

Public [Run in Postman](#)

ENVIRONMENT shikhas heroku

LAYOUT Single Column

LANGUAGE cURL

**SAVS****Introduction****POST** Signup a student**GET** Get Student Information**POST** Signup a student for club member**POST** Signup a student for club head**POST** Signup a student for club head 2**POST** Create a new club**POST** Create a new club 2**PUT** Add a member happy case**POST** Propose Event 1**POST** Propose Event 2**GET** View proposed events**POST** Register for event1**POST** Register for event2**GET** View all upcoming events**GET** View registered events**GET** See all the clubs and role in the club ...**GET** See all the clubs and role in the club ...**GET** Get all clubs**PUT** Edit club happy case**GET** Get Club by id**DEL** Delete club happy case**PUT** Edit Event**GET** View event

SAVS

This is a complete test suite for testing all endpoints for SAVS app. To run this collection -

- click on the collection name to view the home page.
- click on the Run button at top right corner
- all the tests will be run one by one.

Note: the tests need to be run in the specified order.

POST Signup a student

```
https://savs-project-final.herokuapp.com/student
```

Here we test that a student can sign up on the app. The student we sign up here is a sample student (Relevant for tests ahead)

BODY raw

```
{
  "name": "TestStudent",
  "email_id": "test_student@columbia.edu",
  "college": "Fu Foundation",
  "department": "Computer Science"
}
```

Example Request

Signup a student

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student' \
--data-raw '{
  "name": "TestStudent",
  "email_id": "test_student@columbia.edu",
  "college": "Fu Foundation",
  "department": "Computer Science"
}'
```

GET Get Student Information

```
https://savs-project-final.herokuapp.com/student/test_student@columbia.edu
```

We test that the admin/club member/club head/student can view the details of the student.

Example Request

Get Student Information

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/student/test_student@
```

POST Signup a student for club member

```
https://savs-project-final.herokuapp.com/student
```

Here we sign up a student that acts as club member later in the tests.

BODY raw

```
{
  "name": "TestClubMember",
  "email_id": "test_club_member@columbia.edu",
  "college": "Fu Foundation",
  "department": "Computer Science"
}
```

Example Request

Signup a student for club member

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student' \
```

```
--data=raw '{  
  "name": "TestClubMember",  
  "email_id": "test_club_member@columbia.edu",  
  "college": "Fu Foundation",  
  "department": "Computer Science"  
}'
```

POST Signup a student for club head

<https://savs-project-final.herokuapp.com/student>

Here we sign up a student that acts as club head for TestClub1 later in the tests.

BODY raw

```
{  
  "name": "TestClubHead",  
  "email_id": "test_club_head@columbia.edu",  
  "college": "Fu Foundation",  
  "department": "Computer Science"  
}'
```

Example Request

Signup a student for club head

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student' \  
--data=raw '{  
  "name": "TestClubHead",  
  "email_id": "test_club_head@columbia.edu",  
  "college": "Fu Foundation",  
  "department": "Computer Science"  
}'
```

POST Signup a student for club head 2

<https://savs-project-final.herokuapp.com/student>

Here we sign up a student that acts as club head for TestClub2 later in the tests.

BODY raw

```
{  
  "name": "TestClubHead2",  
  "email_id": "test_club_head_2@columbia.edu",  
  "college": "Fu Foundation",  
  "department": "Computer Science"  
}'
```

Example Request

Signup a student for club head 2

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student' \  
--data=raw '{  
  "name": "TestClubHead2",  
  "email_id": "test_club_head_2@columbia.edu",  
  "college": "Fu Foundation",  
  "department": "Computer Science"  
}'
```

POST Create a new club

<https://savs-project-final.herokuapp.com/student/club>

Here we test creation of a new club Club1 by a student. Email id is used to check if the student exists and is eligible to create a club.

BODY raw

