

# HW #2 - GitHub Service

Git Repo Link: [Git Repo](#)

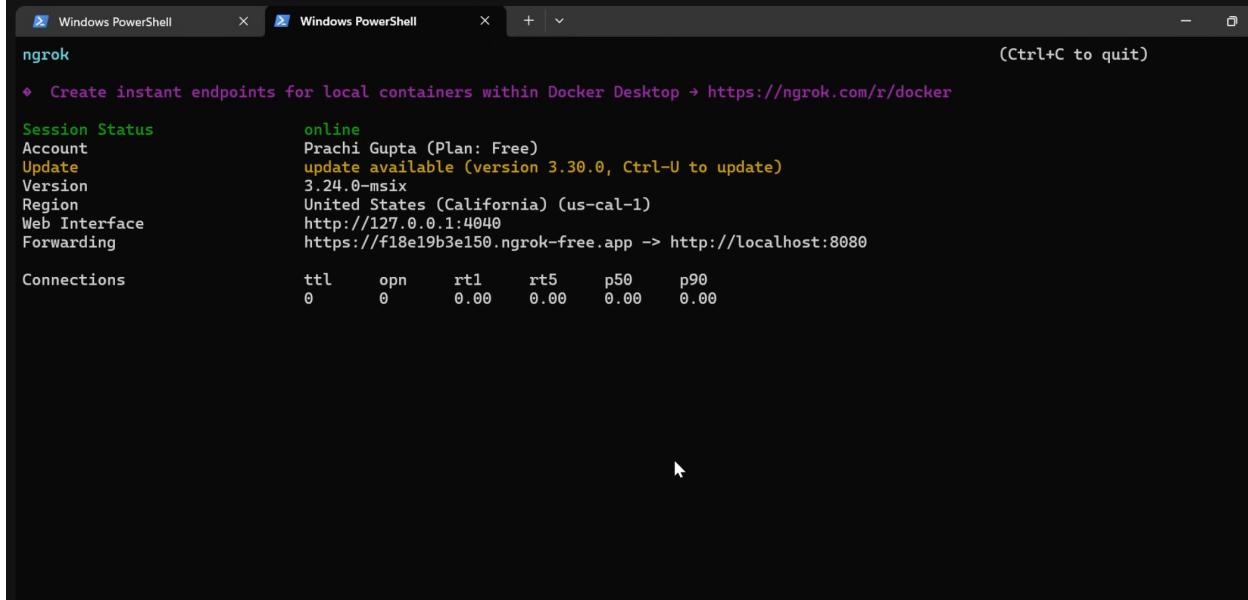
**Team : EnthuPrise**

Jaya Vyas – 019144463

Prachi Gupta – 019106594

Soham Raj Jain - 019139796

Tamizh Selvan Manivannan – 019148896

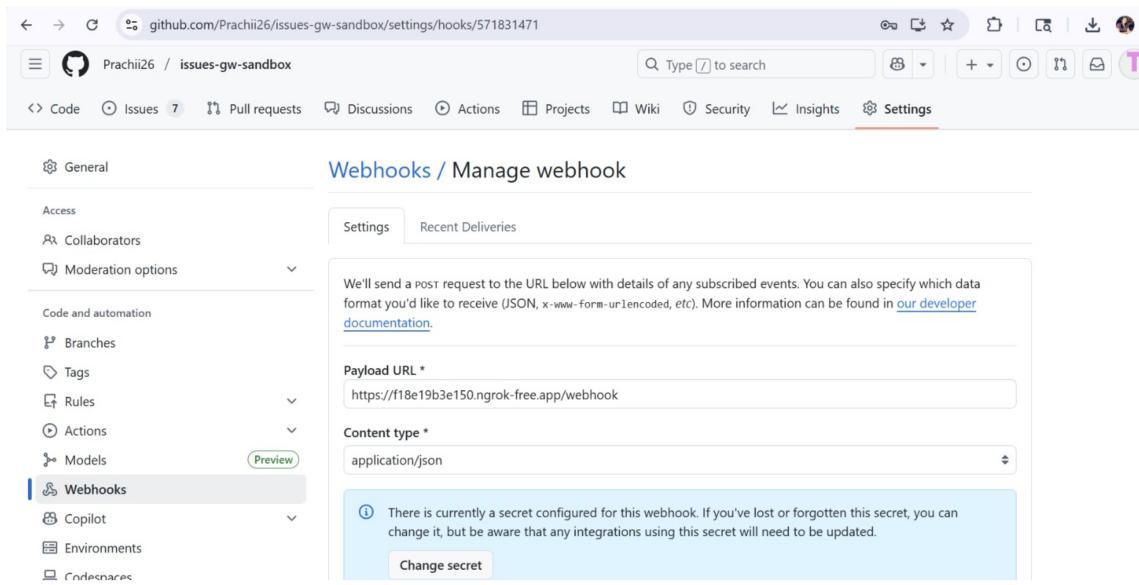


```
Windows PowerShell x Windows PowerShell x + - (Ctrl+C to quit)

ngrok
◆ Create instant endpoints for local containers within Docker Desktop → https://ngrok.com/r/docker

Session Status      online
Account             Prachi Gupta (Plan: Free)
Update              update available (version 3.30.0, Ctrl-U to update)
Version             3.24.0-msix
Region              United States (California) (us-cal-1)
Web Interface       http://127.0.0.1:4040
Forwarding          https://f18e19b3e150.ngrok-free.app → http://localhost:8080

Connections         ttl     opn      rsl      rt5      p50      p90
                    0       0       0.00    0.00    0.00    0.00
```



