**Friend Management API**

This POC is developed to provide Friend management system with provision to connect users, to find list of friends of an user, common friends between 2 users, provisioning user to subscribe to another user etc. This functionalities are enables with following APIs.

1. To connect 2 users
2. To get list of friends of given user
3. To get list of common friends between 2 users
4. To enable subscription from requester to target
5. To enable un subscription from requester to target
6. To get list of subscribers

**Technology Stack**

**Spring Boot**

* This helps quickly start development of standalone application
* Helpful to develop restful web services
* Supports JDBCTemplate for communicating with database

**Swagger**

* provides UI to test rest end points
* Swagger UI is configured for the app and it is available:

<https://sps-demo.cfapps.io/friendmanagement/swagger-ui.html>

**PCF –Pivotal Cloud Foundry**

The Application is deployed on cloud using PCF(Pivotal Cloud foundry) .It can be accessed via the below url (https://sps-demo.cfapps.io/friendmanagement/sps/) and the path(context) for all the api is / friendmanagement/sps

**Database**-

H2, in memory database which is fast & serves the need of this POC.

**Postman**

To test Rest end points

postman path to access API would be <https://sps-demo.cfapps.io/friendmanagement/sps/> For example: To access /add endpoint, the URL should be: https://sps-demo.cfapps.io/friendmanagement/sps/add

**Maven**

Build Tool to build application

**STS**

(Spring Tool Suite) – editor for development

**List of REST Endpoints and Explanation**

1. To connect 2 users

Path – /add

Postman- https://sps-demo.cfapps.io/friendmanagement/sps/add

Input - {

"requestor": "santosh@gmail.com",

"target": "reema@gmail.com"

}

Sample output

{

"status": "Success",

"message": "Successfully connected"

}

Error handling

400 – Bad request

1. To get list of friends of given user

Path – /showFriendList

Postman - https://sps-demo.cfapps.io/friendmanagement/sps/showFriendlist

Input - {

"email": "preeta@gmail.com"

}

Sample output

{

"status": "Success",

"count": 1,

"friends": [

"som@gmail.com"

]

}

Error handling

400 – Bad request

1. To get list of common friends between 2 users

Path – /showCommonFriends

Postman - https://sps-demo.cfapps.io/friendmanagement/sps/showCommonFriends

Input - {

"friends": [

"preeta@gmail.com","santosh@gmail.com"

]

}

Sample output

{

"status": "Success",

"count": 1,

"friends": [

"som@gmail.com"

]

}

Error handling

400 – Bad request

1. To enable subscription from requester to target

Path – /subscriber

Postman - https://sps-demo.cfapps.io/friendmanagement/sps/subscribe

Input - {

"requestor": "preeta@gmail.com",

"target": "ankush@gmail.com"

}

Sample output

{

"status": "Success",

"message": "Subscribed successfully"

}

Error handling

400 – Bad request

1. To enable unsubscription from requester to target

Path – /unsubscriber

Postman - https://sps-demo.cfapps.io/friendmanagement/sps/unsubscribe

Input - {

"requestor": "preeta@gmail.com",

"target": "ankush@gmail.com"

}

Sample output

{

"status": "Success",

"message": "Unsubscribed successfully"

}

Error handling

400 – Bad request

1. To get list of subscribers

Path – /showsubscriber

Postman - https://sps-demo.cfapps.io/friendmanagement/sps/showsubscriber

Input - {

"sender": "preeta@gmail.com",

"text": "Hi @abc@gamil.com"

}

Sample output

{

"status": "Success",

"friends": [

"som@gmail.com",

"abc@gamil.com"

]

}

Error handling

400 – Bad request

**Database**

The Database is pre populated with 4 users for testing purpose. Following is table structure

FriendManagement is main table where all friends connections & subscriptions. Unsubscribe table is history table where an entry is made for every unsubscribe API call.

