

API Documentation

Content

1. [Login](#)
2. [Individual Student's info](#)
3. [Group student's info](#)
4. [Individual Teacher's info](#)
5. [Group Teacher's info](#)
6. [Add a student](#)
7. [Modify a student's data](#)
8. [Delete a student](#)
9. [Add a teacher](#)
10. [Modify a teacher's data](#)
11. [Delete a teacher](#)
12. [Change password](#)
13. [Reset password](#)

Login:

Call file: api/login.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers
username: - User name of teacher/ student or COE
password: - Password of the given username.
category: - 0/1/2 (0 for COE, 1 for teacher, 2 for student)

E.g.:

key=WnXsEnsDk&username=GCECTB-R17-3018&password=1234&category=2

(NOTE: here “WnXsEnsDk” is not the actual key, it is just shown for clarity of the example)

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “login successful”

E.g.:

```
{  
    status: "OK",  
    comment: "login successful"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.

“incorrect username: XYZ” – when the username is not correct.

“incorrect password” – when the password is not correct.

“incorrect category: XYZ” – when the category is anything other than 0 or 1 or 2.

“undefined parameter: key” – when the parameter “key” is not given.

“undefined parameter: username” – when the parameter “username” is not given.

“undefined parameter: password” – when the parameter “password” is not given.

“undefined parameter: category” – when the parameter “category” is not given.

“server error occurred” – when some server-side error occurs

E.g.1:

```
{  
  status: "FAIL",  
  comment: "incorrect username: XYZ"  
}
```

E.g.2:

```
{  
  status: "FAIL",  
  comment: "undefined parameter: key"  
}
```

Individual Student's info:

Call file: api/student_info.php

Request type: - http

Method: GET

Input:

key: - Key provided by the API developers

username: - User name of student.

E.g.:

api/student_info.php?key=WnXsEnsDk&username=GCECTB-R17-3018

(NOTE: here "WnXsEnsDk" is not the actual key, it is just shown for clarity of the example)

Output:

Type: json file

- **If successful:**

First parameter "status" value of which will be "OK"

Second parameter "result" value of which will be the information of the student whose username is provided. The result will contain (username, name, department, joining year, pass out year, registration no, email, contact no)

E.g.:

```
{
  status: "OK",
  result: {
    username: "GCECTB-R17-3018",
    name: "RASHED MEHDI",
    department: "CSE",
    joining year: 2017,
    pass out year: 2021,
    registration no: "",
    email: "",
    contact no: ""
  }
}
```

- **If fail:**

First parameter "status" value of which will be "FAIL"

Second parameter "comment" value of which can be either of:

"incorrect key: XYZ" – when the key is not correct.

"incorrect username: XYZ" – when the username is not correct.

"undefined parameter: key" – when the parameter "key" is not given.

"undefined parameter: username" – when the parameter "username" is not given.

E.g.1:

Request:

api/student_info.php?key=WnXsEnsDk&username=GCECTB-R17-3018

Output:

```
{
    status: "FAIL",
    comment: "incorrect key: WnXsEnsDk"
}
```

E.g.2:

Request:

api/student_info.php?key=WnXsEnsDk&uname=GCECTB-R17-3018

```
{
    status: "FAIL",
    comment: "undefined parameter: username"
}
```

Group Student info:

Call file: api/student_group_info.php

Request type: - http

Method: GET

Input:

key: -	Key provided by the API developers
stream: -	can be either of CSE/IT/CT or 1 for all the department
batch_passout_year: -	passing out year of the batch (e.g. 2021) or 1 for all the passing out year
joining_year: -	joining year (e.g. 2017) or 1 for all the joining year

NOTE: *joining_year* includes all the student that has joined in the year given, this will not include a year lag student. *batch_passout_year* on the other hand will include the year lag student and will display all the students who are expected to pass out in the given year.

E.g.1:

api/student_group_info.php?key=WnXsEnsDk&stream=1&batch_passout_year=1&joining_year=2017

This will give all the student data who have joined in the year 2017.

E.g.2:

api/student_group_info.php?key=WnXsEnsDk&stream=CSE&batch_passout_year=2021&joining_year=1

This will give all the student data who will pass out in the year 2021 and the stream is CSE.

(NOTE: here “WnXsEnsDk” is not the actual key, it is just shown for clarity of the example)

Output:

Type: json file

- **If successful:**

First parameter “**status**” value of which will be “**OK**”

Second parameter “**result**” value of which will be an array of information of the students who met the given criteria. The array will contain (username, name, department, joining year, pass_out_year, registration no, email, contact no)

E.g.1:

Request:

api/student_group_info.php?key=WnXsEnsDk&stream=1&batch_passout_year=1&joining_year=2017

Output:

