authorized to do this functionality").
* All datatypes of the inputs in the request body should be valid.

**Request**:

Functionality: login the staff member to the system

Route: /login

Request type: POST

Request body: {“username”:”HRTester”,“password”:”password123”}

**Response**:

Functionality: get an authentication token if the login credentials are valid/correct.

Route: /login

Request type: POST

Response example:

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6Im1vaGFtZWQuZWxzYXllZGFsaTIiLCJyb2xlIjoiSFIiLCJpYXQiOjE2MDg4OTUyMDcsImV4cCI6MTYwODkwNjAwN30.NGhEUuqRN00yjhK8O2XzPDCt9jQRwPVM\_91zaMTlJ4c

------------------------------------------------------------------------------------

12- Add Location:

Functionality: Adding locations to the system

Route: /hrAddLocation

Request type: POST

Request body: {location\_name:”C7.202",

location\_type: ”office", capacity: 6, occupied places: 4}

Response example : a message saying that a location is added successfully. Incase the location exists before another message appears indicating that a location already exists.

-----------------------------------------------------------------------------------

13-Update Location:

Functionality: updating locations that already exist in the system.

Route: /hrUpdateLocation

Request type: PUT

Request body:{“location\_name": ”C7.202", “updatedLocation\_name": ”C7.201", “updatedLocation\_type": ”Tutorial Room", “updatedCapacity": 25 , “updatedOccupiedPlaces" : 0}

Response:

* Make sure “updatedOccupiedPlaces" are always less than or equal to “updatedCapacity" as it will send a message of (“Occupied places can not be greater than capacity of the location")
* Otherwise, a message of (“Location is updated!") is sent.

--------------------------------------------------------------------------------

14- Delete Location:

Functionality: deleting locations that already exist in the system.

Route: /hrDeleteLocation

Request type: DELETE

Request body:{“location\_name": ”C7.202"}

Response:

* a message of (“Location is deleted") is sent
* Incase a location does not exist it will send a message of (“Can not find Location").

------------------------------------------------------------------------------------

15- Add Faculty:

Functionality: adding faculties in the system.

Route: /hrAddFaculty

Request type: POST

Request body:{“name": ”Engineering", “department": []}

Response:

* If the faculty is already on the system, a message of (“Faculty already exists!") is sent.
* Other than that, a message of (“Faculty is added!") is sent.

------------------------------------------------------------------------------------

16- Update Faculty:

Functionality: updating faculties’ names that already exist in the system.

Route: /hrUpdateFaculty

Request type: PUT

Request body:{“facultyName":”Engineering", “updatedName": ”Eng"}

Response:

* If “facultyName" is not found, a message of (“Can not find faculty") is sent.
* If the “updatedName" already exists, a message of (“Faculty already exists") is sent.
* Otherwise, message of (“Faculty is updated") is sent.

------------------------------------------------------------------------------------

17- Delete Faculty:

Functionality: deleting faculties that already exist in the system.

Route: /hrDeleteFaculty

Request type: DELETE

Request body:{“facultyName":”Engineering"}

Response:

* If “facultyName" is not found, message of (“Cannot find faculty").
* Otherwise, a message of (“Faculty deleted")

------------------------------------------------------------------------------------

18- Add department:

Functionality: adding departments to faculties that already exist in the system.

Route: /hrAddDepartment

Request type: POST

Request body:{“facultyName":”Engineering" , “departmentName": ”MET"}

Response:

* If “facultyName" is not found, a message of (“There is no faculty with that name") is sent.
* If the department sent in the request body is the same as the one on the system under a certain faculty, a message of (“Department already exists in that faculty") is sent.
* Otherwise, a message of (“Department added successfully") is sent.

------------------------------------------------------------------------------------

19- Update department:

Functionality: updating departments to faculties that already exist in the system.

Route: /hrUpdateDepartment

Request type: PUT

Request body:{“facultyName":”Engineering" , “departmentName": ”MET", ”newdepartmentName": ”MET11"}

Response:

* If the “facultyName" is not found, a message of (“There is no faculty with that name") is sent.
* If the “departmentName" entered is not found, message of (“There is no existing department with the name you entered") is sent.
* Otherwise, a message of (“Department is updated") is sent.

------------------------------------------------------------------------------------

20- delete department:

Functionality: deleting departments in faculties that already exist in the system.

Route: /hrDeleteDepartment

Request type: DELETE

Request body:{“facultyName":”Engineering" , “departmentName": ”MET"}

Response:

* If the “facultyName" is not found, a message of (“There is no faculty with that name") is sent.
* If the “departmentName" entered is not found, message of (“There is no existing department with the name you entered") is sent.
* Otherwise, a message of (“Department is deleted") is sent.

------------------------------------------------------------------------------------

21- add course

Functionality: adding courses under certain departments that already exist in the system.

Route: /hrAddCourse

Request type: POST

Request body:{“departmentName": ”MET",”courseName": ”CSEN505"}

Response:

* If the “departmentName" entered is not found, message of (“There is no existing department with the name you entered") is sent.
* If the “courseName" entered already exists under that department, message of (“The course already exists in that department!") is sent.
* Otherwise, a message of (“A course is added") is sent.

------------------------------------------------------------------------------------

22-update course:

Functionality: updating courses’ names under certain departments that already exist in the system.

Route: /hrUpdateCourse

Request type: PUT

Request body:{“facultyName": ”Engineering", “departmentName": ”MET",”courseName": ”CSEN505",”newCourseName": “CSEN701"}

Response:

* If the “facultyName" is not found, a message of (“There is no faculty with that name") is sent.
* If the “departmentName" entered is not found, message of (“There is no existing department with the name you entered") is sent.
* If the “courseName" is not found, a message of (“There is no course with that name")
* Otherwise, a message of (“The course is updated") is sent.

------------------------------------------------------------------------------------

23- delete course

Functionality: deleting courses in departments faculties that already exist in the system.

Route: /hrDeleteCourse

Request type: DELETE

Request body:{“facultyName":”Engineering" , “departmentName": ”MET" , “courseName": ”CSEN701"}

Response:

* If the “facultyName" is not found, a message of (“There is no faculty with that name") is sent.
* If the “departmentName" entered is not found, message of (“There is no existing department with the name you entered") is sent.
* If the “courseName" is not found, a message of (“There is no course with that name")
* Otherwise, a message of (“The course is deleted") is sent.

------------------------------------------------------------------------------------

24- add staff

Functionality: adding staff to the system.

