

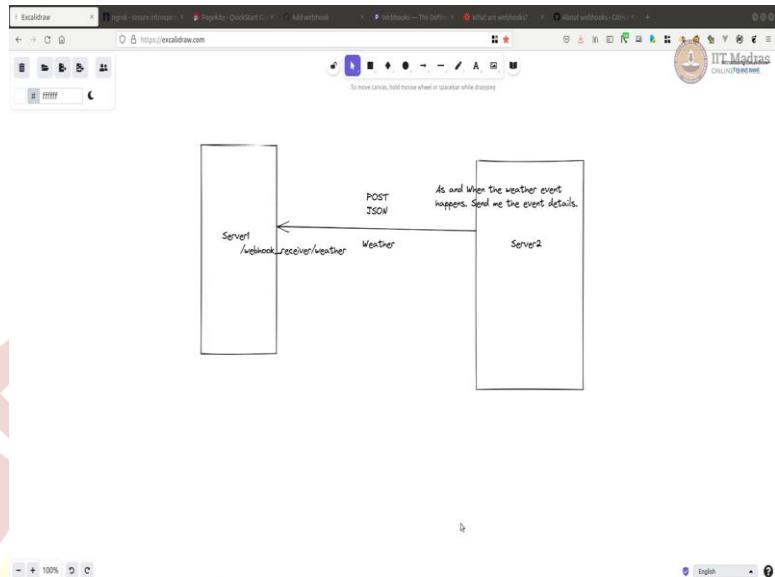
IIT Madras

ONLINE DEGREE

Modern Application Development - II
Professor. Thejes G. N.
Software Consultant
Indian Institute of Technology, Madras

How to use Webhooks?

(Refer Slide Time: 0:14)



Welcome to the modern application development 2 screencast. In this screencast, we will see how to use webhooks for a server to communication with another server. You will need an Linux development environment and the project that we have been using until now. And then you will need internet connection and some tools that needs to be installed to please pause and install the tools when asked you to do.

Now let us get into what is webhooks? There are many ways the server talks to a server in on internet. Let us say this is server 1 or your application 1 talks to another server 2 usually through API's let us say this is server 1, it is asking another server for weather. And this 1 responds with weather details JSON. So, this is usually server 1 talks to server 2. Now there could be a situation where you want to know from server 2, if there are any important weather events, for example, there is heavy rains, or the temperature drop, or temperature going high or there is wind, etcetera etcetera.

Now, in your survey, you can keep calling every 5 minutes and checking the weather saying please tell me if there are any weather event in extreme weather events. If it is there, then it will send back if it is not there, it will not. Now there could be calls you have to make every minute, possibly, to check if it is an important event. And that is a waste call. Because you do not want to make that call, like because most of the times, there are no major weather events. But you cannot stop checking, because it is so important. So, you have to keep checking, and most of the times it will return nothing.

So, this calls on internet becomes 0, the load on the server 1 and server 2 also becomes a lot. So, for such event or in such cases, we use webhooks, what we do is instead of asking for weather and it responding, we will give a server to a query saying whenever there is a change in weather, or whenever there is a weather event, please send me the details whenever that happens. So, there is no request response, a synchronous call, there will be only one call. It is called a webhook. So, this happily goes away. And there will be only 1, and it will send the weather as in when the weather event happens.

So, the note thing is, as and when the weather event happens, send me the event details. Usually, the sending happens over the internet itself, it is HTTP usually happens to through usually, but it can happen through other methods too, but usually post and usually posted JSON, it can post request parameters or XML or even simple text. But often JSON is used to endpoint here on the receiver side, like probably like they would have issued webhooks receiver for whether this will be the endpoint.

And then this URL will be given to server 2, so that is server 2 knows, it should call server 1 dot webhook, underscore receiver, slash weather and post to that endpoint. And so that server 1 get it. So, what is the advantage? The advantage here is you do not have to keep making calls to get an update. And you will get update as soon as possible as soon as the server to has it and sends it. It is actually less load for both of them. And it is easier to program you do not need to write loops or you do not need any time inquiries, all of that.

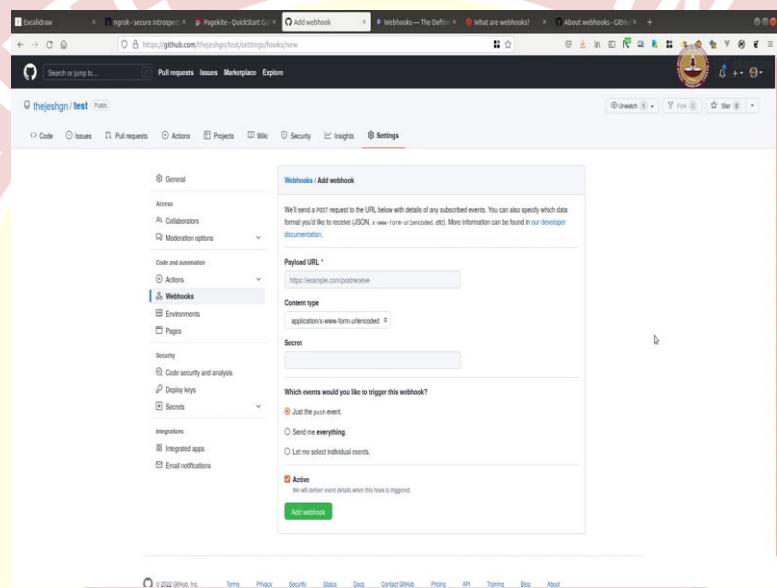
So, this whole architecture or structure is called the webhooks because you are hooking into the event. Now let us see how to implement it. First of all, let us see who are all the providers who provide this. Sometimes the providers are whether providers like we saw here, sometimes it could be payment providers, let us say you create a payment request on let us say, razor pay, or any other stripe or any other player, who actually processes the payment, if the payment is successful, they will send you an webhook saying payment complete.

Once you get a details of payments are complete, then you can do an action of sending an email to your customer saying, we have received all your payments here resolved or confirmed, or whatever. Or if you are sending a courier, this is a courier provider, let us say we India post or dtdc, or, or any of those providers. And they keep updating the status of the courier that it has reached station 2, station 3, station 4, and that you can receive on your server to show your clients that if they are shipping, like E commerce 1, you can always show

them an update that your courier has been checked, your career has reached center 2, center 3, center 4.

Instead of actually making calls every time to their server, you actually get updates from them, you store it and from there you show to your end user. So, these are all real world scenarios that you can check out, like, the interesting ones to explore are for example, stripe dot com, webhooks or razorpay dot com to a webhook, etcetera, or even GitHub and GitLab has webhooks.

(Refer Slide Time: 6:44)



Here, I have a repository called test it just for testing. And within the repository, I can have a webhook where I can configure who should receive what kind of events. Now, to receive events, you will have to write the receiving point.

(Refer Slide Time: 7:00)

The image shows two screenshots of a Sublime Text editor window. The top screenshot shows a file browser on the left with a tree view of a project structure. The bottom screenshot shows the content of a Python file named 'controller.py'.

```
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
• iit-webhooks
• iit_code
• __pycache__
• __init__.py
• config.py
• controllers.py
• database.py
• models.py
• tasks.py
• validation.py
• workers.py
• db_directory
• static
• templates
□ celerybeat_schedule
□ devlog.log
□ local_beat.sh
□ local_gunicorn.sh
□ local_nsqd.sh
□ local_workers.sh
□ main.py
□ README.md
□ requirements.txt
□ README.md

Cancel
Name: webhooks.py
Search
Name Size Modified
pycache_ 2.7 KB Yesterday
controllers.py 1.5 KB Yesterday
config.py 1.9 KB Yesterday
workers.py 379 bytes 17 Feb
app.py 1.4 KB 10 Feb
models.py 1.8 KB 10 Feb
validation.py 732 bytes 9 Feb
database.py 155 bytes 27 Sep 2021
__init__.py 0 bytes 27 Sep 2021

Documents
Downloads
Music
Pictures
Videos
Work
Software
Backups
Code
Other Locations
```

```
-File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
• iit-webhooks
• iit_code
• __pycache__
• __init__.py
• config.py
• controllers.py
• database.py
• models.py
• tasks.py
• validation.py
• workers.py
• db_directory
• static
• templates
□ celerybeat_schedule
□ devlog.log
□ local_beat.sh
□ local_gunicorn.sh
□ local_nsqd.sh
□ local_workers.sh
□ main.py
□ README.md
□ requirements.txt
□ README.md

controller.py
1 from flask import Flask, request
2 from flask import render_template
3 from main import app as app
4 from application.models import Article
5 from flask_security import login_required, roles_accepted, roles_required
6 from application import tasks
7 from flask_ssl import ssl
8
9 print("in controller app", app)
10
11 @app.route("/hello", methods=["GET"])
12 def hello():
13     return "World"
14
15 @app.route("/", methods=["GET", "POST"])
16 def articles():
17     app.logger.info("Inside get all articles using info")
18     articles = Article.query.all()
19     app.logger.debug("Inside get all articles using debug")
20     return render_template("articles.html", articles=articles)
21
22 @app.route("/articles_by<user_name>", methods=["GET", "POST"])
23 @login_required
24 def articles_by_author(user_name):
25     articles = Article.query.filter(Article.authors.any(username=user_name))
26     return render_template("articles_by_author.html", articles=articles, username=user_name)
27
28
29 @app.route("/article_like<article_id>", methods=["GET", "POST"])
30 def like(article_id):
31     print(article_id)
    article_id = article_id.replace('-', '_').replace(' ', '_').lower().strip()

Source: 4 Python
```

So, here, it is like I said, it is going to be just a simple endpoint with post and JSON and we will have some validation, I will show how to do validation, but there is no authentication as such, but there will be header parameters that can be sent from the server 2 to server 1, to make sure that the data is coming from server 2 and not some random person on the internet. So, we will do keep try to implement one weather receiver, which is simple home controller, but I am not going to put it into controller, we can just put it into just as a separate.

(Refer Slide Time: 7:45)

The image displays three separate screenshots of a Sublime Text editor window, each showing a different Python file from a Flask application. The files are:

- `application/webhooks.py`: This file contains logic for handling webhooks. It imports Flask, request, render_template, and Article from the main application. It also imports login_required, roles_accepted, and roles_required from flask_security. The code includes a reference block for external JavaScript files (bootstrap.bundle.js, bootstrap.esm.js, bootstrap.js) and a function that parses JSON request data, validates it, and calls an asynchronous job. It returns "OK" with a 200 status code.
- `application/controllers.py`: This file contains routes and logic for articles. It imports Flask, request, render_template, Article, login_required, roles_accepted, roles_required, and tasks from the main application. It also imports Article from application.models. The code defines routes for getting all articles, getting articles by author, and liking an article. It uses app.logger.info and app.logger.debug for logging.
- `application/webhooks.py`: This file is identical to the one in the first screenshot, showing the same import statements and a function that prints "in controller app" and returns "World".

The Sublime Text interface shows the project structure on the left, which includes folders for .gitignore, .htaccess, .travis.yml, config, controllers, database, models, tasks, validation, and webhooks, along with static files like favicon.ico, index.html, robots.txt, and styles.css. The status bar at the bottom of each window indicates the file type as Python and the number of characters (e.g., 25 characters selected, 24 lines, 200 characters selected).

Because this is going to be like an integration point for your application with many other applications. So, I try to keep it separate. Let us do one thing webhook receiver for GitHub. So, once I receive, I will get some headers, I will get some request dot JSON the whatever they want to send and I want to validate. If everything goes well, either I can process the data here, or if the data is a lot on, it is going to take a longer time, I can call an async job. I can call task dot something and calling a sync job. And also I can alert the user if it is required saying I have received successful payment or successful commit to whatever that is.

(Refer Slide Time: 8:45)

The image shows two screenshots demonstrating the implementation of a GitHub webhook receiver. The top screenshot is a Sublime Text editor window showing the code for 'webhooks.py'. The code defines a Flask application with a route for GitHub webhooks, parses the JSON payload, and prints it. The bottom screenshot is a terminal window showing the execution of the application and its startup logs, which includeunicorn worker booting and database initialization messages.

```

File Edit Selection Find View Goto Tools Project Preferences Help
File Edit Selection Find View Goto Tools Project Preferences Help
Folders
    • iitk-webhooks
        • __init__.py
        • __main__.py
        • __apache_
        • __application
            • __apache_
                • __init__.py
                • config.py
                • controller.py
                • database.py
                • models.py
                • tasks.py
                • validation.py
            • __webhooks__
                • __init__.py
            • __init__.py
        • __static__
        • __templates__
            • configed_schedule
            • debug.log
            • local_base.ah
            • local_gunicorn.ah
            • local_uwsgi.ah
            • local_werk.ah
            • main.py
            • nginx.conf
            • requirements.txt
            • requirements.txt
Code: Sublime Text
Line 17 Column 10: Colored Throwing
File Edit View Search Terminal Help
=====
Welcome to the setup. This will setup the local virtual env.
And then it will install all the required python libraries.
You can rerun this without any issues.
-----
Enabling virtual env
[2022-02-24 08:22:12 +0530] [31493] [INFO] Starting gunicorn 20.1.0
[2022-02-24 08:22:12 +0530] [31493] [INFO] Listening at: http://127.0.0.1:8000 (31493)
[2022-02-24 08:22:12 +0530] [31493] [INFO] Using worker: gevent
[2022-02-24 08:22:12 +0530] [31496] [INFO] Booting worker with pid: 31496
[2022-02-24 08:22:12 +0530] [31497] [INFO] Booting worker with pid: 31497
[2022-02-24 08:22:12 +0530] [31498] [INFO] Booting worker with pid: 31498
[2022-02-24 08:22:12 +0530] [31499] [INFO] Booting worker with pid: 31499
stage
Starting stage
pushed config
DB Init
DB Init complete
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
Create app completed
In controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>

```

```

thej@una:~/code/work/flask-webhooks$ code
File Edit View Search Terminal Help
In controller app <Flask 'main'>
Create app complete
In controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
[2022-02-24 08:35:34 +0530] [31493] [INFO] Handling signal: int
[2022-02-24 08:35:35 +0530] [31496] [INFO] Worker exiting (pid: 31496)
[2022-02-24 08:35:35 +0530] [31499] [INFO] Worker exiting (pid: 31499)
[2022-02-24 08:35:35 +0530] [31498] [INFO] Worker exiting (pid: 31498)
Traceback (most recent call last):
  File "/home/thej/.code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/monkey.py", line 893, in _shutdown
    sleep()
  File "/home/thej/.code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/hub.py", line 159, in sleep
    time.sleep(get())
  File "src/gevent/walter.py", line 143, in gevent_c_walter.Walter.get
  File "src/gevent/walter.py", line 154, in gevent_c_walter.Walter.get
  File "src/gevent/greenlet/primitives.py", line 61, in gevent_c_greenlet.primitives.SwitchOutGreenletWithLoop.switch
  File "src/gevent/greenlet/primitives.py", line 61, in gevent_c_greenlet.primitives.SwitchOutGreenletWithLoop.switch
  File "src/gevent/greenlet/primitives.py", line 65, in gevent_c_greenlet.primitives.SwitchOutGreenletWithLoop.switch
  File "src/gevent/greenlet/primitives.py", line 35, in gevent_c_greenlet.primitives..greenlet.switch
  File "src/gevent/greenlet.py", line 986, in gevent_c_greenlet.Greenlet.run
  File "/home/thej/.code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/unicorn/workers/base.py", line 198, in handle_quit
    sys.exit(0)
SystemExit: 0
2022-02-24 08:35:35 +0530 <greenlet.greenlet object at 0x7fc84c51f700 (otid=0x7fc84e29f5a0) current active started main> failed with SystemExit
[2022-02-24 08:35:35 +0530] [31497] [INFO] Worker exiting (pid: 31497)
[2022-02-24 08:35:35 +0530] [31493] [INFO] Shutting down: Master
thej@una:~/code/work/flask-webhooks$ code$ clear

```

So, I am going to save this and start the let us print this, and start the server, I think have already started.

(Refer Slide Time: 9:11)

```

File Edit Selection Find View Go To Tools Project Preferences Help
FOLDERS
* .git
* .idea
* .env
* __pycache__
* application
  __pycache__
    __init__.py
    celery.py
    config.py
    controllers.py
    database.py
    models.py
    tasks.py
    validation.py
    webhooks.py
    workers.py
* __pycache__.d
* static
* templates
  __pycache__.d
    __init__.py
    debug.log
    local_base.j2
    local_base.jinja2
    local_jinja2.j2
    local_jinja2.jinja2
    local_setup.j2
    local_workers.j2
* main.py
* README.md
* requirements.txt
* README.md

File Edit Selection Find View Go To Tools Project Preferences Help
+{code/work/flask-webhooks/code/main.py (task-webhook)} - Sublime Text
63     return app, api, celery
64
65     app, api, celery = create_app()
66
67
68     # Add all restful controllers
69     from application.api import ArticleLikesAPI
70     api.add_resource(ArticleLikesAPI, "/api/article_likes", "/api/article_likes/<int:article"
71
72
73     from application.api import TestAPI
74     api.add_resource(TestAPI, "/api/test")
75
76
77     @sse.before_request
78     def check_access():
79         if request.args.get("channel") == "analytics" and not g.user.is_admin():
80             abort(403)
81
82     # This is for streaming
83     app.register_blueprint(sse, url_prefix='/stream')
84
85     # Import all the controllers so they are loaded
86     from application.controllers import *
87
88     @app.errorhandler(404)
89     def page_not_found(e):
90         # note that we set the 404 status explicitly
91         return render_template('404.html'), 404
92
93

