GUC Staff Members Functionalities

Server is listening on port 3000

Mongoose.connect is index.js file so u can change the connection string there

You can run the project with any of these commands(node index.js, npm init, nodemon index.js)

• Register first hr member

Functionality: Login with unique email and password

Route: /hr/seedfirsthr

Request type: get

Response: all info of first hr member

• Log in with a unique email and a password

Functionality: Login with unique email and password

Route: /authentication/login

Request type: POST

Request body: {“email”:”bassel@gmail.com”, “password”:”bassel123”}

Response: Successful Login and token (both in Json object).

Example:

{"Msg":“Welcome you logged in successfully”,“token”:”dsafrinaersd.sadfgrgafd.eqwqrfdvbxc”}

Response body:{ {"Msg":“Welcome you logged in successfully”,“token”:”dsafrinaersd.sadfgrgafd.eqwqrfdvbxc”}}

• Log out of the system

Functionality:

Route: /authentication/logout

Request type: POST

Response:”Logged out"

Description: Adds the token to a black list so it won't work again :)

• View Profile

Functionality:view the users profile

Route: /authentication/profile

Request type: GET

Response: users profile as Json object

Response Example:{“id”:”ac-1”,”name”:”bassel”,”faculty”:”Engineering”,....etc}

Request body empty cause we can get it with the token

• **Edit Profile**

Functionality:allows user to edit some of his profile attributes

Route: /authentication/profile

Request Type: PUT

Request body:{“email”:”bassel2@gmail.com”,”dayoff”:”tuesday”}

Response:”Profile edited successfully”

• Reset Password

Functionality: user edits his password

Route: /authentication/resetpassword

Request Type: post

Request body:{“newpassword”:”123”}

Response:”password updated”

- **Sign in**

Functionality: records user signing in to campus

Route: /authentication/signin

Request Type: post

Response:”sign in recorded”

•Sign out

Functionality: records user signing out of campus (adds the working hours)

Route: /authentication/signout

Request Type: post

Response:”sign out recorded”

- **View all/month attendance record**

ViewAll

Functionality: views attendance records of the user

Route: /authentication/viewAllAttendance

Request Type: get

Response:array of attendance records in json objects

Example:[

```
 {
      "signs": [ {"signin": "10:36:26" },{  "signin": "10:37:32"  },  {"signout":
"10:37:44"

         },

         { "signout": "10:37:45" }, { "signin": "10:38:00" },  {  "signout": "10:39:00"} ],
"id": "ci-4", "date": "2020-12-21", "dayname": "Monday"

   },

   {

      "signs": [ {  "signin": "10:38:00" },{  "signout": "10:39:00" }

      ], "id": "ci-4","date": "2020-12-20",  "dayname": "sunday"}

]
```

<u>ViewMonth</u>

Functionality: views attendance of a certain month

Route: /authentication/viewMonthAttendance

Request Type: get

Request body:{“month”:”12”}

Response:all attendance records available in that month

Example:[

```
   {

      "signs": [ {"signin": "10:36:26" },{  "signin": "10:37:32"  },  {"signout":
"10:37:44"

         },

         { "signout": "10:37:45" }, { "signin": "10:38:00" },  {  "signout": "10:39:00"} ],
"id": "ci-4", "date": "2020-12-21", "dayname": "Monday"

   },

   {

      "signs": [ {  "signin": "10:38:00" },{  "signout": "10:39:00" }

      ], "id": "ci-4","date": "2020-12-20",  "dayname": "sunday"}
```

]

• view missing days

Functionality: view missing days

Route: /hr/viewMissingdays

Request type: Post

Request body:

Response: "Missing days: 2"

• view missing/extra hours

Functionality: view missing days

Route: /hr/viewHours

Request type: Post

Request body:

Response: "

Missing hours: 2

Extra hours: 3"