Route: /hrAddStaff

Request type: POST

Request body:{“username":”HRTester",

”password":”pass123",

”role":”HR",

”gender":”female",

”displayName":”tester",

”imgPath":”<https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg>",

”dayOff":”Tuesday",”courseName":”CSEN101",”salary":”200",”faculty":”MET",”department":”officeLocation":”C7.101"}

Response:

* If the “username" given already exists, a message of (“Username taken") is sent.
* If added staff member’s “role" is “HR", and the “dayOff" is not set to “Saturday", a message of (“HR members always have Saturday as their dayoff!") is sent.
* If “courseName" is be assigned to any value except null, a message of (“you are not allowed to assign courses to new staff!") is sent.
* If “officeLocation" is not on the system, message of (“Location not found") is sent.
* If the type of“officeLocation" is not “Office", a message of (“The following location is not an office!") is sent.
* If the capacity of “officeLocation" is full, a message of (“the following location is full, please choose another one!") is sent.
* If “role" is not given, message of (“you have to specify the role!") is sent.
* Otherwise, a message of (“staff member is added") is sent.

------------------------------------------------------------------------------------

25- update staff

Functionality: updating staff members in the system.

Route: /hrUpdateStaff

Request type: PUT

Request body:{“username":”HRTester",”oldOffice" :”C7.101",”office":”C6.202",”role":”CC",”dayOff":”Monday",”faculty":”Engineering",”department":”MET",”gender":”female",”displayName":”tester"}

Response:

* If “username" is not found, a message of (“Cannot find staff member") is sent.
* If “oldOffice" or “office" are not found, a message of (“Location not found") is sent.
* If “office" is full, a message of (“Location is already full, please choose another one!") is sent.
* If an “HR" member tries to change his/her or other HR members “dayOff", a message of(“you can’t change your dayOff or other HRs dayOff") is sent.
* Otherwise, a message of (“staff updated!") is sent.

------------------------------------------------------------------------------------

26- delete staff.

Functionality: deleteing staff members in the system.

Route: /hrDeleteStaff

Request type: DELETE

Request body:{“username":”HRTester",”officeName":”C7.101"}

Response:

* If “username" is not found, a message of (“Cannot find staff member") is sent.
* If “officeName" does not match the “username" actual office, a message of (“Wrong Location") is sent.
* If “officeName"does not exist on the system, a message of (“Location not found") is sent.
* Otherwise, a message of (“Staff member deleted") is sent.

------------------------------------------------------------------------------------

27- update salary.

Functionality: updating staff members’ salary in the system.

Route: /hrUpdateSalary

Request type: PUT

Request body:{“username":”HRTester",”newSalary": 5000}

Response:

* If “username" is not found, a message of (“User not found") is sent.
* Otherwise, a message of (“Salary is updated") is sent.

------------------------------------------------------------------------------------

28- view any staff member attendance record

Functionality: Showing staff members’ attendance record in the system.

Route: /hrViewAttendance

Request type: GET

Request body:{“username":”mohamed.elsayedali2"}

Response: an array of objects showing the sign in and sign out time.

Example: [

{ "\_id": "5fe595d1f801123150f683cb",

"signIn": "2020-12-25T07:27:56.161Z",

"signOut": "2020-12-25T07:33:37.941Z" },

{ "\_id": "5fe5996d7b2cd31a28cc098e",

"signIn": "2020-12-25T07:35:42.613Z",

"signOut": "2020-12-25T07:49:01.560Z" }]

-----------------------------------------------------------------------------------

29- manually add a missing signIn/signOut record

Functionality: adding a signIn/Out record to the attendance in the system.

Route: /hrAddSignInSignOut

Request type: POST

Request body:{“staffUsername":”mohamed.elsayedali2",”signIn": ”December 29, 2020 6:00:00", “signOut": ”December 29, 2020 22:00:00"}

Response:

* If “staffUsername" is not found, a message of (“User does not exist") is sent.
* Otherwise, a message of (“Record added successfully") is sent.

------------------------------------------------------------------------------------

30 – viewing staff members with missing hours and missing days.

Functionality: adding a signIn/Out record to the attendance in the system.

Route: /hrViewMissingHoursMissingDays

Request type: GET

Response example:

{ "profile": { "gender": "male", "displayName": "Helmy", "imgPath": "<https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg>" }, "date": "2020-12-14T19:05:08.783Z", "missingDays": [ "2020-12-25T07:27:56.161Z" ], "\_id": "5fd7b76481344639f46da7b6", "attendance": [ { "\_id": "5fe595d1f801123150f683cb", "signIn": "2020-12-25T07:27:56.161Z", "signOut": "2020-12-25T07:33:37.941Z" }, { "\_id": "5fe5996d7b2cd31a28cc098e", "signIn": "2020-12-25T07:35:42.613Z", "signOut": "2020-12-25T07:49:01.560Z" }, { "\_id": "5fe59ab626f4540e603790d7", "signIn": "2020-12-25T07:54:26.891Z", "signOut": "2020-12-25T07:54:30.950Z" }], "id": "hr-1", "username": "mohamed.elsayedali2", "email": "[mohamed.elsayedali2@guc.edu.eg](mailto:mohamed.elsayedali2@guc.edu.eg)", "password": "$2a$10$WaAQvV6hp20yz8PE6mcvjOGw5oDLgMnZ/OmheJ93tiiZhscs8J3KS", "role": "HR", "signedIn": false, "schedule": [ { "\_id": "5fdd091b1594c745d09840a5", "day": "Monday", "slots": [ { "\_id": "5fdd091b1594c745d09840a6", "course": "csen703", "start": "11:45" }, { "\_id": "5fdf7fda41b2f62240c77a7e", "course": "csen703", "location": "c3.101", "start": "81:15", "end": "9:45" },]], "\_\_v": 0, "department": "dep1", "accidentalDays": 6, "annualBalance": 7.5, "annualLastAdded": "2020-12-15T20:56:03.787Z", "dayOff": "Saturday", "notifications": 7, "signIn": "2020-12-25T10:31:24.198Z", "extraMinutes": 216, "minutes": 504, "missingMinutes": 0}]

------------------------------------------------------------------------------------

**HOD Requests:**

31-

**Request:**

Functionality: viewing all the staff in the department of the signed in head of department member.

Route: /viewStaffInDepartment

Request Type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality: viewing all the staff in the department of the signed in head of department member.

