

Table of Contents

1. Overview

1.1. Information

1.2. Members of the group

1.3. URI scheme

2. Paths

2.1. POST /login/

2.2. POST /signup/

2.3. GET /getGroups/

2.4. GET

/getGroupsOfAUser/{email}

2.5. GET

/getGroupInfo/{groupCookie}

2.6. GET

/getComments/{groupCookie}

2.7. GET

/judgeGroupJoined/{groupCookie}

2.8. GET

/judgeAdmin/{groupCookie}/{user}

2.9. POST /leaveGroup/

2.10. POST /joinGroup

2.11. GET

/getMembers/{group}

2.12. POST /editUser/

Meet Up Project

1. Overview

1.1. Information

Meet Up Project documentation

1.2. Members of the group

Jing WANG & Jacomo BERLANDE

1.3. URI scheme

Host: localhost:8080/MeetUpBasePath: /accueilSchemes: HTTP

- 2.13. POST /editDescription/
- 2.14. POST /addComment/
- 2.15. POST /deleteUser
- 2.16. POST /deleteGroup/
- 2.17. POST /createGroup/
- 2.18. GET /getUsers/
- 2.19. GET /getUsers/{email}
- 2.20. GET /getUsersGroups/

3. Definitions

- 3.1. Comment
- 3.2. Group
- 3.3. User

2. Paths

2.1. POST /login/

2.1.1. Description

Verify the login

2.1.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	email	Email of the user who wants to login	true	string	
QueryParameter	pwd	Password of the user who login	true	string	

2.1.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success
400	Status 400	Error

2.1.4. Consumes

- application/json

2.2. POST /signup/

2.2.1. Description

It add the firstname, the lastname, the password and eventually the description of the user to the database.

2.2.2. Parameters

Type	Name	Description	Required	Schema	Default
BodyParameter	body		true	User	

2.2.3. Responses

HTTP Code	Description	Schema
200	Status 200	

2.2.4. Consumes

- application/json

2.3. GET /getGroups/

2.3.1. Description

It return all the groups

2.3.2. Responses

HTTP Code	Description	Schema
200	Status 200	Comment

2.4. GET /getGroupsOfAUser/{email}

2.4.1. Description

Get the groups of a user.

2.4.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	email	Email of the user we want the groups	true	string	

2.4.3. Responses

HTTP Code	Description	Schema
200	Status 200	List of Group

2.4.4. Produces

- application/json

2.5. GET /getGroupInfo/{groupCookie}

2.5.1. Description

It returns the informations of a specific group.

2.5.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	commentid	Identifier of the Comment	true	string	

2.5.3. Responses

HTTP Code	Description	Schema
200	Status 200	Group

2.5.4. Produces

- application/json

2.6. GET /getComments/{groupCookie}

2.6.1. Description

Get all the comments of a group.

2.6.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	groupCookie	Name of the group we want the comment.	true	string	

2.6.3. Responses

HTTP Code	Description	Schema
200	Status 200	Comment List

2.6.4. Produces

- application/json

2.7. GET

/judgeGroupJoined/{groupCookie}/{user_email}

2.7.1. Description

To know if a user is in a group.

2.7.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	groupCookie	Name of the group	true	String	
PathParameter	user_email	Name of the user	true	String	

2.7.3. Responses

HTTP Code	Description	Schema
200	Status 200	String

2.7.4. Produces

- application/json

2.8. GET /judgeAdmin/{groupCookie}/{user_email}

2.8.1. Description

To verify if a user is admin of the group

2.8.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	groupCookie	Name of the group	true	String	
PathParameter	user_email	Name of the	true	String	

		user			
--	--	------	--	--	--

2.8.3. Responses

HTTP Code	Description	Schema
200	Status 200	String

2.9. POST /leaveGroup/

2.9.1. Description

Leave a group

2.9.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	groupName	Name of the group	true	string	
QueryParameter	userEmail	Email of the user	true	string	