```



```
-code/work/flask-webhooks/code/main.py - [Task-webhooks] - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
• iit_task_webhooks
  • iit_code
    • iit_env
    • iit_greache_
    • iit_application
      • iit__init__.py
      • iit_greache.py
      • iit_controllers.py
      • iit_models.py
      • iit_tasks.py
      • iit_validation.py
      • iit_webhooks.py
      • iit_workers.py
    • iit_db_directory
    • iit_static
    • iit_templates
      • iit_greache_schedule
        • iit_debug_log
        • iit_local_beat.sh
        • iit_local_greache.sh
        • iit_local_run.sh
        • iit_local_setup.sh
        • iit_local_workers.sh
      • main.py
      • readme.md
      • requirements.txt
      • roadmap.md
  • iit_main.py
  • iit_mainme.md
  • iit_requirements.txt
  • iit_roadmap.md

63     return app, api, celery
64
65 app, api, celery = create_app()
66
67
68 # Add all restful controllers
69 from application.api import ArticleLikesAPI
70 api.add_resource(ArticleLikesAPI, "/api/article_likes", "/api/article_likes<int:article")
71
72
73 from application.api import TestAPI
74 api.add_resource(TestAPI, "/api/test")
75
76
77 @sse.before_request
78 def check_access():
79     if request.args.get("channel") == "analytics" and not g.user.is_admin():
80         abort(403)
81
82 # This is for streaming
83 app.register_blueprint(sse, url_prefix='/stream')
84
85 # Import all the controllers so they are loaded
86 from application.controllers import *
87 from application.controllers import *
88
89 @app.errorhandler(404)
90 def page_not_found(e):
91     # note that we set the 404 status explicitly
92     return render_template('404.html'), 404
93
```

```
-code/work/flask-webhooks/code/main.py - [Task-webhooks] - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
• iit_task_webhooks
  • iit_code
    • iit_env
    • iit_greache_
    • iit_application
      • iit__init__.py
      • iit_greache.py
      • iit_controllers.py
      • iit_models.py
      • iit_tasks.py
      • iit_validation.py
      • iit_webhooks.py
      • iit_workers.py
    • iit_db_directory
    • iit_static
    • iit_templates
      • iit_greache_schedule
        • iit_debug_log
        • iit_local_beat.sh
        • iit_local_greache.sh
        • iit_local_run.sh
        • iit_local_setup.sh
        • iit_local_workers.sh
      • main.py
      • readme.md
      • requirements.txt
      • roadmap.md
  • iit_main.py
  • iit_mainme.md
  • iit_requirements.txt
  • iit_roadmap.md

63     return app, api, celery
64
65 app, api, celery = create_app()
66
67
68 # Add all restful controllers
69 from application.api import ArticleLikesAPI
70 api.add_resource(ArticleLikesAPI, "/api/article_likes", "/api/article_likes<int:article")
71
72
73 from application.api import TestAPI
74 api.add_resource(TestAPI, "/api/test")
75
76
77 @sse.before_request
78 def check_access():
79     if request.args.get("channel") == "analytics" and not g.user.is_admin():
80         abort(403)
81
82 # This is for streaming
83 app.register_blueprint(sse, url_prefix='/stream')
84
85 # Import all the controllers so they are loaded
86 from application.controllers import *
87 from application.controllers import *
88
89 @app.errorhandler(404)
90 def page_not_found(e):
91     # note that we set the 404 status explicitly
92     return render_template('404.html'), 404
93
```

```
thejh@name -code/work/flask-webhooks/code
File Edit View Search Terminal Help
File "/src/grevent/walter.py", line 143, in gevent_c_walter.Walter.get
File "/src/grevent/walter.py", line 154, in gevent_c_walter.Walter.get
File "/src/grevent/grevent_c_greenlet/primitives.pxd", line 35, in gevent_c_grevent_c_greenlet_primitives.gremlin_switch
2022-02-23T19:06:08Z <greenlet.greenlet object at 0x7fb8fbdcb5f0> current active started main> failed with SystemExit
thejh@name -code/work/flask-webhooks/code
File "/src/grevent/gremlin/primitives.py", line 906, in gevent_c_gremlin.Greenlet.run
File "/src/grevent/gremlin/primitives.py", line 61, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.py", line 61, in gevent_c_gremlin.Greenlet.switch
File "/home/thejh/.code/work/Flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 198, in handle_quit
    sys.exit()
File "/src/grevent/gremlin/primitives.py", line 65, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.pxd", line 35, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/grevent_c_greenlet/primitives.pxd", line 35, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/grevent_c_greenlet/primitives.pxd", line 35, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.py", line 906, in gevent_c_gremlin.Greenlet.run
File "/home/thejh/.code/work/Flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 198, in handle_quit
    sys.exit()
SystemExit: 0
2022-02-23T19:06:08Z <greenlet.greenlet object at 0x7fb8fbdcb5f0> current active started main> failed with SystemExit
[2022-02-24 00:36:08 +0530] [1138] [INFO] Worker exiting (pid: 1138)
Traceback (most recent call last):
  File "/home/thejh/.code/work/Flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/hub.py", line 893, in shutdown
    sleep()
  File "/home/thejh/.code/work/Flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/hub.py", line 159, in sleep
    _select([self])
  File "/src/grevent/walter.py", line 143, in gevent_c_walter.Walter.get
  File "/src/grevent/walter.py", line 154, in gevent_c_walter.Walter.get
  File "/src/grevent/gremlin/primitives.py", line 61, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.py", line 65, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.pxd", line 35, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.pxd", line 35, in gevent_c_gremlin.Greenlet.switch
File "/src/grevent/gremlin/primitives.py", line 906, in gevent_c_gremlin.Greenlet.run
File "/home/thejh/.code/work/Flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 198, in handle_quit
    sys.exit()
SystemExit: 0
2022-02-23T19:06:08Z <greenlet.greenlet object at 0x7fb8fbdcb5f0> current active started main> failed with SystemExit
[2022-02-24 00:36:08 +0530] [1131] [INFO] Shutting down: Master
thejh@name -code/work/flask-webhooks/code$ sh local_unicorn.sh
```

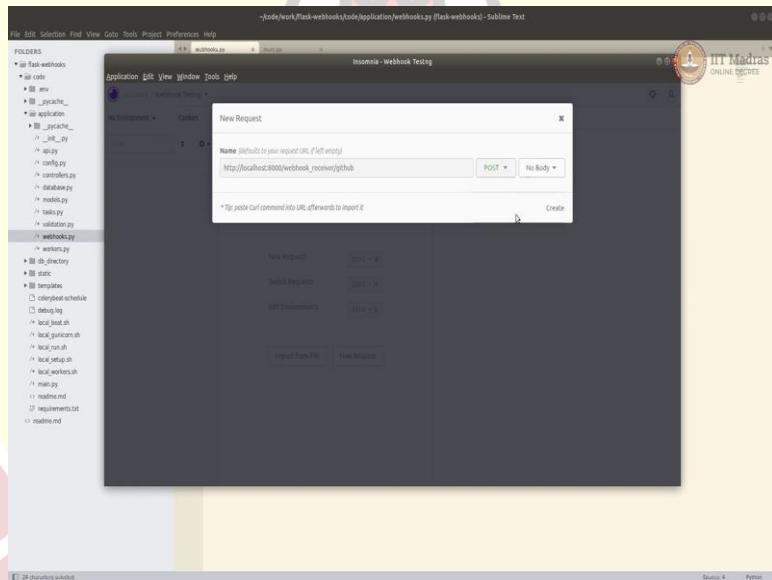
```

the@juma:~/code/work/flask-webhooks$ code
File Edit View Search Terminal Help
    sys.exit()
SystemExit: 0
2022-02-23T19:06:08Z <greenlet.greenlet object at 0x7fb8fbe4b7d0> current active started main> failed with SystemExit
SystemExit: 0
2022-02-24 00:36:08 +0530 [1138] [INFO] Worker exiting (pid: 1138)
Traceback (most recent call last):
  File "/home/the/.code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/monkey.py", line 893, in _shutdown
    sleep()
  File "/src/gevent/greenlet.py", line 154, in gevent_in_c_walter_Walter_get
  File "/src/gevent/greenlet_primitives.py", line 61, in gevent_in_c_preenet_primitives_SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet_primitives.py", line 61, in gevent_in_c_preenet_primitives_SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet_primitives.py", line 65, in gevent_in_c_preenet_primitives_SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet_primitives.pxd", line 35, in gevent_in_c_greennet_primitives_greennet_switch
  File "/src/gevent/greenlet.py", line 906, in gevent_in_c_greennet_primitives_greennet_switch
  File "/home/the/.code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 198, in handle_quit
    sys.exit()
SystemExit: 0
2022-02-23T19:06:08Z <greenlet.greenlet object at 0x7fb8fbe4b7d0> current active started main> failed with SystemExit
[2022-02-24 00:36:08 +0530] [1131] [INFO] Shutting down: Master
the@juma:~/code/work/flask-webhooks$ code5 sh local_unicorn.sh
=====
Welcome to the setup. This will setup the local virtual env.
And then it will install all the required python libraries.
You can rerun this without any issues.
=====
Enabling virtual env
[2022-02-24 00:36:10 +0530] [1217] [INFO] Starting gunicorn 20.1.0
[2022-02-24 00:36:10 +0530] [1217] [INFO] Listening at: http://127.0.0.1:8000 [1217]
[2022-02-24 00:36:10 +0530] [1217] [INFO] Using workers: eventlet
[2022-02-24 00:36:10 +0530] [1220] [INFO] Booting worker with pid: 1220
[2022-02-24 00:36:10 +0530] [1221] [INFO] Booting worker with pid: 1221
[2022-02-24 00:36:10 +0530] [1222] [INFO] Booting worker with pid: 1222
[2022-02-24 00:36:10 +0530] [1223] [INFO] Booting worker with pid: 1223

```

So, the one thing that we are not done is including that in the main you are not important that.

(Refer Slide Time: 9:39)



The screenshot shows a development environment with three main windows:

- Sublime Text Editor:** Displays the file `webhooks.py` containing Python code for a Flask application. The code handles GitHub webhooks, specifically for the `/webhook_receiver/github` endpoint. It prints the received content, gets headers, parses the JSON content, and calls an asynchronous job.
- Insomnia REST Client:** Shows a POST request to `http://localhost:8000/webhook_receive/github`. The JSON payload is `{"Hello": "world"}`. The response status is OK.
- Terminal:** Shows the output of running a virtual environment (`venv`) and starting a unicorn web server. The logs indicate the server is listening on port 8000 and booting multiple workers with pids 1221, 1222, and 1223.

```

from flask import Flask, request
from main import app as app
from application.models import Article
from flask_security import login_required, roles_accepted, roles_required
from flask_sse import sse
print("in controller app", app)

@app.route("/webhook_receiver/github", methods=["POST"])
def webhook_github():
    # get headers
    # parse the content
    content = request.json
    print(content)
    # Validate
    # call async job

    return "OK", 200
  
```

Let us check whether we are going to make a post request to http colon slash slash local host 8000, 8000 what is the endpoint slash web slash GitHub. I am going to make a post request and create the send some JSON data. What I am using here is called an insomniac client, you can search, it is an open source application, usually use for testing API's to make all kinds of post GET request instead of writing code to test.

You can just use this client to make a call. If, to paste this URL here, then I will send OK, I got OK, as a thing. And data here has been received, you can see Hello World. Yes, indeed. So, this endpoint is working. As you can see, there is no authentication.

(Refer Slide Time: 11:08)

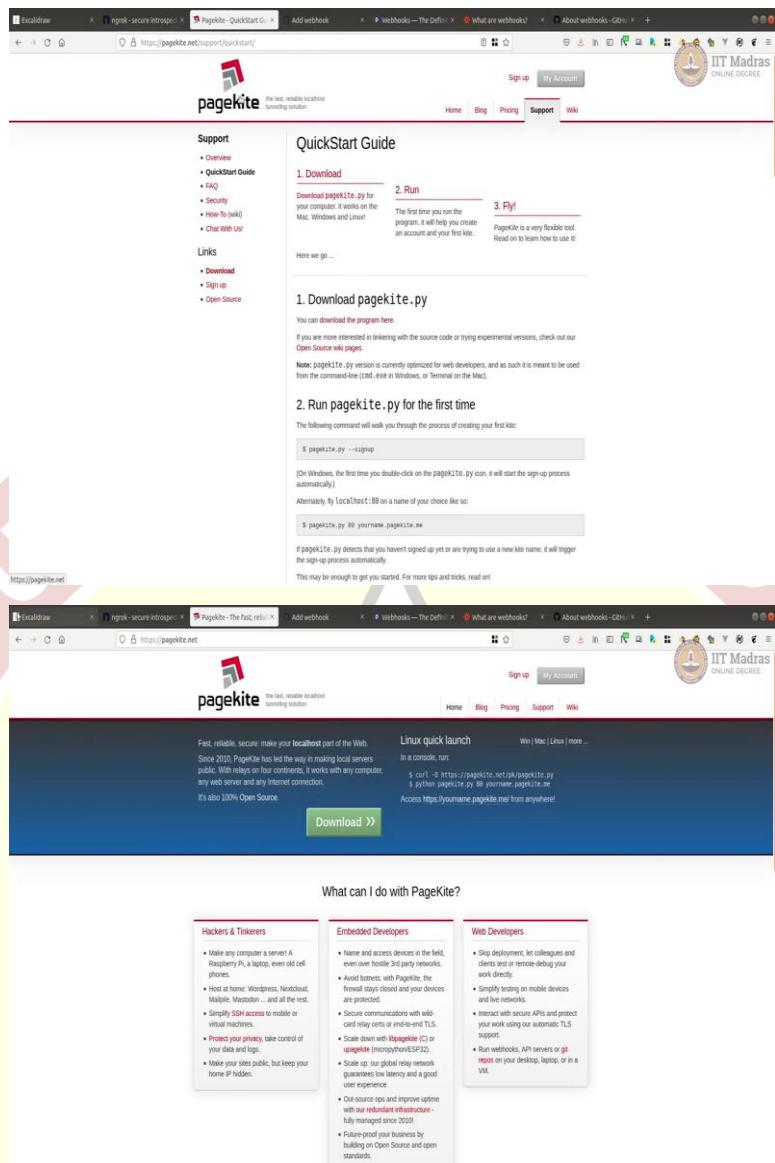
The image contains two screenshots of a web browser showing the configuration and testing of a GitHub webhook.

Screenshot 1: GitHub Webhook Configuration

This screenshot shows the "Webhooks" section of the GitHub settings for the repository "thejeshgn/test". The "Payload URL" is set to "https://example.com/postreceive". The "Content type" is "application/json". The "Secret" field is empty. Under "Which events would you like to trigger this webhook?", the "Just the push event" option is selected. The "Active" checkbox is checked. A green "Add webhook" button is visible at the bottom right.

Screenshot 2: Insomnia - Webhook Testing

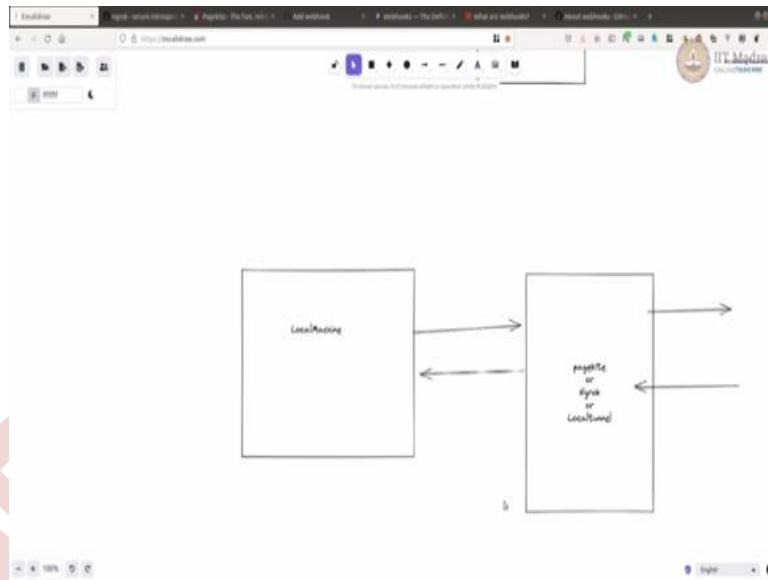
This screenshot shows the Insomnia REST Client interface. It is configured to perform a POST request to "http://localhost:8000/webhook_receive/github". The "Content-Type" header is set to "application/json". The "Send" button is highlighted in purple. The response status is "OK".



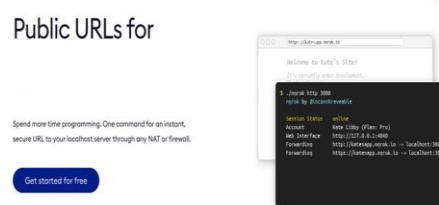
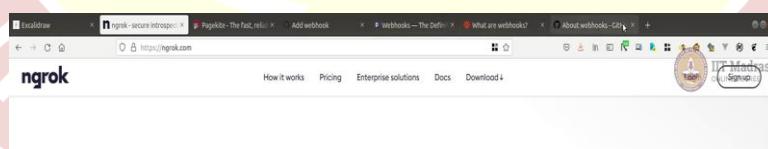
Now, let us go back and set up our GitHub. So, this on this GitHub, whenever there is a commit, someone does something that it is commit raising an issue, or adding an action, whatever they need, I want to know I wanted on my server, I will take, I will have to do something based on that. So, I will do like an application JSON at a given URL. Now, there is a problem. Now I am running locally, there is no way to give that URL here. So, if you are using any online development environment, you could have given directly their path.

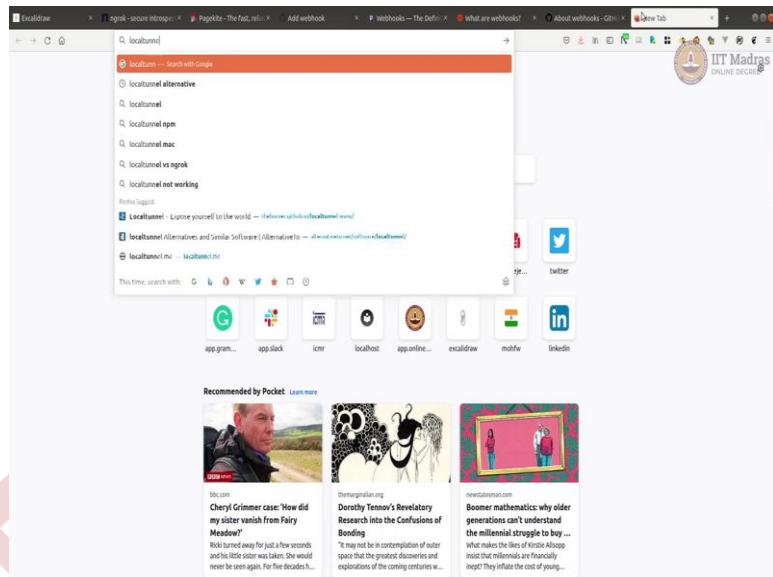
If you are using repple, you could have given the repple production endpoint like your server endpoint. Or if you had deployed, you could give that endpoint, but I am testing locally, and I cannot. So, what do we do now? So, we use services like either page guide, you can look at page guide, there are many services, like this page contains one of the open source 1, which will actually forward local server to remote server with a given domain.