Route: /viewStaffInDepartment

Request type: GET

Response ex: [

{

"date": "2020-12-25T09:09:05.513Z",

"attendance": [],

"missingDays": [],

"courses": [],

"\_id": "5fd9f8e8fc2b27eae9a8e740",

"name": "pola",

"schedule": []

}

]

**32-**

**Request**:

Functionality: Viewing the staff members who teach the course that will be parsed in the body of the request

Route: /viewStaffInCourse

Request type: GET

Request header : {“auth-token” : “the token we got from login route”, “Content-Type”:”application/json”}

Request Body : {“course” : “cs1”}

**Response**:

Functionality: Viewing the staff members who teach the course that will be parsed in the body of the request

Route: /viewStaffInCourse

Request type: GET

Response ex: [

{

"profile": {

"gender": "male"

},

"date": "2020-12-16T12:46:27.377Z",

"attendance": [],

"missingDays": [],

"courses": [

"cs1",

"iet1"

],

"\_id": "5fda01a338b5e71ba479e099",

"id": "43-17507",

"username": "Haytham",

"email": "[Haytham@guc.edu.eg](mailto:Haytham@guc.edu.eg)",

"password": "$2a$10$irLxMM4Cbkx4FPvgqmrD/OLoMLgWqouhE6UYWrgTx0iQILJe9EQeG",

"role": "HR",

"signedIn": false,

"dayOff": "sunday",

"faculty": "Engineering",

"department": "MET",

"salary": 3000,

"schedule": [],

"\_\_v": 0

}

]

**33-**

**Request**:

Functionality: Viewing the day of for all the staff in the department of the signed in Head Of Department

Route: /viewDayOffAllStaff

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}.

**Response**:

Functionality: Viewing the day of for all the staff in the department of the signed in Head Of Department

Route: /viewDayOffAllStaff

Request type: GET

Response ex: [

{

"name": "jordan",

"department": "MET",

"dayOff": "saturday"

},

{

"name": "Haytham",

"department": "MET",

"dayOff": "sunday"

},

{

"name": "Haytham",

"department": "MET",

"dayOff": "sunday"

},

{

"name": "pola5",

"department": "MET",

"dayOff": "saturdan"

}

]

**34-**

**Request**:

Functionality: Viewing the day of for a single staff in the department of the signed in Head Of Department

Route: /viewDayOffSingleStaff

Request type: GET

Request header : {“auth-token” : “the token we got from login route”, “Content-Type”:”application/json”}

Request Body : {“username” : “jordan”}.

**Response**:

Functionality: Viewing the day of for the given staff in the department of the signed in Head Of Department

Route: /viewDayOffSingleStaff

Request type: GET

Response ex:[

{

"name": "jordan",

"dayOff": "saturday"

}

]

**35-**

**Request**:

Functionality: Assign a course instructor to a given course in the department of the signed in head of department

Route: /assignCourseInstructor

Request type: POST

Request header : {“auth-token” : “the token we got from login route”, “Content-Type”:”application/json”}

Request Body : {“courseName” : “cs1”, “instructor”:”Cardone”}.

**Response**:

Functionality: Assign a course instructor to a given course in the department of the signed in head of department

Route: /assignCourseInstructor

Request type: POST

Response ex:[

{

"name": "cs1",

"instructor": "Cardone",

"coordinator": "pola2",

"slots": [

{

"\_id": "5fda2e74fa8ca00e44cc63e7",

"teacher": "pola3",

"location": "b1",

"day": "saturday",

"start": "first",

"end": "second"

}

]

}

]



The instructor of the course named “cs1” was changed to “Cardone”.

**36-**

**Request**:

Functionality: Update a course instructor to a given course in the department of the signed in head of department

Route: /updateCourseInstructor

Request type: POST

Request header : {“auth-token” : “the token we got from login route”, “Content-Type”:”application/json”}

Request Body : {“courseName” : “cs1”, “instructor”:”GC”}.

**Response**:

Functionality: Update a course instructor to a given course in the department of the signed in head of department

Route: /updateCourseInstructor

Request type: POST

Response ex:

[

{

"name": "cs1",

"instructor": "GC",

"coordinator": "pola2",

"slots": [

{

"\_id": "5fda2e74fa8ca00e44cc63e7",

"teacher": "pola3",

"location": "b1",

"day": "saturday",

"start": "first",

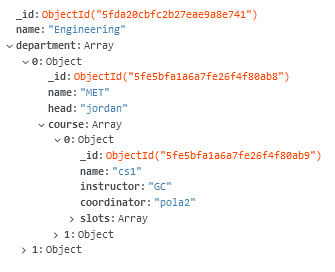
"end": "second"

}

]

}

]



The instructor of the course named “cs1” was changed to “GC”.

**37-**

**Request**:

Functionality: deleting the course instructor in the given course in the body of the request , that course should be in the same department as the signed in head of department.

Route: /deleteCourseInstructor

Request type: POST

Request header : {“auth-token” : “the token we got from login route”, “Content-Type”:”application/json”}

Request Body : {“courseName” : “cs1”}.

**Response**:

Functionality: deleting the course instructor in the given course in the body of the request , that course should be in the same department as the signed in head of department.

Route: /deleteCourseInstructor

Request type: POST

Response ex:[

{

"name": "cs1",

"instructor": "", // the instructor name became an empty String

"coordinator": "pola2",

"slots": [

{

"\_id": "5fda2e74fa8ca00e44cc63e7",

"teacher": "pola3",

"location": "b1",

"day": "saturday",

"start": "first",

"end": "second"

}

]

}

]

**38-**

**Request**:

Functionality:Views the requests sent to the signed in head of department.

Route: /viewRequests

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality:Views the requests sent to the signed in head of department.

Route: /viewRequests

Request type: GET

Response ex:[

[],

[

{

"\_id": "5fdb2f7cfaee86c255b88d82",

"sender": "pola3",

"recipient": "jordan",

"day": "saturdan",

"reason": "sick",

"status": "R",

"rejectionReason": "his reason is a bullshit"

},

{

"\_id": "5fdb3024faee86c255b88d84",

"sender": "pola5",

"recipient": "jordan",

"day": "saturdan",

"reason": "sick",

"status": "A"

}

]

]

**39-**

**Request**:

Functionality:Views which staff member teaches which slots of course offered by the signed in head of departments

Route: /viewTeachingAssignments

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality:Views which staff member teaches which slots of course offered by the signed in head of departments

Route: /viewTeachingAssignments

Request type: GET

Response ex:[

{

"courseName": "cs1",

"Assignments": [

{

"\_id": "5fda2e74fa8ca00e44cc63e7",

"teacher": "pola3",

"location": "b1",

"day": "saturday",

"start": "first",

"end": "second"

}

]

},

{

"courseName": "cs2",

"Assignments": [

{

"\_id": "5fda2e74fa8ca00e44cc63e9",

"teacher": "pola6",

"location": "c1",

"day": "sunday",

"start": "first",

"end": "second"

}

]

}

]

**40-**

**Request**:

Functionality:Views the coverage of each course in the department of the signed in head of departments.

Route: /viewCoursesCoverage

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Functionality:Views the coverage of each course in the department of the signed in head of departments.

Route: /viewCoursesCoverage

Request type: GET

Response ex:[

{

"courseName": "cs1",

"coverage": 1

},

{

"courseName": "cs2",

"coverage": 1

}

]

**41-**

**Request**:

Functionality : Rejects a day Off request sent by a staff member.

