

FLASK API DEVELOPMENT

BY

VENNAM DEVENDHAR

This Project Documentation Submitted to Codegnan

Organization(<https://codegnan.com/>) for Bootcamp Competition JUN-2020

which is organized by Codegnan.

Submitted By :

Name : Vennam Devendhar

Mail-ID : vennam1devendhar@gmail.com

Mobile No : 8143567195

submitted To:

Organization Name : <https://codegnan.com/>

Organization Mai-ID : codegnanceo@gmail.com

Organization mobile No : 6301341478

Project Aim:

This project main aim to be gives the best user interaction to the end user who is going to this API End points and these can be achieved by flasgger package in Python .This API handling the End User requests dynamically and provide the response in time.

The FLASK API is mainly protected with API KEY . If End user does not have valid API KEY ,they are not going to perform some operation with API End Points(EX: ChangePassword,ForgotPassword,DeleteUserName).

The each end point will perform unique tasks for handling requests from the end users.

The Total Number of end points shown below:

GET	/Admin	Admin Page
POST	/ChangePassword	Let's Change Password
POST	/DeleteUserName	Let's Change Password
POST	/ForgotPassword	Forgot Password
GET	/Login	Let's Login Now
POST	/Signup	Sign Up Page
GET	/UserNameAvailability	Let's Check the UserName available or not

Project Main Features:

- 1: This API end Points are secured with API KEY
- 2: The API end Points process the end user requests in dynamic manner.
- 3: Better UI Developed for each and every end point.
- 4: Live API has created for all end points which already shown above.

Link :<https://codegnan.herokuapp.com/apidocs>

Everyone can test the above end point for functionality of application

NOTE: for Forgot_Password end point sending mail has been disabled for above link.

In that case OTP is UserName itself for Changing Password.

- 5: we can hit the API end points by CURL command also.

```
Kindly go and check UserName_availability_end_point for current venna1.  
C:\Users\venna>curl -X POST "http://127.0.0.1:5000/Signup?UserName=vennam2&Password=test&FullName=vennam&Email=vennamdev  
endhar%40gmail.com" -H "accept: application/json"  
vennam2 details are stored successfully  
C:\Users\venna>
```

Introduction :

Flask:

Flask is a lightweight WSGI web application framework.

Installation:

Pip install flask

Response:

```
http://127.0.0.1:5000/UserNameAvailability?UserName=vennam
```

```
UserName is available.
```

Flasgger:

Flasgger also comes embedde with SwaggerUI

We can access <http://localhost:5000/apidocs>

Installation:

Pip install flasgger

Response:

The screenshot shows the SwaggerUI interface for the `/UserNameAvailability` endpoint. The method is `GET`. The description is "Let's Check the UserName available or not". Below the description, it says "This is used for cheching avallibility of UserName". The parameters section shows a required query parameter `UserName` of type `string` with the value `vennam` entered. The responses section shows a `200` response with the description "This will be showing UserName availibilty as Output". The response content type is set to `application/json`.

Name	Description
UserName * required	vennam
string (query)	

Code	Description
200	This will be showing UserName availibilty as Output

Response:

Curl

```
curl -X GET "http://127.0.0.1:5000/UserNameAvailability?UserName=ram" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/UserNameAvailability?UserName=ram
```

Server response

Code	Details
200	<div><p>Response body</p><div>UserName is available.</div><div>Download</div></div> <p>Response headers</p> <pre>content-length: 22 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 13:22:22 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

flask-ngrok.

A secure URL is provided for the server running on Colab/Local host.
Traffic stats can be used to debug in case of any errors in your web app.

Installation:

pip install flask-ngrok

```
Session Status      online
Account             Vennam Devendhar (Plan: Free)
Version             2.3.35
Region             United States (us)
Web Interface       http://127.0.0.1:4040
Forwarding           http://aafe989668ea.ngrok.io -> http://localhost:5000
Forwarding           https://aafe989668ea.ngrok.io -> http://localhost:5000
```

Traffics Analysis:

Available on : <http://127.0.0.1:4040/>

ngrokonline

Inspect

Status

Documentation

Filter by

All Requests

Clear

GET /flasgger_static/favicon-32x32.png	200 OK	56.98ms
GET /apispec_1.json	200 OK	77ms
GET /flasgger_static/lib/jquery.min.js	200 OK	11.97s
GET /flasgger_static/swagger-ui-standalone-preset.js	200 OK	23.86s
GET /flasgger_static/swagger-ui-bundle.js	200 OK	35.1s
GET /flasgger_static/swagger-ui.css	200 OK	8.32s
GET /apidocs/	200 OK	171.36ms
GET /apidocs	308 PERMANENT REDIRECT	18.99ms
GET /favicon.ico	404 NOT FOUND	6ms
GET /	200 OK	7.01ms