HR Functionalities

•Add/update/delete location

Functionality:add/update/delete a location

Route: /hr/locationEdit

Request type: add:post , update:put, delete:delete

//we dont allow updating of roomid

Request body:(in case of add and update) post / put

```
 {
      "room":"c7.205",
      "type":"office",
      "capacity":90
}
```

Request body:(in case of delete) delete

```
 {
      "room":"c7.205"
}
```

Response: add:"Location added, **put**:"Location updated, **delete**:"location deleted"

•Add/update/delete faculty

Functionality:add/update/delete a faculty

Route: /hr/facultyEdit

Request type: add:post , update:put, delete:delete

Request body: (in case of add/delete) post / delete

```
 {
      "name":"managment"
}
```

Request body: (in case of update) put

```
 {
```

“name”:”managment”,”newname”:”engineering”

}

Response: add:”managment added”, **delete**:”managment deleted”,

put:”managment updated to engineering”

•Add/update/delete department under a faculty

Functionality:add/update/delete a faculty

Route: /hr/departmentEdit

Request type: add: Post , update: Put, delete: Delete

Request body: (in case of add) post

{ “faculty”:”engineering”, ”department”:”met” }

Request body: (in case of delete) delete

{ “faculty”:”engineering”, ”department”:”met” }

Request body: (in case of update) put

{ faculty”:”engineering”, ”department”:”met”, ”newdepartment”:”iet” ,”newhod”:”slim”}

Response: add:”met added”, **delete**:”met deleted”, **put**:”met updated to iet”

•add/update/delete a course under a department

Functionality:add / update /delete a faculty

Route: /hr/courseEdit

Request type: add: Post , update: Put, delete: Delete

Request body: (in case of add) Post

{

"Courseid":"cs3", "coursename":"algorithms", "faculty":"engineering", "department":"met", "numberofslots":6

}

Request body: (in case of delete) delete

{ "Courseid":"cs3"}

Request body: (in case of update) put

{//we dont allow updating of courseid

"Courseid":"cs3", "coursename":”objectoriented", "faculty":"business", "department":"bi", "numberofslots":5

}

Response: add:”course added”, **delete**:”course deleted”, **put**:”course updated to iet”

•<u>add new staff member</u>

Functionality: add new staff member

Route: /hr/addUser

Request type: POST

Request body:

```
{
"name":"bassel",
"email":"b2@gmail.com",
"userType":"hr",
"office":"c7.205",
"salary":2000,
"dayoff":"saturday",
"department":"met",
"faculty":"Engineering"
```

}

Response:”User added successfully”

•**<u>update/delete already existing staff</u>**

<u>Update</u>

Functionality: update existing staff member

Route: /hr/editStaffMember

Request type: put

Request body:

```
    {
       “staffid":"am-2"
    "email":"newemail@gmail.com",
    "office":"c7.205",
    "dayoff":"tuesday",
    "department":"iet",
    "faculty":"Engineering"
}
```

Response:”staff member updated”

<u>delete</u>

Functionality: delete existing staff member

Route: /hr/editStaffMember

Request type:delete

Request body:

```
    {    “staffid":"am-2"}
```

Response:”staff member deleted”

•**manually add missing sign in/out (except himself)**

manually add missing sign in

Functionality: manual add sign in for staff member

Route: /hr/addSignin

Request type: post

Request body: {

"staffid":"ci-4",

"year":2020,

"month":12,

"day":20,

"hour":10,

"minute":38

}

Response:"Sign in added for staff member"

manually add missing sign out

Functionality: manual add sign in for staff member

Route: /hr/addSignout

Request type: post

Request body: {

"staffid":"ci-4",

"year":2020,

"month":12,

"day":20,

"hour":10,

```
   "minute":38
}
```

Response:”Sign out added for staff member”

•View any staff member attendance record.