(Refer Slide Time: 12:20)



```
File Edit View Search Terminal Help
Enabling virtual env
[2022-02-24 08:36:10 +0530] [1217] [INFO] Starting unicorn 20.1.0
[2022-02-24 08:36:10 +0530] [1217] [INFO] Listening at: http://127.0.0.1:8000 (1217)
[2022-02-24 08:36:10 +0530] [1217] [INFO] Using worker: gevent
[2022-02-24 08:36:10 +0530] [1220] [INFO] Booting worker with pid: 1220
[2022-02-24 08:36:10 +0530] [1221] [INFO] Booting worker with pid: 1221
[2022-02-24 08:36:10 +0530] [1222] [INFO] Booting worker with pid: 1222
[2022-02-24 08:36:10 +0530] [1223] [INFO] Booting worker with pid: 1223
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
In controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
In controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <Flask 'main'>
In controller app <Flask 'main'>
Create app complete
In controller app <Flask 'main'>
In controller app <Flask 'main'>
{'hello': 'world'}
```

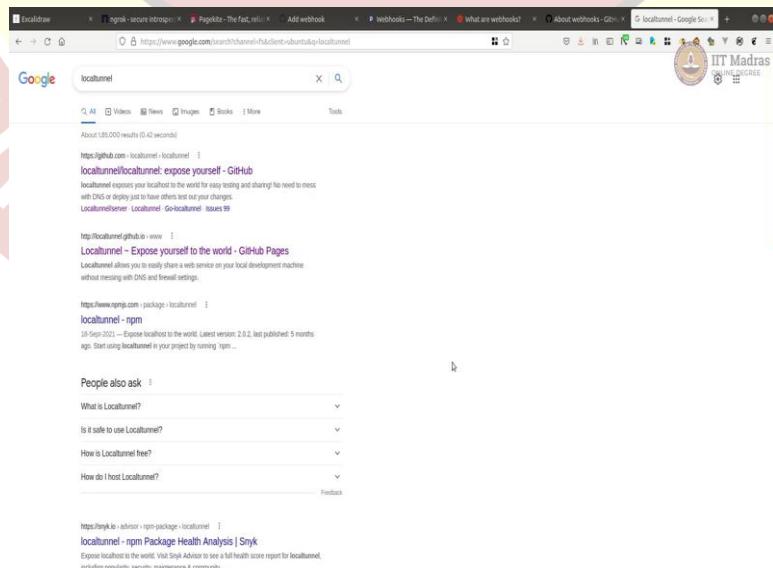


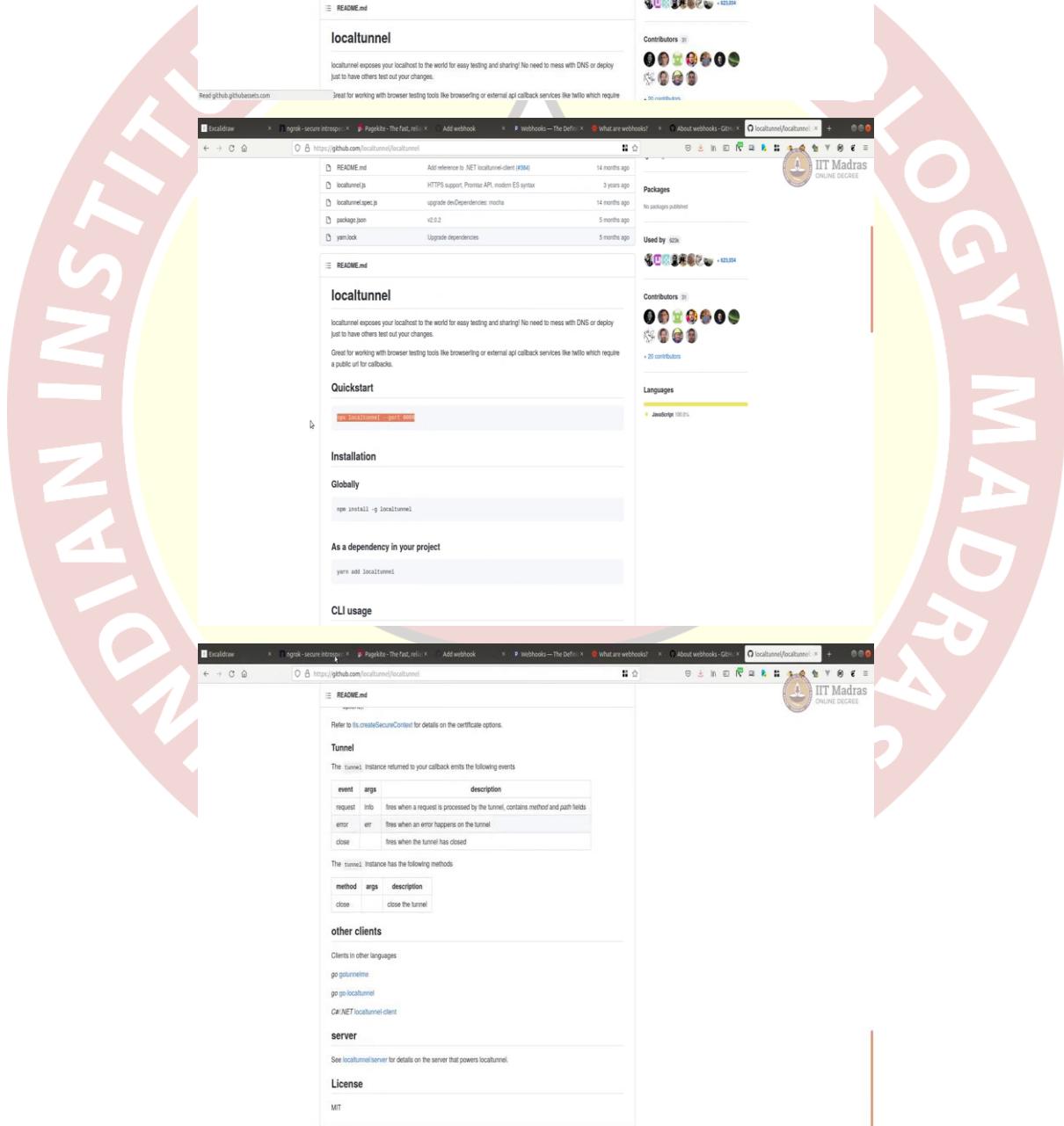


So, it is like hosting server from your machine like if you have like a local machine. And then if you have a service provider, like this middleman, proxy service layer really page guide, or Ngrok, or local tunnel. You can use any of them to expose your local machine ports to the internet. So, that you do not have to be running this domain that is mapped to your local machine or with the static IP. So, that it can be exposed and access through this, I am going to show it to of the service and not through all of them.

But you can check out page guide can checkout Ngrok, or there is local terminal, which is also free, local terminal.

(Refer Slide Time: 13:27)





So, here on GitHub, you can install it and use it. So, it can expose in like Port 800. And then it will give in URL to access that. Usually use it only for testing and do not use it for any production things. So, this is only testing feature.

(Refer Slide Time: 13:50)

```
File Edit View Search Terminal Help
Enabling virtual env
[2022-02-24 08:36:10 +0530] [1217] [INFO] Starting unicorn 2.0.1.0
[2022-02-24 08:36:10 +0530] [1217] [INFO] Listening at: http://127.0.0.1:8000 (1217)
[2022-02-24 08:36:10 +0530] [1217] [INFO] Using worker: gevent
[2022-02-24 08:36:10 +0530] [1220] [INFO] Booting worker with pid: 1220
[2022-02-24 08:36:10 +0530] [1221] [INFO] Booting worker with pid: 1221
[2022-02-24 08:36:10 +0530] [1222] [INFO] Booting worker with pid: 1222
[2022-02-24 08:36:10 +0530] [1223] [INFO] Booting worker with pid: 1223
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <flask 'main'>
in controller app <flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <flask 'main'>
in controller app <flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <flask 'main'>
in controller app <flask 'main'>
Create app complete
in controller app <flask 'main'>
in controller app <flask 'main'>
'Hello': 'world'
```


All Articles

Hello World

Written by: [theinception](#)

Welcome to Flask's documentation. Get started with Installation and then get an overview with the Quickstart. There is also a more detailed Tutorial that shows how to create a small but complete application with Flask. Common patterns are described in the Patterns for Flask section. The rest of the docs describe each component of Flask in detail, with a full reference in the API section.

Flask-SQLAlchemy

Written by:

Flask SQLAlchemy is an extension for Flask that adds support for SQLAlchemy to your application. It aims to simplify using SQLAlchemy with Flask by providing useful defaults and extra helpers that make it easier to accomplish common tasks. See the SQLAlchemy documentation to learn how to work with the ORM in depth. The following documentation is a brief overview of the most common tasks, as well as the features specific to Flask SQLAlchemy.

my new article

Written by: [theinception](#)

my new article content

dummy new article

Written by: [theinception](#)

my dummy new article content

Using relationship

Written by: [theinception](#)

Use relationships to insert. It's easy

2nd Using relationship

Written by: [theinception](#)

2nd Use relationships to insert. It's easy

So, now I am going to expose my local host, which will be a local host 8000 through the service called Ngrok. So, that I am running the command on the other screen, I am not going to show it to you because it kind of exposes some of the privacy and password related issues. But then we have running on the client behind the screen. So, now the same local host is available.

(Refer Slide Time: 14:19)

The image shows three screenshots of a web application interface, likely a Flask application running on localhost:8000. The top screenshot shows a dashboard with several cards: 'Inbox (4,880) - @thejgjgn.com - Thejgjgn.com Mail', 'Projects - Dashboard - GitHub', 'Dashboard - Thejgjgn.com - WordPress', 'WhatsApp - web-whatsapp', 'Project Rapido - dbthejgjgn_com_ufds', 'Twitter - It's what's happening / Twitter', 'Grammarly - Grammarly', 'Slack | online-degree | IITMPOD | 6 new items', and 'Indian Council of Medical Research, New Delhi'. Below the dashboard, there are sections for 'my new article' and 'dummy new article', both written by 'thejgjgn'. A section titled 'Using relationship' follows, with a note: 'Use relationships to insert. It's easy'. The middle screenshot shows an 'Insomnia - Webhook Testing' tool with a POST request to 'http://123f7f813ce.ngrok.io/webhook_recency'. The JSON payload is: { "hello": "world" }. The response is 'OK'. The bottom screenshot shows another 'Insomnia - Webhook Testing' tool with a POST request to 'http://123f7f813ce.ngrok.io/webhook_recency'. The JSON payload is: { "hello": "world" }. The response is 'OK'. Both Insomnia windows show a terminal at the bottom displaying command-line logs related to the application's execution.

On this endpoint temporary, it is not permanent. It is temporary. So, now I can use this and the same here, write and then send. So, I am getting Hello World and OK and you can see I am getting that Hello World here. So, that should pass through now.

(Refer Slide Time: 14:54)

The terminal window shows the following code:

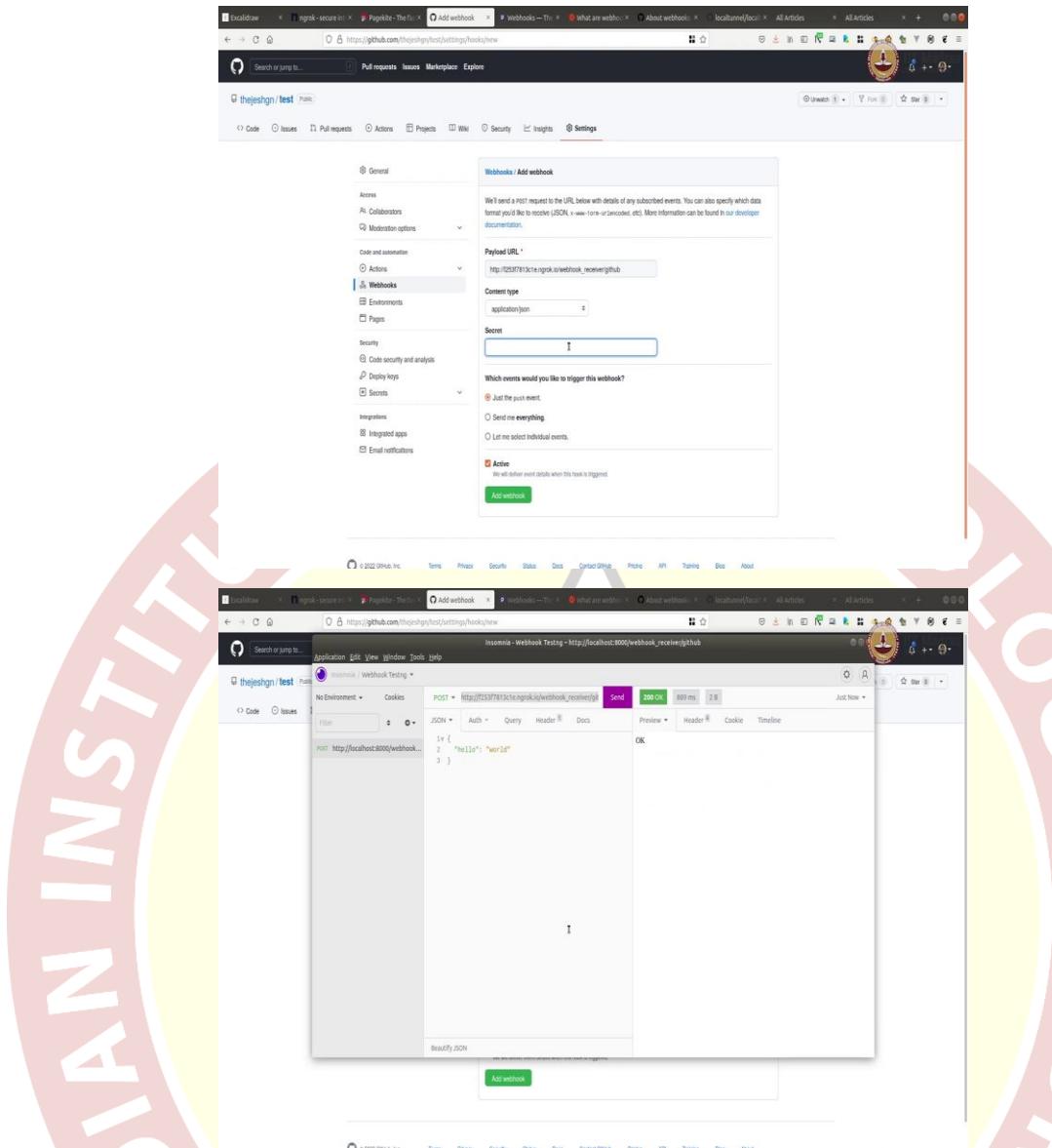
```
File Edit View Search Terminal Help
Starting stage
pushed config
DB Init
DB Init complete
in controller app <-
in controller app <-
Create app complete
in controller app <-
in controller app <-
('hello': 'world')
[2022-02-24 00:40:33] 127.0.0.1 -> http://localhost:8000/webhook...
Traceback (most recent call last):
  File "/home/thebyc/...
    resp.write_file()
  File "/home/thebyc/...
    if not self.sendfile:
  File "/home/thebyc/...
    self.sock.sendfile(
      fileobj,
      offset,
      count)
  File "/home/thebyc/...
    return self.sendfile(
      fileobj,
      offset,
      count)
  File "/home/thebyc/...
    self._check_sendfile_params(
      fileobj,
      offset,
      count)
  File "/home/thebyc/...
    "count must be a positive integer (got %r)".format(count))
ValueError: count must be a positive integer (got 0)
[2022-02-24 00:40:53]
Traceback (most recent call last):
  File "/home/thebyc/...
    resp.write_file()
  File "/home/thebyc/...
    if not self.sendfile:
  File "/home/thebyc/...
    self.sock.sendfile(
      fileobj,
      offset,
      count)
  File "/home/thebyc/...
    return self.sendfile(
      fileobj,
      offset,
      count)
  File "/home/thebyc/...
    self._check_sendfile_params(
      fileobj,
      offset,
      count)
  File "/home/thebyc/...
    "count must be a positive Integer (got %r)".format(count))
ValueError: count must be a positive Integer (got 0)
[ hello: 'world'
]
```

The Insomnia POST request shows a successful response with status 200 OK and the JSON payload:

```
[{"hello": "world"}]
```

The web browser screenshot shows the 'All Articles' page with the following articles listed:

- Hello World
- Flask-SQLAlchemy
- my new article
- dummy new article
- Using relationship
- 2nd Using relationship



Now this is accessible on internet. And it is a temporary URL, so we will use it only for testing. Now I can give that here. And like, I want to choose HTML, JSON format, and not XML, or rather XWW form URL encoded format, I want a JSON format. And then there is a secret, we will see why secret is used? This secret is used to encode or send a header that we can validate that the data is coming from here. Otherwise, anyone could post here.

They can guess the, if they can guess the URL, they can post to this endpoint, how will you decide it came from GitHub? For that, I am just going to give you a dummy password.

(Refer Slide Time: 15:40)

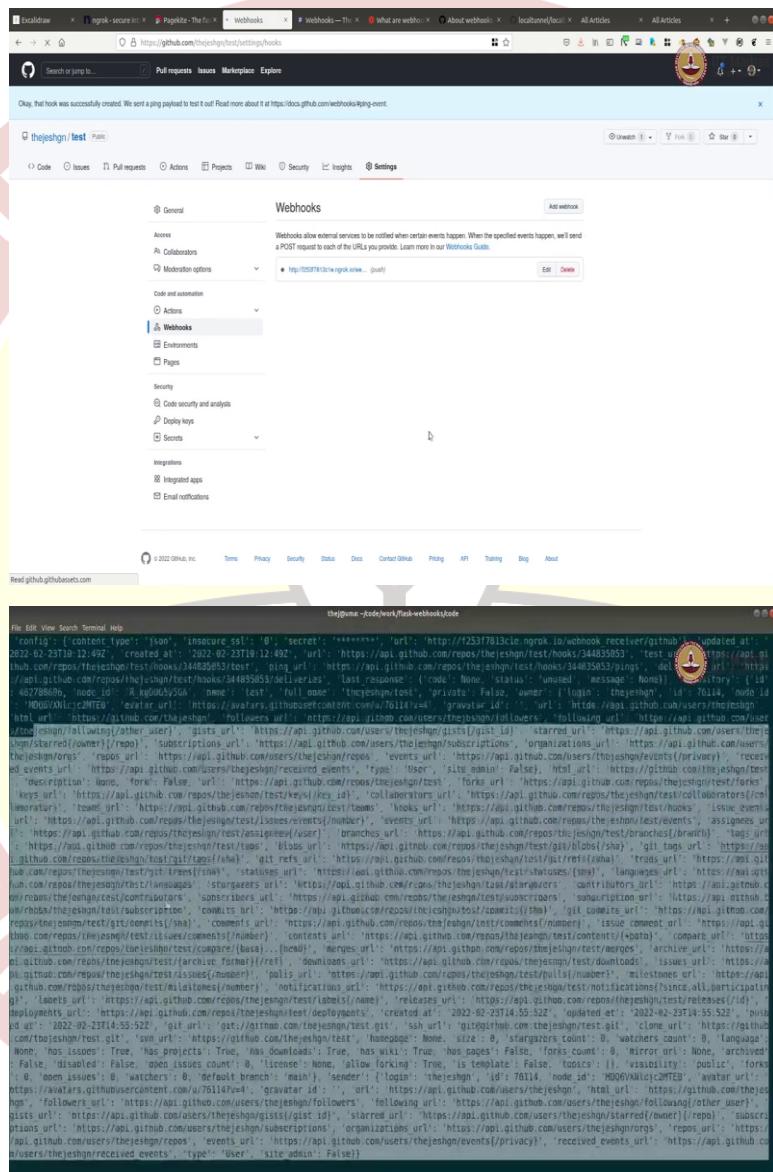
The image shows a desktop environment with three windows open. The central window is a Sublime Text editor displaying a single line of text: "kuaywjsfhfresgd2qu6497uv". To the left of this window is a GitHub repository page for "thejeshgn/test", specifically the "Webhooks" settings. To the right is another identical GitHub "Webhooks" settings page. The GitHub pages show a webhook configuration with the following details:

- Payload URL:** http://2537913.cse.ngrok.io/webhook_recieverghub
- Content type:** application/json
- Secret:** kuaywjsfhfresgd2qu6497uv
- Which events would you like to trigger this webhook?** (radio buttons):
 - Just the push event.
 - Send me everything.
 - Let me select individual events.
- Active:** (checkbox) checked

The Sublime Text editor window has a sidebar showing project files and a main area with the same string "kuaywjsfhfresgd2qu6497uv".

Or like I am going to create like a strong password of random numbers and strings, it is important to have a good password. If you are, or this will work like a password that way. I am going to have it here and create. And this is not to be shared with any1, I am going to remove this repo anyway post this session, because I do not want all of you getting it. Now, as soon as I click.