Route: /rejectDayOffRequest

Request type: POST

Request header : {“auth-token” : “the token we got from login route”,”Content-Type”: “application/json”}

Request Body : {“id” : “5fdb2f7cfaee86c255b88d82

”, “rejectionReason” : “his reason is not reasonable”}.

**Response**:

Functionality : Rejects a day Off request sent by a staff member.

Route: /rejectDayOffRequest

Request type: POST

Response ex:{

"\_id": "5fdb2f7cfaee86c255b88d82",

"sender": "pola3",

"recipient": "jordan",

"day": "saturdan",

"reason": "sick",

"status": "R",

"rejectionReason": "his reason is not reasonable"

}

**42-**

**Request**:

Functionality : Rejects a Leave request sent by a staff member.

Route: /rejectLeaveRequest

Request type: POST

Request header : {“auth-token” : “the token we got from login route”,”Content-Type”: “application/json”}

Request Body : {“id” : “5fdb2f7cfaee86c255b88d82

”, “rejectionReason” : “his reason is unreasonable”}.

**Response**:

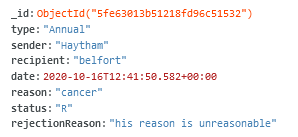
Functionality : Rejects a leave request sent by a staff member.

Route: /rejectLeaveRequest

Request type: POST.

Response Example:

//this is the request after it was rejected



//this is the notification of rejecting the request



**43-**

**Request**:

Functionality : Accepts a day off request sent by a staff member.

Route: /acceptDayOffRequest

Request type: POST

Request header : {“auth-token” : “the token we got from login route”,”Content-Type”: “application/json”}

Request Body : {“id” : “5fdb3024faee86c255b88d84”}.

**Response**:

Functionality : Accepts a day off request sent by a staff member.

Route: /acceptDayOffRequest

Request type: POST.

Response ex:{

"profile": {

"gender": "male"

},

"date": "2020-12-25T17:20:04.690Z",

"attendance": [],

"missingDays": [],

"courses": [],

"\_id": "5fdb857d54abc651f08b7353",

"id": "43-17507",

"username": "pola5",

"email": "[pola5@guc.edu.eg](mailto:pola5@guc.edu.eg)",

"password": "haybro",

"role": "HR",

"dayOff": "saturdan",

"salary": 3000,

"faculty": "Engineering",

"department": "MET",

"schedule": []

}

**44-**

**Request**:

Functionality : Accepts a Leave request sent by a staff member.

Route: /acceptLeaveRequest

Request type: POST

Request header : {“auth-token” : “the token we got from login route”,”Content-Type”: “application/json”}

Request Body : {“id” : “5fdb2f7cfaee86c255b88d82

”}.

**Response**:

Functionality : Accepts a Leave request sent by a staff member.

Route: /acceptLeaveRequest

Request type: POST

Response ex:

//this is the request after it was approved

{

"documents": [],

"\_id": "5fe62925b51218fd96c5152f",

"type": "Annual",

"sender": "grant",

"recipient": "belfort",

"date": "2020-10-16T12:41:50.582Z",

"reason": "cancer",

"status": "A",

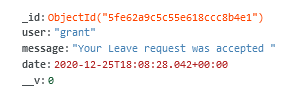
"rejectionReason": ""

}

//this is the staff after the request was approved and his annual balance and notifications changed



//this is the new notification added after acceptance



=================================================================================

Course Instructor functions:-

45-

**Request**:

Functionality: View the coverage of course(s) he/she is assigned to.

Route: /viewCourseCoverage

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body: {“course” : “csen704”}

Routes we use to get to this route:

1. /viewCourses

**Response**:

ex. “Course coverage is: 50%”

46-

**Request**:

Functionality: View the slots' assignment of course(s) he/she is assigned to.

Route: /viewSlotsAssignment

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body: {“course” : “csen703”}

Routes we use to get to this route:

1. /viewCourses

**Response**:

ex.

[ {

"\_id": "5fe521b55a279913705dd0b9",

"location": "C7.301",

"day": "Sunday",

"start": "8:15",

"end": "9:45",

"teacher": "teacher1"

},

{

"\_id": "5fe521b55a279913705dd0ba",

"location": "C7.209",

"day": "Monday",

"start": "10:00",

"end": "11:30"

}

]

47-

**Request:**

Functionality: Assign an academic member to an unassigned slots in course(s) he/she is assigned to.

Route: /assignToSlot

Request type: POST

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{ "course": "csen704",

"teacher": "teacher2",

"location": "C7.205",

"day": "Sunday",

"start": "8:15",

"end": "9:45"

}

Routes we use to get to this route:

1. /viewCourses
2. /viewUnassignedSlots
3. /viewStaffInCourse

**Response:**

If the teacher we want to assign is free:-

* “Slot has been successfully assigned.”
* The schedule of the assigned teacher is updated in addition to course slots.

If the teacher we want to assign is not free:-

* “The chosen teacher is not available for this slot.”

48-

**Request:**

Functionality: Update assignment of academic member in course(s) he/she is assigned to.

Route: /updateSlotAssignment

Request type: POST

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"course": "csen704",

"oldteacher": "teacher2",

"newteacher": "teacher1",

"location": "C7.205",

"day": "Sunday",

"start": "8:15",

"end": "9:45"

}

Routes we use to get to this route:

1. /viewCourses
2. /viewAssignedSlots
3. /viewStaffInCourse

**Response:**

If oldteacher = newteacher :-

* ”Please choose two different teachers.”

if newteacher is available:-

* ”Slot has been successfully updated.”
* The schedules of both newteacher and oldteacher are updated in addition to course slots.

if newteacher is unavailable:-

* “New teacher unavailable for this slot.”

49-

**Request:**

Functionality:-

* Delete assignment of academic member in course(s) he/she is assigned to.
* Remove an assigned academic member in course(s) he/she is assigned to.

Route: /removeSlotAssignment

Request type: DELETE

Request header: {“auth-token” : “the token we got from login route”}

Request type:

{

"course": "csen704",

"teacher": "teacher1",

"location": "C7.205",

"day": "Sunday",

"start": "8:15",

"end": "9:45"

}

Routes we use to get to this route:

1. /viewCourses
2. /viewAssignedSlots
3. /viewStaffInCourse

**Response:**

* “Slot assignment has been successfully removed.”
* The schedule of the removed teacher is updated in addition to course slots.

50-

**Request:**

Functionality: Assign an academic member in each of his/her course(s) to be a course coordinator.