1 minute agoDuration 7.01msIP 2401:4900:22a1:8c2e:3534:5578:b1ec:f8a7

GET /

SummaryHeadersRawBinary

Replay

200 OK

SummaryHeadersRawBinary

11 bytes text/html; charset=utf-8

Welcome All

ngrokonline

Inspect

Status

Documentation

Configuration

Tunnels

online - server 2.3.35

command_line

URL	https://aafe989668ea.ngrok.io
Addr	http://localhost:5000
Inspect	enabled
Proto	https

command_line (http)

URL	http://aafe989668ea.ngrok.io
Addr	http://localhost:5000
Inspect	enabled
Proto	http

Metrics

Connections

tunnel	total	open	/sec 1m	/sec 5m	/sec 15m
command_line	0	0	0.00	0.00	0.00
command_line (http)	10	0	0.05	0.03	0.01
All	10	0	0.05	0.03	0.01

Connection Durations

(in seconds)

tunnel	50%	90%	95%	99%
command_line	0.00	0.00	0.00	0.00
command_line (http)	0.64	34.56	35.40	35.40
All	0.64	34.56	35.40	35.40

5

Global

authtokenPrefix	1cqKi8phERYIvE6xxNrOu1WAYYE
configPathsList	C:\Users\lenna/.ngrok2/ngrok.yml
consoleUi	enabled
consoleUiColor	black
heartbeatInterval	10s
heartbeatTolerance	15s
inspectDbSize	52428800
logFormat	logfmt
logLevel	info
logTarget	false
metadata	no value
proxy	no value
region	us
rootCas	ngrok.com trusted root
serverAddr	tunnel.us.ngrok.com:443

HTTP Requests

tunnel	total	/sec 1m	/sec 5m	/sec 15m
command_line	0	0.00	0.00	0.00
command_line (http)	10	0.03	0.02	0.01
All	10	0.03	0.02	0.01

HTTP Request Durations (in seconds)

tunnel	50%	90%	95%	99%
command_line	0.00	0.00	0.00	0.00
command_line (http)	0.12	33.98	35.10	35.10
All	0.12	33.98	35.10	35.10

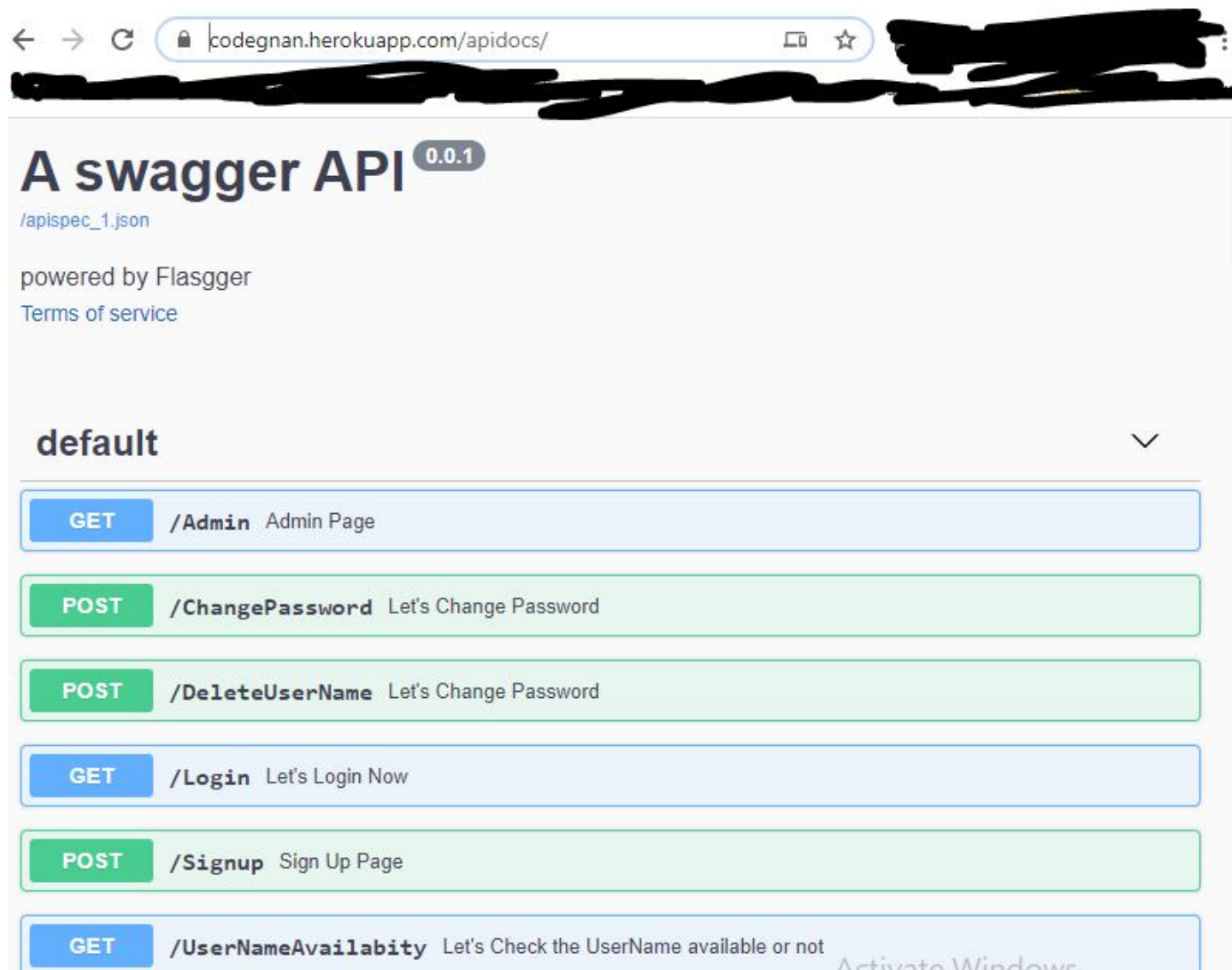
Activate Windows
Go to Settings to activate Windows.