(Refer Slide Time: 16:16)



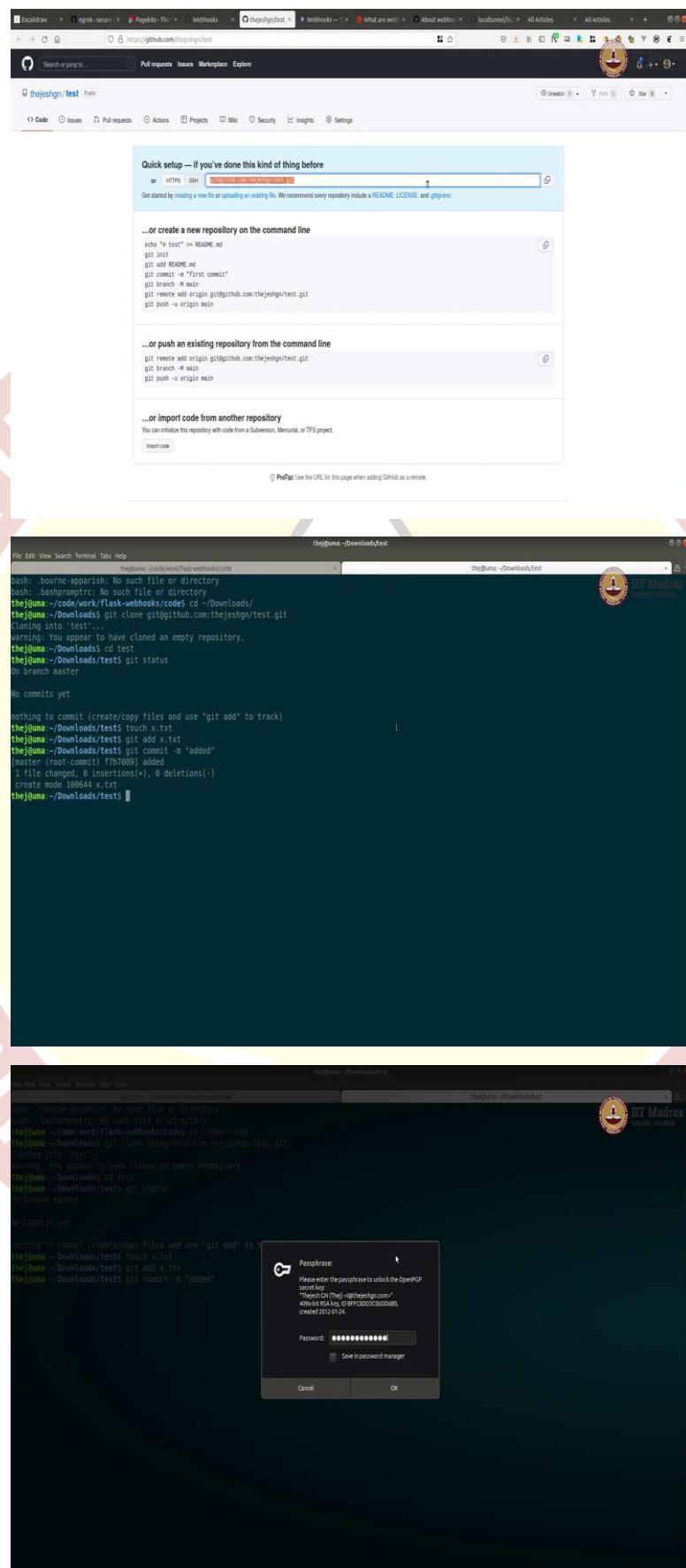
The screenshot shows two terminal windows on a Linux desktop. The top window is a GitHub repository page for 'thejeshgn/test'. The bottom window is a terminal window showing the command 'git pull' being run.

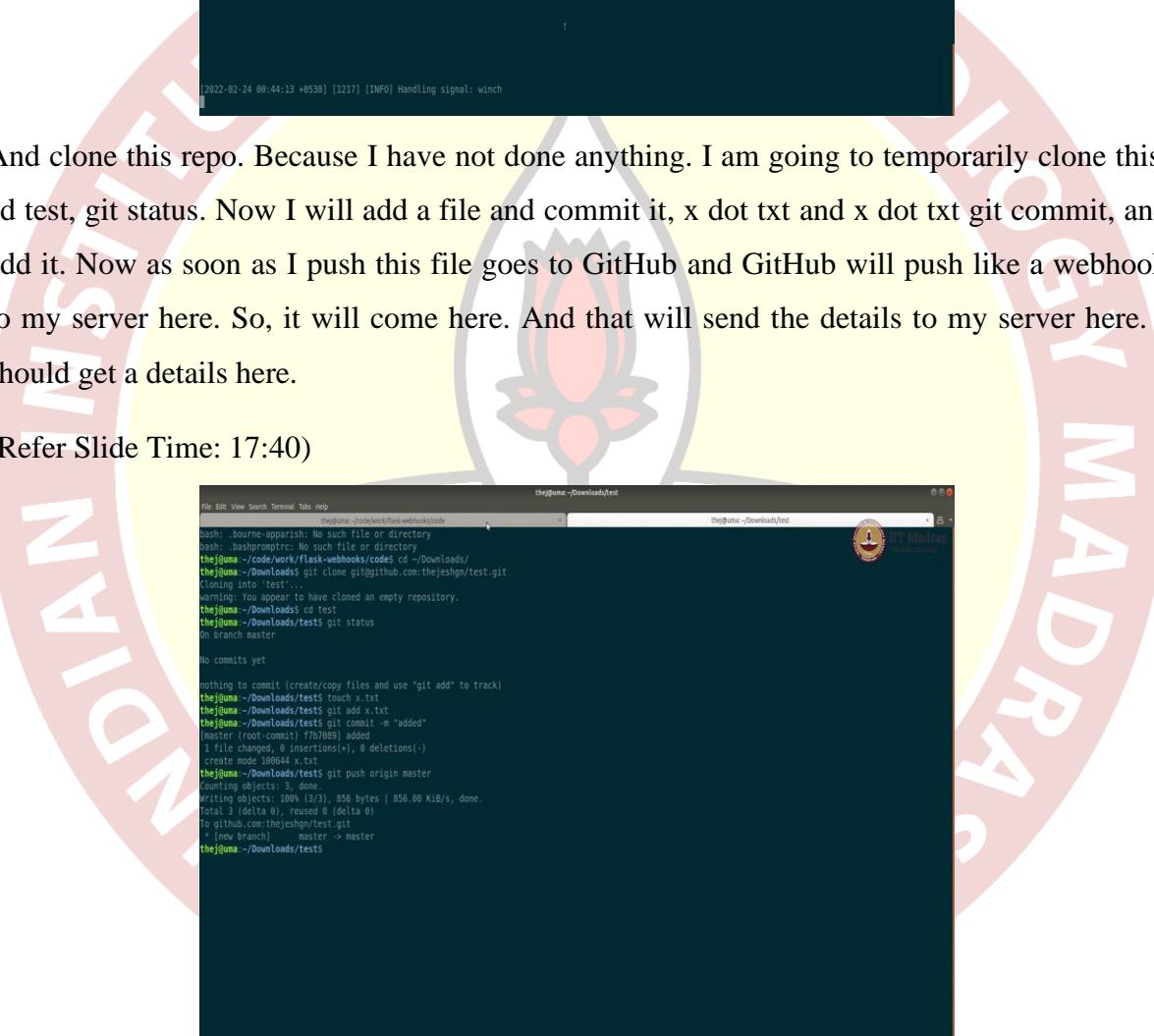
```
thejeshgn@thejeshgn-Lenovo-G50-70:~/code/work/flash-webhooks$ git pull
remote: Enumerating objects: 1, done.
remote: Total 1 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (1/1), done.
From https://github.com/thejeshgn/test
 * [new branch] master     -> origin/master
Branch master updated, and now points to commit 7a9a237c55522e0a23d434f43455522e0a23d434.

```

It is added into the service. Now, whenever there is an event, I should get the details here. So you already got one detail because it got added. Now let us see. Let us do a commit, I am going to open another tab.

(Refer Slide Time: 16:41)





```
thej@juna:~/code/work/flask-webhooks$ curl -X POST https://api.github.com/repos/thejeshgn/test/hooks/1217 --data '{ "name": "webhook", "config": { "url": "http://127.0.0.1:5000/webhook", "content_type": "json" } }' -H "Content-Type: application/json" -H "Authorization: token MDEyOVNjcjCMTE0" -H "User-Agent: Python-HTTPLibrary/0.12.0"
```

[2022-02-24 00:44:13 +0530] [1217] [INFO] Handling signal: winch

And clone this repo. Because I have not done anything. I am going to temporarily clone this, cd test, git status. Now I will add a file and commit it, x dot txt and x dot txt git commit, and add it. Now as soon as I push this file goes to GitHub and GitHub will push like a webhook to my server here. So, it will come here. And that will send the details to my server here. I should get a details here.

(Refer Slide Time: 17:40)

```
thej@juna:~/Downloads$ cd test
```

```
dash:bourne-appraisi: No such file or directory
dash: bash:prompt: No such file or directory
thej@juna:~/code/work/flask-webhooks$ cd ~Downloads/
thej@juna:~/Downloads$ git clone git@github.com:thejeshgn/test.git
Cloning into 'test'...
warning: You appear to have cloned an empty repository.
thej@juna:~/Downloads$ cd test
thej@juna:~/Downloads/test$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use 'git add' to track)
thej@juna:~/Downloads/test$ touch x.txt
thej@juna:~/Downloads/test$ git add x.txt
thej@juna:~/Downloads/test$ git commit -m "added"
[master (root-commit) f7b7089] added
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 108644 x.txt
thej@juna:~/Downloads/test$ git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 896 bytes | 856.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:thejeshgn/test.git
 ! [new branch]   master -> master
thej@juna:~/Downloads/test$
```

INDIAN INSTITUTE

BLOGY MADRAS

```
the@uma:~/code/work/flask-webhooks$ code
File Edit View Search Terminal Help
the@uma:~/code/work/flask-webhooks$ code
n/users/thejeshgn/received_events', 'type': 'User', 'site_admin': False}

[2022-02-24 00:44:13 +0530] [2217] [INFO] Handling signal: winch
[ref]: 'refs/heads/master', 'before': '0000000000000000000000000000000000000000', 'after': 'f7b7089e7e99499bdcfc2fe108947bf8aa0clef', 'repository': {'id': 462780696, 'node_id': 'R_kgDGSG5GA', 'name': 'test', 'full_name': 'thejeshgn/test', 'private': False, 'owner': {'name': 'thejeshgn', 'email': 'i@thejeshgn.com', 'login': 'thejeshgn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTE0', 'avatar_url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar_id': '', 'url': 'https://api.github.com/users/thejeshgn', 'html_url': 'https://github.com/thejeshgn', 'followers_url': 'https://api.github.com/users/thejeshgn/followers', 'following_url': 'https://api.github.com/users/thejeshgn/following/other_user', 'gists_url': 'https://api.github.com/users/thejeshgn/gists/{gist_id}', 'starred_url': 'https://api.github.com/users/thejeshgn/starred{/owner}', 'subscriptions_url': 'https://api.github.com/users/thejeshgn/subscriptions', 'organization_url': 'https://api.github.com/users/thejeshgn/organizations', 'repos_url': 'https://api.github.com/users/thejeshgn/repos', 'events_url': 'https://api.github.com/users/thejeshgn/events{/privacy}', 'received_events_url': 'https://api.github.com/users/thejeshgn/received_events', 'type': 'User', 'site_admin': False}, 'html_url': 'https://github.com/thejeshgn/test', 'description': None, 'fork': False, 'forks_url': 'https://api.github.com/repos/thejeshgn/test/forks', 'collaborators_url': 'https://api.github.com/repos/thejeshgn/test/collaborators{/collaborator}', 'teams_url': 'https://api.github.com/repos/thejeshgn/test/teams', 'hooks_url': 'https://api.github.com/repos/thejeshgn/test/hooks', 'issues_url': 'https://api.github.com/repos/thejeshgn/test/issues{/number}', 'events_url': 'https://api.github.com/repos/thejeshgn/test/events', 'assignees_url': 'https://api.github.com/repos/thejeshgn/test/assignees{/user}', 'branches_url': 'https://api.github.com/repos/thejeshgn/test/branches{/branch}', 'tags_url': 'https://api.github.com/repos/thejeshgn/test/tags', 'blobs_url': 'https://api.github.com/repos/thejeshgn/test/git/blobs{/sha}', 'git_tags_url': 'https://api.github.com/repos/thejeshgn/test/git/tags{/sha}', 'git_refs_url': 'https://api.github.com/repos/thejeshgn/test/git/refs{/sha}', 'trees_url': 'https://api.github.com/repos/thejeshgn/test/git/trees{/sha}', 'statuses_url': 'https://api.github.com/repos/thejeshgn/test/statuses{/sha}', 'languages_url': 'https://api.github.com/repos/thejeshgn/test/languages{/name}', 'contributors_url': 'https://api.github.com/repos/thejeshgn/test/contributors', 'subscribers_url': 'https://api.github.com/repos/thejeshgn/test/subscribers', 'subscription_url': 'https://api.github.com/repos/thejeshgn/test/subscription', 'commits_url': 'https://api.github.com/repos/thejeshgn/test/commits{/sha}', 'git_commits_url': 'https://api.github.com/repos/thejeshgn/test/git/commits{/sha}', 'comments_url': 'https://api.github.com/repos/thejeshgn/test/comments{/number}', 'issue_comment_url': 'https://api.github.com/repos/thejeshgn/test/issues{/comment}', 'contents_url': 'https://api.github.com/repos/thejeshgn/test/contents{/path}', 'compare_url': 'https://api.github.com/repos/thejeshgn/test/compare{/base..{head}}', 'merges_url': 'https://api.github.com/repos/thejeshgn/test/merges', 'archive_url': 'https://api.github.com/repos/thejeshgn/test/archive/{sha}.tar.gz', 'download_url': 'https://api.github.com/repos/thejeshgn/test/download'}
```

```
File Edit Selection Find View Goto Tools Project Preferences Help
Ref: refs/heads/master, before: 00000000 • (flask-webhooks) - Sublime Text
FOLDERS
• flask-webhooks
  • __init__.py
  • __main__.py
  • __pycache__
  • application
    • __init__.py
    • app.py
    • config.py
    • controllers.py
    • database.py
    • models.py
    • tasks.py
    • validation.py
    • webhooks.py
  • tests.py
  • __tests__.py
  • db_directory
  • static
  • templates
    • compiled_schedule
      • debug.log
      • local_base.sh
      • local_gunicorn.sh
      • local_run.sh
      • local_setup.sh
      • local_workers.sh
      • main.py
      • README.md
      • requirements.txt
      • readme.md
  • README.md

1 {ref}: 'refs/heads/master', 'before': '0000000000000000000000000000000000000000', 'after': 'f7b7089e7e99499bdcfc2fe108947bf8aa0clef', 'repository': {'id': 462780696, 'node_id': 'R_kgDGSG5GA', 'name': 'test', 'full_name': 'thejeshgn/test', 'private': False, 'owner': {'name': 'thejeshgn', 'email': 'i@thejeshgn.com', 'login': 'thejeshgn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTE0', 'avatar_url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar_id': '', 'url': 'https://api.github.com/users/thejeshgn', 'html_url': 'https://github.com/thejeshgn/test', 'followers_url': 'https://api.github.com/users/thejeshgn/followers', 'following_url': 'https://api.github.com/users/thejeshgn/following/other_user', 'gists_url': 'https://api.github.com/users/thejeshgn/gists/{gist_id}', 'starred_url': 'https://api.github.com/users/thejeshgn/starred{/owner}', 'subscriptions_url': 'https://api.github.com/users/thejeshgn/subscriptions', 'organizations_url': 'https://api.github.com/users/thejeshgn/orgs', 'repos_url': 'https://api.github.com/repos/thejeshgn/test/forks', 'collaborators_url': 'https://api.github.com/repos/thejeshgn/test/collaborators{/collaborator}', 'teams_url': 'https://api.github.com/repos/thejeshgn/test/teams', 'hooks_url': 'https://api.github.com/repos/thejeshgn/test/hooks', 'issues_url': 'https://api.github.com/repos/thejeshgn/test/issues{/number}', 'events_url': 'https://api.github.com/repos/thejeshgn/test/events', 'assignees_url': 'https://api.github.com/repos/thejeshgn/test/assignees{/user}', 'branches_url': 'https://api.github.com/repos/thejeshgn/test/branches{/branch}', 'tags_url': 'https://api.github.com/repos/thejeshgn/test/tags', 'blobs_url': 'https://api.github.com/repos/thejeshgn/test/git/blobs{/sha}', 'git_tags_url': 'https://api.github.com/repos/thejeshgn/test/git/tags{/sha}', 'git_refs_url': 'https://api.github.com/repos/thejeshgn/test/git/refs{/sha}', 'trees_url': 'https://api.github.com/repos/thejeshgn/test/git/trees{/sha}', 'statuses_url': 'https://api.github.com/repos/thejeshgn/test/statuses{/sha}', 'languages_url': 'https://api.github.com/repos/thejeshgn/test/languages{/name}'}

Expecting value:
line 1 column 468
(char 467)
```

```
File Edit Selection Find View Goto Tools Project Preferences Help
Ref: refs/heads/master, before: 00000000 • (flask-webhooks) - Sublime Text
FOLDERS
• flask-webhooks
  • __init__.py
  • __main__.py
  • __pycache__
  • application
    • __init__.py
    • app.py
    • config.py
    • controllers.py
    • database.py
    • models.py
    • tasks.py
    • validation.py
    • webhooks.py
  • tests.py
  • __tests__.py
  • db_directory
  • static
  • templates
    • compiled_schedule
      • debug.log
      • local_base.sh
      • local_gunicorn.sh
      • local_run.sh
      • local_setup.sh
      • local_workers.sh
      • main.py
      • README.md
      • requirements.txt
      • readme.md
  • README.md

1 {ref}: 'refs/heads/master', 'before': '0000000000000000000000000000000000000000', 'after': 'f7b7089e7e99499bdcfc2fe108947bf8aa0clef', 'repository': {'id': 462780696, 'node_id': 'R_kgDGSG5GA', 'name': 'test', 'full_name': 'thejeshgn/test', 'private': False, 'owner': {'name': 'thejeshgn', 'email': 'i@thejeshgn.com', 'login': 'thejeshgn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTE0', 'avatar_url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar_id': '', 'url': 'https://api.github.com/users/thejeshgn', 'html_url': 'https://github.com/thejeshgn/test', 'followers_url': 'https://api.github.com/users/thejeshgn/followers', 'following_url': 'https://api.github.com/users/thejeshgn/following/other_user', 'gists_url': 'https://api.github.com/users/thejeshgn/gists/{gist_id}', 'starred_url': 'https://api.github.com/users/thejeshgn/starred{/owner}', 'subscriptions_url': 'https://api.github.com/users/thejeshgn/subscriptions', 'organizations_url': 'https://api.github.com/users/thejeshgn/orgs', 'repos_url': 'https://api.github.com/users/thejeshgn/test/forks', 'collaborators_url': 'https://api.github.com/repos/thejeshgn/test/collaborators{/collaborator}', 'teams_url': 'https://api.github.com/repos/thejeshgn/test/teams', 'hooks_url': 'https://api.github.com/repos/thejeshgn/test/hooks', 'issues_url': 'https://api.github.com/repos/thejeshgn/test/issues{/number}', 'events_url': 'https://api.github.com/repos/thejeshgn/test/events', 'assignees_url': 'https://api.github.com/repos/thejeshgn/test/assignees{/user}', 'branches_url': 'https://api.github.com/repos/thejeshgn/test/branches{/branch}', 'tags_url': 'https://api.github.com/repos/thejeshgn/test/tags', 'blobs_url': 'https://api.github.com/repos/thejeshgn/test/git/blobs{/sha}', 'git_tags_url': 'https://api.github.com/repos/thejeshgn/test/git/tags{/sha}', 'git_refs_url': 'https://api.github.com/repos/thejeshgn/test/git/refs{/sha}', 'trees_url': 'https://api.github.com/repos/thejeshgn/test/git/trees{/sha}', 'statuses_url': 'https://api.github.com/repos/thejeshgn/test/statuses{/sha}', 'languages_url': 'https://api.github.com/repos/thejeshgn/test/languages{/name}'}

Expecting value:
line 1 column 468
(char 467)
```

File Edit Selection Find View Goto Tools Project Preferences Help

```
[ref:refs/heads/master, before: 3000000000000000000000000000000000000000 - (Task-webhooks) - Sublime Text
```

FOLDERS

- iit-webhooks
- lib
- lib_coder
- lib_env
- lib_github
- lib_application
- lib_github_
 - .. lib_github_py
 - .. lib_github_env
 - .. lib_github_config.py
 - .. lib_github_controllers.py
 - .. lib_github_database.py
 - .. lib_github_models.py
 - .. lib_github_tasks.py
 - .. lib_github_validation.py
 - .. lib_github_webhooks.py
 - .. lib_github_webhooks.py
- lib_db_directory
- lib_static
- lib_templates
 - calendarSchedule
 - delete_log
 - local_base.sh
 - local_base.sh
 - local_base.sh
 - local_setup.sh
 - local_workers.sh
 - .. main.py
 - .. README.md
 - .. requirements.txt
 - .. README.md

1 { 'ref': 'refs/heads/master', 'before': '3000000000000000000000000000000000000000', 'repository': { 'id': 46278996, 'node_id': 'R_kgDGSV5GA', 'name': 'test', 'full_name': 'thejeshgn/test', 'private': false, 'owner': { 'name': 'thejeshgn', 'email': 'i@thejeshgn.com', 'login': 'thejeshgn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTE0', 'avatar_url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar_id': '', 'url': 'https://api.github.com/users/thejeshgn', 'html_url': 'https://github.com/thejeshgn', 'followers_url': 'https://api.github.com/users/thejeshgn/followers', 'following_url': 'https://api.github.com/users/thejeshgn/following{/other_user}', 'gists_url': 'https://api.github.com/users/thejeshgn/gists{/gist_id}', 'starred_url': 'https://api.github.com/users/thejeshgn/starred{/owner}{/repo}', 'subscriptions_url': 'https://api.github.com/users/thejeshgn/subscriptions', 'organizations_url': 'https://api.github.com/users/thejeshgn/orgs', 'repos_url': 'https://api.github.com/users/thejeshgn/repos', 'events_url': 'https://api.github.com/users/thejeshgn/events{/privacy}', 'received_events_url': 'https://api.github.com/users/thejeshgn/received_events', 'type': 'User', 'site_admin': False}, 'html_url': 'https://github.com/thejeshgn/test', 'description': None, 'fork': False, 'url': 'https://github.com/thejeshgn/test', 'forks_url': 'https://api.github.com/repos/thejeshgn/test/forks', 'keys_url': 'https://api.github.com/repos/thejeshgn/test/keys{/key_id}', 'collaborators_url': 'https://api.github.com/repos/thejeshgn/test/collaborators{/collaborator}', 'teams_url': 'https://api.github.com/repos/thejeshgn/test/teams', 'hooks_url': 'https://api.github.com/repos/thejeshgn/test/hooks', 'issue_events_url': 'https://api.github.com/repos/thejeshgn/test/issues/events{/number}', 'events_url': 'https://api.github.com/repos/thejeshgn/test/events', 'assignees_url': 'https://api.github.com/repos/thejeshgn/test/assignees{/user}', 'branches_url': 'https://api.github.com/repos/thejeshgn/test/branches{/branch}', 'tags_url': 'https://api.github.com/repos/thejeshgn/test/tags', 'blobs_url': 'https://api.github.com/repos/thejeshgn/test/git/blobs{/sha}', 'git_tags_url': 'https://api.github.com/repos/thejeshgn/test/git/tags{/sha}', 'git_refs_url': 'https://api.github.com/repos/thejeshgn/test/git/refs{/sha}', 'trees_url': 'https://api.github.com/repos/thejeshgn/test/git/trees{/sha}', 'statuses_url': 'https://api.github.com/repos/thejeshgn/test/statuses{/sha}', 'languages_url': 'https://api.github.com/repos/thejeshgn/test/languages'}

Source 4 Plain Text

Let us see, git push origin master it got pushed, you see I have received you can actually format this JSON to see what it has sent. Is this related to this event? I think it is not printed as text. So, to print it from Python. So, that is why it is not printed properly. Anyway. So, you can see that I am the 1 who is pushed to the repo.

