

Manual Test Plan(Postman)

Test Version

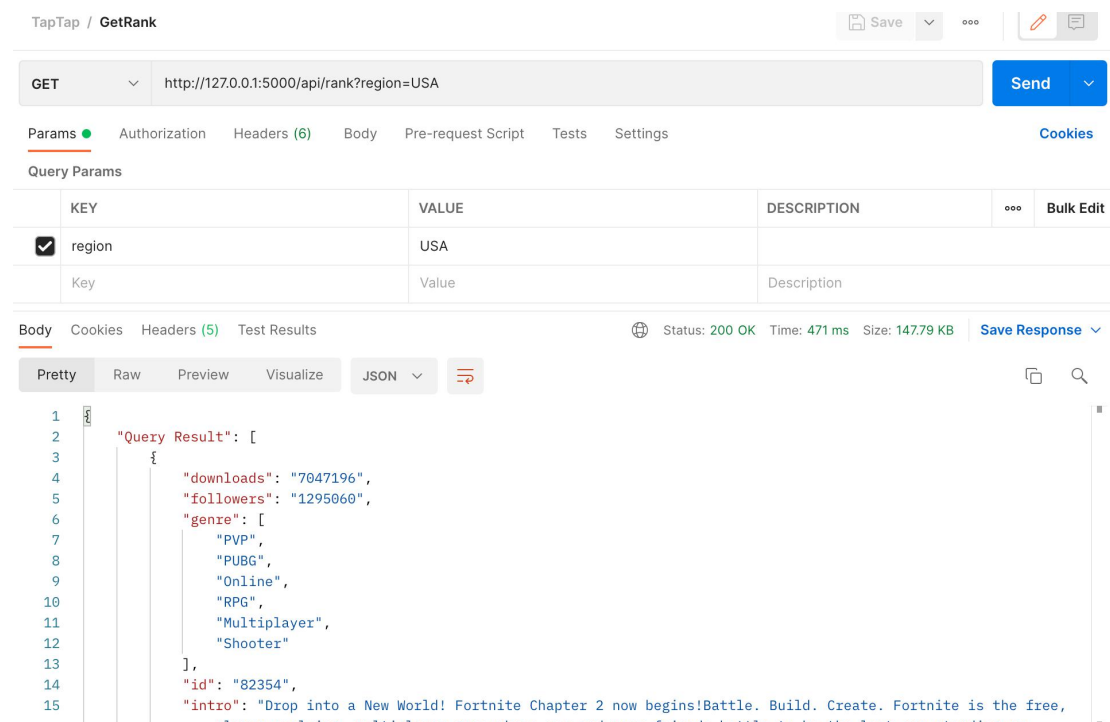
Branch: project-2.0

Environment Setup & Configuration

- All set up in previous manual test plan is also required for this plan.
- Test Tool: Postman. Download here: <https://www.postman.com/>
- Run web_api.py in Pycharm first.

Manual Test Process

(1) Test /api/rank route. The following get request should work. Query result should be the info of top 15 games in this region.



If region is not specified in query argument or incorrect region name is provided, then status code should not be 200 and server should return error message:

GET

http://127.0.0.1:5000/api/rank?

Send

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>					
	Key	Value	Description		

Body

Cookies

Headers (5)

Test Results

Status: 400 BAD REQUEST

Time: 162 ms

Size: 241 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1
2  "Error_message": "Query arg region is not valid"
3
```

GET

http://127.0.0.1:5000/api/rank?region=Apple

Send

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	region	Apple			
	Key	Value	Description		

Body

Cookies

Headers (5)

Test Results

Status: 400 BAD REQUEST

Time: 407 ms

Size: 241 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1
2  "Error_message": "Query arg region is not valid"
3
```

(2) Test /api/genre route. The following get request should work. Query result should be a list of dict containing info of games containing this genre.