```
{
  status: "OK",
  result: [
    {
      username: "GCECTB-R17-3018",
      name: "RASHED MEHDI",
      department: "CSE",
      joining year: 2017,
      pass out year: 2021,
      registration no: "",
      email: "",
      contact no: ""
    },
    {
      username: "GCECTB-R17-1019",
      name: "RISHAV BANERJEE",
      department: "CT",
      joining year: 2017,
      pass out year: 2021,
      registration no: "",
      email: "",
      contact no: ""
    },
    {
      username: "GCECTB-R17-3030",
      name: "SOUMYA MUKHERJEE",
      department: "CSE",
      joining year: 2017,
      pass out year: 2021,
      registration no: "",
      email: "",
      contact no: ""
    },
    {
      username: "GCECTB-R17-2024",
      name: "SARANYA NAHA ROY",
      department: "IT",
      joining year: 2017,
      pass out year: 2022,
      registration no: "",
      email: "",
      contact no: ""
    }
  ]
}
```


- **If fail:**

First parameter “**status**” value of which will be “**FAIL**”

Second parameter “**comment**” value of which can be either of:

“**incorrect key: XYZ**” – when the key is not correct.

“**incorrect stream: XYZ**” – when the stream is other than CSE/IT/CT/1

“**incorrect joining_year: XYZ**” – when the joining year is not 1 or in the format 20XX.

“**incorrect batch_passout_year XYZ**” - when the pass out year is not 1 or in the format 20XX.

“**undefined parameter: key**” – when the parameter “key” is not given.

“**undefined parameter: stream**” – when the parameter stream is not defined.

“**undefined parameter: joining_year**” – when the parameter joining_year is not defined.

“**undefined parameter: batch_passout_year**” - when the parameter batch_passout_year is not defined.

E.g.1:

Request:

api/student_group_info.php?key=WnXsEnsDk&stm=CSE&batch_passout_year=1&joining_year=2017

Output:

```
{
    status: "FAIL",
    comment: "undefined parameter: stream"
}
```

E.g.2:

Request:

api/student_group_info.php?key=WnXsEnsDk&stream=CSE&batch_passout_year=1&joining_year=3017

Output:

```
{
    status: "FAIL",
    comment: "incorrect joining year: 3017"
}
```

Individual Teacher's info:

Call file: api/teacher_info.php

Request type: - http

Method: GET

Input:

key: - Key provided by the API developers

username: - User name of teacher.

E.g.:

api/teacher_info.php?key=WnXsEnsDk&username=GCECT/F/0001

(NOTE: here "WnXsEnsDk" is not the actual key, it is just shown for clarity of the example)

Output:

Type: json file

- **If successful:**

First parameter "status" value of which will be "OK"

Second parameter "result" value of which will be the information of the teacher whose username is provided. The result will contain (username, honorary title, first name, last name, department, designation, address, email, contact no)

E.g.:

```
{
  status: "OK",
  result: {
    username: "GCECTF/0001",
    honorary title: "Mr"
    first name: "SAGAR",
    last name: "MAJUMDAR"
    department: "CSE",
    designation: "Assistant professor",
    address: ""
    email: "",
    contact no: ""
  }
}
```

- **If fail:**

First parameter "status" value of which will be "FAIL"

Second parameter "comment" value of which can be either of:

"incorrect key: XYZ" – when the key is not correct.

"incorrect username: XYZ" – when the username is not correct.

"undefined parameter: key" – when the parameter "key" is not given.

“undefined parameter: username” – when the parameter “username” is not given.

E.g.1:

Request:

api/teacher_info.php?key=WnXsEnsDk&username=GCECTF/0001

Output:

```
{
  status: "FAIL",
  comment: "incorrect key: WnXsEnsDk"
}
```

E.g.2:

Request:

api/student_info.php?key=WnXsEnsDk&username=GCECTB-R17-3018

```
{
  status: "FAIL",
  comment: "incorrect username: GCECTB-R17-3018"
}
```

Group Teacher info:

Call file: api/teacher_group_info.php

Request type: - http

Method: GET

Input:

key: -	Key provided by the API developers
stream: -	can be either of CSE/IT/CT or 1 for all the department

E.g.:

api/teacher_group_info.php?key=WnXsEnsDk&stream=CSE

This will give all the teacher's data who are in CSE department.

(NOTE: here "WnXsEnsDk" is not the actual key, it is just shown for clarity of the example)

Output:

Type: json file

- **If successful:**

First parameter "status" value of which will be "OK"

Second parameter "result" value of which will be an array of information of the teachers who met the given criteria. The array will contain (username, honorary title, first name, last name, department, designation, email, contact no)

E.g.1:

Request:

api/teacher_group_info.php?key=WnXsEnsDk&stream=CSE

Output:

```
{
  status: "OK",
  result: [
    {
      username: "GCECTF/0001",
      honorary title: "Mr"
      first name: "SAGAR",
      last name: "MAJUMDAR"
      department: "CSE",
      designation: "Assistant professor",
      address: ""
      email: "",
      contact no: ""
    }
  ],
}
```

```

{
    username: "GCECTF/0002",
    honorary title: "Mr"
    first name: "PARTHA",
    last name: "HALDAR"
    department: "CSE",
    designation: "Assistant professor",
    address: ""
    email: "",
    contact no: ""
},
{
    username: "GCECTF/0001",
    honorary title: "Mr"
    first name: "KINGSHUK",
    last name: "CHATERJEE"
    department: "CSE",
    designation: "Assistant professor",
    address: ""
    email: "",
    contact no: ""
},
{
    username: "GCECTF/0001",
    honorary title: "Mr"
    first name: "ALOK",
    last name: "MAJUMDAR"
    department: "CSE",
    designation: "Assistant professor",
    address: ""
    email: "",
    contact no: ""
}
]
}

```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.

“incorrect stream: XYZ” – when the stream is other than CSE/IT/CT/1

“undefined parameter: key” – when the parameter “key” is not given.

“undefined parameter: stream” – when the parameter stream is not defined.

E.g.1:

Request:

api/teacher_group_info.php?key=WnXsEnsDk&stm=CSE

Output:

```
{  
    status: "FAIL",  
    comment: "undefined parameter: stream"  
}
```

E.g.2:

Request:

api/teacher_group_info.php?key=WnXsEnsDk&stream=CSTE

Output:

```
{  
    status: "FAIL",  
    comment: "incorrect stream: CSTE"  
}
```

Add a Student:

Call file: api/add_student.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers
username: - User name of student.
password: - Password of the new student's account.

Output:

Type: json file

- **If successful:**

First parameter "status" value of which will be "OK"

Second parameter "comment" value of which will be "student added successfully"

E.g.:

```
{
    status: "OK",
    comment: "student added successfully"
}
```

- **If fail:**

First parameter "status" value of which will be "FAIL"

Second parameter "comment" value of which can be either of:

"incorrect key: XYZ" – when the key is not correct.

"username: XYZ already exists" – when the username already exists.

"undefined parameter: key" – when the parameter "key" is not given.

"undefined parameter: username" – when the parameter "username" is not given.

"undefined parameter: password" – when the parameter "password" is not given.

"server error occurred" – when some server error occurs.

E.g.1:

Output:

```
{
    status: "FAIL",
    comment: "username: GCECTB-R17-3018 already exists"
}
```

E.g.2:

```
{
    status: "FAIL",
    comment: "undefined parameter: password"
}
```

Modify a Student's data:

Call file: api/modify_student.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers
username: - User name of student.
name: - Name of the student.
department: - department of the student which should be either CSE or IT or CT (cannot be null)
registration_no: - registration no of the student or null if not provided
joining_year: - joining year of the student (cannot be null)
email: - email of the student or null if not provided
contact_no: - contact no of the student or null if not provided

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “record successfully updated”

E.g.:

```
{  
    status: "OK",  
    comment: "record successfully updated"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.
“incorrect username: XYZ” – when the username does not exist.
“undefined parameter: key” – when the parameter “key” is not given.
“undefined parameter: username” – when the parameter “username” is not given.
“undefined parameter: name” – when the parameter “name” is not given.
“undefined parameter: department” – when the parameter “department” is not given.
“incorrect department: XYZ” – when the parameter “department” is other than CSE or IT or CT.
“undefined parameter: registration_no” – when the parameter “registration_no” is not given.

“undefined parameter: joining_year” – when the parameter “joining_year” is not given.

“incorrect joining_year: XYZ” – when the parameter “joining_year” is not in the format 20XX.

“undefined parameter: joining_year” – when the parameter “joining_year” is not given.

“undefined parameter: email” – when the parameter “email” is not given.

“undefined parameter: contact_no” – when the parameter “contact_no” is not given.

“server error occurred” – when some server error occurs.

E.g.1:

```
{  
    status: "FAIL",  
    comment: "username: GCECTB-R17-3018 does not exist"  
}
```

E.g.2:

```
{  
    status: "FAIL",  
    comment: "undefined parameter: joining_year"  
}
```

Delete a student:

Call file: api/delete_student.php

Request type: - http

Method: GET

Input:

key: - Key provided by the API developers
username: - User name of student.

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “student deleted”

E.g.:

```
{  
    status: "OK",  
    comment: "student deleted"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.

“incorrect username: XYZ” – when the username does not exist.

“undefined parameter: key” – when the parameter “key” is not given.

“undefined parameter: username” – when the parameter “username” is not given.

E.g.1:

Output:

```
{  
    status: "FAIL",  
    comment: "username: GCECTB-R17-3018 does not exist"  
}
```

E.g.2:

```
{  
    status: "FAIL",  
    comment: "undefined parameter: key"  
}
```

Year lag a student:

Call file: api/year_lag_student.php

Request type: - http

Method: GET

Input:

key: - Key provided by the API developers
username: - User name of student.

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “student deleted”

E.g.:

```
{
    status: "OK",
    comment: "record updated successfully"
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.

“incorrect username: XYZ” – when the username does not exist.

“undefined parameter: key” – when the parameter “key” is not given.

“undefined parameter: username” – when the parameter “username” is not given.

E.g.1:

Output:

```
{
    status: "FAIL",
    comment: "username: GCECTB-R17-3018 does not exist"
}
```

E.g.2:

```
{
    status: "FAIL",
    comment: "undefined parameter: key"
}
```

Add a Teacher:

Call file: api/add_teacher.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers
username: - User name of teacher.
password: - Password of the new teacher's account.
stream: - Stream of the teacher.

Output:

Type: json file

- **If successful:**

First parameter "status" value of which will be "OK"

Second parameter "comment" value of which will be "teacher added successfully"

E.g.:

```
{  
    status: "OK",  
    comment: "teacher added successfully"  
}
```

- **If fail:**

First parameter "status" value of which will be "FAIL"

Second parameter "comment" value of which can be either of:

"incorrect key: XYZ" – when the key is not correct.

"username: XYZ already exists" – when the username already exists.

"undefined parameter: key" – when the parameter "key" is not given.

"undefined parameter: username" – when the parameter "username" is not given.

"undefined parameter: password" – when the parameter "password" is not given.

"server error occurred" – when some server error occurs.

E.g.1:

Output:

```
{  
    status: "FAIL",  
    comment: "username: GCECTF/0001 already exists"  
}
```

E.g.2:

```
{  
    status: "FAIL",  
    comment: "undefined parameter: password"  
}
```

}

Modify a Teacher's data:

Call file: api/modify_teacher.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers
username: - User name of teacher.
name: - Name of teacher.
department: - department of the teacher which should be either CSE or IT or CT or BSEH
(cannot be null)
designation: - designation of the teacher.
address: - address of the teacher or null if not provided
email: - email of the student or null if not provided
contact_no: - contact no of the student or null if not provided

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “record successfully updated”