Functionality: views attendance records of selected staff member

Route: /hr/viewStaffAttendance

Request Type: get

Request Body:{“staffid”:”ci-4”}

Response:array of attendance records in json objects of the staff member

```
[
   {
      "signs": [{ "signin": "13:6:6"  },{ "signout": "13:6:30"}  ],
     "id": "ci-4",    "date": "2020-12-20",     "dayname": "Sunday"
   }
]
```

•view staff members with missing days

Functionality: view staff members that have missing days

Route: /hr/staffmissingdays

Request type: get

Request body:

Response: [

```
   {
      "missingdays": 1,"name": "biso", "id": "hr-1"
```

},

{ "missingdays": 2, "name": "biso", "id": "hr-2"

}

]

•<u>**view staff members with missing hours**</u>

Functionality: view staff members that have missing hours

Route: /hr/staffmissinghours

Request type: get

Request body:

Response: [

{"missinghours": 1, "name": "biso", "id": "hr-1"

},

{"missinghours": 2, "name": "biso", "id": "hr-2" }

]

•<u>**update salary of staff member**</u>

Functionality:update salary of staff member by id

Route: /hr/editStaffSalary

Request type: Post

Request body: {“staffid”:”hr-2”,”newsalary”:”20000”}

Response: “staff member salary updated to 20000”

•Hr assigns hod(update his profile to hod)

Functionality:update salary of staff member by id

Route: /hr/assignhod

Request type: Post

Request body: {“staffid”:”am-2”,”department”:”met”}

Response: “hod assigned”

//this is done automatically whenever it 19:00

//see schedule method at end of login_route.js

//i just added this route if you want to test it

•update users hours

Functionality:updates all the users hours

Route: hr/updatehours

Request type: Post

Console log: “users hours updated”

//this is update missing days also done automatically but this is a route if you want to test it

//you can see the automatic code at the end of the login_route.js file

•update users missing days

Functionality:updates all the users hours

Route: hr/updatemissingdays

Request type: Post

response: “users missing days updated”

COURSE CORDINATOR

•View \slot linking" request(s) from academic members linked to his/her course.

Functionality: view all requests in the cc courses

Route: cc/viewslotlinking

Request type: get

Response:

```
[
   {
      "status": "pending",
      "type": "SlotLinkingRequest",
      "id": "r-1", "senderid": "am-1", "destinationid": "cc-1", "slot": 1, n"course":
"cs3","date": "2020-12-23T09:48:38.660Z", "createdAt":
"2020-12-23T09:48:38.666Z","updatedAt": "2020-12-23T09:48:38.666Z","__v": 0
   },
   {
      "status": "pending","type": "SlotLinkingRequest","id": "r-1", "senderid":
"am-1", "destinationid": "cc-1",
      "slot": 1, "course": "cs4","date": "2020-12-23T09:49:01.222Z", "createdAt":
"2020-12-23T09:49:01.225Z", "updatedAt": "2020-12-23T09:49:01.225Z", "__v": 0
   }
]
```

•Accept slot linking request

Functionality: accept a slot linking request

Route: cc/acceptslotlinking

Request type: post

Request body:

```
{
   "Id":"r1",  "Senderid":"cc-1",   "course":"cs3"
}
```

Response: “request accepted and the slot has been updated”

•reject slot linking request

Functionality: reject a slot linking request

Route: cc/rejectslotlinking

Request type: post

Request body:{

"Id":"r1", "Senderid":"cc-1", "course":"cs3"

}

Response:"request rejected"

•add/update/delete course slot

Add

Functionality: add course slot

Route: cc/editcourseschedule

Request type: post

Request body: {

"Courseid":"cs4", "Day":"sunday", "Slot":3, "location":"c3.404"

}

Response:"Slot added"

Delete:

Functionality: delete course slot

Route: cc/editcourseschedule

Request type: delete

Request body: {

"Courseid":"cs4", "Day":"sunday", "Slot":3, "location":"c3.404"

}

Response:"Slot deleted"

update:

Functionality: dupdate course slot

Route: cc/editcourseschedule

Request type: put

Request body: {

"Courseid":"cs4", "Day":"wednesday", "Slot":3, "location":"c3.404"

}

Response:"Slot updated"

Academic member

- **Send replacement request**

Functionality: send a linking request

Route: am1/sendlinkingrequest

Request type: post

Request body:{{ "id":"r1", "course": "cs3", "day": "monday", "slot": 1, "location": "c3.2", "destinationid":"cc-1"

}

Response: "Linking request send";

- **View replacement requests**

Functionality: send a linking request

Route: am1/viewlinkingrequest

Request type: get

Response:

```
[
  {
    "status": "pending",
    "type": "SlotLinkingRequest",
```

"_id": "5fe33115f1efab3448b1d8ce", "id": "r1", "course": "cs3","day": "monday","slot": 1, "location": "c3.2","destinationid": "cc-1", "date": "2020-12-23T11:59:17.422Z", "senderid": "cc-1",

"createdAt": "2020-12-23T11:59:17.429Z","updatedAt": "2020-12-23T11:59:17.429Z",

"__v": 0

}

]

HOD ROUTES

- **Assign a course instructor for each course in his department.**

Route: /hod/courseinstructor

Request type: POST

Request body: {"insid":"am-1", "courseid":"csen"}

meaning:(insid=> instructor id you want to assign,courseid=>is the targeted course)

Response:A-done(in case of valid data) or B-please double check enterd data(in case of invalid data)

- **update a course instructor for each course in his department.**

Route: /hod/courseinstructor

Request type: put

Request body: {"insidold":"am-1","insidnew":"am-2", "courseid":"csen"}

meaning:(insidold=>the old instructor id you want to assign,insidnew=>the new instructor id you want to assign,courseid=>is the target course)

Response:A-done(in case of valid data) or B-please double check enterd data(in case of invalid data)

- **delet a course instructor for each course in his department.**

Route: /hod/courseinstructor

Request type: delet

Request body: `{"insid":"am-1","courseid":"csen"}`

meaning:(insid=>the instructor id you want to assign,courseid=>is the target course)

Response:A-done(in case of valid data) or B-please double check enterd data(in case of invalid data)

- **View all the staff in his/her department**

Route: /hod/staffindept

Request type: get

Request body: {}

Request body meaning:(all the required data will be from payload of the token)

Response example:A-please double check entered data (in case of invalid data)

,B-[

```
    {
        "dayoff": "saturday",
        "hourspermonth": 164.8,
        "annualleavebalance": 2.5,
        "gender": "male",
        "notifications": [],
        "name": "hod1",
        "id": "hod-1",
        "email": "hod1@weza.com",
        "office": "req.body.office",
        "userType": "hod",
        "department": "cs",
        "__v": 0 }]
```

Response meaning:list of all users in same department

- **View all the staff per course**

Route: /hod/staffincourse

Request type: get

Request body:`{"courseid":"csen"}`

Request body meaning:(courseid=>id of target course)

Response example:A-please double check entered data (in case of invalid data)

,B-[

```
{

    "dayoff": "saturday",

    "hourspermonth": 164.8,

    "annualleavebalance": 2.5,

    "gender": "male",

    "notifications": [],

    "name": "hod1",

    "id": "hod-1",

    "email": "hod1@weza.com",

    "office": "req.body.office",

    "userType": "hod",

    "department": "cs",

    "__v": 0 }]
```

Response meaning:list of all users in same course

- **View the day off of all the staff in his/her department.**

Route: /hod/getdoffall

Request type: get

Request body:{}

Request body meaning:(all the required data will be from payload of the token)

Response example: A-please double check entered data (in case of invalid data)

```
,B-[{
        "dayoff": "sunday",
        "name": "am1",
        "id": "am-12"
    },
    {
        "dayoff": "saturday",
        "name": "am2",
        "id": "am-2"
    },
    {
        "dayoff": "saturday",
        "name": "ci1",
        "id": "ci-5"
    }
]
```