 Ask a question

Live API End Points:

The live API has created and accessed by given below URL

<https://codegnan.herokuapp.com/apidocs/>



CURL Request:

Curl

```
curl -X GET "https://codegnan.herokuapp.com/Login?UserName=raj&Password=test" -H "accept: application/json"
```

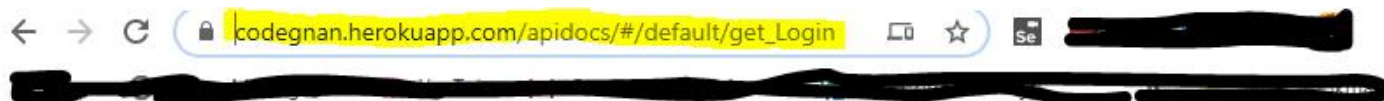
Request URL

<https://codegnan.herokuapp.com/Login?UserName=raj&Password=test>

Response:

Response body

```
{
  "result": [
    {
      "UserName": "raj",
      "Password": "098f6bcd4621d373cade4e832627b4f6",
      "FullName": "Raj kumar",
      "Email": "raj@gmail.com"
    }
  ]
}
```



Responses

Response content type

application/json

Curl

```
curl -X GET "https://codegnan.herokuapp.com/Login?UserName=raj&Password=test" -H "accept: application/json"
```

Request URL

<https://codegnan.herokuapp.com/Login?UserName=raj&Password=test>

Server response

Code

Details

200

Response body

```
{
  "result": [
    {
      "UserName": "raj",
      "Password": "098f6bcd4621d373cade4e832627b4f6",
      "FullName": "Raj kumar",
      "Email": "raj@gmail.com"
    }
  ]
}
```

Download

Activate Windows

Postman:

GET <https://codegnan.herokuapp.com/Login?UserName=raj&Password=test> Send

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

Query Params

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> UserName	raj	
<input checked="" type="checkbox"/> Password	test	
Key	Value	Description

Body Cookies Headers (6) Test Results Status: 200 OK Time: 1359 ms Size: 306 B Sav

Pretty Raw Preview Visualize JSON

```
1 {
2   "result": [
3     {
4       "UserName": "raj",
5       "Password": "098f6bcd4621d373cade4e832627b4f6",
6       "FullName": "Raj kumar",
7       "Email": "raj@gmail.com"
8     }
9   ]
10 }
```

Activate Windows
Go to Settings to activate W

API END-POINTS:

1. GET/UserNameAvailability :

Method 1:(Without Swagger UI)

Case1: If UserName is not taken by anyone

Parameters:

UserName :String

```
http://127.0.0.1:5000/UserNameAvailability?UserName=vennam
```

