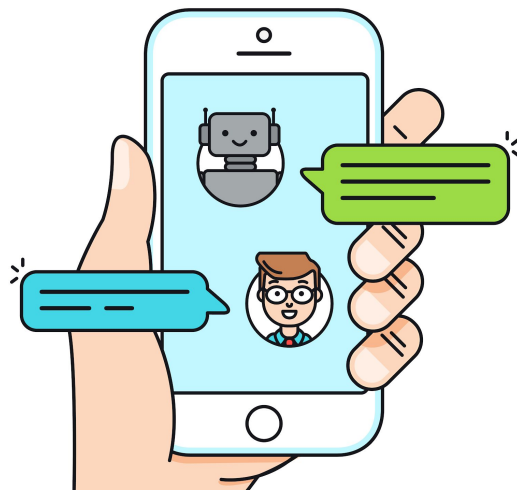




SMT203 Smart City Systems and Management AY 2019/2020 T2 - G2



Exchinder
API User Guide & Documentation
Version 1.0

By Team 4

Members:

Lee Wan Ning

Luo Junru

Ng Xun Jie

Ong Yan Ning

Toh Wang Ting

Yiap Mei Lin Andrea

Table of content

1. Exchinder Overview	3
2. Exchinder API overview	3
3. Making API Calls	4
What API calls can you make?	4
Generic steps on how to make API calls using Postman	4
4. API Documentation	5
4.1 PostIntRecord	5
4.2 GetMatchIntRecords	7
4.3 GetUniversities	9
4.4 GetCountries	10
4.5 UpdateIntRecord	11
5. Error Handling	13
Contact Us	13

1. Exchinder Overview

Exchinder aims to help SMU students find fellow companions when they plan to study abroad for exchange programs. Students will engage with Exchinder telegram bot to find like-minded friends to go on exchange together before they make their decisions as to which university and country to go.

2. Exchinder API overview

Application Program Interfaces (APIs) facilitate data sharing while applications and services communicate with one another. Exchinder API uses a RESTful API design and it provides you with the information of countries available for exchange, the available universities under each respective country and a telegram group chat link for you to join depending on your desired country and university indicated.

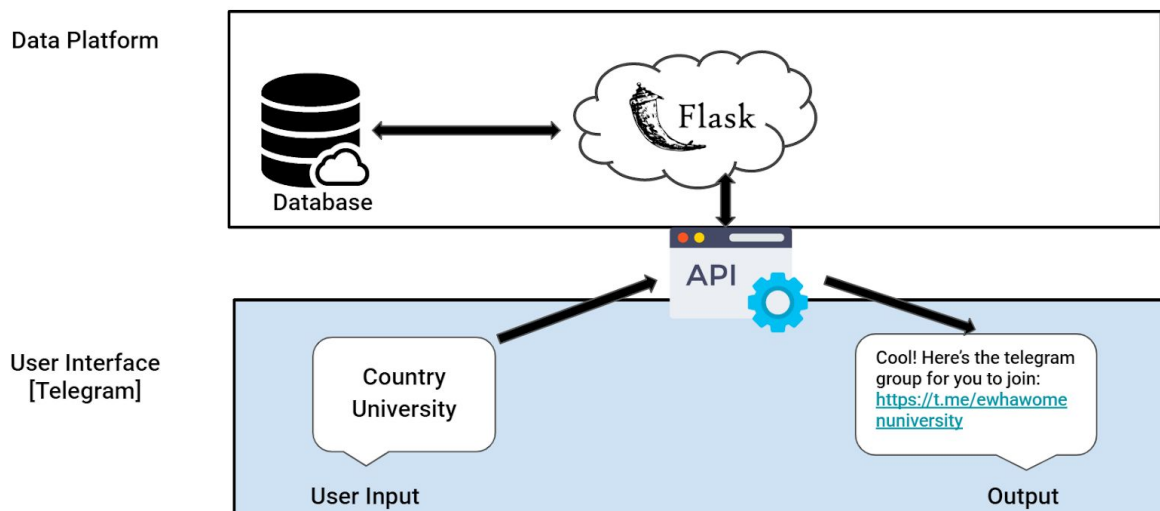


Figure 1

Figure 1 illustrates the system of our project, whereby everything is connected to the API. The user input needs to be processed by the Exchinder API which then communicates with our Exchinder App hosted on Flask server and relevant data will be extracted/changed in our Exchinder database. After which, relevant information will be sent from the database back to the Exchinder App then go through the Exchinder API and finally, the user will receive an output in the form of a text message from the Telegram bot.