, Response meaning:list of all users in same dept along with their dayoff

- **View the day off of a single staff in his/her department.**

Route: /hod/`getdoffsingle`

Request type: get

Request body:`{"userid":"am-2"}`

Request body meaning:(the target user)

Response example:A-please double check entered data (in case of invalid data)

,B-[

```
    {
        "dayoff": "saturday",
        "name": "am2",
        "id": "am-2"
    }
]
```

Response meaning:uaer name id and dayoff

View the coverage of each course in his/her department.

Route: /hod/`coursecoverage`

Request type: get

Request body:`{"courseid":"csen1"}`

Request body meaning:(the target course)

Response example:A-please double check entered data (in case of invalid data)

,B-{

```
    "coverage": 0.5
```

}Response meaning:coverage of target course

• View teaching assignments (which staff members teach which slots) of course offered by his department.

Route: /hod/`teachingassignments`

Request type: get

Request body:`{"courseid":"csen1"}`

Request body meaning:(the target course)

Response example:A-please double check entered data (in case of invalid data)

```
,B-[
    {
        "replacment": [],
        "_id": "5fddd67291bf7b3e80a8eeda",
        "courseid": "csen1",
        "day": "tuesday",
        "slot": 2,
        "userid": "am-2",
        "location": "d4",
        "__v": 0
    }]
```

Response meaning:list of all assignments for each course

View all the change day requests of sent by staff members in his/her department.

Route: /hod/`dayoffrequest`

Request type: get

Request body:`{}`

Request body meaning:(all data will be from jwt payload)

Response example:[

```
    {
        "status": "pending",
        "type": "ChangeDayoffRequest",
        "_id": "5fe5dcda47a5853cccbf910d",
        "id": "cjhhjh",
        "senderid": "am-2",
        "destinationid": "hod-1",
        "dayoff": "thursday",
        "date": "2020-12-25T12:36:42.774Z",
        "createdAt": "2020-12-25T12:36:42.777Z",
        "updatedAt": "2020-12-25T12:36:42.777Z",
        "__v": 0
    }
]
```

Response meaning:list of all requests

Accept a change day of request sent by staff members in his/her department.

Route: /hod/`dayoffrequest`

Request type: post

Request body:`{"reqid":"cjhhjh"}`

Request body meaning:(target request id)

Response example:A-please double check entered data (in case of invalid data)

,B-done (data are valid)

rejecta change day of request sent by staff members in his/her department.

Route: /hod/`dayoffrequest`

Request type: delet

Request body:`{"reqid":"cjhhjh","comment":"cjhhjhmbhbhjhjb"}`

Request body meaning:(target request id and optional comment)

Response example:A-please double check entered data (in case of invalid data)

,B-done (data are valid)

View all leave requests of sent by staff members in his/her department.

Route: /hod/`leaverequests`

Request type: get

Request body:`{}`

Request body meaning:(all data will be from jwt payload)

Response example:`[`

```
    {

        "status": "pending",

        "type": "annualleaverequest",

        "_id": "5fe4e6c5948f21644c461fb1",

        "id": "jbjh",

        "senderid": "am-2",

        "destinationid": "hod-1",

        "targetdate": "2020-12-25T13:19:37.143Z",
        "reason": "wdsvafvojndvjknsdfkvnlekjfvdn",
        "date": "2020-12-24T19:06:45.050Z",
        "createdAt": "2020-12-24T19:06:45.054Z",
        "updatedAt": "2020-12-24T19:06:45.054Z",
        "__v": 0
    },
    {
        "status": "pending",
        "type": "annualleaverequest",
        "_id": "5fe4e723d7689136cc4db7a9",
        "id": "jbjdgfgh",
        "senderid": "am-2",
        "destinationid": "hod-1",
        "targetdate": "2020-12-25T13:19:37.143Z",
        "reason": "wdsvafvojndvjknsdfkvnlekjfvdn",
        "date": "2020-12-24T19:08:19.269Z",
        "createdAt": "2020-12-24T19:08:19.272Z",
        "updatedAt": "2020-12-24T19:08:19.272Z",
        "__v": 0
    }]
```

Response meaning:list of all requests

Accept a leave request sent by staff members in his/her department.