(Refer Slide Time: 18:36)

File Edit Selection Find View Goto Tools Project Preferences Help

```
[ref:refs/heads/master, before: 3000000000000000000000000000000000000000 - (Task-webhooks) - Sublime Text
```

FOLDERS

- iit-webhooks
- lib
- lib_coder
- lib_env
- lib_github
- lib_application
- lib_github_
 - .. lib_github_py
 - .. lib_github_env
 - .. lib_github_config.py
 - .. lib_github_controllers.py
 - .. lib_github_database.py
 - .. lib_github_models.py
 - .. lib_github_tasks.py
 - .. lib_github_validation.py
 - .. lib_github_webhooks.py
 - .. lib_github_webhooks.py
- lib_db_directory
- lib_static
- lib_templates
 - calendarSchedule
 - delete_log
 - local_base.sh
 - local_base.sh
 - local_base.sh
 - local_setup.sh
 - local_workers.sh
 - .. main.py
 - .. README.md
 - .. requirements.txt
 - .. README.md

1 { 'ref': 'refs/heads/master', 'before': '3000000000000000000000000000000000000000', 'repository': { 'id': 46278996, 'node_id': 'R_kgDGSV5GA', 'name': 'test', 'full_name': 'thejeshgn/test', 'private': false, 'owner': { 'name': 'thejeshgn', 'email': 'i@thejeshgn.com', 'login': 'thejeshgn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTE0', 'avatar_url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar_id': '', 'url': 'https://api.github.com/users/thejeshgn', 'html_url': 'https://github.com/thejeshgn', 'followers_url': 'https://api.github.com/users/thejeshgn/followers', 'following_url': 'https://api.github.com/users/thejeshgn/following{/other_user}', 'gists_url': 'https://api.github.com/users/thejeshgn/gists{/gist_id}', 'starred_url': 'https://api.github.com/users/thejeshgn/starred{/owner}{/repo}', 'subscriptions_url': 'https://api.github.com/users/thejeshgn/subscriptions', 'organizations_url': 'https://api.github.com/users/thejeshgn/orgs', 'repos_url': 'https://api.github.com/users/thejeshgn/repos', 'events_url': 'https://api.github.com/users/thejeshgn/events{/privacy}', 'received_events_url': 'https://api.github.com/users/thejeshgn/received_events', 'type': 'User', 'site_admin': False}, 'html_url': 'https://github.com/thejeshgn/test', 'description': None, 'fork': False, 'url': 'https://github.com/thejeshgn/test', 'forks_url': 'https://api.github.com/repos/thejeshgn/test/forks', 'keys_url': 'https://api.github.com/repos/thejeshgn/test/keys{/key_id}', 'collaborators_url': 'https://api.github.com/repos/thejeshgn/test/collaborators{/collaborator}', 'teams_url': 'https://api.github.com/repos/thejeshgn/test/teams', 'hooks_url': 'https://api.github.com/repos/thejeshgn/test/hooks', 'issue_events_url': 'https://api.github.com/repos/thejeshgn/test/issues/events{/number}', 'events_url': 'https://api.github.com/repos/thejeshgn/test/events', 'assignees_url': 'https://api.github.com/repos/thejeshgn/test/assignees{/user}', 'branches_url': 'https://api.github.com/repos/thejeshgn/test/branches{/branch}', 'tags_url': 'https://api.github.com/repos/thejeshgn/test/tags', 'blobs_url': 'https://api.github.com/repos/thejeshgn/test/git/blobs{/sha}', 'git_tags_url': 'https://api.github.com/repos/thejeshgn/test/git/tags{/sha}', 'git_refs_url': 'https://api.github.com/repos/thejeshgn/test/git/refs{/sha}', 'trees_url': 'https://api.github.com/repos/thejeshgn/test/git/trees{/sha}', 'statuses_url': 'https://api.github.com/repos/thejeshgn/test/statuses{/sha}', 'languages_url': 'https://api.github.com/repos/thejeshgn/test/languages'}

2 { 'received_events': [{ 'type': 'push', 'base_commit': null, 'create': true, 'decrements': 0, 'distinct': true, 'force': false, 'forced': false, 'head_commit': { 'id': 'fb7089e7e99499bdcfc2fef108947bf8aa0c1ef', 'tree_id': 'c3fe1f0484edb75ca3c45e49ba19d820c84c179' }, 'commit': { 'id': 'fb7089e7e99499bdcfc2fef108947bf8aa0c1ef', 'tree_id': 'c3fe1f0484edb75ca3c45e49ba19d820c84c179' }, 'distinct': true, 'message': 'added', 'timestamp': '2022-02-24T00:43:58+05:30', 'url': 'https://github.com/thejeshgn/test/commit/fb7089e7e99499bdcfc2fef108947bf8aa0c1ef' }, { 'type': 'push', 'base_commit': null, 'create': true, 'decrements': 0, 'distinct': true, 'force': false, 'forced': false, 'head_commit': { 'id': 'fb7089e7e99499bdcfc2fef108947bf8aa0c1ef', 'tree_id': 'c3fe1f0484edb75ca3c45e49ba19d820c84c179' }, 'commit': { 'id': 'fb7089e7e99499bdcfc2fef108947bf8aa0c1ef', 'tree_id': 'c3fe1f0484edb75ca3c45e49ba19d820c84c179' }, 'distinct': true, 'message': 'added', 'timestamp': '2022-02-24T00:43:58+05:30', 'url': 'https://github.com/thejeshgn/test/commit/fb7089e7e99499bdcfc2fef108947bf8aa0c1ef' }, { 'type': 'push', 'base_commit': null, 'create': true, 'decrements': 0, 'distinct': true, 'force': false, 'forced': false, 'head_commit': { 'id': 'fb7089e7e99499bdcfc2fef108947bf8aa0c1ef', 'tree_id': 'c3fe1f0484edb75ca3c45e49ba19d820c84c179' }, 'commit': { 'id': 'fb7089e7e99499bdcfc2fef108947bf8aa0c1ef', 'tree_id': 'c3fe1f0484edb75ca3c45e49ba19d820c84c179' }, 'distinct': true, 'message': 'added', 'timestamp': '2022-02-24T00:43:58+05:30', 'url': 'https://github.com/thejeshgn/test/commit/fb7089e7e99499bdcfc2fef108947bf8aa0c1ef' }], 'removed': [], 'modified': [] }

Source 4 Plain Text

File Edit Selection Find View Goto Tools Project Preferences Help

104-04-2007, git@github.com:thejeshn/test.git, ssh url: https://github.com/thejeshn/test.git, svn url: https://github.com/thejeshn/test/, homepage: None, size: 0, stargazers count: 0, watchers count: 0, language: None, has issues: True, has projects: True, has downloads: True, has wiki: True, has pages: False, forks count: 0, mirror url: None, archived: False, disabled: False, open issue count: 0, license: None, allow forking: True, is template: False, topics: [], visibility: 'public', forks: 0, open issues: 0, watchers: 0, default branch: 'master', stargazers: 0, master branch: 'master', pusher: {'name': 'thejeshn', 'email': 'i@thejeshn.com'}, sender: {'login': 'thejeshn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTED0', 'avatar url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar id': '', 'url': 'https://api.github.com/users/thejeshn', 'html url': 'https://github.com/thejeshn', 'followers url': 'https://api.github.com/users/thejeshn/followers', 'following url': 'https://api.github.com/users/thejeshn/following/other user', 'gists url': 'https://api.github.com/users/thejeshn/gists/gist id', 'starred url': 'https://api.github.com/users/thejeshn/starred/owner/repo', 'subscriptions url': 'https://api.github.com/users/thejeshn/subscriptions', 'organizations url': 'https://api.github.com/users/thejeshn/orgs', 'repos url': 'https://api.github.com/users/thejeshn/repos', 'events url': 'https://api.github.com/users/thejeshn/events/privacy', 'received events url': 'https://api.github.com/users/thejeshn/received events', type: 'User', site admin: False}, created: True, deleted: False, forced: False, base ref: None, compare: 'https://github.com/thejeshn/test/commit/f7b7089e7e99499bdcf2fe108947bf8aa0a1c79', distinct: True, message: 'added', timestamp: '2022-02-24T00:43:58+05:30', url: 'https://github.com/Thejeshn/test/commit/f7b7089e7e99499bdcf2fe108947bf8aa0a1c79', author: {'name': 'Thejes GN', 'email': 'i@thejeshn.com', 'username': 'thejeshn'}, committer: {'name': 'Thejes GN', 'email': 'i@thejeshn.com', 'username': 'thejeshn'}, added: ['x.txt'], removed: [], modified: []}, head commit: {'id': 'f7b7089e7e99499bdcf2fe108947bf8aa0a1c79', 'tree id': 'c3fe1f0484ed75ca3c45e49ba19d820c84c179', 'distinct': True, 'message': 'added', 'timestamp': '2022-02-24T00:43:58+05:30', 'url': 'https://github.com/thejeshn/test/commit/f7b7089e7e99499bdcf2fe108947bf8aa0a1c79'}]

File Edit Selection Find View Goto Tools Project Preferences Help

1645643667, git url: 'git@github.com:thejeshn/test.git', ssh url: 'git@github.com:thejeshn/test.git', clone url: 'https://github.com/thejeshn/test.git', svn url: 'https://github.com/thejeshn/test/', homepage: None, size: 0, stargazers count: 0, watchers count: 0, language: None, has issues: True, has projects: True, has downloads: True, has wiki: True, has pages: False, forks count: 0, mirror url: None, archived: False, disabled: False, open issue count: 0, license: None, allow forking: True, is template: False, topics: [], visibility: 'public', forks: 0, open issues: 0, watchers: 0, default branch: 'master', stargazers: 0, master branch: 'master', pusher: {'name': 'thejeshn', 'email': 'i@thejeshn.com'}, sender: {'login': 'thejeshn', 'id': 76114, 'node_id': 'MDQ6VXNlcj2MTED0', 'avatar url': 'https://avatars.githubusercontent.com/u/76114?v=4', 'gravatar id': '', 'url': 'https://api.github.com/users/thejeshn', 'html url': 'https://github.com/thejeshn', 'followers url': 'https://api.github.com/users/thejeshn/followers', 'following url': 'https://api.github.com/users/thejeshn/following/other user', 'gists url': 'https://api.github.com/users/thejeshn/gists/gist id', 'starred url': 'https://api.github.com/users/thejeshn/starred/owner/repo', 'subscriptions url': 'https://api.github.com/users/thejeshn/subscriptions', 'organizations url': 'https://api.github.com/users/thejeshn/orgs', 'repos url': 'https://api.github.com/users/thejeshn/repos', 'events url': 'https://api.github.com/users/thejeshn/events/privacy', 'received events url': 'https://api.github.com/users/thejeshn/received events', type: 'User', site admin: False}, created: True, deleted: False, forced: False, base ref: None, compare: 'https://github.com/thejeshn/test/commit/f7b7089e7e99499bdcf2fe108947bf8aa0a1c79', distinct: True, message: 'added', timestamp: '2022-02-24T00:43:58+05:30', url: 'https://github.com/Thejeshn/test/commit/f7b7089e7e99499bdcf2fe108947bf8aa0a1c79', author: {'name': 'Thejes GN', 'email': 'i@thejeshn.com', 'username': 'thejeshn'}, committer: {'name': 'Thejes GN', 'email': 'i@thejeshn.com', 'username': 'thejeshn'}, added: ['x.txt'], removed: [], modified: []}, head commit: {'id': 'f7b7089e7e99499bdcf2fe108947bf8aa0a1c79', 'tree id': 'c3fe1f0484ed75ca3c45e49ba19d820c84c179', 'distinct': True, 'message': 'added', 'timestamp': '2022-02-24T00:43:58+05:30', 'url': 'https://github.com/thejeshn/test/commit/f7b7089e7e99499bdcf2fe108947bf8aa0a1c79'}

And all the details of the repo, etcetera, etcetera here. Including my email, what, who was on who committed all of those details.

(Refer Slide Time: 18:47)

The screenshots show the GitHub webhook settings page for the repository 'thejeyshgn/test'. The first screenshot shows the basic configuration with an active webhook. The subsequent two screenshots show the raw JSON payload of the webhook event, which includes details about the push to the 'master' branch.

Recent Deliveries

Headers

```
Request URL: http://7231f7813cae.ngrok.io/webhook_receiver/github
Request method: POST
Accept: */*
Content-Type: application/json
User-Agent: GitHub-Hookshot/7af2776
X-GitHub-Delivery: 61d5fe5f-94c1-11ec-8684-7ed54013f67a
X-GitHub-Event: push
X-GitHub-Hook-ID: 34405583
X-GitHub-Hook-Installation-Target-ID: 40278906
X-GitHub-Hook-Installation-Target-Type: repository
X-Hub-Signature: 258:sha256=c981d741639c78402b5323acec557285e4fa3d9401a3e7e9691678871432
```

Payload

```
{
  "ref": "refs/heads/master",
  "before": "8000000000000000000000000000000000000000",
  "after": "717079091eb94990dc12fe188947bf8a6c5",
  "repository": {
    "id": 4279906,
    "node_id": "M_1gQ0GDSU5A",
    "name": "test",
    "full_name": "thejeyshgn/test",
    "private": false,
    "owner": {
      "login": "thejeyshgn",
      "email": "thejeyshgn.com",
      "type": "Organization"
    }
  },
  "event": "push",
  "created": true
}
```

Recent Deliveries

Headers

```
Request URL: http://7231f7813cae.ngrok.io/webhook_receiver/github
Request method: POST
Accept: */*
Content-Type: application/json
User-Agent: GitHub-Hookshot/7af2776
X-GitHub-Delivery: 61d5fe5f-94c1-11ec-8684-7ed54013f67a
X-GitHub-Event: push
X-GitHub-Hook-ID: 34405583
X-GitHub-Hook-Installation-Target-ID: 40278906
X-GitHub-Hook-Installation-Target-Type: repository
X-Hub-Signature: 258:sha256=c981d741639c78402b5323acec557285e4fa3d9401a3e7e9691678871432
```

Payload

```
{
  "ref": "refs/heads/master",
  "before": "8000000000000000000000000000000000000000",
  "after": "717079091eb94990dc12fe188947bf8a6c5",
  "repository": {
    "id": 4279906,
    "node_id": "M_1gQ0GDSU5A",
    "name": "test",
    "full_name": "thejeyshgn/test",
    "private": false,
    "owner": {
      "login": "thejeyshgn",
      "email": "thejeyshgn.com",
      "type": "Organization"
    }
  },
  "event": "push",
  "created": true
}
```

Recent Deliveries

Headers

```
Request URL: http://7231f7813cae.ngrok.io/webhook_receiver/github
Request method: POST
Accept: */*
Content-Type: application/json
User-Agent: GitHub-Hookshot/7af2776
X-GitHub-Delivery: 61d5fe5f-94c1-11ec-8684-7ed54013f67a
X-GitHub-Event: push
X-GitHub-Hook-ID: 34405583
X-GitHub-Hook-Installation-Target-ID: 40278906
X-GitHub-Hook-Installation-Target-Type: repository
X-Hub-Signature: 258:sha256=c981d741639c78402b5323acec557285e4fa3d9401a3e7e9691678871432
```

Payload

```
{
  "ref": "refs/heads/master",
  "before": "8000000000000000000000000000000000000000",
  "after": "717079091eb94990dc12fe188947bf8a6c5",
  "repository": {
    "id": 4279906,
    "node_id": "M_1gQ0GDSU5A",
    "name": "test",
    "full_name": "thejeyshgn/test",
    "private": false,
    "owner": {
      "login": "thejeyshgn",
      "email": "thejeyshgn.com",
      "type": "Organization"
    }
  },
  "event": "push",
  "created": true
}
```

```

Request URL: https://7292f9f5c5a.ngrok.io/webhook_receiver/github
Request Method: POST
Request Headers:
Accept: */*
Content-Type: application/json
User-Agent: GitHub-Webhook/7292f9f5c5a.ngrok.io
X-GitHub-Delivery: 61fb8fe8-8a4c-8864-7ed5d13f7a
X-GitHub-Event: push
X-GitHub-Repo-ID: 34481582
X-GitHub-Signature: sha1=dd4debea6cd0801a5d5020ca944007c5476
X-Hub-Signature-256: sha256=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X-Request-Id: 2022-02-01T14:43:19Z
X-Ratelimit-Limit: 5000
X-Ratelimit-Remaining: 4999
X-Ratelimit-Reset: 1643700000
X-Ratelimit-Used: 1
X-Timestamp: 1643700000
X-XSS-Protection: 1; mode=block