```
{  
  "emailId": "test_club_head@columbia.edu",  
  "club": {  
    "name": "Test Club 1",  
    "head": "test_club_head@columbia.edu",  
    "category": "Test Category 1",  
    "description": "Test Club Description 1"  
  }  
}'
```

Example Request Create a new club

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student/club' \
--data-raw '{
  "emailId": "test_club_head@columbia.edu",
  "club": {
    "name": "Test Club 1",
    "head": "test_club_head@columbia.edu",
    "category": "Test Category 1",
    "description": "Test Club Description 1"
  }
}'
```

[View More](#)

POST Create a new club 2

<https://savs-project-final.herokuapp.com/student/club>

Here we test creation of a new club Club 2 by a student. Email id is used to check if the student exists and is eligible to create a club. This will be used in tests later.

BODY raw

```
{
  "emailId": "test_club_head_2@columbia.edu",
  "club": {
    "name": "Test Club 2",
    "head": "test_club_head_2@columbia.edu",
    "category": "Test Category 2",
    "description": "Test Club Description 2"
  }
}
```

Example Request Create a new club 2

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student/club' \
--data-raw '{
  "emailId": "test_club_head_2@columbia.edu",
  "club": {
    "name": "Test Club 2",
    "head": "test_club_head_2@columbia.edu",
    "category": "Test Category 2",
    "description": "Test Club Description 2"
  }
}'
```

[View More](#)

PUT Add a member happy case

<https://savs-project-final.herokuapp.com/member/1>

Here we test that the club head is able to add a member to their club . The email id in payload is used to verify that the person trying to add a member is head of the club infact.

BODY raw

```
{
  "emailId": "test_club_head@columbia.edu",
  "student_email_id": "test_club_member@columbia.edu"
}
```

Example Request Add a member happy case

```
curl --location --request PUT 'https://savs-project-final.herokuapp.com/member/1' \
--data-raw '{
  "emailId": "test_club_head@columbia.edu",
  "student_email_id": "test_club_member@columbia.edu"
}'
```

POST Propose Event 1

<https://savs-project-final.herokuapp.com/events>

Here we test that a club member is able to propose an event.The email id is payload is used to verify whether the person is actually member of the club.

BODY raw

```
{  
    "emailId": "test_club_member@columbia.edu",  
    "event": {  
        "name": "TestEvent1",  
        "club_id": 1,  
        "start_timestamp": "2022-12-03 09:30:00",  
        "end_timestamp": "2022-12-05 00:00:00",  
        "location": "New York City",  
        "max_registration": 125,  
        "description": "Winter Hack", View More .021",  
        "fee": 10.  
    }  
}
```

Example Request

Propose Event 1

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/events' \  
--data-raw '{  
    "emailId": "test_club_member@columbia.edu",  
    "event": {  
        "name": "TestEvent1",  
        "club_id": 1,  
        "start_timestamp": "2022-12-03 09:30:00",  
        "end_timestamp": "2022-12-05 00:00:00",  
        "location": "New York City", View More  
        "max_registration": 125,  
    }  
}'
```

POST Propose Event 2

<https://savs-project-final.herokuapp.com/events>

Here we test that a club member is able to propose another event. This will be used for testing later.

BODY raw

```
{  
    "emailId": "test_club_member@columbia.edu",  
    "event": {  
        "name": "TestEvent2",  
        "club_id": 1,  
        "start_timestamp": "2020-12-03 09:30:00",  
        "end_timestamp": "2020-12-05 00:00:00",  
        "location": "New York City",  
        "max_registration": 125,  
        "description": "Winter Hack", View More .021",  
        "fee": 10.  
    }  
}
```

Example Request

Propose Event 2

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/events' \  
--data-raw '{  
    "emailId": "test_club_member@columbia.edu",  
    "event": {  
        "name": "TestEvent2",  
        "club_id": 1,  
        "start_timestamp": "2020-12-03 09:30:00",  
        "end_timestamp": "2020-12-05 00:00:00",  
        "location": "New York City", View More  
        "max_registration": 125,  
    }  
}'
```

GET View proposed events

https://savs-project-final.herokuapp.com/events?created_by=2

Here the club member can view the events they have proposed so far. Query parameter filters the results for particular club member.

PARAMS

created_by 2

Example Request

View proposed events

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/events?created_by=2'
```

POST Register for event1

https://savs-project-final.herokuapp.com/student/register_event/1

Here we test whether a student is able to register for event1.