3. Making API Calls

API calls can be made in various ways. In this API documentation we will introduce 2 ways of making API calls. The first method is via Python coding scripts, which is recommended for more advanced users. For users that have no prior knowledge on how to make API calls or users unfamiliar with Python programming language, we would recommend the usage of [Postman](#), a third-party application that makes API calling easier.

What API calls can you make?

Method name	HTTP Method	Description
PostIntRecord	POST	Create a new interest record, to be updated to the database
GetMatchIntRecords	GET	Returns matching record in the database, if any
GetUniversities	GET	Returns list of universities for a particular country selected
GetCountries	GET	Returns list of all countries
UpdateIntRecord	PUT	Let user update the DB with the newly created Telegram group URL.

Generic steps on how to make API calls using Postman

Steps to making an API call:

1. Download and install the Postman [here](#).
2. Change the HTTP Method to the one of your choice. E.g POST
3. Enter the URL in the field request URL.
4. Navigate to the “Body” Headers and below it, select the radio button “raw” and “JSON” from the dropdown list.
5. Enter the parameters for the request. Please refer to the [API documentation below](#) for the parameters for the various requests
6. Click on the Send button.

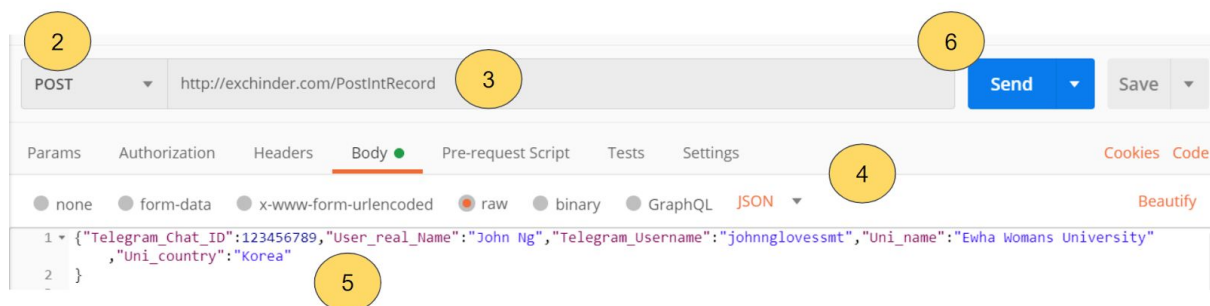


Figure 2

Figure 2 shows an example of using Postman to make API calls. The steps are illustrated above with the number showing where to click for each step.

4. API Documentation

The following lists all real-time datasets that are served out via APIs. Specification for each API can be found in the rest of this document.

4.1 PostIntRecord

URL:	http://exchinder.com/PostIntRecord
Description:	Create a new interest record based on student's interested country and university

Request				
Parameters	Description	Type	Mandatory	Example
Telegram_Chat_ID	The chat_id of individual students, so that the bot can update them through messages.	Integer	Yes	123456789
User_real_Name	Real name of the student	String	Yes	John Ng
Telegram_username	Telegram username of the student	String	Yes	johnnglovessmt
Uni_name	Name of University	String	Yes	Ewha Womans University

Uni_country	Country of University	String	Yes	Korea
Response				
Attributes	Description	Type	Example	
status	Notify the user whether the request is successfully sent	String	Interest record successfully created	

Sample API Request and Response

API Request

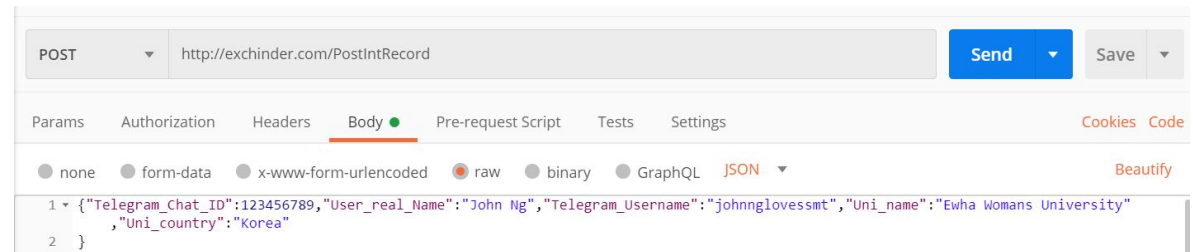
Via Python Script:

```
import requests
import json

params = {"Telegram_Chat_ID": 123456789, "User_real_Name": "John Ng",
"Telegram_username": "johnnglovessmt", "Uni_name": "Ewha Womans University", "Uni_country": "Korea"}

interestRecord = requests.post(url="http://exchinder.com/PostIntRecord",
params=params).json()
```

Via Postman:



API Response

Via Python Script:

```
{
  "status": "Interest record successfully created"
}
```

Via Postman:

```
{
  "status": "Interest record successfully created"
}
```

4.2 GetMatchIntRecords

URL:	http://exchinder.com/GetMatchIntRecords
Description:	Returns matching record in the database, if any.