Response:

```
UserName is available.
```

Case2: If UserName is already registered by someone

Parameters:

UserName :String

```
http://127.0.0.1:5000/UserNameAvailability?UserName=vennam
```

Response:

```
UserName is already taken kindly choose another one
```

Method 2:(With Swagger UI)

Parameters:
 UserName :String

GET

/UserNameAvailability

Let's Check the UserName available or not

This is used for cheching avallability of UserName

Parameters

Try it out

Name	Description
UserName * required string (query)	<input type="text" value="vennam"/>

Responses

Response content type application/json

Code	Description
200	This will be showing UserName avaliabilty as Output

Activate Windows
Go to Settings to activate Windows.

- Click on “Try it out”

GET

/UserNameAvailability

Let's Check the UserName available or not

This is used for cheching avallability of UserName

Parameters

Cancel

Name	Description
UserName * required string (query)	<input type="text" value="vennam"/>

Execute

Responses

Response content type application/json

Code	Description
200	This will be showing UserName avaliabilty as Output

Activate Windows

- Enter UserName and click Execute

Case1: If UserName is already registered by someone

Response:

Curl

```
curl -X GET "http://127.0.0.1:5000/UserNameAvailability?UserName=vennam" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/UserNameAvailability?UserName=vennam
```

Server response

Code	Details
200	<p>Response body</p> <p>UserName is already taken kindly choose another one</p> <p>Response headers</p> <pre>content-length: 51 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 10:47:24 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

Responses

Code	Description
200	This will be showing UserName availability as Output

Activate Windows
Go to Settings to activate Windows.

Case2: If UserName is not taken by anyone

Response:

Curl

```
curl -X GET "http://127.0.0.1:5000/UserNameAvailability?UserName=ram" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/UserNameAvailability?UserName=ram
```

Server response

Code	Details
200	<p>Response body</p> <p>UserName is available.</p> <p>Response headers</p> <pre>content-length: 22 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 13:22:22 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

2. POST/Signup:

Method 1:(Without Swagger UI)

Parameters:

UserName : String
Password :String
FullName :String
Email :String

Request URL

```
http://127.0.0.1:5000/Signup?UserName=vennam1&Password=test&FullName=vennam&Email=vennamdevendhar%40gmail.com
```

Case1: If UserName is already registered by someone

Response:

```
kindly go and check UserName_availability end_point for current vennam1.
```

Case2: If UserName is Unique.

Response:

```
vennam12 details are stored successfully
```

Method 2:(With Swagger UI)

Parameters:

UserName : String
Password :String
FullName :String
Email :String

POST /Signup Sign Up Page

This is using for signup

Parameters Try it out

Name	Description
UserName * required string (query)	<input type="text" value="UserName"/>
Password * required string (query)	<input type="text" value="Password"/>
FullName * required string (query)	<input type="text" value="FullName"/>
Email * required string (query)	<input type="text" value="Email"/>

Activate Windows
Go to Settings to activate Windows.

- Click on “Try it out”
- Enter Details correctly and click **Execute**

POST /Signup Sign Up Page

This is using for signup

Parameters Cancel

Name	Description
UserName * required string (query)	<input type="text" value="vennam"/>
Password * required string (query)	<input type="text" value="test"/>
FullName * required string (query)	<input type="text" value="vennam"/>
Email * required string (query)	<input type="text" value="vennamdevendhar@gmail.com"/>

Activate Windows
Go to Settings to activate Windows.

Execute

Case1: If UserName is already registered by someone

Response:

Responses Response content type application/json

Curl

```
curl -X POST "http://127.0.0.1:5000/Signup?UserName=vennam&Password=test&FullName=vennam&Email=vennamdevendhar%40gmail.com" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/Signup?UserName=vennam&Password=test&FullName=vennam&Email=vennamdevendhar%40gmail.com
```

Server response

Code	Details
200	<div>Response body<div>kindly go and check UserName_availability_end_point for current vennam.</div><div>Download</div></div> <div>Response headers<div>content-length: 69 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 10:58:54 GMT server: Werkzeug/1.0.0 Python/3.7.6</div></div>

Responses Activate Windows
Go to Settings to activate Windows.

Code	Description
------	-------------

Case2: If UserName is Unique.