Route: /hod/`leaverequests`

Request type: post

Request body:`{"requestid":"jbjdgfgh"}`

Request body meaning:(target request id)

Response example:A-please double check entered data (in case of invalid data)

,B-done (data are valid)

rejecta leave request sent by staff members in his/her department.

Route: /hod/`leaverequests`

Request type: delet

Request body:`{"requestid":"jbjh","comment":"cjhhjhmbhbhjhjb"}`

Request body meaning:(target request id)

Response example:A-please double check entered data (in case of invalid data)

,B-done (data are valid)

Course Instructor Routes

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

Route: /ci/coursecoverage

Request type: get

Request body:`{"courseid":["csen1"]}`

Request body meaning:(list of all courses he gives)

Response example:A-please double check entered data (in case of invalid data)

,B-`[`

```
    {
        "courseid": "csen1",
        "coverage": 0
    }
]
```

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

Route: /ci/`slotsassignment`

Request type: get

Request body:`{"courseid":["csen1"]}`

Request body meaning:(list of all courses he gives)

Response example:A-please double check entered data (in case of invalid data)

,B-`[`

```
    {
        "replacment": [],
        "_id": "5fddd67291bf7b3e80a8eeda",
        "courseid": "csen1",
        "day": "tuesday",
        "slot": 2,
        "userid": "am-2",
        "location": "d4",

        "__v": 0

    }]
```

- **View all the staff in his/her department**

Route: /ci/`staffindept`

Request type: get

Request body: {}

Request body meaning:(all the required data will be from payload of the token)

Response example:A-please double check entered data (in case of invalid data)

,B-[

```
    {

        "dayoff": "saturday",

        "hourspermonth": 164.8,

        "annualleavebalance": 2.5,

        "gender": "male",

        "notifications": [],

        "name": "hod1",

        "id": "hod-1",

        "email": "hod1@weza.com",

        "office": "req.body.office",

        "userType": "hod",

        "department": "cs",

        "__v": 0 }]
```

Response meaning:list of all users in same department

- **View all the staff per course**

Route: /ci/`staffincourse`

Request type: get

Request body:`{"courseid":["csen"]}`

Request body meaning:(courseid=>list of ids of target courses)

Response example:A-please double check entered data (in case of invalid data)

,B-`[{`

```
        "dayoff": "saturday",

        "hourspermonth": 164.8,

        "annualleavebalance": 0,

        "gender": "male",

        "notifications": [],

        "_id": "5fdf4ac9f0211f50f815368b",

        "name": "ci1",

        "id": "am-1",

        "password": "$2b$10$tcHabu9k2nqZ5736/Ih7Du2S5qb.LfKH.X4Fw7ceo4/Bw3x3ef0km",

        "email": "ci1@weza.com",

        "salary": 60,

        "office": "req.body.office",

        "userType": "ci",

        "department": "cs",

        "__v": 0

    }
```

]

Response meaning:list of all users in same course

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

Route: /ci/`assignacmemtoslot`

Request type: post

Request body:
```
{"courseid":"csen1",
  "amid":"am-12",
  "day":"sunday",
  "slot":3,
  "location":"c3-102",
  "replace":1
}
```

Request body meaning:(courseid=>id of target course

amid=>the id academic memper you want to place,

day=>the day of desired slot,

slot=>the slot number of desired slot,

location=>location of the slot

replace=>indicates wether you want to replace any oblications the AM has that conflict with this slot(1=>replace anyway,0=>do not replace in case of conflict)