Request				
Parameters	Description	Type	Mandatory	Example
Uni_name	Name of University	String	Yes	Ewha Womans University
Uni_country	Country of University	String	Yes	Korea
Response				
Attributes	Description	Type	Example	
Telegram_Username	Telegram username of the student	String	johnlovessmt	
User_real_Name (previous student)	Real name of the previous student	String	John Ng	
student_chat_ID	Chat ID of the student (recipient for messages). For example, if Peter is the first student interested in University A and later Jane shows the same interest, Peter will be informed by the bot.	Integer	123456789	
Uni_name	Name of University	String (VARCHAR)	Ewha Womans University	

Sample API Request and Response

API Request

Via Python Script:

```
import requests
import json

params = {"Uni_name": "Ewha Womans University", "Uni_country": "Korea"}

getMatchRecords = requests.get(url="http://exchinder.com/GetMatchIntRecords", params=params)
```

Via Postman:

The screenshot shows the Postman interface for a GET request. The URL bar contains the URL: `http://exchinder.com/GetMatchIntRecords?Uni_name=Ewha Womans University&Uni_country=Korea`. Below the URL bar, the 'Query Params' tab is selected, displaying a table with the following data:

KEY	VALUE	DESCRIPTION	*** Bulk Edit
<input checked="" type="checkbox"/> Uni_name	Ewha Womans University		
<input checked="" type="checkbox"/> Uni_country	Korea		

API Response

Via Python Script:

```
{
  "students":
  [
    {
      "Telegram_Username": "johnlovessmt",
      "User_real_Name": "John Ng",
      "student_chat_ID": 123456789,
      "Uni_name": "Ewha Womans University"
    }
  ]
}
```

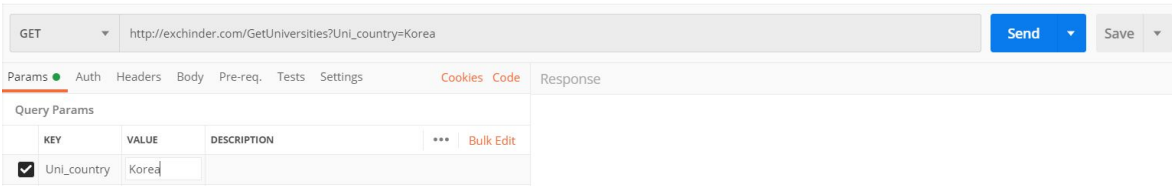
Via Postman:

```
{
  "students":
  [
    { "Telegram_Username": "johnnglovessmt", "User_real_Name": "John Ng", "student_chat_ID": 123456789, "Uni_name": "Ewha Womans University" }
  ]
}
```


4.3 GetUniversities

URL:	http://exchinder.com/GetUniversities
Description:	Returns list of universities for a particular country selected

Request				
Parameters	Description	Type	Mandatory	Example
Uni_country	Country of University	String	Yes	Korea
Response				
Attributes	Description	Type	Example	
Uni_name	Name of University	List of String	Ewha Womans University, Yonsei University, Seoul National University	

Sample API Request and Response
<p>API Request</p> <p><u>Via Python Script:</u></p> <pre>import requests import json params={"Uni_country":"Korea"} UniRecords= requests.get(url="http://exchinder.com/GetUniversities", params=params).json()</pre> <p><u>Via Postman:</u></p> 
<p>API Response</p> <p><u>Via Python Script:</u></p>

```
{
  "Universities":
    [
      {
        "Uni_name": ["Ehwa Womans University", "Yonsei University", "Seoul National University"]
      }
    ]
}
```

Via Postman:

```
{
  "universities":
    [
      {
        "Uni_name":["Ewha Womans University","Yonsei University","Seoul National University"]
      }
    ]
}
```