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/Signup?UserName=vennam1&Password=test&FullName=vennam&Email=vennamdevendhar%40gmail.com" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/Signup?UserName=vennam1&Password=test&FullName=vennam&Email=vennamdevendhar%40gmail.com
```

Server response

Code	Details
200	<div>Response body<div>vennam1 details are stored successfully</div><div>Download</div></div> <div>Response headers<div>content-length: 39 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:01:07 GMT server: Werkzeug/1.0.0 Python/3.7.6</div></div>

Responses Activate Windows
Go to Settings to activate Windows.

Code	Description
200	The output returns whether the user details successfully added or not

3. GET/Login :

Method 1:(Without Swagger UI)

Parameters:

UserName : String

Password :String

Request URL

```
http://127.0.0.1:5000/Login?UserName=venna&Password=test
```

Case1: If End user enters correct details

Response:

```
{
  "result": [
    {
      "UserName": "vennam1",
      "Password": "098f6bcd4621d373cade4e832627b4f6",
      "FullName": "vennam",
      "Email": "vennamdevendhar@gmail.com"
    }
  ]
}
```

Case1: If End user enters incorrect details

Response:

Response body

```
{
  "Error": "Invalid UserName and Password not available in Database."
}
```


Method 2:(With Swagger UI)

Parameters:
 UserName : String
 Password :String

GET

/Login

Let's Login Now

This is using for Login

Parameters

Try it out

Name	Description
UserName * required string (query)	<input type="text" value="UserName"/>
Password * required string (query)	<input type="text" value="Password"/>

Responses

Response content type

application/json

Code	Description
200	The output returns the end user details

Activate Windows
Go to Settings to activate Windows.

- Click on “Try it out”
- Enter Details correctly and click **Execute**

This is using for Login

Parameters

Cancel

Name	Description
UserName * required string (query)	<input type="text" value="vennam1"/>
Password * required string (query)	<input type="text" value="test"/>

Execute

Clear

Case1: If End user enters correct details

Response:

Curl

```
curl -X GET "http://127.0.0.1:5000/Login?UserName=vennam1&Password=test" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/Login?UserName=vennam1&Password=test
```

Server response

Code	Details
200	<div><p>Response body</p><pre>{ "result": [{ "UserName": "vennam1", "Password": "098f6bcd4621d373cade4e832627b4f6", "FullName": "vennam", "Email": "vennamdevendhar@gmail.com" }] }</pre><p>Download</p><p>Response headers</p><pre>content-length: 145 content-type: application/json date: Tue, 09 Jun 2020 11:15:17 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Activate Windows
Go to Settings to activate Windows

Case2: If End user enters incorrect details

Response:

Request URL

```
http://127.0.0.1:5000/Login?UserName=venna&Password=test
```

Server response

Code	Details
200	<div><p>Response body</p><pre>{ "Error": "Invalid UserName and UserName not available in Database." }</pre><p>Download</p><p>Response headers</p><pre>content-length: 70 content-type: application/json date: Tue, 09 Jun 2020 11:17:16 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Responses

Code	Description
200	The output returns the end user details

Activate Windows

4. POST/ForgotPassword:

Method 1:(Without Swagger UI)

Parameters:

UserName : String

Header :API KEY

Request URL

```
http://127.0.0.1:5000/ForgotPassword?UserName=vennam1
```

Case1: If username is valid and API key is not Valid

Response:

```
Enter correct API KEY for Authentication.
```

Case2: If username and API key are Valid

Response:

```
Mail has been sent to Registered mail id.
```

Case2: If username is not valid and API key is valid

Reponse:

```
venna details not found kindly enter valid UserName.
```

Method 2:(With Swagger UI):

Parameters:
 UserName : String
 Header :API KEY

POST

/ForgotPassword

Forgot Password

This is using for forgot Password.

Parameters

Try it out

Name	Description
UserName * required string (query)	<input type="text" value="UserName"/>
API_KEY * required string (header)	<input type="text" value="API_KEY - an authorization header"/>

Responses

Response content type

application/json

Code	Description
200	The output returns message whether mail has been sent or not

Activate Windows
Go to Settings to activate Windows.

- Click on “Try it out”
- Enter Details correctly and click **Execute**

POST

/ForgotPassword

Forgot Password

This is using for forgot Password.

Parameters

Cancel

Name	Description
UserName * required string (query)	<input type="text" value="vennam1"/>
API_KEY * required string (header)	<input type="text" value="W09AU4KifLhAm-USx2sQl7xT2icnXsBU9g"/>

Execute

Responses

Response content type

application/json

Code	Description
200	The output returns message whether mail has been sent or not

Activate Windows
Go to Settings to activate Windows.

Case1: If username is valid and API key is not Valid

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/ForgotPassword?UserName=vennam1" -H "accept: application/json" -H "API_KEY: W09AU4KifLhAm-USx2sQI7xT2icnXs8U9"
```

Request URL

```
http://127.0.0.1:5000/ForgotPassword?UserName=vennam1
```

Server response

Code	Details
200	<div><p>Response body</p><p>Enter correct API KEY for Authentication.</p><p>Download</p></div> <div><p>Response headers</p><p>content-length: 41 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:30:26 GMT server: Werkzeug/1.0.0 Python/3.7.6</p></div>

Responses

Activate Windows
Go to Settings to activate Windows.