Route: /assignCoordinator

Request type: POST

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"coordinatorName": "teacher1",

"course": "csen704"

}

Routes we use to get to this route:

1. /viewCourses
2. /viewStaffInCourse

**Response:**

* “Coordinator has been successfully assigned.”
* The role of the newly assigned coordinator is changed to “CC”.

51-

**Request:**

Functionality: View courses the instructor is assigned to.

Route: /viewCourses

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

**Response:** ex.

**[**

**"csen703",**

**"csen704"**

**]**

52-

**Request:**

Functionality: View course slots that are unassigned.

Route: /viewUnassignedSlots

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"course": "csen703"

}

Routes we use to get to this route:

1. /viewCourses

**Response:** ex.

**[**

**{**

**"\_id": "5fe521b55a279913705dd0ba",**

**"location": "C7.209",**

**"day": "Monday",**

**"start": "10:00",**

**"end": "11:30"**

**}**

**]**

53-

**Request:**

Functionality: View course slots that are assigned.

Route: /viewAssignedSlots

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"course": "csen703"

}

Routes we use to get to this route:

1. /viewCourses

**Response:** ex.

[

{

"\_id": "5fe521b55a279913705dd0b9",

"location": "C7.301",

"day": "Sunday",

"start": "8:15",

"end": "9:45",

"teacher": "teacher1"

}

]

54-

**Request:**

Functionality: View all the staff in a given course.

Route: /viewStaffInCourse

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"course": "csen703"

}

Routes we use to get to this route:

1. /viewCourses

**Response:** ex.

[

{

"profile": {

"gender": "female",

"displayName": "Helmy",

"imgPath": "<https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg>"

},

"date": "2020-12-14T19:05:08.783Z",

"attendance": [],

"missingDays": [

"2020-12-25T07:27:56.161Z"

],

"courses": [

"csen703"

],

"\_id": "5fd7b76481344639f46da7b6",

"id": "hr-1",

"username": "mohamed.elsayedali2",

"email": "[mohamed.elsayedali2@guc.edu.eg](mailto:mohamed.elsayedali2@guc.edu.eg)",

"password": "$2a$10$WaAQvV6hp20yz8PE6mcvjOGw5oDLgMnZ/OmheJ93tiiZhscs8J3KS",

"role": "CI",

"signedIn": false,

"schedule": [

{

"\_id": "5fdd091b1594c745d09840a5",

"day": "Monday",

"slots": [

{

"\_id": "5fdd091b1594c745d09840a6",

"course": "csen703",

"start": "11:45"

},

{

"\_id": "5fdf7fda41b2f62240c77a7e",

"course": "csen703",

"location": "c3.101",

"start": "81:15",

"end": "9:45"

},

{

"\_id": "5fdfbdc8051b0218b816fb7b",

"course": "csen703",

"location": "c3.101",

"start": "81:15",

"end": "9:45"

},

{

"\_id": "5fdfc07be168902fb88e3cf3",

"course": "csen703",

"location": "c3.101",

"start": "8:15",

"end": "9:45"

}

]

},

{

"\_id": "5fdd0a4c2ea5633d188cfc9e",

"day": "Tuesday",

"slots": []

},

{

"\_id": "5fe54292c04c3e33c0c665b8",

"day": "Wednesday",

"slots": []

},

{

"\_id": "5fe543a0c04c3e33c0c665c8",

"day": "Thursday",

"slots": []

}

],

"\_\_v": 0,

"department": "dep1",

"accidentalDays": 6,

"annualBalance": 7.5,

"annualLastAdded": "2020-12-15T20:56:03.787Z",

"dayOff": "Saturday",

"notifications": 7,

"signIn": "2020-12-25T10:31:24.198Z",

"extraMinutes": 216,

"minutes": 504,

"missingMinutes": 0

},

{

"profile": {

"gender": "female",

"displayName": "Helmy",

"imgPath": "<https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg>"

},

"date": "2020-12-14T19:05:08.783Z",

"attendance": [],

"missingDays": [],

"courses": [

"csen703",

"csen704"

],

"\_id": "5fe5cbc76e354e3afc6c2098",

"id": "hr-1",

"username": "instructor1",

"email": "[mohamed.elsayedali2@guc.edu.eg](mailto:mohamed.elsayedali2@guc.edu.eg)",

"password": "$2a$10$WaAQvV6hp20yz8PE6mcvjOGw5oDLgMnZ/OmheJ93tiiZhscs8J3KS",

"role": "CI",

"signedIn": false,

"schedule": [

{

"\_id": "5fdd091b1594c745d09840a5",

"day": "Monday",

"slots": [

{

"\_id": "5fdd091b1594c745d09840a6",

"course": "csen703",

"start": "11:45"

},

{

"\_id": "5fdd091b1594c745d09840a7",

"course": "csen704",

"replacement": "2020-12-18T19:55:07.921Z"

},

{

"\_id": "5fdd091b1594c745d09840a8",

"course": "csen705",

"start": "8:15",

"replacement": "2020-12-22T00:00:00.000Z"

},

{

"\_id": "5fdf7fda41b2f62240c77a7e",

"course": "csen703",

"location": "c3.101",

"start": "81:15",

"end": "9:45"

},

{

"\_id": "5fdfbdc8051b0218b816fb7b",

"course": "csen703",

"location": "c3.101",

"start": "81:15",

"end": "9:45"

},

{

"\_id": "5fdfc07be168902fb88e3cf3",

"course": "csen703",

"location": "c3.101",

"start": "8:15",

"end": "9:45"

}

]

},

{

"\_id": "5fdd0a4c2ea5633d188cfc9e",

"day": "Tuesday",

"slots": [

{

"\_id": "5fdd0a4c2ea5633d188cfc9f",

"course": "csen703",

"start": "8:15",

"replacement": "2020-12-22T00:00:00.000Z"

}

]

},

{

"\_id": "5fe54292c04c3e33c0c665b8",

"day": "Wednesday",

"slots": []

},

{

"\_id": "5fe543a0c04c3e33c0c665c8",

"day": "Thursday",

"slots": []

}

],

"\_\_v": 0,

"department": "department1",

"accidentalDays": 6,

"annualBalance": 5,

"annualLastAdded": "2020-11-15T20:56:03.787Z",

"dayOff": "Saturday",

"notifications": 7,

"signIn": "2020-12-25T10:31:24.198Z",

"extraMinutes": 0,

"minutes": 504,

"missingMinutes": 0

},

{

"date": "2020-12-25T11:56:00.379Z",

"attendance": [],

"missingDays": [],

"courses": [

"csen704",

"csen703"

],

"\_id": "5fe5d35067efa448c40268db",

"username": "teacher1",

"faculty": "faculty1",

"department": "department1",

"schedule": [

{

"\_id": "5fe60f6adc51b635acd1c0fa",

"day": "Sunday",

"slots": [

{

"\_id": "5fe60f6adc51b635acd1c0f9",

"course": "csen703",

"location": "C7.301",

"start": "8:15",

"end": "9:45"

}

]

},

{

"\_id": "5fe61f85dc51b635acd1c119",

"day": "Monday",

"slots": []

},

{

"\_id": "5fe5e1e7fe5d3a2384317972",

"day": "Tuesday",

"slots": [

{

"\_id": "5fe60f14dc51b635acd1c0f1",

"course": "csen704",

"location": "C5.105",

"start": "11:45",

"end": "1:15"

}

]

},

{

"\_id": "5fe5e1d4fe5d3a2384317968",

"day": "Wednesday",

"slots": []

},

{

"\_id": "5fe5e1d4fe5d3a2384317969",

"day": "Thursday",

"slots": []

}

],

"\_\_v": 0,

"role": "CC"

}

]

55-

**Request:**

Functionality: View all the staff in his/her department.