GET ▼ http://127.0.0.1:5000/api/genre?genre=Pixel Send ▼

Params ● Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	genre	Pixel			
	Key	Value	Description		

Body Cookies Headers (5) Test Results ⊕ Status: 200 OK Time: 213 ms Size: 90.05 KB Save Response ▼

Pretty Raw Preview Visualize JSON ▼ ⌵

```
1  {
2    "Query Result": [
3      {
4        "downloads": "785730",
5        "followers": "152938",
6        "genre": [
7          "Online",
8          "Multiplayer",
9          "Sandbox",
10         "Open World",
11         "Pixel",
12         "Freedom",
13         "Creative",
14         "DIY",
15         "Building"
16       ]
17     }
18   ]
19 }
```

Bootcamp Runner Trash

If genre is not specified in query argument or incorrect genre name is provided, then status code should not be 200 and server should return error message:

TapTap / GetGenre Save ▼ ... ✎ 💬

GET ▼ http://127.0.0.1:5000/api/genre Send ▼

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (5) Test Results ⊕ Status: 400 BAD REQUEST Time: 9 ms Size: 240 B Save Response ▼

Pretty Raw Preview Visualize JSON ▼ ⌵

```
1  {
2    "Error_message": "Query arg genre is not valid"
3  }
```

GET ▼ http://127.0.0.1:5000/api/genre?genre=null Send ▼

Params ● Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	genre	null			
	Key	Value	Description		

Body Cookies Headers (5) Test Results ⌐ Status: 400 BAD REQUEST Time: 55 ms Size: 240 B Save Response ▼

Pretty Raw Preview Visualize JSON ▼ ⌐ 🔍

```

1  [
2    "Error_message": "Query arg genre is not valid"
3  ]

```

(3) Test /api/game route. The following get request should work. Body should contain list of game id for query. Content type must be application/json. Query result is info of all requested games.

GET ▼ http://127.0.0.1:5000/api/game Send ▼

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings Cookies

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL ● JSON ▼ Beautify

1 ["2247", "214145"]

Body Cookies Headers (5) Test Results ⌐ Status: 200 OK Time: 106 ms Size: 16.74 KB Save Response ▼

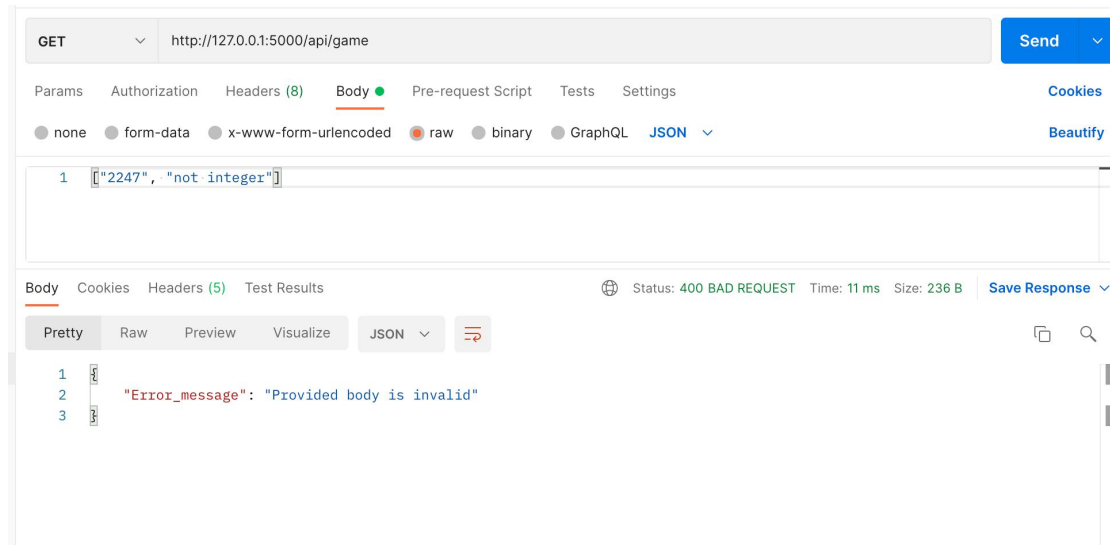
Pretty Raw Preview Visualize JSON ▼ ⌐ 🔍

```

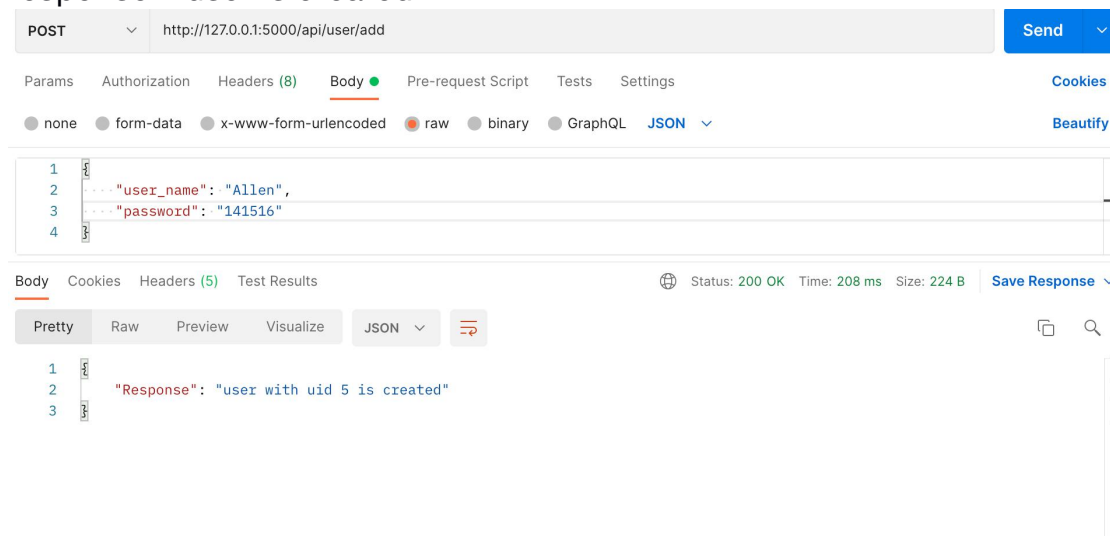
1  [
2    "Query Result": [
3      {
4        "downloads": "1093355",
5        "followers": "252324",
6        "genre": [
7          "Fighting",
8          "Action",
9          "ACG",
10         "Naruto",
11         "Side-scroll",
12         "Anime",
13         "JRPG"
14       ],
15       "id": "2247",