github.com/Prachii26/issues-gw-sandbox/settings/hooks/571831471

Prachii26 / issues-gw-sandbox

Type [ ] to search

Code Issues 7 Pull requests Discussions Actions Projects Wiki Security Insights Settings

General

Webhooks / Manage webhook

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Models

Webhooks

Copilot

Environments

Codepaces

Settings

Recent Deliveries

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL \*

https://f18e19b3e150.ngrok-free.app/webhook

Content type \*

application/json

There is currently a secret configured for this webhook. 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

# HW #2 - GitHub Service

The screenshot shows the GitHub settings page for a repository. The URL is [github.com/Prachii26/issues-gw-sandbox/settings/hooks/571831471](https://github.com/Prachii26/issues-gw-sandbox/settings/hooks/571831471). The page lists several types of events that can be hooked:

- Discussion**: Discussion created, edited, closed, reopened, pinned, unpinned, locked, unlocked, transferred, answered, unanswered, labeled, unlabeled, had its category changed, or was deleted.
- alerts**: Code scanning alert dismissal request was created, or received a response (approved or rejected).
- Dismissal requests for secret scanning alerts**: Secret scanning alert dismissal request was created, cancelled, or received a response.
- Forks**: Repository forked.
- Issue dependencies**: Issue dependencies - such as blocked by or blocking - added or removed. This event is not yet public. If you would like to receive payloads for this event, you will need to explicitly choose it here.
- Issue comments**: Issue comment created, edited, or deleted. (Selected)
- Issues**: Issue opened, edited, deleted, transferred, pinned, unpinned, closed, reopened, assigned, unassigned, labeled, unlabeled, milestone, demilestone, locked, unlocked, typed, or untyped.
- Labels**: Label created, edited or deleted.
- Merge groups**: Merge Group requested checks, or was destroyed.
- Meta**: This particular hook is deleted.

The screenshot shows a Windows PowerShell terminal window with two tabs open. The current tab displays the following command output:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\sarth> docker run --rm -p 8080:8080 --env-file .\env issues-gw:latest
docker: open .\env: The system cannot find the file specified.

Run 'docker run --help' for more information
PS C:\Users\sarth> cd C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw
PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw> docker run --rm -p 8080:8080 --env-file .\env issues-gw:latest
INFO: Started server process [1]
INFO: Waiting for application startup.
2025-09-26 23:15:44 [info    ] startup_complete
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8080 (Press CTRL+C to quit)
```

## HW #2 - GitHub Service

```
Windows PowerShell      Windows PowerShell      + 
docker: open .\env: The system cannot find the file specified.