BODY raw

```
{  
  "emailId": "test_student@columbia.edu"  
}
```

Example Request

Register for event1

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student/register_event/  
--data-raw '{  
  "emailId": "test_student@columbia.edu"  
}'
```

POST Register for event2

https://savs-project-final.herokuapp.com/student/register_event/2

Here we test whether the student is able to register for multiple events.

BODY raw

```
{  
  "emailId": "test_student@columbia.edu"  
}
```

Example Request

Register for event2

```
curl --location --request POST 'https://savs-project-final.herokuapp.com/student/register_event/  
--data-raw '{  
  "emailId": "test_student@columbia.edu"  
}'
```

GET View all upcoming events

https://savs-project-final.herokuapp.com/student/get_upcoming_events

Here the student can view the events they have registered to and are upcoming i.e. the past events will not show up here.

BODY raw

```
{  
  "emailId": "test_student@columbia.edu"  
}
```

Example Request

View all upcoming events

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/student/get_upcoming_  
--data-raw '{  
  "emailId": "test_student@columbia.edu"  
}'
```

GET View registered events

https://savs-project-final.herokuapp.com/student/get_registered_events

Here the student can view all the events they have registered to, irrespective of the status.

BODY raw

```
{  
  "emailId": "test_student@columbia.edu"  
}
```

Example Request

View registered events

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/student/get_registered' \
--data-raw '{
  "emailId":"test_student@columbia.edu"
}'
```

GET See all the clubs and role in the club for club member

https://savs-project-final.herokuapp.com/student/get_roles

Here a student can check what their role is in various clubs. The example used here is of a club member.

BODY raw

```
{\n  \"emailId\":\"test_club_member@columbia.edu\"\n}'
```

Example Request

See all the clubs and role in the club for club member

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/student/get_roles' \
--data-raw '{
  "emailId":"test_club_member@columbia.edu"
}'
```

GET See all the clubs and role in the club for club head

https://savs-project-final.herokuapp.com/student/get_roles

Here a student can check what their role is in various clubs. The example used here is of a club head.

BODY raw

```
{\n  \"emailId\":\"test_club_head@columbia.edu\"\n}'
```

Example Request

See all the clubs and role in the club for club head

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/student/get_roles' \
--data-raw '{
  "emailId":"test_club_head@columbia.edu"
}'
```

GET Get all clubs

<https://savs-project-final.herokuapp.com/clubs>

Here we test if a student can view all the clubs in the college.

BODY raw

```
{\n  \"emailId\":\"test_student@columbia.edu\"\n}'
```

Example Request

Get all clubs

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/clubs' \
--data-raw '{
  \"emailId\":\"test_student@columbia.edu"
}'
```

PUT Edit club happy case

<https://savs-project-final.herokuapp.com/clubs/2>

Here we test if the club head can edit the club. The email id in payload is used to verify if the person is club head itself.

BODY raw

```
{  
    "emailId": "test_club_head_2@columbia.edu",  
    "club": {  
        "description": "Test Club Description edit"  
    }  
}
```

Example Request

Edit club happy case

```
curl --location --request PUT 'https://savs-project-final.herokuapp.com/clubs/2' \  
--data-raw '{  
    "emailId": "test_club_head_2@columbia.edu",  
    "club": {  
        "description": "Test Club Description edit"  
    }  
}'
```

GET Get Club by id

<https://savs-project-final.herokuapp.com/clubs/2>

here we test if a student can view details for a particular club given the id. Notice the description for club2 has been modified in last test.