Case2: If username and API key are Valid

Reponse:

Curl

```
curl -X POST "http://127.0.0.1:5000/ForgotPassword?UserName=vennam1" -H "accept: application/json" -H "API_KEY: W09AU4KifLhAm-USx2sQI7xT2icnXs8U9g"
```

Request URL

```
http://127.0.0.1:5000/ForgotPassword?UserName=vennam1
```

Server response

Code	Details
200	<div><p>Response body</p><p>Mail has been sent to Registered mail id.</p><p>Download</p></div> <div><p>Response headers</p><p>content-length: 41 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:27:59 GMT server: Werkzeug/1.0.0 Python/3.7.6</p></div>

Responses

Case2: If username is not valid and API key is Valid:

Response:

Curl

curl -X POST "http://127.0.0.1:5000/ForgotPassword?UserName=venna" -H "accept: application/json" -H "API_KEY: WD9AU4KiflHAm-USx2sQI7xT2icnXsBU9g"

Request URL

http://127.0.0.1:5000/ForgotPassword?UserName=venna

Server response

Code	Details
200	<div>Response body</div> <div>venna details not found kindly enter valid UserName.</div> <div>Download</div> <div>Response headers</div> <div>content-length: 52 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:32:54 GMT server: Werkzeug/1.0.0 Python/3.7.6</div>

Responses

Code	Description
200	The output returns message whether mail has been sent or not

Activate Windows
Go to Settings to activate Windows.

5. POST/ChangePassword:

Method 1:(Without Swagger UI)

Parameters:

UserName : String
Header :API KEY
OTP :Number
NewPassword:String

Request URL

```
http://127.0.0.1:5000/ChangePassword?UserName=vennaaa&OTP=7878&NewPassword=test1
```

Case1: If username does not exists

Response:

```
vennaaa does not exist , kindly enter correct UserName.
```

Case2: If username or OTP wrong

Response:

```
Something went wrong check the OTP or UserName!!!!
```

Case3: If APIKEY wrong

Response:

```
Enter correct API KEY for Authentication.
```

Case4: If everything correct

Response:

```
Password updated successfully in the database based on UserName.
```

Method 2:(With Swagger UI)

Parameters:

UserName : String

Header :API KEY

POST /ChangePassword Let's Change Password

This is using for Changing Password based on UserName and validation of OTP

Parameters Try it out

Name	Description
UserName * required string (query)	<input type="text" value="UserName"/>
OTP * required number (query)	<input type="text" value="OTP"/>
NewPassword * required string (query)	<input type="text" value="NewPassword"/>
API_KEY * required string (header)	an authorization header <input type="text" value="API_KEY - an authorization header"/>

Activate Windows
Go to Settings to activate Windows

- Click on “**Try it out**”
- Enter Details correctly and click **Execute**

Case1: If username does not exists

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/ChangePassword?UserName=vennaaa&OTP=78788NewPassword=test1" -H "accept: application/json" -H "API_KEY: W09AU4KiflhAm-USx2sQI7xT2icnXsBU9"
```

Request URL

```
http://127.0.0.1:5000/ChangePassword?UserName=vennaaa&OTP=78788NewPassword=test1
```

Server response

Code	Details
200	<div><p>Response body</p><pre>vennaaa does not exist , kindly enter correct UserName.</pre><p>Download</p><p>Response headers</p><pre>content-length: 55 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:37:01 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Responses

Code	Description
------	-------------

Case2: If username or OTP wrong

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/ChangePassword?UserName=vennam&OTP=78788NewPassword=test1" -H "accept: application/json" -H "API_KEY: W09AU4KiflhAm-USx2sQI7xT2icnXsBU9g"
```

Request URL

```
http://127.0.0.1:5000/ChangePassword?UserName=vennam&OTP=78788NewPassword=test1
```

Server response

Code	Details
200	<div><p>Response body</p><pre>Something went wrong check the OTP or UserName!!!!</pre><p>Download</p><p>Response headers</p><pre>content-length: 50 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:40:57 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Responses