E.g.:

```
{  
    status: "OK",  
    comment: "record successfully updated"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.
“incorrect username: XYZ” – when the username does not exist.
“undefined parameter: key” – when the parameter “key” is not given.
“undefined parameter: username” – when the parameter “username” is not given.
“undefined parameter: name” – when the parameter “fname” is not given.
“undefined parameter: department” – when the parameter “department” is not given.
“undefined parameter: designation” – when the parameter “designation” is not given.
“incorrect department: XYZ” – when the parameter “department” is other than CSE or IT or CT or BSEH.
“undefined parameter: address” – when the parameter “address” is not given.

“undefined parameter: email” – when the parameter “email” is not given.
“undefined parameter: contact_no” – when the parameter “contact_no” is not given.
“server error occurred” – when some server error occurs.

E.g.1:

```
{  
    status: "FAIL",  
    comment: "username: GCECTF/0001 does not exist"  
}
```

E.g.2:

```
{  
    status: "FAIL",  
    comment: "undefined parameter: email"  
}
```

Delete a Teacher:

Call file: api/delete_teacher.php

Request type: - http

Method: GET

Input:

key: - Key provided by the API developers
username: - User name of teacher.

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “teacher deleted”

E.g.:

```
{  
    status: "OK",  
    comment: "teacher deleted"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“incorrect key: XYZ” – when the key is not correct.

“incorrect username: XYZ” – when the username does not exist.

“undefined parameter: key” – when the parameter “key” is not given.

“undefined parameter: username” – when the parameter “username” is not given.

E.g.1:

Output:

```
{  
    status: "FAIL",  
    comment: "username: GCECTF/0001 does not exist"  
}
```

E.g.2:

```
{  
    status: "FAIL",  
    comment: "undefined parameter: key"  
}
```

Change Password:

Call file: api/change_password.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers
username: - Username.
old_password – Old password
new_password – New password
category - should be either 0 or 1 or 2 (0 for COE, 1 for teacher, 2 for student)

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “password updated successfully”

E.g.:

```
{  
  status: "OK",  
  comment: "password updated successfully"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“undefined parameter: key” – when the parameter “key” is not given.

“incorrect key: XYZ” – when the key is not correct.

“undefined parameter: username” – when the parameter “username” is not given.

“incorrect username: XYZ” – when the username does not exist.

“undefined parameter: old_password” – when the parameter “old_password” is not given.

“incorrect old password” – when old password is not correct.

“undefined parameter: new_password” – when the parameter “new_password” is not given.

“undefined parameter: category” – when the parameter “category” is not given.

“incorrect category: XYZ” – when category is other than 0 or 1 or 2.

E.g.1:

Output:

```
{  
  status: "FAIL",
```



```
        comment: "incorrect username: GCECTF/0001"  
    }  
}
```

E.g.2:

```
{  
    status: "FAIL",  
    comment: "incorrect category: 4"  
}
```

Reset Password:

Call file: api/reset_password.php

Request type: - http

Method: POST

Input:

key: - Key provided by the API developers

username: - Username.

new_password – New password

category - should be either 0 or 1 or 2 (0 for COE, 1 for teacher, 2 for student)

Output:

Type: json file

- **If successful:**

First parameter “status” value of which will be “OK”

Second parameter “comment” value of which will be “password updated successfully”

E.g.:

```
{  
  status: "OK",  
  comment: "password updated successfully"  
}
```

- **If fail:**

First parameter “status” value of which will be “FAIL”

Second parameter “comment” value of which can be either of:

“undefined parameter: key” – when the parameter “key” is not given.

“incorrect key: XYZ” – when the key is not correct.

“undefined parameter: username” – when the parameter “username” is not given.

“incorrect username: XYZ” – when the username does not exist.

“undefined parameter: new_password” – when the parameter “new_password” is not given.

“undefined parameter: category” – when the parameter “category” is not given.

“incorrect category: XYZ” – when category is other than 0 or 1 or 2.

E.g.1:

Output:

```
{  
  status: "FAIL",  
  comment: "incorrect username: GCECTF/0001"  
}
```

E.g.2:

```
{  
  status: "FAIL",  
  comment: "incorrect category: 4"  
}
```