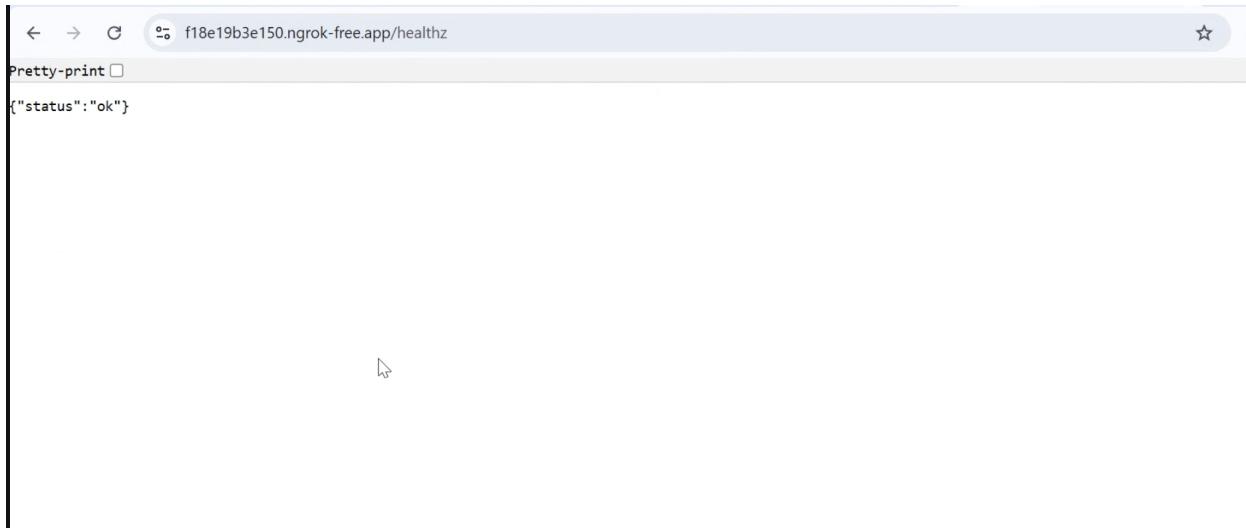
Run 'docker run --help' for more information
PS C:\Users\sarth> cd C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw
PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw> docker run --rm -p 8080:8080 --env-file .\env issues-gw:latest
INFO: Started server process [1]
INFO: Waiting for application startup.
2025-09-26 23:15:44 [info] startup_complete
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8080 (Press CTRL+C to quit)
INFO: 127.0.0.1:44738 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:45948 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:45948 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:55276 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:43578 - "GET /webhook/healthz HTTP/1.1" 404 Not Found
INFO: 172.17.0.1:43578 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:46142 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:33870 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:33870 - "GET /healthz HTTP/1.1" 200 OK
INFO: 127.0.0.1:56468 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:40440 - "GET /public/openapi.yaml HTTP/1.1" 200 OK
INFO: 127.0.0.1:47616 - "GET /healthz HTTP/1.1" 200 OK
INFO: 127.0.0.1:41572 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:56682 - "GET /issues?state=open&per_page=10 HTTP/1.1" 200 OK
INFO: 127.0.0.1:60756 - "GET /healthz HTTP/1.1" 200 OK
INFO: 127.0.0.1:43778 - "GET /healthz HTTP/1.1" 200 OK
INFO: 172.17.0.1:52764 - "POST /issues/3/comments HTTP/1.1" 201 Created
2025-09-26 23:20:06 [info] webhook_ack           action=created delivery_id=565f4de0-9b2f-11f0-9f1f-5a656b06c9cc gh_eve
nt=issue_comment issue_number=3
INFO: 172.17.0.1:52778 - "POST /webhook HTTP/1.1" 204 No Content
INFO: 127.0.0.1:39652 - "GET /healthz HTTP/1.1" 200 OK
```

### Health check



# HW #2 - GitHub Service

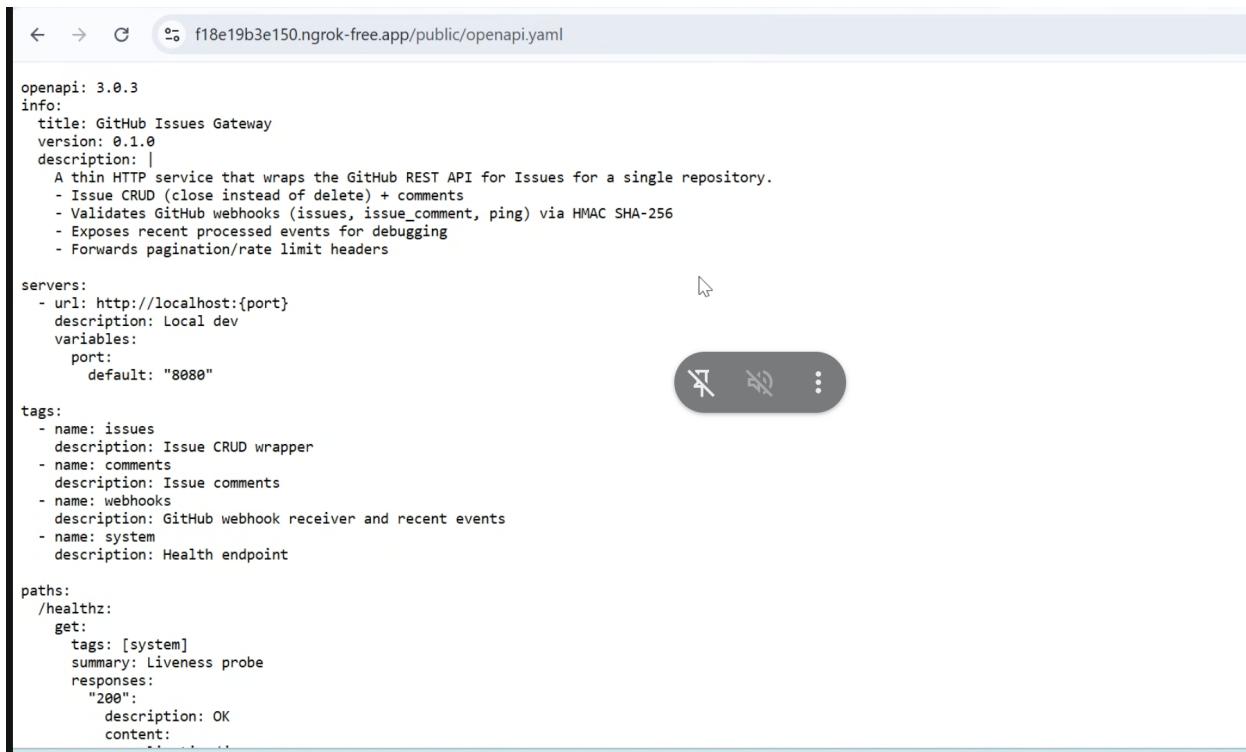
## Health check



A screenshot of a web browser window. The address bar shows the URL `f18e19b3e150.ngrok-free.app/healthz`. Below the address bar is a toolbar with icons for back, forward, and search. A dropdown menu labeled "Pretty-print" is open. The main content area displays a JSON response:

```
{"status": "ok"}
```

## Openapi.yml



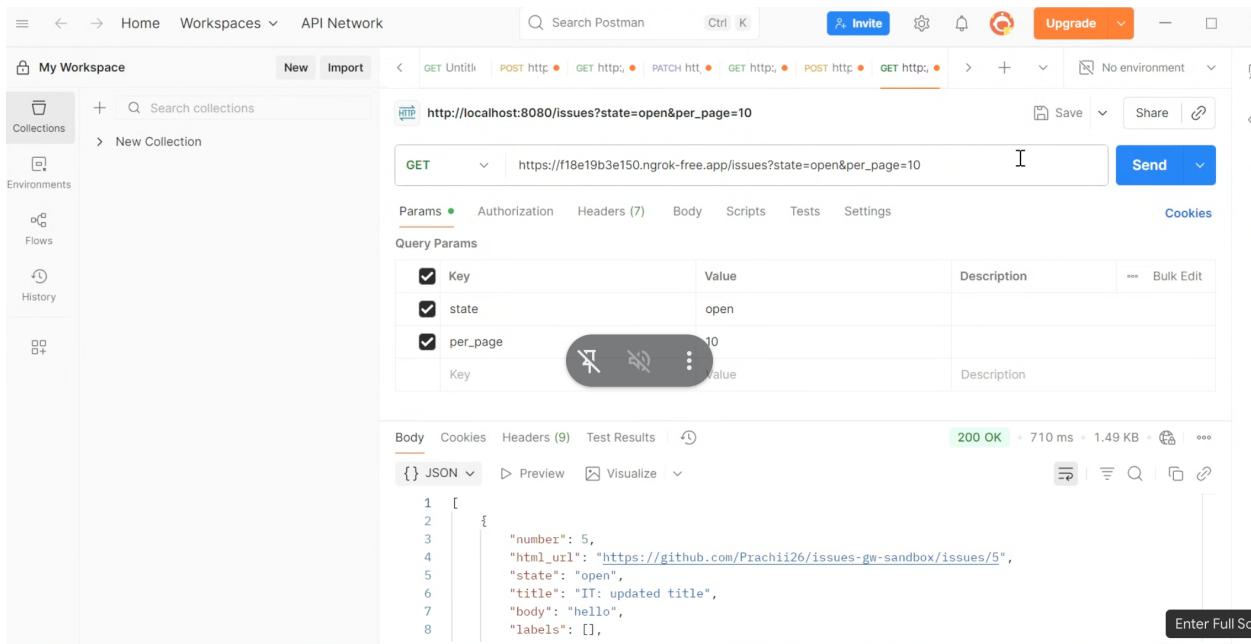
A screenshot of a web browser window. The address bar shows the URL `f18e19b3e150.ngrok-free.app/public/openapi.yaml`. Below the address bar is a toolbar with icons for back, forward, and search. A dropdown menu is open, showing options like "X" and "...". The main content area displays the contents of the `openapi.yaml` file:

```
openapi: 3.0.3
info:
  title: GitHub Issues Gateway
  version: 0.1.0
  description: |
    A thin HTTP service that wraps the GitHub REST API for Issues for a single repository.
    - Issue CRUD (close instead of delete) + comments
    - Validates GitHub webhooks (issues, issue_comment, ping) via HMAC SHA-256
    - Exposes recent processed events for debugging
    - Forwards pagination/rate limit headers
servers:
  - url: http://localhost:{port}
    description: Local dev
    variables:
      port:
        default: "8080"
tags:
  - name: issues
    description: Issue CRUD wrapper
  - name: comments
    description: Issue comments
  - name: webhooks
    description: GitHub webhook receiver and recent events
  - name: system
    description: Health endpoint
paths:
  /healthz:
    get:
      tags: [system]
      summary: Liveness probe
      responses:
        "200":
          description: OK
          content:
            application/json:
              schema:
                type: object
                properties:
                  status:
                    type: string
                    enum:
                      - ok

```

# HW #2 - GitHub Service

## Postman GET



My Workspace

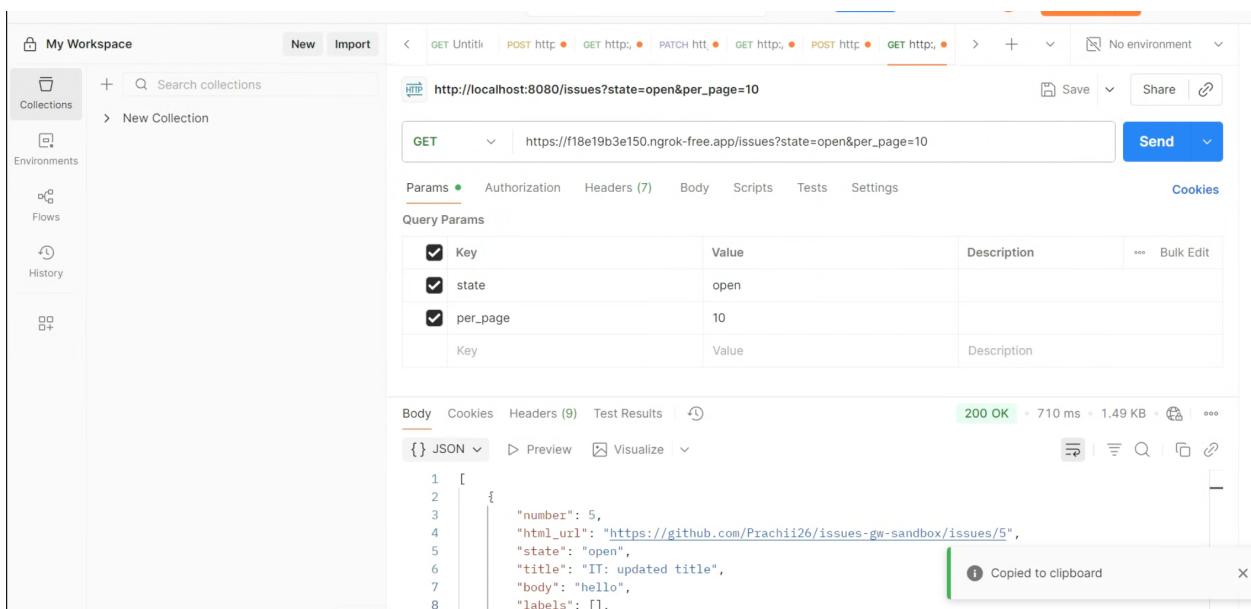
GET [https://f18e19b3e150.ngrok-free.app/issues?state=open&per\\_page=10](https://f18e19b3e150.ngrok-free.app/issues?state=open&per_page=10)

Params

Key	Value	Description	Bulk Edit
state	open		
per_page	10		

Body