```

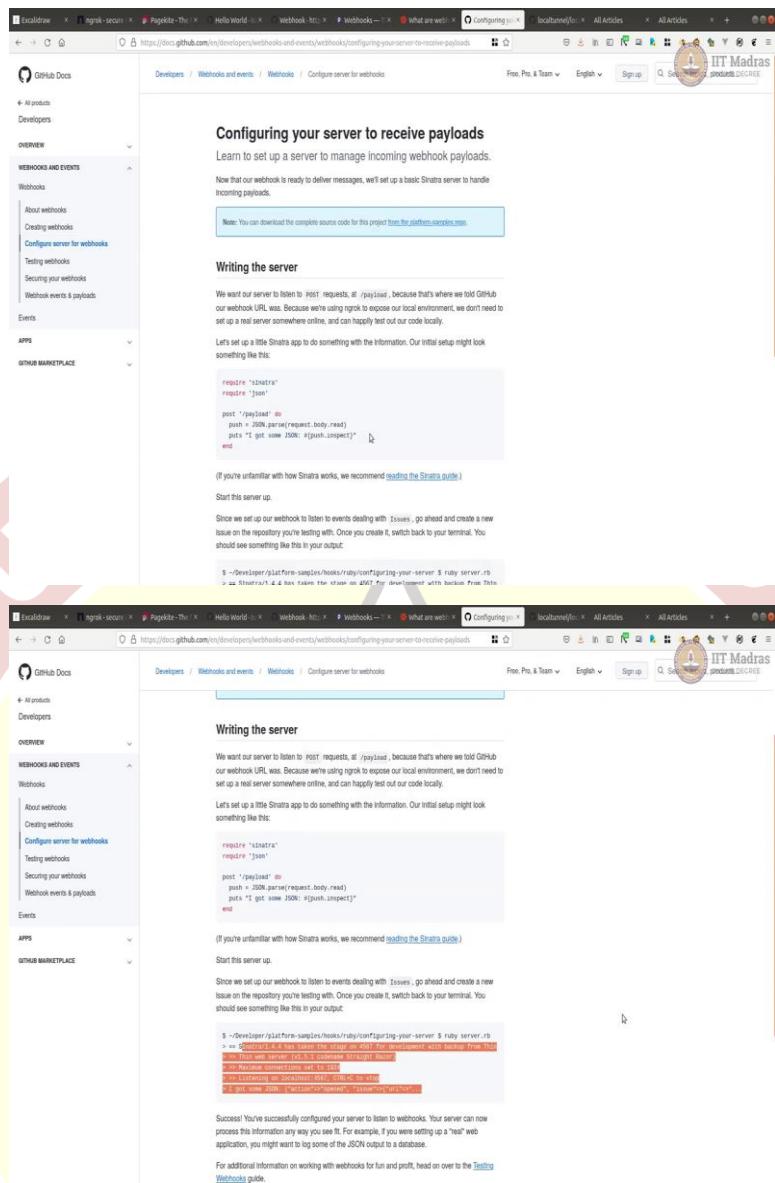
```

{
  "ref": "refs/heads/master",
  "after": "0000000000000000000000000000000000000000",
  "repository": {
    "id": 34481582,
    "name": "keyfeng/test",
    "full_name": "keyfeng/test",
    "owner": {
      "login": "keyfeng",
      "name": "Key Feng",
      "email": "88keyfengs.com",
      "url": "https://github.com/keyfeng"
    }
  }
}

```

So, this is the one which got pushed, you can see that these are the headers it is sending. And we should verify this. Based on you should get this from the header in our code, and we should generate it, and then we should check.

(Refer Slide Time: 19:06)



Let us say in webhook, in GitHub thing they will show, they will tell you how to validate it. Usually all of them tell you or any kind of providers. Here it is, how to receive and then how to validate.

• सद्विर्भवति कर्मजा •

(Refer Slide Time: 19:29)

Screenshot 1: Setting your secret token

In this article: Setting your secret token, Validating payloads from GitHub.

for security reasons.

Once your server is configured to receive payloads, it'll listen for any payload sent to the endpoint you configured. For security reasons, you probably want to limit requests to those coming from GitHub. There are a few ways to go about this—for example, you could opt to allow requests from GitHub's IP address—but a far easier method is to set up a secret token and validate the information.

The webhook REST API enable you to manage repository, organization, and app webhooks. You can use this API to list webhook deliveries for a webhook, or get and redeliver an individual delivery for a webhook, which can be integrated into an external app or service. You can also use the REST API to change the configuration of the webhook. For example, you can modify the payload URL, content type, SSL verification, and secret. For more information, see:

- Repository Webhooks REST API
- Organization Webhooks REST API
- GitHub App Webhooks REST API

Setting your secret token

You'll need to set up your secret token in two places: GitHub and your server.

To set your token in GitHub:

- 1 Navigate to the repository where you're setting up your webhook.
- 2 Fill out the Secret text box. Use a random string with high entropy (e.g., by taking the output of `ruby -rsecurerandom -e "puts SecureRandom.hex(32)"` at the terminal).

Content type: application/x-www-form-urlencoded

Secret:

By default, we verify SSL certificate when delivering payloads. Disable SSL verification

Which events would you like to trigger this webhook?

Screenshot 2: Validating payloads from GitHub

In this article: Click Update Webhook, Setting your secret token, Validating payloads from GitHub.

Next, set up an environment variable on your server that stores this token. Typically, this is as simple as running:

```
$ export SECRET_TOKEN=your_token
```

Never hardcode the token into your app!

Validating payloads from GitHub

When your secret token is set, GitHub uses it to create a hash signature with each payload. This hash signature is included with the headers of each request as `X-Hub-Signature=256`.

Note: For behavior-compatibility, we also include the `X-Hub-Signature` header that is generated using the SHA-1 hash function. If possible, we recommend that you use the `X-Hub-Signature=256` header for improved security. The example below demonstrates using the `X-Hub-Signature=256` header.

For example, if you have a basic server that listens for webhooks, it might be configured similar to this:

```
require 'securerandom'
require 'json'

post '/payload' do
  request.body.read
  push = JSON.parse(request.body.read)
  "I got some JSON: #{push.inspect}"
end
```

The intention is to calculate a hash using your `SECRET_TOKEN`, and ensure that the result matches the hash from GitHub. GitHub uses an HMAC hex digest to compute the hash, so you could reconfigure your server to look a little like this:

```
require 'json'
require 'digest'
require 'base64'

post '/payload' do
  request.body.read
  payload_body = request.body.read
  push = JSON.parse(payload_body)
  push['signature'] = Digest::SHA256.new.base64digest("[" + push.to_json + "]")
  "I got some JSON: #{push.inspect}"
end
```

The intention is to calculate a hash using your `SECRET_TOKEN`, and ensure that the result matches the hash from GitHub. GitHub uses an HMAC hex digest to compute the hash, so you could reconfigure your server to look a little like this:

```
post '/payload' do
  request.body.read
  payload_body = request.body.read
  push = JSON.parse(payload_body)
  push['signature'] = Digest::SHA256.new.base64digest("[" + push.to_json + "]")
  "I got some JSON: #{push.inspect}"
end

def verify_signature(payload_body)
  signature = "sha256=" + OpenSSL::HMAC.digest('sha256', ENV['SECRET_TOKEN'], payload_body)
  return false if "Signature didn't match" unless Rack::Utils.secure_compare(signature, push['signature'])
end
```

Note: Webhook payloads can contain unicode characters. If your language and server implementation specifies a character encoding, ensure that you handle the payload as UTF-8.

Your language and server implementations may differ from this example code. However, there are a number of very important things to point out:

- No matter which implementation you use, the hash signature starts with `sha256=`, using the key of your secret token and your payload body.
- Using a plain `==` operator is **not advised**. A method like `==> ==<==` performs a “constant time” string comparison, which helps mitigate certain timing attacks against regular equality operators.

The intention is to calculate a hash using your `SECRET_TOKEN`, and ensure that the result matches the hash from GitHub. GitHub uses an HMAC hex digest to compute the hash, so you could reconfigure your server to look a little like this:

```

post '*' payload do
  request.body.read
  payload.body = request.body.read
  verify_signature(payload.body)
  push = JSON.parse(payload.body)
  "I got some JSON: #{push.inspect}"
end

def verify_signature(payload_body)
  signature = OpenSSL::Digest::SHA256.new().hexdigest(payload_body).lstrip
  return false, "$signature didn't match!" unless Rack::Utils.secure_compare(signature,
    payload_body)
end

```

Note: Webhook payloads can contain unicode characters. If your language and server implementation specifies a character encoding, ensure that you handle the payload as UTF-8.

Your language and server implementations may differ from this example code. However, there are a number of very important things to point out:

- No matter which implementation you use, the hash signature starts with `sha256-`, using the key of your secret token and your payload body.
- Using a plain `==` operator is **not advised**. A method like `secure_compare` performs a “constant time” string comparison, which helps mitigate certain timing attacks against regular equality operators.

And click on the webhook validating the payload from GitHub. So, you should actually read the body and verify the signature like you. It uses a HMAC. So, they would have given some example code that you can use to verify here.

(Refer Slide Time: 19:51)

When your secret token is set, GitHub uses it to create a hash signature with each payload. This hash signature is included with the headers of each request as `X-Hub-Signature=256`.

Note: For backward-compatibility, we also include the `X-Hub-Signature` header that is generated using the SHA-1 hash function. If possible, we recommend that you use the `X-Hub-Signature=256` header for improved security. The example below demonstrates using the `X-Hub-Signature=256` header.

For example, if you have a basic server that listens for webhooks, it might be configured similar to this:

```

require 'sinatra'
require 'openssl'

get '/' do
  payload = params['payload']
  push = JSON.parse(payload)
  "I got some JSON: #{push.inspect}"
end

```

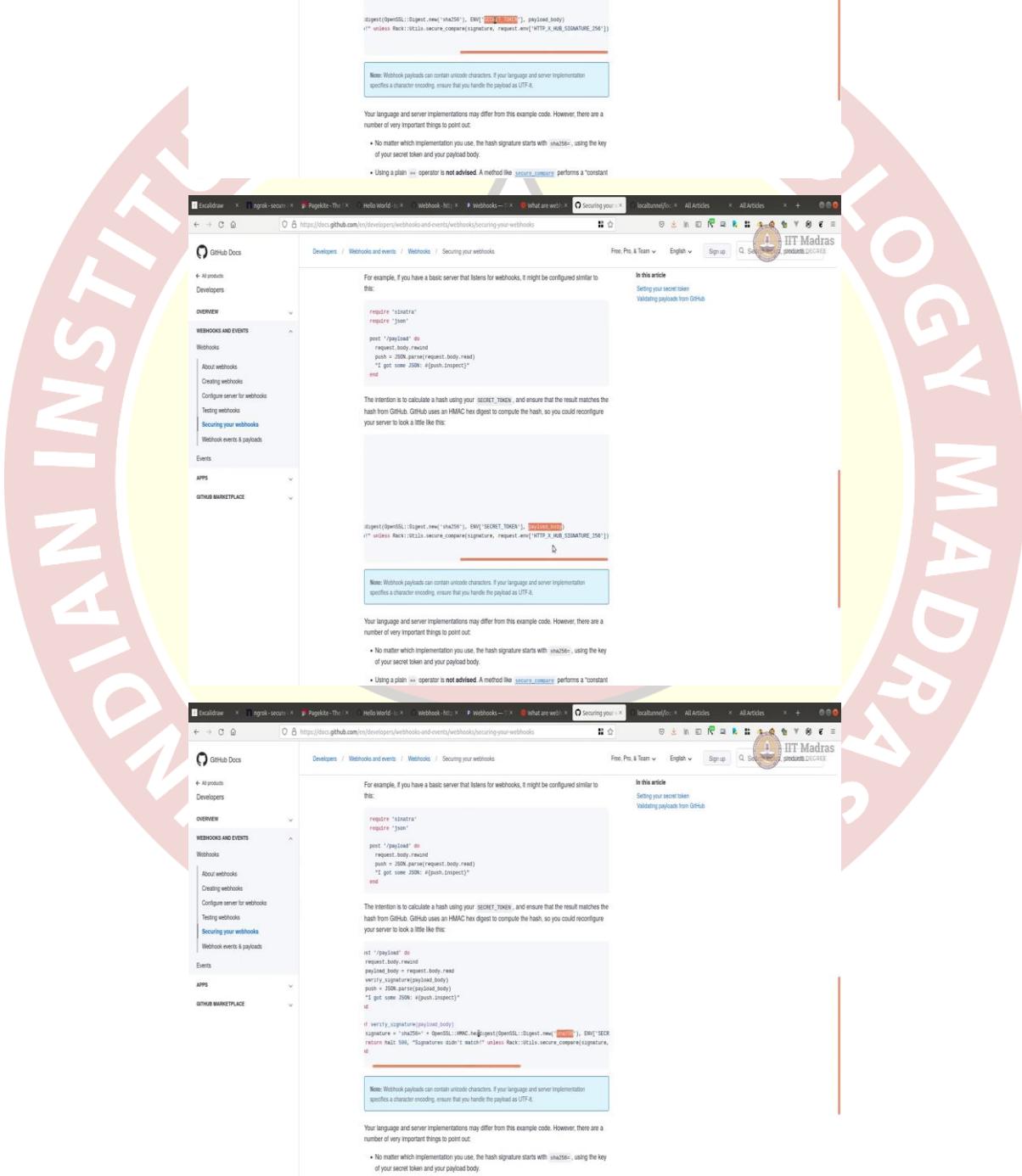
The intention is to calculate a hash using your `SECRET_TOKEN`, and ensure that the result matches the hash from GitHub. GitHub uses an HMAC hex digest to compute the hash, so you could reconfigure your server to look a little like this:

```

post '*' payload do
  request.body.read
  payload.body = request.body.read
  verify_signature(payload.body)
  push = JSON.parse(payload.body)
  "I got some JSON: #{push.inspect}"
end

def verify_signature(payload_body)
  signature = OpenSSL::Digest::SHA256.new().hexdigest(payload_body).lstrip
  return false, "$signature didn't match!" unless Rack::Utils.secure_compare(signature,
    payload_body)
end

```



```

require 'sinatra'
require 'json'

post '/payload' do
  request.body.read
  push = JSON.parse(request.body.read)
  "I got some JSON: #{push.inspect}"
end

The intention is to calculate a hash using your SECRET_TOKEN, and ensure that the result matches the
hash from GitHub. GitHub uses an HMAC hex digest to compute the hash, so you could reconfigure
your server to look a little like this:

get '/payload' do
  request.body.read
  payload_body = request.body.read
  verfify_signature(payload_body)
  push = JSON.parse(payload_body)
  "I got some JSON: #{push.inspect}"
end

if verify_signature(payload_body)
  signature = OpenSSL::HMAC.hexdigest(OpenSSL::Digest.new('sha256'), SECRET_TOKEN, payload_body)
  return halt 500, "Signatures didn't match" unless Rack::Utils.secure_compare(signature, verfify_signature(payload_body))
else
  halt 403, "Signature verification failed"
end

```

Note: Webhook payloads can contain unicode characters. If your language and server implementation specifies a character encoding, ensure that you handle the payload as UTF-8.

In this article

- Setting your secret token
- Validating your payloads from GitHub

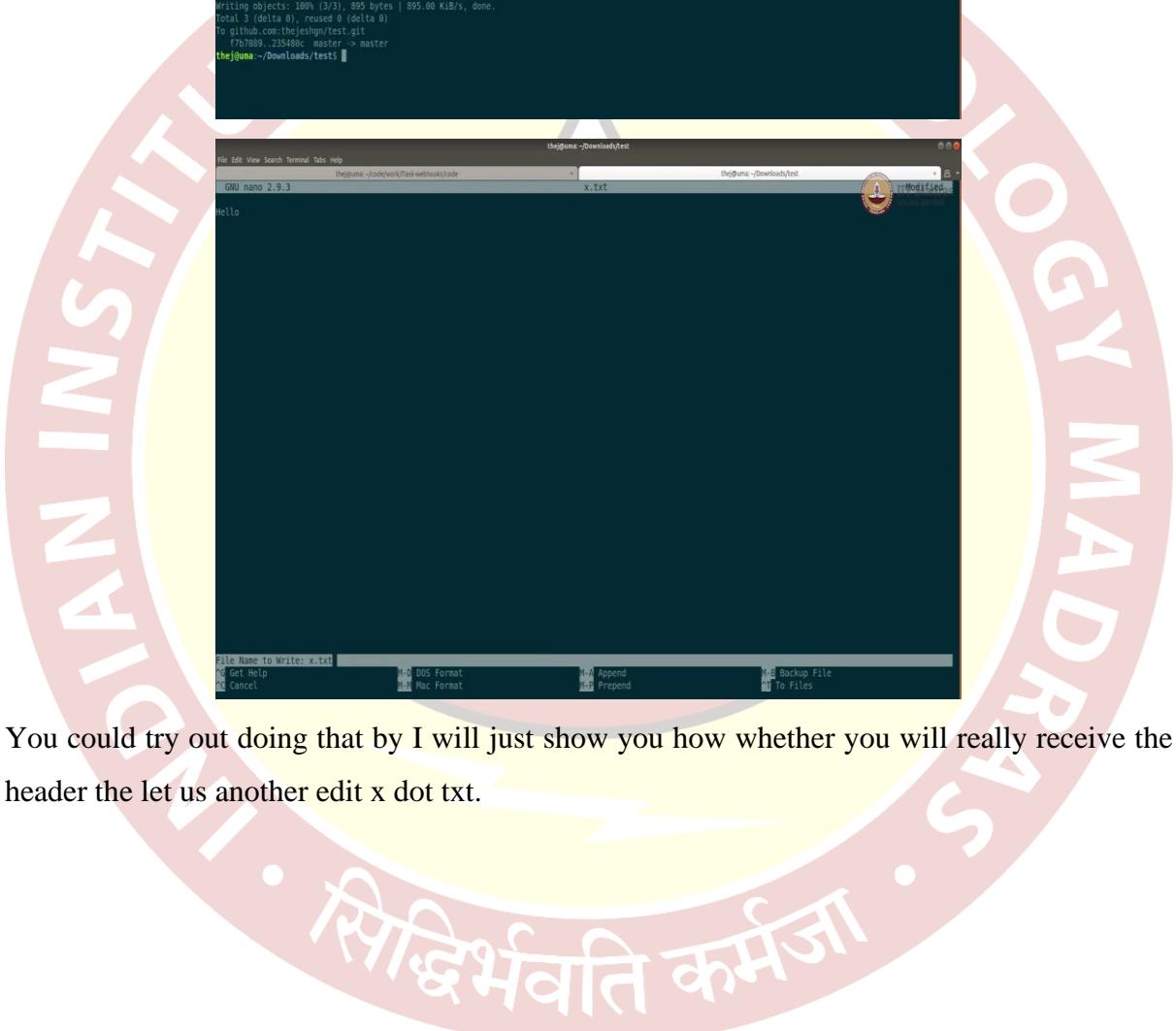
I think they are given Ruby code, but you could, you could actually use any other language as well. What they are doing is here taking the secret token and payload body and then creating a sha 256 of it and making a HMAC and then that is what they are comparing.