Route: /viewStaffInDepartment

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"department": "department1"

}

**Response:**

**[**

**{**

**"profile": {**

**"gender": "female",**

**"displayName": "Helmy",**

**"imgPath": "**[**https://image.freepik.com/free-photo/image-human-brain\_99433-298.jpg**](https://image.freepik.com/free-photo/image-human-brain_99433-298.jpg)**"**

**},**

**"date": "2020-12-14T19:05:08.783Z",**

**"attendance": [],**

**"missingDays": [],**

**"courses": [**

**"csen703",**

**"csen704"**

**],**

**"\_id": "5fe5cbc76e354e3afc6c2098",**

**"id": "hr-1",**

**"username": "instructor1",**

**"email": "**[**mohamed.elsayedali2@guc.edu.eg**](mailto:mohamed.elsayedali2@guc.edu.eg)**",**

**"password": "$2a$10$WaAQvV6hp20yz8PE6mcvjOGw5oDLgMnZ/OmheJ93tiiZhscs8J3KS",**

**"role": "CI",**

**"signedIn": false,**

**"schedule": [**

**{**

**"\_id": "5fdd091b1594c745d09840a5",**

**"day": "Monday",**

**"slots": [**

**{**

**"\_id": "5fdd091b1594c745d09840a6",**

**"course": "csen703",**

**"start": "11:45"**

**},**

**{**

**"\_id": "5fdd091b1594c745d09840a7",**

**"course": "csen704",**

**"replacement": "2020-12-18T19:55:07.921Z"**

**},**

**{**

**"\_id": "5fdd091b1594c745d09840a8",**

**"course": "csen705",**

**"start": "8:15",**

**"replacement": "2020-12-22T00:00:00.000Z"**

**},**

**{**

**"\_id": "5fdf7fda41b2f62240c77a7e",**

**"course": "csen703",**

**"location": "c3.101",**

**"start": "81:15",**

**"end": "9:45"**

**},**

**{**

**"\_id": "5fdfbdc8051b0218b816fb7b",**

**"course": "csen703",**

**"location": "c3.101",**

**"start": "81:15",**

**"end": "9:45"**

**},**

**{**

**"\_id": "5fdfc07be168902fb88e3cf3",**

**"course": "csen703",**

**"location": "c3.101",**

**"start": "8:15",**

**"end": "9:45"**

**}**

**]**

**},**

**{**

**"\_id": "5fdd0a4c2ea5633d188cfc9e",**

**"day": "Tuesday",**

**"slots": [**

**{**

**"\_id": "5fdd0a4c2ea5633d188cfc9f",**

**"course": "csen703",**

**"start": "8:15",**

**"replacement": "2020-12-22T00:00:00.000Z"**

**}**

**]**

**},**

**{**

**"\_id": "5fe54292c04c3e33c0c665b8",**

**"day": "Wednesday",**

**"slots": []**

**},**

**{**

**"\_id": "5fe543a0c04c3e33c0c665c8",**

**"day": "Thursday",**

**"slots": []**

**}**

**],**

**"\_\_v": 0,**

**"department": "department1",**

**"accidentalDays": 6,**

**"annualBalance": 5,**

**"annualLastAdded": "2020-11-15T20:56:03.787Z",**

**"dayOff": "Saturday",**

**"notifications": 7,**

**"signIn": "2020-12-25T10:31:24.198Z",**

**"extraMinutes": 0,**

**"minutes": 504,**

**"missingMinutes": 0**

**},**

**{**

**"date": "2020-12-25T11:56:00.379Z",**

**"attendance": [],**

**"missingDays": [],**

**"courses": [**

**"csen704",**

**"csen703"**

**],**

**"\_id": "5fe5d35067efa448c40268db",**

**"username": "teacher1",**

**"faculty": "faculty1",**

**"department": "department1",**

**"schedule": [**

**{**

**"\_id": "5fe60f6adc51b635acd1c0fa",**

**"day": "Sunday",**

**"slots": [**

**{**

**"\_id": "5fe60f6adc51b635acd1c0f9",**

**"course": "csen703",**

**"location": "C7.301",**

**"start": "8:15",**

**"end": "9:45"**

**}**

**]**

**},**

**{**

**"\_id": "5fe61f85dc51b635acd1c119",**

**"day": "Monday",**

**"slots": []**

**},**

**{**

**"\_id": "5fe5e1e7fe5d3a2384317972",**

**"day": "Tuesday",**

**"slots": [**

**{**

**"\_id": "5fe60f14dc51b635acd1c0f1",**

**"course": "csen704",**

**"location": "C5.105",**

**"start": "11:45",**

**"end": "1:15"**

**}**

**]**

**},**

**{**

**"\_id": "5fe5e1d4fe5d3a2384317968",**

**"day": "Wednesday",**

**"slots": []**

**},**

**{**

**"\_id": "5fe5e1d4fe5d3a2384317969",**

**"day": "Thursday",**

**"slots": []**

**}**

**],**

**"\_\_v": 0,**

**"role": "CC"**

**},**

**{**

**"date": "2020-12-25T11:56:00.379Z",**

**"attendance": [],**

**"missingDays": [],**

**"courses": [**

**"csen704"**

**],**

**"\_id": "5fe5d35067efa448c40268dc",**

**"username": "teacher2",**

**"faculty": "faculty1",**

**"department": "department1",**

**"schedule": [**

**{**

**"\_id": "5fe60f84dc51b635acd1c109",**

**"day": "Sunday",**

**"slots": [**

**{**

**"\_id": "5fe60f84dc51b635acd1c108",**

**"course": "csen704",**

**"location": "C7.205",**

**"start": "8:15",**

**"end": "9:45"**

**}**

**]**

**},**

**{**

**"\_id": "5fe5e1d4fe5d3a238431796b",**

**"day": "Monday",**

**"slots": []**

**},**

**{**

**"\_id": "5fe5e1d4fe5d3a238431796c",**

**"day": "Tuesday",**

**"slots": []**

**},**

**{**

**"\_id": "5fe5e1d4fe5d3a238431796d",**

**"day": "Wednesday",**

**"slots": []**

**},**

**{**

**"\_id": "5fe5e1d4fe5d3a238431796e",**

**"day": "Thursday",**

**"slots": []**

**}**

**],**

**"\_\_v": 0**

**}**

**]**

56-

**Request:**

Functionality: Assign an academic member to an instructor’s course.