```
[{"number": 5, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/5", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 6, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/6", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 7, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/7", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 8, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/8", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 9, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/9", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 10, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/10", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}]
```



My Workspace

GET [https://f18e19b3e150.ngrok-free.app/issues?state=open&per\\_page=10](https://f18e19b3e150.ngrok-free.app/issues?state=open&per_page=10)

Params

Key	Value	Description	Bulk Edit
state	open		
per_page	10		

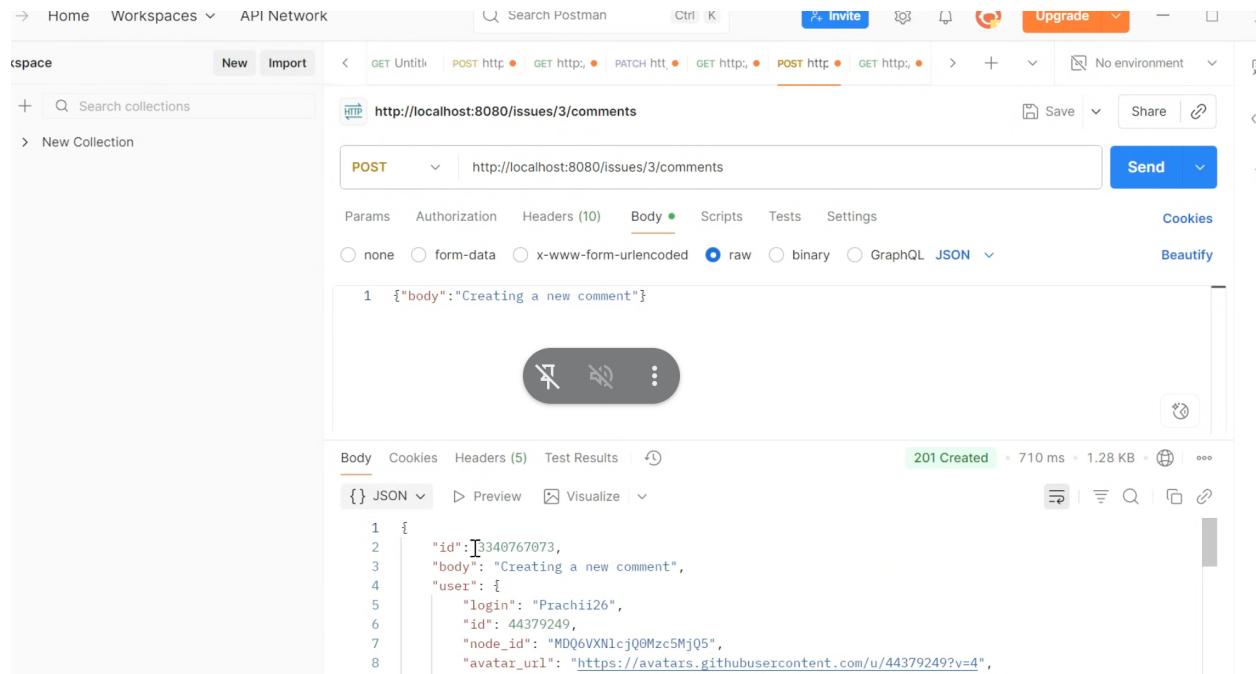
Body