```

If body contain invalid data structure or content type is incorrect, then status code should not be 200 and server should return error message:



(4) Test /api/user/add route. The following post request should work. Body should contain a dict specifying user name and password. Content type must be application/json. Server should return success response if user is created:



If body contain invalid data structure or content type is incorrect, then status code should not be 200 and server should return error message:

POST http://127.0.0.1:5000/api/user/add

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 {
2   "invalid_key": "Allen",
3   "password": "141516"
4 }
```

Body Cookies Headers (5) Test Results Status: 400 BAD REQUEST Time: 21 ms Size: 236 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "Error_message": "Provided body is invalid"
3 }
```

(5) Test /api/user/password route. The following get request should work. User should specify id in parameter. Query result is password of this user

GET http://127.0.0.1:5000/api/user/password?id=3

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	3			
	Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 200 OK Time: 346 ms Size: 208 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "Query Result": "141516"
3 }
```

If id is not specified in query argument or invalid id(user not exist) is provided, then status code should not be 200 and server should return error message:

GET <http://127.0.0.1:5000/api/user/password?id=7> [Send](#)

Params [Authorization](#) [Headers \(6\)](#) [Body](#) [Pre-request Script](#) [Tests](#) [Settings](#) [Cookies](#)

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> id	7			
Key	Value	Description		

Body [Cookies](#) [Headers \(5\)](#) [Test Results](#) [Status: 400 BAD REQUEST](#) [Time: 56 ms](#) [Size: 237 B](#) [Save Response](#)

[Pretty](#) [Raw](#) [Preview](#) [Visualize](#) [JSON](#) [Send](#)

```
1 {
2   "Error_message": "Query arg id is not valid"
3 }
```

(6) Test /api/user/password route. The following put request should work. User should specify id and password in parameter. Server should return success response

PUT <http://127.0.0.1:5000/api/user/password?id=1&password=123> [Send](#)

Params [Authorization](#) [Headers \(7\)](#) [Body](#) [Pre-request Script](#) [Tests](#) [Settings](#) [Cookies](#)

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> id	1			
<input checked="" type="checkbox"/> password	123			
Key	Value	Description		

Body [Cookies](#) [Headers \(5\)](#) [Test Results](#) [Status: 200 OK](#) [Time: 131 ms](#) [Size: 237 B](#) [Save Response](#)

[Pretty](#) [Raw](#) [Preview](#) [Visualize](#) [JSON](#) [Send](#)