Route: /assignAcademicMember

Request type: POST

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"teacher": "teacher1",

"course": "csen703"

}

Routes we use to get to this route:

1. /viewCourses

**Response:**

* “Academic member has been successfully added to your course.”

57-

**Request:**

Functionality: View course slots.

Route: /viewCourseSlots

Request type: GET

Request header: {“auth-token” : “the token we got from login route”}

Request body:

{

"course": "csen703"

}

Routes we use to get to this route:

1. /viewCourses

**Response:**

**[**

**{**

**"\_id": "5fe521b55a279913705dd0b9",**

**"location": "C7.301",**

**"day": "Sunday",**

**"start": "8:15",**

**"end": "9:45",**

**"teacher": "teacher1"**

**},**

**{**

**"\_id": "5fe521b55a279913705dd0ba",**

**"location": "C7.209",**

**"day": "Monday",**

**"start": "10:00",**

**"end": "11:30"**

**}**

**]**

===================================================

**4.3 Course Coordinator Functionalities**

58–

**Request**:

Functionality: View “slot linking" request(s) from academic members linked to his/her course.

**Route: /viewSlotLinkingRequests**

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Array of requests . Example of a single request:

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"course": "csen703",

"day": "Monday",

"start": "8:15",

"end": "9:45",

"location": "c3.101",

"status": "P",

"\_\_v": 0

}

59–

**Request**:

Functionality: Reject “slot linking” request

**Route: /rejectSlotLinking**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”: “5fdf377e8fedba3350b43ebf”}

**Response**:

The request, ex : {

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"course": "csen703",

"day": "Monday",

"start": "8:15",

"end": "9:45",

"location": "c3.101",

"status": "R",

"\_\_v": 0

}

60–

**Request**:

Functionality: Accept “slot linking” request

**Route: /acceptSlotLinking**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”: “5fdf377e8fedba3350b43ebf”}

**Response**:

The request, ex : {

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"course": "csen703",

"day": "Monday",

"start": "8:15",

"end": "9:45",

"location": "c3.101",

"status": "A",

"\_\_v": 0

}

61–

**Request**:

Functionality: Add/update/delete course slot(s) in his/her course. (4 Routes)

**Route 1: /viewCoursesOfCoordinator**

Explanation: Gets all courses which the user is assigned as a coordinator.

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Array of course names. Example of a single element:

{“csen703”}

**Route 2: /addCourseSlot**

Explanation: Adds a slot in a course given a course name, a day, a start, end time and a location.

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “courseName”: “csen703”, “day”:”Monday”, “start”:”8:15”, “end”:”9:45”, “location”: “c7.205”}

**Response**:

“Slot added successfully”

**Route 3: /updateCourseSlot**

Explanation: updates a slot in a course given a course name, old day, old start time, old location, new Teacher, new day, new start time, new end time and new location.

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “courseName”: “csen703”, “oldSlotDay”:”Monday”, “oldSlotStart”:”8:15”, “oldLocation”:”c7.205”, “newTeacher”: “Soubra”, “newSlotDay”:”Tuesday”, “newSlotStart”:”10:00”, “newSlotEnd”:”11:30”, “newLocation”:”c5.102”}

**Response**:

“Slot updated successfully”

**Route 4: /deleteCourseSlot**

Explanation: deletes a slot in a course given a course name, a day, a start time and location.

Request type: DELETE

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “courseName”: “csen703”, “slotDay”:”Monday”, “slotStart”:”8:15”, “location”:”c7.205”}

**Response**:

“Slot deleted successfully”

**4.4 Academic member Functionalities**

62 –

**Request**:

Functionality: View their schedule.

**Route: /viewSchedule**

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Array of day objects with nested slots. Example of a single object:

{

"\_id": "5fdd091b1594c745d09840a5",

"day": "Monday",

"slots": [

{

"\_id": "5fdf7fda41b2f62240c77a7e",

"course": "csen703",

"location": "c3.101",

"start": "8:15",

"end": "9:45"

}

]

}

63–

**Request**:

Functionality: Send replacement. (3 Routes)

**Route 1: /viewSlotsOnDate**

Request type: GET

Explanation: When given a date, it retrieves the slots of the personal schedule of the sender on the day that corresponds to the given date.

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “date”: “2020-12-25”}

**Response**:

Array of slots. Example of a single slot:

{

"\_id": "5fdf7fda41b2f62240c77a7e",

"course": "csen703",

"location": "c3.101",

"start": "8:15",

"end": "9:45"

}

**Route 2: /viewStaffInCourse**

Explanation: from the results of the previous route we pick a slot and pass the course name to this route to retrieve the staff that teach this course.

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “course”: “csen703”}

**Response**:

Array of usernames. Example of a single element: “Soubra”

**Route 3: /sendReplacement**

Explanation: Provided with a date, a slot and a staff from the previous routes. Now we can send a replacement request.

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “senderDate”: “2020-12-25”, “course”:”csen703”, “start”:”8:15”, “end”:”9:45”, “location”:”c7.205”, “recipient”:”Soubra”}

**Response**:

Returns request created . Example of a created request:

{ "\_id": "5fe5f60edf1914246cb5e358",

"sender": "staff1",

"recipient": "staff2",

"status": "P",

"slot": {

"course": "csen703",

"start": "8:15",

"end": "9:45",

"location": "c7.205",

"replacement": "2020-12-20T22:00:00.000Z"},

"\_\_v": 0}

64–

Functionality: View sent “replacement” requests.

**Route: /viewSentRequests**

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “type”: “replacement”, “status”:””}

//status takes one of 3 values (“R”,”P”,”A”). If left empty then all sent “replacement” requests will be retrieved.

**Response**:

Array of requests. Example of a single request:

{ "\_id": "5fe5f60edf1914246cb5e358",

"sender": "staff1",

"recipient": "staff2",

"status": "P",

"slot": {

"course": "csen703",

"start": "8:15",

"end": "9:45",

"location": "c7.205",

"replacement": "2020-12-20T22:00:00.000Z"},

"\_\_v": 0}

65–

Functionality: View received “replacement” requests.