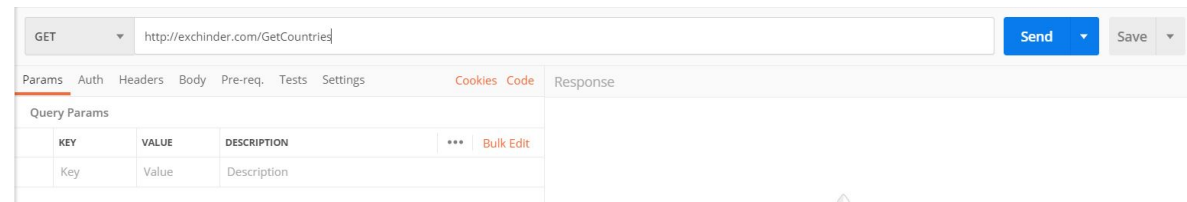
4.4 GetCountries

URL:	http://exchinder.com/GetCountries
Description:	Returns list of all countries when user request for it

Request				
Parameters	Description	Type	Mandatory	Example
NIL	NIL	NIL	NIL	NIL
Response				
Attributes	Description	Type	Example	
Uni_country	Country Names	List of string	South Korea, Taiwan	

Sample API Request and Response
<p>API Request</p> <p><u>Via Python Script:</u></p> <pre>import requests import json UniCountries=requests.get(url="http://exchinder.com/GetCountries").json()</pre>

Via Postman:



API Response

Via Python Script:

```
{
  "Unicountries":
    [
      {
        "Uni_country": ["South Korea", "Taiwan"]
      }
    ]
}
```

Via Postman:

```
{
  "Unicountries":
    [
      {
        "Uni_country": ["South Korea", "Taiwan"]
      }
    ]
}
```

4.5 UpdateIntRecord

URL:	http://exchinder.com/UpdateIntRecord
Description:	Let user update the DB with the newly created Telegram group URL

Request				
Parameters	Description	Type	Mandatory	Example
Group_URL	URL of Telegram group chat	String (VARCHAR)	Yes	t.me/ewhawomen suniversityexchan

				ge
University_ID	ID of university in database	Integer	Yes	35
Response				
Attributes	Description	Type	Example	
status	Notify the user whether the request is successfully sent	String	successfully updated	

Sample API Request and Response

API Request

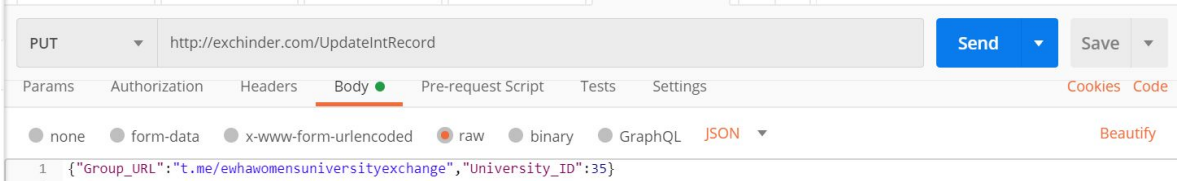
Via Python Script:

```
import requests
import json

json = {"Group_URL": "t.me/ewhawomensuniversityexchange",
        "University_ID": 35}

updateRecord = requests.put(url="http://exchinder.com/UpdateIntRecord",
                             json=json)
```

Via Postman:

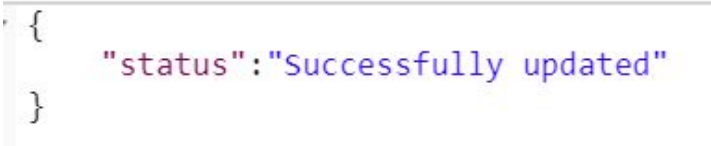


API Response

Via Python Script:

```
{
    "status": "Successfully updated"
}
```

Via Postman:



5. Error Handling

Status Code	Description
1xx: Informational	It means the request has been received and the process is continuing.
2xx: Success	It means the action was successfully received, understood, and accepted.
3xx: Redirection	<div>3xx: Redirection</div> <div>It means further action must be taken in order to complete the request.</div>
4xx: Client Error	It means the request contains incorrect syntax or cannot be fulfilled.
5xx: Server Error	It means the server failed to fulfill an apparently valid request.

Contact Us

We are more than happy to provide assistance if you need it.

For further support on handling errors faced when calling APIs. Please contact us at

Email: ExchinderITSupport@gmail.com

Hotline: 8888 8505

For any other general enquiries, please contact us at

Email: ExchinderGenEng@gmail.com

Hotline: 8888 8888