BODY raw

```
{  
    "emailId": "test_student@columbia.edu"  
}
```

Example Request

Get Club by id

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/clubs/2' \  
--data-raw '{  
    "emailId": "test_student@columbia.edu"  
}'
```

DEL Delete club happy case

<https://savs-project-final.herokuapp.com/clubs/2>

here we test whether a club head can delete the club. We verify that the person is in fact the head using the payload.

BODY raw

```
{  
    "emailId": "test_club_head_2@columbia.edu"  
}
```

Example Request

Delete club happy case

```
curl --location --request DELETE 'https://savs-project-final.herokuapp.com/clubs/2' \  
--data-raw '{  
    "emailId": "test_club_head_2@columbia.edu"  
}'
```

PUT Edit Event

<https://savs-project-final.herokuapp.com/events/2>

Here we test if the club member can edit an event for their club.

BODY raw

```
{  
    "emailId": "test_club_member@columbia.edu",  
    "event": {  
        "name": "TestEvent2",  
        "club_id": 1,  
        "start_timestamp": "2021-12-03 09:30:00",  
        "end_timestamp": "021-12-05 00:00:00",  
        "location": "New York City",  
        "max_registration": 305,  
        "description": "Test Change", View More  
        "fee": 10.  
    }  
}
```

Example Request

Edit Event

```
curl --location --request PUT 'https://savs-project-final.herokuapp.com/events/2' \  
--data-raw '{  
    "emailId": "test_club_member@columbia.edu",  
    "event": {  
        "name": "TestEvent2",  
        "club_id": 1,  
        "start_timestamp": "2021-12-03 09:30:00",  
        "end_timestamp": "021-12-05 00:00:00",  
        "location": "New York City", View More  
        "max_registration": 305,  
    }  
}'
```

GET View event

<https://savs-project-final.herokuapp.com/events/2>

here we test if anyone can view the event details given the event id.

Example Request

View event

```
curl --location --request GET 'https://savs-project-final.herokuapp.com/events/2'
```

PUT Edit club Invalid request

<https://savs-project-final.herokuapp.com/clubs/5>

This is an error test for editing the club. If anyone other than club head tries to edit the club, we throw error that it is an Invalid request.

BODY raw

```
{  
    "emailId": "testemail3@columbia.edu",  
    "club": {  
        "description": "this is a test change"  
    }  
}
```

Example Request

Edit club Invalid request

```
curl --location --request PUT 'https://savs-project-final.herokuapp.com/clubs/5' \  
--data-raw '{  
    "emailId": "testemail3@columbia.edu",  
    "club": {  
        "description": "this is a test change"  
    }  
}'
```

DEL Delete club Invalid Request

<https://savs-project-final.herokuapp.com/clubs/6>

This is an error test for deleting the club. If anyone other than club head tries to delete the club, we throw error that it is an Invalid request.

BODY raw

```
{  
    "emailId": "testemail4@columbia.edu"  
}
```

Example Request

Delete club Invalid Request

```
curl --location --request DELETE 'https://savs-project-final.herokuapp.com/clubs/6' \  
--data-raw '{  
    "emailId": "testemail4@columbia.edu"  
}'
```

PUT Add a member bad request

<https://savs-project-final.herokuapp.com/member/>

This is an error test for adding member to the club. If anyone other than club head tries to add member to the club, we throw error that it is an Invalid request.

BODY raw

```
{  
    "emailId": "testemail1@columbia.edu",  
    "student_email_id": "testemail1@columbia.edu"  
}
```

Example Request

Add a member bad request

```
curl --location --request PUT 'https://savs-project-final.herokuapp.com/member/5' \  
--data-raw '{  
    "emailId": "testemail1@columbia.edu",  
    "student_email_id": "testemail1@columbia.edu"  
}'
```