Response example:A-please double check entered data (in case of invalid data)

,B-this member has an obligation (in case of conflict and replace =0), C-done

Update an academic member to an unassigned slots in course(s) he/she is assigned to

Route: /ci/`assignacmemtoslot`

Request type: put

Request body:

```
{"courseid":"csen1",

  "amidold":"am-12",

  "amidnew":"am-12",

  "day":"sunday",

  "slot":3,

  "location":"c3-102",

  "replace":1

}
```

Request body meaning:(courseid=>id of target course

amidnew=>the id academic memper you want to place,

amidold=>the id of the old member,

day=>the day of desired slot,

slot=>the slot number of desired slot,

location=>location of the slot

replace=>indicates wether you want to replace any oblications the new AM has that conflict with this slot(1=>replace anyway,0=>do not replace in case of conflict)

Response example:A-please double check entered data (in case of invalid data)

,B-this member has an obligation (in case of conflict and replace =0), C-done

Delet an academic member to an unassigned slots in course(s) he/she is assigned to

Route: /ci/assignacmemtoslot

Request type: delet

Request body:
```
{"courseid":"csen1",
  "amid":"am-12",
  "day":"sunday",
  "slot":3,
  "location":"c3-102",
  }
```

Request body meaning:(courseid=>id of target course

amid=>the id academic memper you want to place,

day=>the day of desired slot,

slot=>the slot number of desired slot,

location=>location of the slot

Response example:A-please double check entered data (in case of invalid data)

,B-done

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

Route: /ci/`removememfromcourse`

Request type: delet

Request body:
```
{"courseid":"csen1",
  "amid":"am-12"}
```

Request body meaning:(courseid=>id of target course

amid=>the id academic memper you want to place

Response example:A-please double check entered data (in case of invalid data)

,B-done

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

Route: /ci/assigncoordinator

Request type: post

Request body:{"courseid":["csen1"],

```
  "amid":"am-12"}
```

Request body meaning:(courseid=>list of ids of target courses

amid=>the id academic memper you want to place

Response example:A-please double check entered data (in case of invalid data)

,B-done

Academic member routes

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

Route: /am/viewschedule

Request type: get

Request body:{}

Request body meaning:all data we be from jwt payload

Response example:A-please double check entered data (in case of invalid data)

,B-(list of all obligations the member has)[

```
    {
        "replacment": [],
        "_id": "5fddd67291bf7b3e80a8eeda",
        "courseid": "csen1",
        "day": "tuesday",
        "slot": 2,
        "userid": "am-2",
        "location": "d4",
        "__v": 0
    }
]
```

- **Send/view “replacement” request(s).**

Route: /am/`replacmentrequest`

Request type: post

Request body:

```
{"requestid":"a",
  "targetdate":"2020-12-25T13:19:37.143+00:00",
  "reciverid":"am-3",
  "courseid":"csen"}
```

Request body meaning:requestid=>prefered req id from user(unique),

targetdate=>desired date for replacement,

reciverid=> id of receiving member,

courseid=>id of course to reblace

Response example:A-please double check entered data (in case of invalid data)

,B-done

• view “replacement” request(s).

Route: /am/replacmentrequest

Request type: get

Request body:{}

Response example:A-please double check entered data (in case of invalid data)

,B-[

```
    {
        "status": "pending",
        "type": "ReplacementRequest",
        "_id": "5fe61740c6402b3008891651",
        "id": "aiaiir",
        "senderid": "am-1",
        "destinationid": "hod-1",
        "targetdate": "2020-12-29T13:19:37.143Z",
        "course": "csen1",
        "replacementid": "am-2",
        "date": "2020-12-25T16:45:52.414Z",
        "createdAt": "2020-12-25T16:45:52.417Z",

        "updatedAt": "2020-12-25T16:45:52.417Z",

        "__v": 0

    }

]
```

• Change their day off by sending a “change day off” request (automatically sent to HOD), and optionally leave a reason.