2.9.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success
400	Status 400	Error

2.9.4. Consumes

- application/json

2.10. POST /joinGroup/

2.10.1. Description

Join a group

2.10.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	groupName	Name of the	true	string	

		group			
QueryParameter	userEmail	Email of the user	true	string	

2.10.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success
400	Status 400	Error

2.10.4. Consumes

- application/json

2.11. GET /getMembers/{group}

2.11.1. Description

It returns all the email's members of a group

2.11.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	group	Name of the group	true	string	

2.11.3. Responses

HTTP Code	Description	Schema
200	Status 200	List of String

2.11.4. Produces

- application/json

2.12. POST /editUser/

2.12.1. Description

Update the database with the new values of the user with this email

2.12.2. Parameters

Type	Name	Description	Required	Schema	Default
BodyParameter	body		true	User	

2.12.3. Responses

HTTP Code	Description	Schema
200	Status 200	

2.12.4. Consumes

- application/json

2.13. POST /editDescription/

2.13.1. Description

Modify the description of a group

2.13.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	group	Name of the group	true	string	
QueryParameter	description	The new description	true	string	

2.13.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success
400	Status 400	Error

2.14. POST /addComment

2.14.1. Description

Add a new comment

2.14.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	username	Email of the user	true	string	
QueryParameter	group	Name of the group	true	string	
QueryParameter	comment	The comment to add	true	string	

2.14.3. Responses

HTTP Code	Description	Schema
200	Status 200	Group
400	Status 400	No Content

2.14.4. Consumes

- application/json

2.15. POST /deleteUser

2.15.1. Description

Delete a user

2.15.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	email	Email of the user to delete	true	string	

2.15.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success
200	Status 200	Error

2.15.4. Consumes

- application/json

2.16. POST /deleteGroup/

2.16.1. Description

Delete a group

2.16.2. Parameters

Type	Name	Description	Required	Schema	Default
QueryParameter	group	Name of the group	true	string	

2.16.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success

400	Status 400	Error
-----	------------	-------

2.16.4. Consumes

- application/json

2.17. POST /createGroup/

2.17.1. Description

Create a new group

2.17.2. Parameters

Type	Name	Description	Required	Schema	Default
BodyParameter	body		false	Group	

2.17.3. Responses

HTTP Code	Description	Schema
200	Status 200	Success

400	Status 400	Error
-----	------------	-------

2.17.4. Consumes

- application/json

2.18. GET /getUsers/

2.18.1. Description

Get all users

2.18.2. Responses

HTTP Code	Description	Schema
200	Status 200	List of User

2.19. GET /getUser/{email}

2.19.1. Description

Get a specific user

2.19.2. Parameters

Type	Name	Description	Required	Schema	Default
PathParameter	email	Name of the user	true	string	

2.19.3. Responses

HTTP Code	Description	Schema
200	Status 200	User

2.20. GET /getUsersGroups/

2.20.1. Description

Return the users and the group the user belongs to

2.20.2. Responses

HTTP Code	Description	Schema
-----------	-------------	--------

200	Status 200	String
-----	------------	--------

2.20.3. Produces

- application/json

3. Definitions

3.1. Comment

Name	Description	Required	Schema	Default
comment	Content of the comment	true	string	
users_Email	Email of the user who did the comment	true	string	
groups_Name	Name of the	true	string	

	group where the comment is sent			
presentTime	When the comment is sent	true	integer (int64)	

3.2. Group

Name	Description	Required	Schema	Default
name	Name of the group	true	string	
description	Description of the group	true	string	
admin_email	Email of the admin of the group	true	string	

3.3. User

Name	Description	Required	Schema	Default
email	Email of the user	true	string	
firstname	Firstname of the user	true	string	
lastname	Lastname of the user	true	string	
password	Password of the user	true	string	
biography	Biography of the user	false	string	

Edited by Jing WANG & Jacomo BERLANDE