```
1 {
2   "Response": "user with uid 1 changes password to 123"
3 }
```

If id or password is not specified in query argument or invalid id(user not exist) is provided, then status code should not be 200 and server should return error message:

PUT ▼ http://127.0.0.1:5000/api/user/password?id=7&password=123 Send ▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	7			
<input checked="" type="checkbox"/>	password	123			
	Key	Value	Description		

Body Cookies Headers (5) Test Results 🌐 Status: 400 BAD REQUEST Time: 51 ms Size: 237 B Save Response ▼

Pretty Raw Preview Visualize JSON ▼ 🔍

```
1
2  "Error_message": "Query arg id is not valid"
3
```

(7) Test /api/user/info route. The following get request should work. User should specify id in parameter. Query result is all info of this user

GET ▼ http://127.0.0.1:5000/api/user/info?id=1 Send ▼

Params ● Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	1			
	Key	Value	Description		

Body Cookies Headers (5) Test Results 🌐 Status: 200 OK Time: 93 ms Size: 318 B Save Response ▼

Pretty Raw Preview Visualize JSON ▼ 🔍

```
1
2  "Query Result": [
3    {
4      "favourite_list": [],
5      "id": "1",
6      "password": "123",
7      "user_name": "Allen"
8    }
9  ]
10
```

If id is not specified in query argument or invalid id(user not exist) is provided, then status code should not be 200 and server should return error message:

GET http://127.0.0.1:5000/api/user/info?id=7 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	7			
	Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 400 BAD REQUEST Time: 58 ms Size: 237 B Save Response

Pretty Raw Preview Visualize JSON

```

1  "Error_message": "Query arg id is not valid"
2
3

```

(8) Test /api/user/favourite route. The following put request should work. User should specify user_id and game_id in parameter. Server should return success response

PUT http://127.0.0.1:5000/api/user/favourite?user_id=2&game_id=2247 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	user_id	2			
<input checked="" type="checkbox"/>	game_id	2247			
	Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 200 OK Time: 302 ms Size: 233 B Save Response

Pretty Raw Preview Visualize JSON

```

1  "Response": "Game 2247 added to user 2 favourite"
2
3

```

If id or password is not specified in query argument or invalid id(user/game not exist) is provided, then status code should not be 200 and server should return error message:

PUT http://127.0.0.1:5000/api/user/favourite?user_id=2&game_id=3 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	user_id	2			
<input checked="" type="checkbox"/>	game_id	3			
	Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 400 BAD REQUEST Time: 161 ms Size: 251 B Save Response

Pretty Raw Preview Visualize JSON Copy Search

```
1 {  
2   "Error_message": "Query arg user_id/game_id are not valid"  
3 }
```

(9) Test /api/user/favourite route. The following delete request should work. User should specify user_id and game_id in parameter. Server should return success response

DELETE http://127.0.0.1:5000/api/user/favourite?user_id=2&game_id=2247 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	user_id	2			
<input checked="" type="checkbox"/>	game_id	2247			
	Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 200 OK Time: 246 ms Size: 236 B Save Response

Pretty Raw Preview Visualize JSON Copy Search

```
1 {  
2   "Response": "Game 2247 remove from user 2 favourite"  
3 }
```

If id or password is not specified in query argument or invalid id(user/game not exist) is provided, then status code should not be 200 and server should return error message:

DELETE ▼ http://127.0.0.1:5000/api/user/favourite?user_id=2&game_id=2247 Send ▼

Params ● Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	user_id	2			
<input checked="" type="checkbox"/>	game_id	2247			
	Key	Value	Description		

Body Cookies Headers (5) Test Results ⊕ Status: 400 BAD REQUEST Time: 238 ms Size: 257 B Save Response ▼

Pretty Raw Preview Visualize JSON ▼ ⌵

```

1  {
2    "Error_message": "User doesn't have this game in favourite list"
3  }

```

(10) Test /api/search route. The following get request should work. User should specify query string in parameter. Query result is all info of games satisfying this query string

GET ▼ http://127.0.0.1:5000/api/search?query=Genre: RPG AND IJRPG AND Card; Rating: > 7.0 Send ▼

Params ● Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	query	Genre: RPG AND IJRPG AND Card; Rating: > 7.0			
	Key	Value	Description		

Body Cookies Headers (5) Test Results ⊕ Status: 200 OK Time: 66 ms Size: 8.17 KB Save Response ▼

Pretty Raw Preview Visualize JSON ▼ ⌵