Route: /am/changedayoff

Request type: post

Request body:{"requestid":"a",

  "newday":"monday"}

Request body meaning:requestid=>prefered req id from user(unique),

newday=>desired day off(must be lower case),

Response example:A-please double check entered data (in case of invalid data)

,B-done

• Submit annual leave request

Route: /am/annualleaverequest

Request type: post

Request body:{"requestid":"a",

  "targetdate":"2020-12-25T13:19:37.143+00:00",

  "replacmentid":"am-3",

```
  "reason":"hohohoho"}
```

Request body meaning:`requestid=>prefered req id from user(unique),`

`targetdate=>desired target date ,`

`replacmentid=>id of replacing member(optional) ,`

`reason=>any desired reason(optional)`

Response example:A-please double check entered data (in case of invalid data)

,B-done

- **Submit Accidental leave request**

Route: /am/`accidentalleave`

Request type: post

Request body:`{"requestid":"a",`

```
  "targetdate":"2020-12-25T13:19:37.143+00:00",
    "reason":"hohohoho"}
```

Request body meaning:`requestid=>prefered req id from user(unique),`

`targetdate=>desired target date ,`

`reason=>any desired reason(optional)`

Response example:A-please double check entered data (in case of invalid data)

,B-done

- **Submit Sickleave request**

Route: /am/`sickleave`

Request type: post

Request body:
```
{"requestid":"a",
  "targetdate":"2020-12-25T13:19:37.143+00:00",
  "document":"am-3",
  "reason":"hohohoho"}
```

Request body meaning:requestid=>prefered req id from user(unique),

targetdate=>desired target date ,

document=>any required document

reason=>any desired reason(optional)

Response example:A-please double check entered data (in case of invalid data) ,B-done

- **Submit Maternity request**

Route: /am/maternityleave

Request type: post

Request body:
```
{"requestid":"a",
  "targetdate":"2020-12-25T13:19:37.143+00:00",
  "document":"am-3",
  "reason":"hohohoho"}
```

Request body meaning:requestid=>prefered req id from user(unique),

targetdate=>desired target date ,

document=>any required document

reason=>any desired reason(optional)

Response example:A-please double check entered data (in case of invalid data)

,B-done

• Submit Compensation request

Route: /am/Compensationleave

Request type: post

Request body:{"requestid":"a",

"targetdate":"2020-12-25T13:19:37.143+00:00",

"reason":"hohohoho"}

Request body meaning:requestid=>prefered req id from user(unique),

targetdate=>desired target date ,

reason=>any desired reason(optional)

Response example:A-please double check entered data (in case of invalid data)

,B-done

• Notified whenever their requests are accepted or rejected.

Route: /am/getnotifications

Request type: get

Request body:{}

Response example:A-please double check entered data (in case of invalid data)

,B-["request :A is assepted ","request :jbjh is rejected "]

- **View the status of all submitted requests**

Route: /am/getrequests

Request type: get

Request body:{}

Response example:A-please double check entered data (in case of invalid data)

,B-[

```
    {
        "status": "rejected",
        "type": "annualleaverequest",
        "_id": "5fe4e6c5948f21644c461fb1",
        "id": "jbjh",
        "senderid": "am-2",
        "destinationid": "hod-1",
        "targetdate": "2020-12-25T13:19:37.143Z",
        "reason": "wdsvafvojndvjknsdfkvnlekjfvdn",
        "date": "2020-12-24T19:06:45.050Z",
        "createdAt": "2020-12-24T19:06:45.054Z",
        "updatedAt": "2020-12-25T13:18:33.179Z",
        "__v": 0,
        "comment": "cjhhjhmbhbhjhjb"
    }
```

- **cancel request**

Route: /am/cancelrequest

Request type: delet

Request body:{"requestid":"a"}

Response example:A-please double check entered data (in case of invalid data)

,B-done