Friend Management REST APIs

Table of Contents

Chapter 1. Overview

REST APIs for Friend Management.

1.1. Version information

Version : 1.0.0-SNAPSHOT

1.2. Contact information

Contact : Alvin Flores
Contact Email : alvinjavierflores@gmail.com

1.3. License information

License : Open
Terms of service : null

1.4. URI scheme

Host : localhost
BasePath : /api/v1/

1.5. Tags

- friends-controller : Friends Controller

Chapter 2. Architecture

AWS cloud Services

2.1. Services

Rewards Services
== Rewards

Chapter 3. Resources

3.1. Friends-controller

Friends Controller

3.1.1. Create User Profile

```
POST /friends/
```

Description

Persists a user profile in database

Parameters

Type	Name	Description	Schema
Body	**profile** *required*	User profile details	UserProfileParam

Responses

HTTP Code	Description	Schema
201	Data saved successfully	Status
400	Invalid parameters	ErrorMessage
422	Saving failed	ErrorMessage

Consumes

- `application/json`

Produces

- `application/json`

3.1.2. Get Registered Email Adresses

```
GET /friends/
```

Description

Returns the list of registered email addresses

Responses

HTTP Code	Description	Schema
200	Email list was successfully retrieved	EmailList

Consumes

- `application/json`

Produces

- `application/json`

3.1.3. Create Friend Connection

```
POST /friends/connection/
```

Description

Create a friend connection between two email address

Parameters

Type	Name	Description	Schema
Body	**friends** *required*	Friend connection details	FriendsParam

Responses

HTTP Code	Description	Schema
201	Data saved successfully	Status
400	Invalid parameters	ErrorMessage
422	Saving failed	ErrorMessage

Consumes

- `application/json`

Produces

- `application/json`

3.1.4. Create Block Connection

```
POST /friends/connection/block/
```

Description

Blocked updates from requestor to target email address, no friend connection can be done once the target is blocked

Parameters

Type	Name	Description	Schema
Body	**blockParam** *required*	action details	UserActionParam

Responses

HTTP Code	Description	Schema
201	Data saved successfully	Status
400	Invalid parameters	ErrorMessage
422	Saving failed	ErrorMessage

Consumes

- application/json

Produces

- application/json

3.1.5. Get Friend Connections

```
POST /friends/connection/list/
```

Description

Returns the list of friend connection for a specific email address

Parameters

Type	Name	Description	Schema
Body	**userEmailParam** *required*	User email address	UserEmailParam

Responses

HTTP Code	Description	Schema
200	Data retrieved successfully	FriendsList

HTTP Code	Description	Schema
400	Invalid parameters	ErrorMessage
422	Email address not registered	ErrorMessage

Consumes

- `application/json`

Produces

- `application/json`

3.1.6. Create Subscription

```
POST /friends/connection/subscribe/
```

Description

Creates a subscription between target and requestor email address

Parameters

Type	Name	Description	Schema
Body	**subscriptionP aram** *required*	action details	UserActionParam

Responses

HTTP Code	Description	Schema
201	Data saved successfully	Status
400	Invalid parameters	ErrorMessage
422	Saving failed	ErrorMessage

Consumes

- `application/json`

Produces

- `application/json`

3.1.7. Post Message

```
POST /friends/message/
```

Description

Returns a list of email address eligible to receive message updates

Parameters

Type	Name	Description	Schema
Body	**msgParam** *required*	Message update details	PostMessageParam

Responses

HTTP Code	Description	Schema
200	Retrieved data successfully	UpdateList
400	Invalid parameters	ErrorMessage

Consumes

- application/json

Produces

- application/json

Chapter 4. Definitions

4.1. EmailList

Name	Schema
emailList *optional*	< string > array

4.2. ErrorMessage

Name	Schema
error *optional*	string

4.3. FriendsList

Name	Schema
count *optional*	integer
friends *optional*	< string > array
status *optional*	boolean

4.4. FriendsParam

Name	Schema
friends *optional*	< string > array

4.5. PostMessageParam

Name	Schema
sender *required*	string
text *required*	string

4.6. Status

Name	Schema
status *optional*	boolean

4.7. UpdateList

Name	Schema
recepients *optional*	< string > array
status *optional*	boolean

4.8. UserActionParam

Name	Schema
requestor *required*	string
target *required*	string

4.9. UserEmailParam

Name	Schema
email *required*	string

4.10. UserProfileParam

Name	Schema
emailId *required*	string
fullname *optional*	string