Case3: If API KEY wrong

Response:

Request URL

```
http://127.0.0.1:5000/ChangePassword?UserName=vennam80TP=78788NewPassword=test1
```

Server response

Code	Details
200	<p>Response body</p> <pre>Enter correct API KEY for Authentication.</pre> <p>Response headers</p> <pre>content-length: 41 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:43:41 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

Responses

Code	Description
200	The output returns whether the Password changed sucessfully or not

Case4: If evrything correct

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/ChangePassword?UserName=vennam1&OTP=26913&NewPassword=test1" -H "accept: application/json" -H "API_KEY: W09AU4KifLhAm-USx2sQI7xT2icnXsBU9g"
```

Request URL

```
http://127.0.0.1:5000/ChangePassword?UserName=vennam1&OTP=26913&NewPassword=test1
```

Server response

Code	Details
200	<p>Response body</p> <pre>Password updated successfully in the database based on UserName.</pre> <p>Response headers</p> <pre>content-length: 64 content-type: text/html; charset=utf-8 date: Tue, 09 Jun 2020 11:46:12 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

Responses

Code	Description
200	The output returns whether the Password changed sucessfully or not

Activate Windows
Go to Settings to activate Windows.

6. POST/DeleteUserName

Method 1:(Without Swagger UI)

Parameters:

UserName : String
Header :API KEY
Password:String

Request URL

```
http://127.0.0.1:5000/DeleteUserName?UserName=vennaa&Password=test1
```

Server response

Case1: If username not available

Response:

```
{  
  "Message": "UserName not available ."  
}
```

Case2: If API KEY wrong

Response:

Response body

```
{  
  "Message": "Enter API KEY for Authentication ."  
}
```

Case 3 : If user details are wrong:

Response:

```
{  
  "Message": "Invalid UserName or Password , kindly check ."  
}
```

Case4: If everything correct

Response:

```
"Deleted UserName from database based on UserName."
```

Method 2:(With Swagger UI)

Parameters:

UserName : String
Header :API KEY
Password:String

POST /DeleteUserName Let's Change Password

This is using for Changing Password based on UserName and validation of OTP

Parameters [Try it out](#)

Name	Description
UserName * required string (query)	<input type="text" value="UserName"/>
Password * required string (query)	<input type="text" value="Password"/>
API_KEY * required string (header)	an authorization header <input type="text" value="API_KEY - an authorization header"/>

- Click on “Try it out”
- Enter Details correctly and click **Execute**

Case1: If username not available

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/DeleteUserName?UserName=vennaa&Password=test1" -H "accept: application/json" -H "API_KEY: W09AU4K1fLhAm-USx2sQI7xT2icnXsBU9"
```

Request URL

```
http://127.0.0.1:5000/DeleteUserName?UserName=vennaa&Password=test1
```

Server response

Code	Details
200	<div>Response body<pre>{ "Message": "UserName not available ."} </pre>Download</div> <div>Response headers<pre>content-length: 39 content-type: application/json date: Tue, 09 Jun 2020 11:50:13 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Responses

Code	Description
------	-------------

Activate Windows
Go to Settings to activate Windows.

Case2: If API KEY wrong

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/DeleteUserName?UserName=vennam&Password=test1" -H "accept: application/json" -H "API_KEY: W09AU4KifLhAm-USx2sQITxT2icnXsBU9"
```

Request URL

```
http://127.0.0.1:5000/DeleteUserName?UserName=vennam&Password=test1
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "Message": "Enter API KEY for Authentication ." }</pre> <p>Response headers</p> <pre>content-length: 49 content-type: application/json date: Tue, 09 Jun 2020 11:52:41 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

Responses

Code	Description
200	The output returns whether the Password changed successfully or not

Activate Windows
Go to Settings to activate Windows.