**Route: /viewReceivedReplacement**

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Array of requests. Example of a single request:

{ "\_id": "5fe5f60edf1914246cb5e358",

"sender": "staff1",

"recipient": "staff2",

"status": "P",

"slot": {

"course": "csen703",

"start": "8:15",

"end": "9:45",

"location": "c7.205",

"replacement": "2020-12-20T22:00:00.000Z"},

"\_\_v": 0}

66–

Functionality: reject a “replacement” request.

**Route: /rejectReplacement**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”: “5fdf377e8fedba3350b43ebf”}

**Response**:

“Request rejected successfully”

67–

Functionality: accept a “replacement” request.

**Route: /acceptReplacement**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”: “5fdf377e8fedba3350b43ebf”}

**Response**:

“Request accepted successfully”

68–

Functionality: Send a “slot linking” request. (2 Routes)

**Route 1: /viewCoursesInDepartment**

Request type: GET

Explanation: Gets all courses offered by the department of the user, with slots nested inside each course.

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Array of course objects having slots array nested inside. Example of a single course object:

{

“name”: “csen703”,

“instructor”: [“Soubra”, “Slim”],

“coordinator”: ”staff1”,

“slots”: [object]

}

slot object example =>

{

“teacher”: “Soubra”,

“location”: “c7.205”,

“day”: ”Monday”,

“start”: “8:15”,

“end”: “9:45”

}

**Route 2: /sendSlotLinking**

Request type: POST

Explanation: we choose a course and a slot from the previous route.

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “course”: “csen703”, “day”:”Monday”, “start”:”8:15”, “end”:”9:45”, “location”:”c7.205”}

**Response**:

Request created, Example of a “slot linking” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"course": "csen703",

"day": "Monday",

"start": "8:15",

"end": "9:45",

"location": "c3.101",

"status": "R",

"\_\_v": 0

}

69–

Functionality: Send a “change day-off” request.

**Route: /sendChangeDayOff**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “day”: “Monday”, “reason”:”reason1”}

**Response**:

Request created, Example of a “change day-off” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"day": "Monday",

"reason": "reason1",

"status": "R",

"rejectionReason": "reason2",

"\_\_v": 0

}

70–

Functionality: Send an “annual leave” request.

**Route: /sendAnnualRequest**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “date”: “2020-12-25”, “reason”:”reason1”, “replacers”: [“staff1”, “staff2”]}

**Response**:

Request created, Example of an “annual leave” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"type": "annual",

"date": "2020-12-25T22:00:00.000Z",

"status": "P",

"reason": "reason1",

“replacers”: [“staff1”, “staff2”],

“documents”: [],

"\_\_v": 0

}

71–

Functionality: Send an “accidental leave” request.

**Route: /sendAccidentalRequest**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “date”: “2020-12-25”, “reason”:”reason1”}

**Response**:

Request created, Example of an “annual leave” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"type": "accidental",

"date": "2020-12-25T22:00:00.000Z",

"status": "P",

"reason": "reason1",

“replacers”: [],

“documents”: [],

"\_\_v": 0

}

72–

Functionality: Send a “sick leave” request.

**Route: /sendSickRequest**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “date”: “2020-12-25”, “reason”:”reason1”, “documents”: [“document1”, “document2”]}

**Response**:

Request created, Example of an “sick leave” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"type": "sick",

"date": "2020-12-25T22:00:00.000Z",

"status": "P",

"reason": "reason1",

“replacers”: [],

“documents”: [“document1”, “document2”],

"\_\_v": 0

}

73 –

Functionality: Send a “maternity leave” request.

**Route: /sendMaternityRequest**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “date”: “2020-12-25”, “reason”:”reason1”, “documents”: [“document1”, “document2”]}

**Response**:

Request created, Example of an “maternity leave” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"type": "maternity",

"date": "2020-12-25T22:00:00.000Z",

"status": "P",

"reason": "reason1",

“replacers”: [],

“documents”: [“document1”, “document2”],

"\_\_v": 0

}

74–

Functionality: Send a “compensation leave” request.

**Route: /sendCompensationRequest**

Request type: POST

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “absentDate”: “2020-12-25”, “compensationDate”:”2020-12-28”, “reason”: “reason1”}

**Response**:

Request created, Example of an “compensation leave” request :

{

"\_id": “5fdf377e8fedba3350b43ebf”,

"sender": "staff1",

"recipient": "staff2",

"type": "compensation",

"date": "2020-12-25T22:00:00.000Z",

"status": "P",

"reason": "reason1",

“replacers”: [],

“documents”: [],

"\_\_v": 0

}

75–

Functionality: show notifications. (2 Routes)

**Route 1: /getNotificationCount**

Request type: GET

// gets count of unseen notifications

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

2

**Route 2: /viewNotifications**

Request type: GET

Request header : {“auth-token” : “the token we got from login route”}

**Response**:

Array of notification objects sorted descendingly on date (latest appear first). Example of a notification:

{

“user”: “staff1”,

“message”: “staff2 accepted your replacement request ...”,

“date”: “2020-12-25T19:42:05.442+00:00”

}

76–

Functionality: view status of all submitted requests.

**Route: /viewSentRequests**

Request type: GET

// body has two inputs =>

1- type can have values [“replacement”, “slotLinking”, “dayoff”, “leave”].

2- status can have values [“P”, ”A”, ”R”].

“P” means pending, “A” means accepted, “R” means rejected

For example {“type”:”leave”, “status”:”P”} shows all pending leave requests.

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “type”:”replacement”, “status”:”A”}

**Response**:

Array of replacement requests. An example of a replacement request:

{ "\_id": "5fe5f60edf1914246cb5e358",

"sender": "staff1",

"recipient": "staff2",

"status": "A",

"slot": {

"course": "csen703",

"start": "8:15",

"end": "9:45",

"location": "c7.205",

"replacement": "2020-12-20T22:00:00.000Z"},

"\_\_v": 0}

77–

Functionality: cancel a “day-off” request.

**Route: /cancelDayOffRequest**

Request type: DELETE

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”:“5fdf377e8fedba3350b43ebf”}

**Response**:

“Request canceled successfully”

78–

Functionality: cancel a “slot linking” request.

**Route: /cancelSlotLinking**

Request type: DELETE

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”:“5fdf377e8fedba3350b43ebf”}

**Response**:

“Request canceled successfully”

79–

Functionality: cancel a “leave” request.

**Route: /cancelLeaveRequest**

Request type: DELETE

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”:“5fdf377e8fedba3350b43ebf”}

**Response**:

“Request canceled successfully”

80–

Functionality: cancel a “replacement” request.

**Route: /cancelReplaceRequest**

Request type: DELETE

Request header : {“auth-token” : “the token we got from login route”}

Request body: { “requestId”:“5fdf377e8fedba3350b43ebf”}

**Response**:

“Request canceled successfully”