```

1  {
2    "Query Result": [
3      {
4        "downloads": "17448",
5        "followers": "69720",
6        "genre": [
7          "Idle",
8          "Hang up",
9          "Card",
10         "RPG"
11       ],
12       "id": "150204",
13       "intro": "*** SUMMON - COLLECT - LEVEL UP ***Over 100 heroes in seven factions. Build the ultimate
               formations and battle your way to the top. Celestials, Hypogeans, Dimensionals are the rarest. ***
               STRATEGIC BATTLE ***Equip your heroes and strategize unbeatable lineups to overcome your enemies in

```

If query is not specified in parameter or invalid query string is provided, then status code should not be 200 and server should return error message:

GET

▼

http://127.0.0.1:5000/api/search

Send ▼

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body

Cookies

Headers (5)

Test Results

⌐ Status: 400 BAD REQUEST Time: 12 ms Size: 238 B Save Response ▼

Pretty

Raw

Preview

Visualize

JSON ▼

≡

1

2

3

12345678910111213141516171819202122232425262728293031323334353637383940414243444546474849505152535455565758596061626364656667686970717273747576777879808182838485868788899091929394959697989910010110210310410510610710810911011111211311411511611711811912012112212312412512612712812913013113213313413513613713813914014114214314414514614714814915015115215315415515615715815916016116216316416516616716816917017117217317417517617717817918018118218318418518618718818919019119219319419519619719819920020120220320420520620720820921021121221321421521621721821922022122222322422522622722822923023123223323423523623723823924024124224324424524624724824925025125225325425525625725825926026126226326426526626726826927027127227327427527627727827928028128228328428528628728828929029129229329429529629729829930030130230330430530630730830931031131231331431531631731831932032132232332432532632732832933033133233333433533633733833934034134234334434534634734834935035135235335435535635735835936036136236336436536636736836937037137237337437537637737837938038138238338438538638738838939039139239339439539639739839940040140240340440540640740840941041141241341441541641741841942042142242342442542642742842943043143243343443543643743843944044144244344444544644744844945045145245345445545645745845946046146246346446546646746846947047147247347447547647747847948048148248348448548648748848949049149249349449549649749849950050150250350450550650750850951051151251351451551651751851952052152252352452552652752852953053153253353453553653753853954054154254354454554654754854955055155255355455555655755855956056156256356456556656756856957057157257357457557657757857958058158258358458558658758858959059159259359459559659759859960060160260360460560660760860961061161261361461561661761861962062162262362462562662762862963063163263363463563663763863964064164264364464564664764864965065165265365465565665765865966066166266366466566666766866967067167267367467567667767867968068168268368468568668768868969069169269369469569669769869970070170270370470570670770870971071171271371471571671771871972072172272372472572672772872973073173273373473573673773873974074174274374474574674774874975075175275375475575675775875976076176276376476576676776876977077177277377477577677777877978078178278378478578678778878979079179279379479579679779879980080180280380480580680780880981081181281381481581681781881982082182282382482582682782882983083183283383483583683783883984084184284384484584684784884985085185285385485585685785885986086186286386486586686786886987087187287387487587687787887988088188288388488588688788888989089189289389489589689789889990090190290390490590690790890991091191291391491591691791891992092192292392492592692792892993093193293393493593693793893994094194294394494594694794894995095195295395495595695795895996096196296396496596696796896997097197297397497597697797897998098198298398498598698798898999099199299399499599699799899910001001100210031004100510061007100810091010101110121013101410151016101710181019102010211022102310241025102610271028102910301031103210331034103510361037103810391040104110421043104410451046104710481049105010511052105310541055105610571058105910601061106210631064106510661067106810691070107110721073107410751076107710781079108010811082108310841085108610871088108910901091109210931094109510961097109810991100110111021103110411051106110711081109111011111112111311141115111611171118111911201121112211231124112511261127112811291130113111321133113411351136113711381139114011411142114311441145114611471148114911501151115211531154115511561157115811591160116111621163116411651166116711681169117011711172117311741175117611771178117911801181118211831184118511861187118811891190119111921193119411951196119711981199120012011202120312041205120612071208120912101211121212131214121512161