(Refer Slide Time: 20:15)

```

thej@uma:~/code/work/flask-webhooks$ code
thej@uma:~/code/work/flask-webhooks$ cd flask-webhooks
thej@uma:~/code/work/flask-webhooks$ unicorn -c config/unicorn.rb -D
[2022-02-24 08:47:17 +0530] [3936] [INFO] Starting unicorn 20.1.0
[2022-02-24 08:47:17 +0530] [3936] [INFO] Listening at: http://127.0.0.1:8000 (3936)
[2022-02-24 08:47:17 +0530] [3936] [INFO] Using worker: gevent
[2022-02-24 08:47:17 +0530] [3939] [INFO] Booting worker with pid: 3939
[2022-02-24 08:47:17 +0530] [3944] [INFO] Booting worker with pid: 3944
[2022-02-24 08:47:17 +0530] [3945] [INFO] Booting worker with pid: 3945
[2022-02-24 08:47:17 +0530] [3946] [INFO] Booting worker with pid: 3946
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <Flask 'main'>
in controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <Flask 'main'>
Create app complete
in controller app <Flask 'main'>
in controller app <Flask 'main'>
in controller app <Flask 'main'>

```



the@juna:~/Downloads/test\$ the@juna:~/code\$ cd /Downloads/

```

dash: .bourne-apparish: No such file or directory
dash: bashpromptrc: No such file or directory
the@juna:~/Downloads/test$ cd test
the@juna:~/Downloads/test$ git clone git@github.com:thejeshgn/test.git
Cloning into 'test'
warning: You appear to have cloned an empty repository.
the@juna:~/Downloads/test$ cd test
the@juna:~/Downloads/test$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
the@juna:~/Downloads/test$ touch x.txt
the@juna:~/Downloads/test$ git add x.txt
the@juna:~/Downloads/test$ git commit -m "added"
[master (root-commit) f7b7089] added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 x.txt
the@juna:~/Downloads/test$ git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 856 bytes | 856.00 Kib/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:thejeshgn/test.git
 * [new branch] master -> master
the@juna:~/Downloads/test$ rm x.txt
the@juna:~/Downloads/test$ git add x.txt
the@juna:~/Downloads/test$ git commit -m "part 2"
[master 235480c] part 2
1 file changed, 1 insertion(+)
the@juna:~/Downloads/test$ git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 855 bytes | 895.00 Kib/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:thejeshgn/test.git
 f7b7089..235480c master -> master
the@juna:~/Downloads/test$ 
```

You could try out doing that by I will just show you how whether you will really receive the header the let us another edit x dot txt.

(Refer Slide Time: 20:53)

```
thejh@thejh-Vostro-3460:~/code/work/flask-webhooks$ code
Enabling virtual env
[2022-02-24 00:47:17 +0530] [3936] [INFO] Starting gunicorn 20.1.0
[2022-02-24 00:47:17 +0530] [3936] [INFO] Listening at: http://127.0.0.1:8000 (3936)
[2022-02-24 00:47:17 +0530] [3936] [INFO] Using worker: sync
[2022-02-24 00:47:17 +0530] [3939] [INFO] Booting worker with pid: 3939
[2022-02-24 00:47:17 +0530] [3944] [INFO] Booting worker with pid: 3944
[2022-02-24 00:47:17 +0530] [3945] [INFO] Booting worker with pid: 3945
[2022-02-24 00:47:17 +0530] [3946] [INFO] Booting worker with pid: 3946
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <Flask 'main'>
in controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <Flask 'main'>
in controller app <Flask 'main'>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
in controller app <Flask 'main'>
Create app complete
in controller app <Flask 'main'>
in controller app <Flask 'main'>
in controller app <Flask 'main'>
```

All Articles

Hello World

Written by [thejh@902](#)

Welcome to Flask's documentation. Get started with installation and then get an overview with the Quickstart. There is also a more detailed tutorial that shows how to create a small but complete application with Flask. Common patterns are described in the Patterns for Flask section. The rest of the docs describe each component of Flask in detail, with a full reference in the API section.

Flask-SQLAlchemy

Written by [thejh@902](#)

Flask SQLAlchemy is an extension for Flask that adds support for SQLAlchemy to your application. It aims to simplify using SQLAlchemy with Flask by providing useful defaults and extra helpers that make it easier to accomplish common tasks. See the SQLAlchemy documentation to learn how to work with the DBM in depth. The following documentation is a brief overview of the most common tasks, as well as the features specific to Flask-SQLAlchemy.

my new article

Written by [thejh@902](#)

my new article content

dummy new article

Written by [thejh@902](#)

my dummy new article content

Using relationship

Written by [thejh@902](#)

Use relationships to insert. It's easy

2nd Using relationship

Written by [thejh@902](#)

2nd Use relationships to insert. It's easy

https://github.com/thejh/test/settings/hooks/344833033

general

Access

Collaborators

Moderation options

code and automation

Branches

Actions

Webhooks

Environments

Pages

Security

Code security and analysis

Deploy keys

Secrets

Integrations

Integrated apps

Email notifications

We'll send a POST request to the URL below with details of any subscription. You can also specify which data format you'd like to receive (JSON, XML, YAML, URL-encoded, etc). More information can be found in our developer documentation.

Payload URL:

https://thejh-test.ngrok.io/webhook/repo_update

Content type:

application/json

Secret:

If you've lost or forgotten this secret, you can change it, but be aware that any integrations using this secret will need to be updated. — Change Secret

Which event would you like to trigger this webhook?

Just the push

Send me everything

Let me select individual events

Active

We will deliver events when this hook is triggered.

Upload webhook Delete webhook

Recent Deliveries

event

Request Response Headers

Completed in 0.77 seconds

The screenshot shows the GitHub Webhooks settings page for a repository named "ngrok-secru". The "Payload URL" field contains "http://127.0.0.1:4041/webhook_receiver/github". The "Content type" is set to "application/json". A yellow box highlights the "Secret" field, which is empty. Below it, the "Which event would you like to trigger this webhook?" section has "Just the push" selected. The "Active" checkbox is checked, and the "Update webhook" button is visible.

Then proceed that because my Ngrok has been done. Let me start the Ngrok let us check whether the Ngrok is working it seems to be working let us change your check whether it is the same Ngrok because Ngrok changes it every time you started again. Update webhook.

(Refer Slide Time: 21:38)

The screenshot shows the GitHub Webhooks settings page for a repository named "ngrok-secru". The "Payload URL" field contains "http://127.0.0.1:4041/webhook_receiver/github". The "Content type" is set to "application/json". A yellow box highlights the "Secret" field, which is empty. Below it, the "Which events would you like to trigger this webhook?" section has "Just the push event" selected. The "Active" checkbox is checked, and the "Update webhook" button is visible.

INDIAN INSTITUTE

LOGY MADRAS

The screenshot shows a browser window with several tabs open, including 'Ingrid - Secure', 'Hello World', 'Webhook - Webhooks', and 'What are we?'. A modal dialog box is centered, asking 'Would you like to deliver this webhook?' with options 'Yes' and 'No'. Below the modal, the 'Recent Deliveries' section is visible, showing a single delivery from 'thejeshgn/test' to 'http://127.0.0.1:5000/webhook/receiver/github'. The 'Payload' section displays the JSON payload of the webhook event.

```
{
  "ref": "refs/heads/master",
  "before": "77d7089e79499bdcfc2f1e108847b7baa0c1ef",
  "after": "235480c701389d7246336654c251bd0bedc35b55",
  "repository": {
    "id": 46278006,
    "node_id": "MDEwNTMlCzMTB",
    "name": "thejeshgn/test",
    "private": false,
    "owner": {
      "login": "thejeshgn",
      "id": 7614,
      "node_id": "R_kgDSGcI2MTE",
      "avatar_url": "https://avatars.githubusercontent.com/u/7614/?q=4",
      "gravatar_id": "",
      "url": "https://api.github.com/users/thejeshgn",
      "html_url": "https://github.com/thejeshgn",
      "followers_url": "https://api.github.com/users/thejeshgn/followers",
      "following_url": "https://api.github.com/users/thejeshgn/following/other_user",
      "gists_url": "https://api.github.com/users/thejeshgn/gists",
      "starred_url": "https://api.github.com/users/thejeshgn/starred",
      "subscriptions_url": "https://api.github.com/users/thejeshgn/subscriptions",
      "organizations_url": "https://api.github.com/users/thejeshgn/orgs",
      "received_events_url": "https://api.github.com/users/thejeshgn/received_events",
      "events_url": "https://api.github.com/users/thejeshgn/events",
      "received_events_url": "https://api.github.com/users/thejeshgn/received_events",
      "events_url": "https://api.github.com/users/thejeshgn/events"
    },
    "fork": false,
    "url": "https://github.com/thejeshgn/test",
    "html_url": "https://github.com/thejeshgn/test",
    "clone_url": "https://github.com/thejeshgn/test.git",
    "ssh_url": "git@github.com:thejeshgn/test.git",
    "git_url": "https://github.com/thejeshgn/test.git",
    "git_tag_url": "https://github.com/thejeshgn/test/git/tags/{sha}",
    "git_refs_url": "https://github.com/thejeshgn/test/git/refs/{sha}",
    "trees_url": "https://github.com/thejeshgn/test/trees/{sha}",
    "statuses_url": "https://github.com/thejeshgn/test/statuses/{sha}",
    "languages_url": "https://github.com/thejeshgn/test/languages",
    "stargazers_url": "https://github.com/thejeshgn/test/stargazers",
    "contributors_url": "https://github.com/thejeshgn/test/contributors",
    "subscribers_url": "https://github.com/thejeshgn/test/subscribers",
    "subscription_url": "https://github.com/thejeshgn/test/subscription",
    "commits_url": "https://github.com/thejeshgn/test/commits/{sha}",
    "git_commits_url": "https://github.com/thejeshgn/test/git/commits/{sha}",
    "comments_url": "https://github.com/thejeshgn/test/comments/{number}",
    "issue_comment_url": "https://github.com/thejeshgn/test/issue_comments/{number}",
    "events_url": "https://github.com/thejeshgn/test/events",
    "hooks_url": "https://github.com/thejeshgn/test/hooks",
    "teams_url": "https://github.com/thejeshgn/test/teams",
    "branches_url": "https://github.com/thejeshgn/test/branches",
    "tags_url": "https://github.com/thejeshgn/test/tags/{sha}"
  }
}
```

Below the payload, two terminal windows show the command-line output of running a Python script to handle the GitHub webhook. The first terminal shows the code and the second shows the execution results.

```
thejeshgn:~/code/work/flash-webhooks$ python app.py
[2022-02-24 09:51:44 +0530] [3944] [ERROR] Error handling request
Traceback (most recent call last):
  File "/home/thejeshgn/.code/work/flash-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base_async.py", line 113, in handle_request
    resp.write_file(respiter)
  File "/home/thejeshgn/.code/work/flash-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/http/wsgi.py", line 385, in write_file
    if not self.sendFile(respiter):
  File "/home/thejeshgn/.code/work/flash-webhooks/code/.env/lib/python3.7/site-packages/gevent/_socket3.py", line 375, in sendfile
    self.sock.sendfile(respiter.fileno, count=countbytes)
  File "/home/thejeshgn/.code/work/flash-webhooks/code/.env/lib/python3.7/site-packages/gevent/_socket3.py", line 490, in sendfile
    return self._sendfile_use_sendfile(offset, count)
  File "/home/thejeshgn/.code/work/flash-webhooks/code/.env/lib/python3.7/site-packages/gevent/_socket3.py", line 420, in _sendfile_use_sendfile
    self._check_sendfile_params(file, offset, count)
  File "/home/thejeshgn/.code/work/flash-webhooks/code/.env/lib/python3.7/site-packages/gevent/_socket3.py", line 466, in _check_sendfile_params
    count = self._check_sendfile_params(file, offset, count)
ValueError: count must be a positive integer (got 0)
[2022-02-24 09:51:44 +0530] [3944] [INFO] 127.0.0.1 - "GET / HTTP/1.1" 200 144
thejeshgn:~/code/work/flash-webhooks$
```

```
thejeshgn:~/code/work/flash-webhooks$ python app.py
[2022-02-24 09:51:44 +0530] [3944] [INFO] 127.0.0.1 - "GET / HTTP/1.1" 200 144
thejeshgn:~/code/work/flash-webhooks$
```

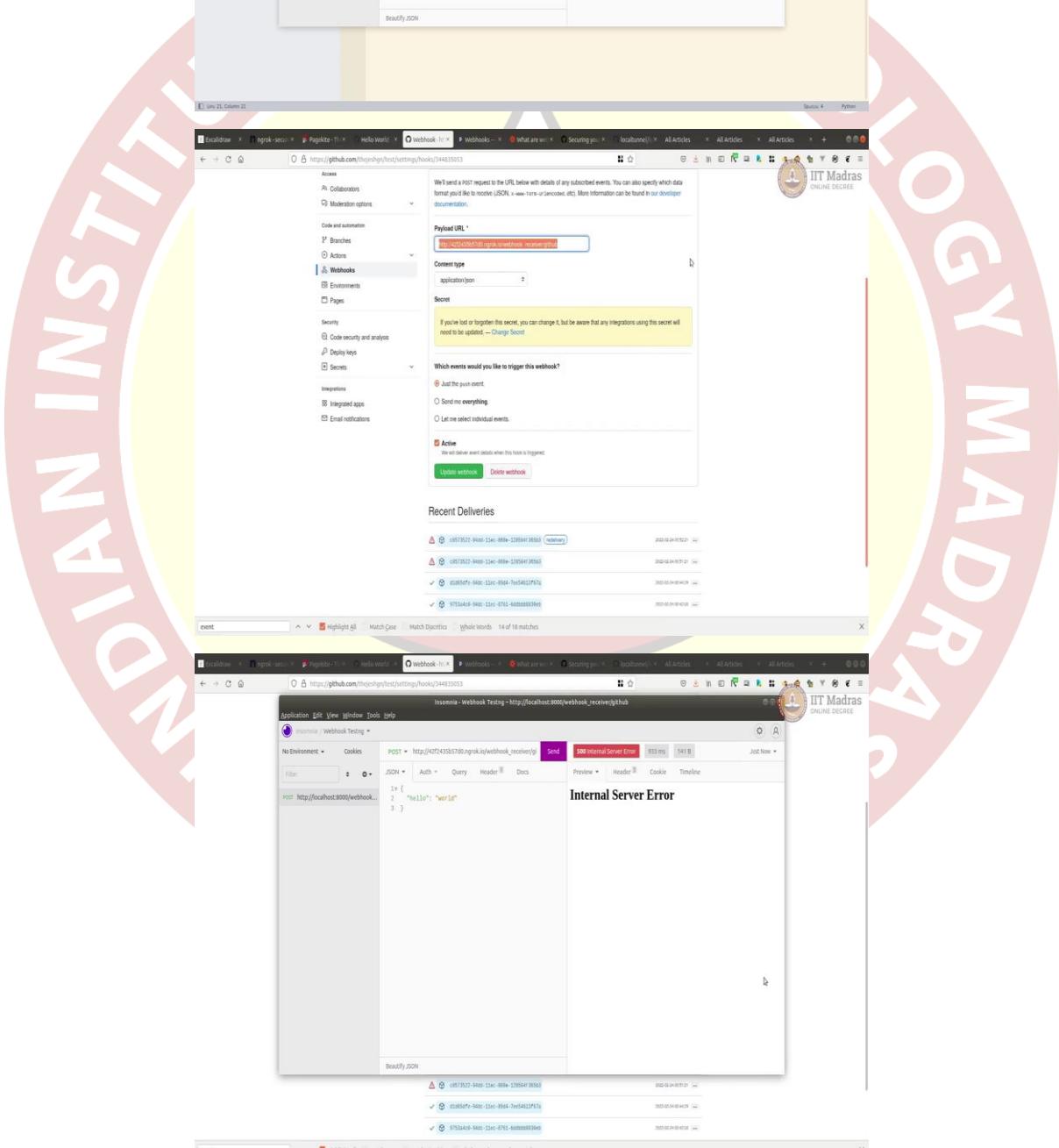
```
thejh@thejh-mbp:~/code/work/flask-webhooks$ code
thejh@thejh-mbp:~/code/work/flask-webhooks$ cd webhooks
thejh@thejh-mbp:~/code/work/flask-webhooks$ git pull
remote: Counting objects: 100% (15/15), done.
remote: Total 15 (delta 0), reused 0 (delta 0)
From https://github.com/thejehgn/test
 * [new branch] test --> origin/test
Branch 'test' set up to track remote branch 'test' from 'origin'.
thejh@thejh-mbp:~/code/work/flask-webhooks$ curl -X POST -H "Content-Type: application/json" -d '{"name": "Thejesh GN", "email": "l@thejehgn.com", "username": "thejehgn", "added": {}, "removed": {}, "modified": ["x.txt"]}' https://api.github.com/repos/thejehgn/test/hooks
{
  "id": 235480c7f01389d74633654c251bd8e6c35055,
  "name": "Thejesh GN",
  "url": "https://api.github.com/repos/thejehgn/test/hooks/235480c7f01389d74633654c251bd8e6c35055",
  "events": [
    "push"
  ],
  "active": true,
  "secret": "235480c7f01389d74633654c251bd8e6c35055",
  "created": "2022-02-24T09:51:14+05:30",
  "updated": "2022-02-24T09:51:14+05:30",
  "deleted": false,
  "forced": false,
  "base_ref": null,
  "compare": "https://github.com/thejehgn/test/commit/235480c7f01389d74633654c251bd8e6c35055...235480c7f01389d74633654c251bd8e6c35055",
  "distinct": true,
  "message": "part 2",
  "timestamp": "2022-02-24T09:51:14+05:30",
  "url": "https://github.com/thejehgn/test/commit/235480c7f01389d74633654c251bd8e6c35055"
}
[2022-02-24 09:52:21 +0530] [2946] [ERROR] Error handling request /webhook/receiver/github
Traceback (most recent call last):
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 85, in handle_request
    self.handle_request(listener_name, req, client, addr)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/gevent.py", line 127, in handle_request
    super().handle_request(listener_name, req, sock, addr)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 108, in handle_request
    responder = self.wsgi(environ, resp.start_response)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/app.py", line 2088, in __call__
    return self.wsgi(environ, start_response)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/app.py", line 2073, in wsgi_app
    response = self.handle_exception(e)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/_restful/_init_.py", line 271, in error_router
    return original_handler(e)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/app.py", line 2870, in wsgi_app
    response = self.full_dispatch_request()
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/app.py", line 1515, in full_dispatch_request
    rv = self.handle_user_exception(e)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/_restful/_init_.py", line 271, in error_router
    return original_handler(e)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/app.py", line 1513, in full_dispatch_request
    rv = self.dispatch_request(e)
File "/home/thejehgn/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/flask/app.py", line 1499, in dispatch_request
    endpoint = self.find_endpoint(self.view_functions.rule.endpoint)(**req.view_args)
File "/home/thejehgn/code/work/flask-webhooks/code/application/webhooks.py", line 20, in webhook_github
    print(request.header)
AttributeError: 'Request' object has no attribute 'header'
```

Let us see here. I think it has got I can just resend again to see Redeliver, yes Redeliver. You can see that you have got. I think there are some error and how it is printing, but that is fine. You can see the JSON has been printed here. I will be sent it and then it is not printing the header. Request object has no attribute called header.

(Refer Slide Time: 22:16)

```
-code/work/flask-webhooks/code/application/webhooks.py (Flask-webhooks) - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
• iit-flask-webhooks
  • iit_code
  • iit_env
  • iit_gunicorn_
  • iit_application
    • iit_apache_
      - .htaccess
      - config.py
      - controllers.py
      - database.py
      - models.py
      - salts.py
      - setup_and_run.py
    - webhooks
      - webhook_github.py
      - webhook_main.py
  • iit_db_directory
  • iit_static
  • iit_templates
    - controller_schedule
    - default.html
    - location.html
    - login.html
    - logout.html
    - user.html
  - lock_setup.sh
  - lock_workers.sh
  - main.py
  - readme.md
  - requirements.txt
  - readme.md

webhooks.py
1  from flask import Flask, request
2  from flask import render_template
3  from main import app
4  from application.models import Article
5  from flask_security import login_required, roles_accepted, roles_required
6  from application import tasks
7  from flask_sse import sse
8
9  print("in controller app", app)
10
11 @app.route("/webhook_receiver/github", methods=["POST"])
12 def webhook_github():
13     # get headers
14
15     # parse the content
16     content = request.json
17     print(content)
18     # Validate
19     print("-----")
20     print(request.headers)
21     # call async job
22
23
24
25
[Line 25, Column 21] Sun 27 Feb 2022 09:52:44 AM IST
```



It is it headers, it is headers let us try once more. That is not the URL. Again, I made the same mistake let us rework to this is something that you need to be sure. And you are developing locally that your URL is same.