Case 3 : IF user details are wrong

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/DeleteUserName?UserName=vennam&Password=test1" -H "accept: application/json" -H "API_KEY: W09AU4KifLhAm-USx2sQITxT2icnXsBU9g"
```

Request URL

```
http://127.0.0.1:5000/DeleteUserName?UserName=vennam&Password=test1
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "Message": "Invalid UserName or Password , kindly check ." }</pre> <p>Response headers</p> <pre>content-length: 61 content-type: application/json date: Tue, 09 Jun 2020 11:54:26 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre>

Case4: If everything correct

Response:

Curl

```
curl -X POST "http://127.0.0.1:5000/DeleteUserName?UserName=vennam1&Password=test1" -H "accept: application/json" -H "API_KEY: W09AU4KifLhAm-USx2sQI7xI2icnXsBU9g"
```

Request URL

```
http://127.0.0.1:5000/DeleteUserName?UserName=vennam1&Password=test1
```

Server response

Code	Details
------	---------

200	
-----	--

Response body

```
{
  "Error": "Deleted UserName from database based on UserName."
}
```

Download

Response headers

```
content-length: 62
content-type: application/json
date: Tue, 09 Jun 2020 11:55:51 GMT
server: Werkzeug/1.0.0 Python/3.7.6
```

Response

7. GET/Admin:

Method 1:(Without Swagger UI)

Parameters:

UserName : String

Password:String

Request URL

```
http://127.0.0.1:5000/Admin?UserName=venna&Password=ad
```

Case1: If admin details are wrong

Response:

Response body

```
{  
  "Error": "Kindly enter correct admin UserName or admin Password ."  
}
```

Case2: If other user details are correct

Response:

```
{  
  "Error": "You are not admin for this API ."  
}
```

Case3: If admin details are correct

Response:

```
{
  "result": [
    {
      "UserName": "vennam",
      "Password": "21232f297a57a5a743894a0e4a801fc3",
      "FullName": "vennam",
      "Email": "vennam1435@gmail.com",
      "otp": null
    },
    {
      "UserName": "raj",
      "Password": "098f6bcd4621d373cade4e832627b4f6",
      "FullName": "Raj kumar",
      "Email": "raj@gmail.com",
      "otp": null
    },
    {
      "UserName": "vennam2",
      "Password": "098f6bcd4621d373cade4e832627b4f6",
      "FullName": "vennam",
      "Email": "vennamdevendhar@gmail.com",
      "otp": null
    }
  ],
}
```

Method 2:(With Swagger UI)

Parameters:

UserName : String

Password:String

GET /Admin Admin Page

This can be used by only ADMIN

Parameters
Try it out

Name	Description
UserName * required string (query)	<input type="text" value="UserName"/>
Password * required string (query)	<input type="text" value="Password"/>

Responses
Response content type application/json

Code	Description
200	The output returns as all user details

- Click on “Try it out”
- Enter Details correctly and click **Execute**

Case1: If admin details are wrong

Response:

Curl

```
curl -X GET "http://127.0.0.1:5000/Admin?UserName=venna&Password=ad" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/Admin?UserName=venna&Password=ad
```

Server response

Code	Details
200	<div><p>Response body</p><pre>{ "Error": "Kindly enter correct admin UserName or admin Password ." }</pre><p>Download</p><p>Response headers</p><pre>content-length: 69 content-type: application/json date: Tue, 09 Jun 2020 12:01:21 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Responses

Activate Windows

Case2: If other user details are correct

Response:

Curl

```
curl -X GET "http://127.0.0.1:5000/Admin?UserName=raj&Password=test" -H "accept: application/json"
```

Request URL

```
http://127.0.0.1:5000/Admin?UserName=raj&Password=test
```

Server response

Code	Details
200	<div><p>Response body</p><pre>{ "Error": "You are not admin for this API ." }</pre><p>Download</p><p>Response headers</p><pre>content-length: 45 content-type: application/json date: Tue, 09 Jun 2020 12:08:59 GMT server: Werkzeug/1.0.0 Python/3.7.6</pre></div>

Responses

Activate Windows

Go to Settings to activate Windows.

Code	Description
------	-------------

Case3: If admin details are correct

Response:

Request URL	
http://127.0.0.1:5000/Admin?UserName=vennam&Password=admin	
Server response	
Code	Details
200	<div>Response body</div> <pre>{ "result": [{ "UserName": "vennam", "Password": "21232f297a57a5a743894a0e4a801fc3", "FullName": "vennam", "Email": "vennam1435@gmail.com", "otp": null }, { "UserName": "raj", "Password": "098f6bcd4621d373cade4e832627b4f6", "FullName": "Raj kumar", "Email": "raj@gmail.com", "otp": null }, { "UserName": "vennam2", "Password": "098f6bcd4621d373cade4e832627b4f6", "FullName": "vennam", "Email": "vennamdevendhar@gmail.com", "otp": null }, { "UserName": "vennam12", "Password": "098f6bcd4621d373cade4e832627b4f6", "FullName": "vennam", "Email": "vennam12@gmail.com", "otp": null }] }</pre>

Activate V
Go to Setting