(Refer Slide Time: 22:52)

```
the@una:~/code/work/flask-webhooks/code
```

```
the@una:~/code/work/flask-webhooks/code
```

```
[2022-02-24 00:53:24 +0530] [INFO] Worker exiting (pid: 3945)
[2022-02-24 00:53:24 +0530] [INFO] Worker exiting (pid: 3946)
[2022-02-24 00:53:24 +0530] [INFO] Worker exiting (pid: 3939)
[2022-02-24 00:53:24 +0530] [INFO] Worker exiting (pid: 3944)
[2022-02-24 00:53:24 +0530] [INFO] Worker exiting (pid: 3944)
Traceback (most recent call last):
  File "/home/thej/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/hub.py", line 893, in shutdown
    sleep()
  File "/home/thej/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/walter/get.py"
  File "/src/gevent/walter.py", line 143, in gevent_c_walter.Walter.get
  File "/src/gevent/walter.py", line 154, in gevent_c_walter.Walter.get
  File "/src/gevent/greenlet primitives.py", line 61, in gevent_c_greennet primitives.SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet primitives.py", line 61, in gevent_c_greennet primitives.SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet primitives.py", line 65, in gevent_c_greennet primitives.SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet primitives.py", line 35, in gevent_c_greennet primitives._greenlet_switch
  File "/src/gevent/greenlet.py", line 986, in gevent_c_greennet primitives.Greenlet.run
  File "/home/thej/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 198, in handle_quit
    sys.exit(0)
SystemExit: 0
2022-02-23T19:23:24Z <greenlet.greenlet object at 0x7fc7971d87d0 (otid=0x7fc798f585a0) current active started main> failed with SystemExit
Traceback (most recent call last):
  File "/home/thej/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/hub.py", line 893, in shutdown
    sleep()
  File "/home/thej/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gevent/walter/get.py"
  File "/src/gevent/walter.py", line 143, in gevent_c_walter.Walter.get
  File "/src/gevent/walter.py", line 154, in gevent_c_walter.Walter.get
  File "/src/gevent/greenlet primitives.py", line 61, in gevent_c_greennet primitives.SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet primitives.py", line 61, in gevent_c_greennet primitives.SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet primitives.py", line 65, in gevent_c_greennet primitives.SwitchOutGreenletWithLoop.switch
  File "/src/gevent/greenlet primitives.py", line 35, in gevent_c_greennet primitives._greenlet_switch
  File "/src/gevent/greenlet.py", line 986, in gevent_c_greennet primitives.Greenlet.run
  File "/home/thej/code/work/flask-webhooks/code/.env/lib/python3.7/site-packages/gunicorn/workers/base.py", line 198, in handle_quit
    sys.exit(0)
SystemExit: 0
2022-02-24 00:53:25 +0530] [INFO] Shutting down: Master
[2022-02-24 00:53:25 +0530] [INFO] Shutting down: Master
the@una:~/code/work/flask-webhooks/code$
```



```
the@una:~/code/work/flask-webhooks/code
```

```
the@una:~/code/work/flask-webhooks/code
```

```
DB Init complete
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <>
stage
Starting stage
pushed config
DB Init
DB Init complete
In controller app <>
Create app complete
In controller app <>
In controller app <>
In controller app <>
stage
Starting stage
pushed config
DB Init
DB Init complete
Create app complete
In controller app <>
Create app complete
In controller app <>
In controller app <>
In controller app <>
In controller app <>
('hello : "world"')
Host: 4272435957d0.n.grokko.io
User-Agent: insomnia
Content-Length: 21
Accept: */*
Content-Type: application/json
X-Forwarded-For: 49.205.139.122
X-Forwarded-Proto: http
Accept-Encoding: gzip
```

POST → http://4272435957d0.n.grokko.io/webhook_receiver/g/hub

200 OK

OK

JSON

Auth

Query

Header

Docs

Preview

Header

Cookie

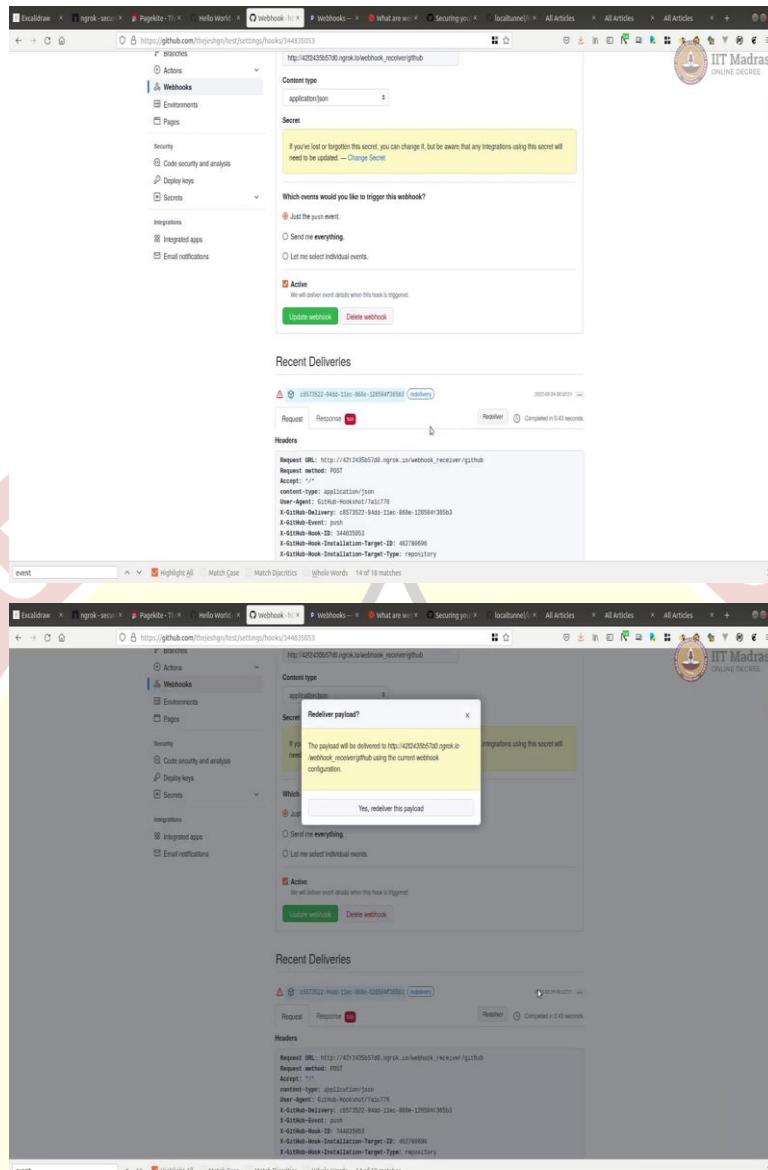
Timeline

Just Now

http://localhost:8000/webhook...

body

body



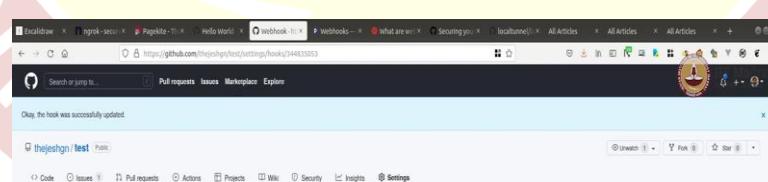
It is something has to be restarted. It is working now. So, now, either I can commit and push on small for testing purpose, I can just click on redeliver. Let it got redelivered.

(Refer Slide Time: 23:20)

INDIAN INSTITUTE

```
thejeshgn:~/code/work/Task-webhooks/code
curl -X POST https://github.com/thejeshgn/test/hooks/344835053 -H 'Content-Type: application/json' -d '{ "name": "Thejesh GN", "email": "1@thejeshgn.com", "username": "thejeshgn", "added": [], "removed": [], "modified": ["x.txt"] }'
```

```
thejeshgn:~/code/work/Task-webhooks/code
curl -X POST https://github.com/thejeshgn/test/hooks/344835053 -H 'Content-Type: application/json' -d '{ "name": "Thejesh GN", "email": "1@thejeshgn.com", "username": "thejeshgn", "added": [], "removed": [], "modified": ["x.txt"] }'
```



General

Address

code and automation

Integrations

Actions

Webhooks

Environments

Pages

Security

Code security and analysis

Deploy keys

Secrets

Events

Just the push event.

Send me everything.

Let me select individual events.

We will deliver event details when this hook is triggered.

event Highlight Match Case Match Diacritics Whole Words 14 of 18 matches

BLOGY MADRAS

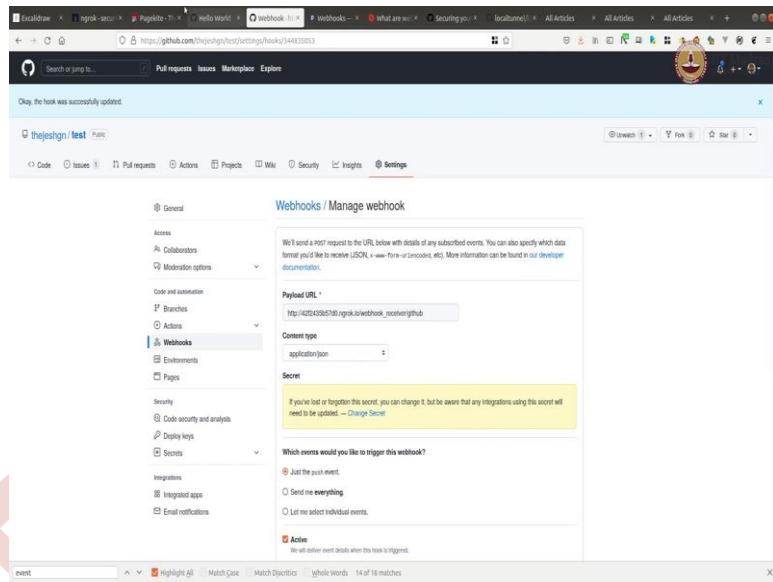
So, you can see that it has received headers 2. Now usually use this signature sha 1 or sha 256 to validate. For to validate you have to get the JSON body and create an HMAC to actually compare and validate. I will leave that as an exercise that is important to do.

(Refer Slide Time: 23:47)

The image shows a terminal window and a code editor side-by-side. The terminal window at the bottom displays a GitHub webhook payload. The code editor at the top shows a Python script named 'webhooks.py' which defines a webhook endpoint for GitHub.

```

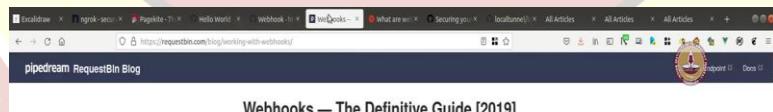
File Edit Selection Find View Cells Tools Project Preferences Help
File Edit Selection Find View Cells Tools Project Preferences Help
FOLDERS
  * all Task webhooks
    * all code
      * all
      * __init__.py
      * __main__.py
      * __pycache__
      * __pycache__.py
      * application
        * __init__.py
        * config.py
        * controllers.py
        * database.py
        * models.py
        * tasks.py
        * validation.py
      * webhooks.py
        * webhooks.py
      * db_directory
      * dbc
      * dbm
      * dbms
      * migrations
        * generated_schedule
        * deleted_log
        * local_test.sh
        * local_github.sh
        * local_run.sh
        * local_setup.sh
        * local_workers.sh
        * migrate
        * README.md
        * requirements.txt
        * readme.md
  13 characters selected
  File Edit View Search Terminal tab: code
  the@the: ~code/work/flask-webhooks/code
  https://github.com/thejeshgn/test/milestones/number?ref=api
  {
    "milestones": [
      {
        "url": "https://api.github.com/repos/thejeshgn/test/milestones/number",
        "notifications_url": "https://api.github.com/repos/thejeshgn/test/notifications_url",
        "labels_url": "https://api.github.com/repos/thejeshgn/test/labels{/name}",
        "releases_url": "https://api.github.com/repos/thejeshgn/test/releases{/id}",
        "self": true,
        "html_url": "https://github.com/thejeshgn/test",
        "id": 1645628152,
        "node_id": "MDM9cl05L2V0ZWQxNzIw",
        "created_at": "2022-07-23T14:35:52Z",
        "updated_at": "2022-07-23T14:35:52Z",
        "pushed_at": "2022-07-23T14:35:52Z",
        "git_url": "https://api.github.com/repos/thejeshgn/test/git",
        "git_tag_url": "https://api.github.com/repos/thejeshgn/test/git_tags{/sha}",
        "archive_url": "https://api.github.com/repos/thejeshgn/test/archives{/sha}",
        "tarball_url": "https://api.github.com/repos/thejeshgn/test/tarball{/sha}",
        "homepage": null,
        "size": 0,
        "stargazers_count": 0,
        "watchers_count": 0,
        "language": null,
        "has_issues": true,
        "has_projects": true,
        "has_downloads": true,
        "has_wiki": true,
        "has_pages": false,
        "forks_count": 0,
        "mirror_url": null,
        "archived": false,
        "disabled": false,
        "open_issues_count": 1,
        "license": null,
        "allow_forking": true,
        "is_template": false,
        "topics": [],
        "visibility": "public",
        "forks": 0,
        "open_issues": 1,
        "watchers": 0,
        "default_branch": "master",
        "stargazers": 0,
        "master_branch": "master",
        "pusher": {
          "name": "thejeshgn"
        },
        "email": "igthejeshgn.com",
        "sender": {
          "login": "thejeshgn",
          "id": 76114,
          "node_id": "MDQ6VXNlcjczMTk",
          "avatar_url": "https://avatars.githubusercontent.com/u/76114?v=4",
          "gravatar_id": "",
          "url": "https://api.github.com/users/thejeshgn",
          "html_url": "https://github.com/thejeshgn",
          "followers_url": "https://api.github.com/users/thejeshgn/followers",
          "following_url": "https://api.github.com/users/thejeshgn/following{/other_user}",
          "gists_url": "https://api.github.com/users/thejeshgn/gists{/gist_id}",
          "starred_url": "https://api.github.com/users/thejeshgn/starred{/repo}",
          "subscriptions_url": "https://api.github.com/users/thejeshgn/subscriptions",
          "organizations_url": "https://api.github.com/users/thejeshgn/orgs",
          "events_url": "https://api.github.com/users/thejeshgn/events{/privacy}",
          "received_events_url": "https://api.github.com/users/thejeshgn/received_events"
        }
      }
    ],
    "repository": {
      "url": "https://api.github.com/repos/thejeshgn/test",
      "clone_url": "https://github.com/thejeshgn/test.git",
      "ssh_url": "git@github.com:thejeshgn/test.git",
      "git_url": "https://git<...>@github.com/thejeshgn/test.git",
      "tag": "v0.0.1"
    }
  }
  Host: 42f243505708.ngrok.io
  User-Agent: GitHub-Hookshot/Talc776
  Content-Length: 6916
  Accept: */*
  Content-Type: application/json
  X-Forwarded-For: 127.0.0.1, 115.115
  X-Forwarded-Port: 443
  X-GitHub-Del-Paste: 0522-94d-llec-868e-12859af3d5fb
  X-GitHub-Eve: Red-Only
  X-GitHub-Hoo: Preferences: 5653
  X-GitHub-Hoo: NewWindow: Ion-Target-Id: 462788696
  X-GitHub-Hoo: NewTab: Ion-Target-Type: repository
  X-GitHub-Signal: Show-Member: 3f794ad88ccfd76bf4479e781ffabaa901f708c0fd2060615b90de2a90512cf94c449b4b
  X-GitHub-Signature: 0352: sha256-68afcd19fcc7570ffabaa901f708c0fd2060615b90de2a90512cf94c449b4b
  Accept-Encoding: gzip
  
```



So, I have written a thing here, validate you have to get from the headers, you have to get this like for example, GitHub key is equal to request dot headers. And then form using this body, form that to JSON and then generate the key and compare that key to this key and they should match. That is the logic.

And you can have you can use pre-built functions. Most providers please review. So, here I have shown one example with GitHub, similar examples of possible with many other providers and most of the integrations that you do with external partners on the internet will be through webhooks.

(Refer Slide Time: 24:46)



Webhooks — The Definitive Guide [2019]

April 18, 2019
Looking to learn more about webhooks? You've come to the right place. We'll cover everything you need to know about working with webhooks in 2019.

To get the most out of this guide, you'll need a basic understanding of HTTP. If you need a refresher, check out our [HTTP glossary](#), or see the references [here](#) and [here](#).

Have a specific question about webhooks? Skip to that section below, or [reach out!](#) if you can't find the answer you're looking for.

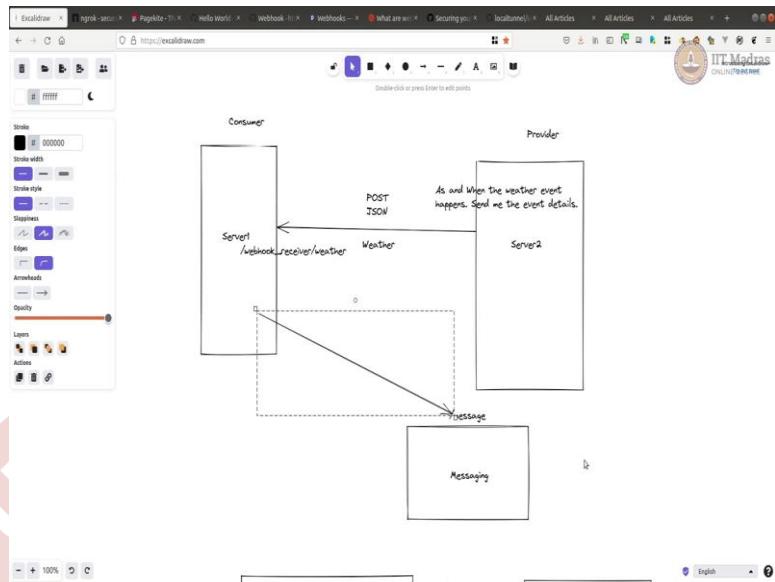
- What Is A Webhook?
- Webhooks vs APIs – What's The Difference?
 - Why Information Loss Can Be A Problem
 - What To Use Webhooks, APIs, Or Both
- How Do I Process Webhook Requests?
- How Do I Test & Debug A Webhook?
 - Other Useful Tools For Debugging Webhooks
- How Do I Secure A Webhook Integration?
 - Use HTTPS
 - Pass Secrets Or Tokens
 - Accept Only Signed Requests
 - Whitelist The Webhook Provider's IP Addresses
- Example: Send New GitHub Issues To Discord
 - Step 1 – Create A RequestBin URL For Debugging
 - Step 2 – Create A Test GitHub Repository
 - Step 3 – Enable Webhook Events For New Issues
 - Step 4 – Use RequestBin To Make Sure Everything Looks OK
 - Step 5 – Configure A Discord Channel And Incoming Webhook Integration
 - Step 6 – Configure GitHub To Send Issues To Discord
 - Step 7 – Improve This Integration
- Who Summits Webhooks?



Now this is a good tutorial at the request pin that you can go through which also explain what it means and what are the tools and what is the use of it. How do you set up, how do you test all of that.

And this is one of the most important features of the web, in the sense of, how people have invented new protocols on the web, to suit the requirements. And webhooks are used a lot to share the event across the system. And I hope you will get in, end up getting used. I hope you will end up using webhooks in your application to receive and process the data. There is another part of the web hooks, where actually you can send out the web hooks, you can be the provider.

(Refer Slide Time: 25:49)



Here, if you see this is the provider. And this is the consumer. Because we are assuming, but what if there is case there are events that you are generating and someone else is interested in it? For example, a messaging service. Let us say there is a messaging service. And that could be Google Talk webhooks. Let us say this is Google Room or Google Talk or some other messaging service, which also gives a webhook called message.

And then you could be pushing details to that based on the updates in your application. And that case, you become the provider and this becomes the consumer. So, you could try out that to trying to become provider. But in most cases you will be a consumer. And since you will be integrating with multiple other providers. I hope you will end webhooks, and it is fun. And it is the easiest way to integrate services on the web. Thank you so much for watching.