```
[{"number": 5, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/5", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 6, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/6", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 7, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/7", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 8, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/8", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 9, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/9", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}, {"number": 10, "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/10", "state": "open", "title": "IT: updated title", "body": "hello", "labels": []}]
```

Copied to clipboard

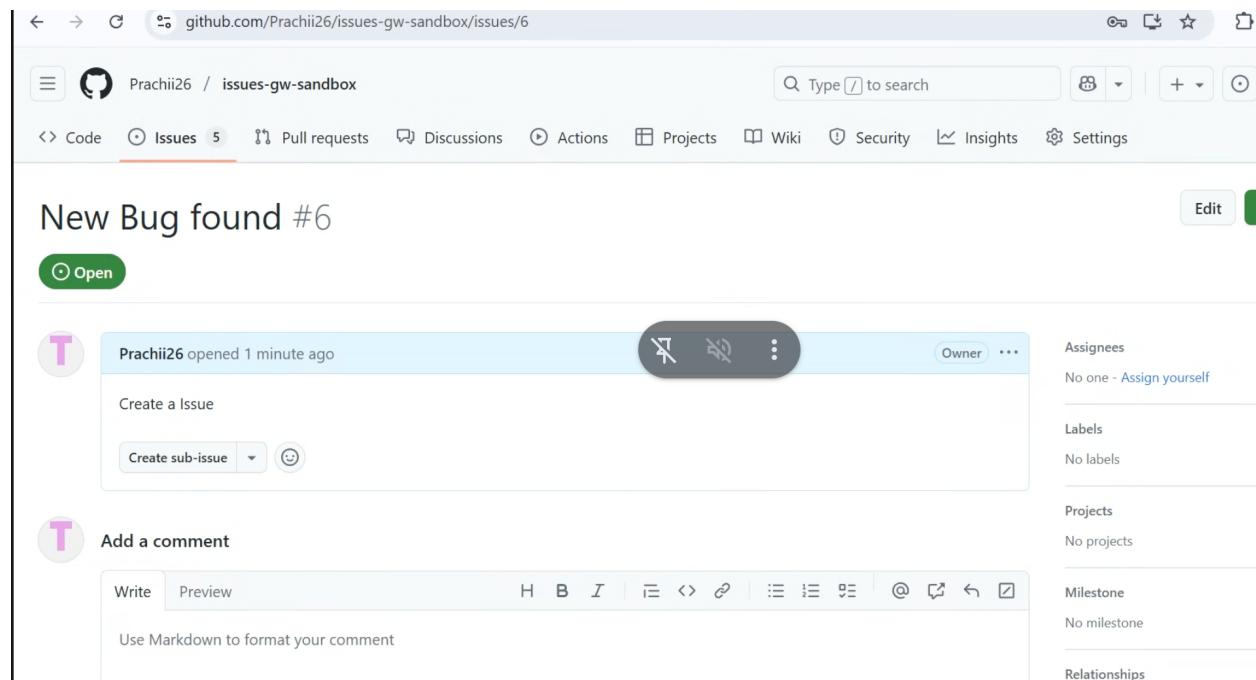
# HW #2 - GitHub Service

## POSTING new comment



The screenshot shows the Postman interface. A collection named 'cspace' is selected. A new POST request is being made to `http://localhost:8080/issues/3/comments`. The request body is set to raw JSON with the value `{"body": "Creating a new comment"}`. The response status is `201 Created`, and the response body is a JSON object containing:

```
1 {  
2   "id": 1340767073,  
3   "body": "Creating a new comment",  
4   "user": {  
5     "login": "Prachii26",  
6     "id": 44379249,  
7     "node_id": "MDQ6VXNlcjQ0Mzc5MjQ5",  
8     "avatar_url": "https://avatars.githubusercontent.com/u/44379249?v=4",  
9   }  
10 }
```



The screenshot shows a GitHub issue page for issue #6. The title is `New Bug found #6`. The 'Issues' tab is selected, showing 5 issues. The right sidebar contains the following fields:

- Assignees: No one - [Assign yourself](#)
- Labels: No labels
- Projects: No projects
- Milestone: No milestone
- Relationships: None

New comment POSTED

# HW #2 - GitHub Service

The screenshot shows a GitHub issue page for a demo issue. The first comment, posted 1 minute ago, says "Hello from my FastAPI gateway!". The second comment, posted now, says "Creating a new comment". On the right side of the page, there are sections for "No milestone", "Relationships", "Development", "Notifications", and "Participants". A "Code with age" button is also present.

## Update operation

The screenshot shows a Postman collection named "ns" with a "New Collection" option. A request is being built for "http://localhost:8080/issues/1" using the PATCH method. The "Body" tab is selected, showing a raw JSON payload: {"state": "closed"}. The response is a 200 OK status with 755 ms duration and 396 B size. The response body is identical to the request body: {"state": "closed"}.

# HW #2 - GitHub Service

The screenshot shows a GitHub issue page for issue #1. The title is "Testing Issue #1". A purple button indicates it is "Closed". The issue was opened by "Prachi26" last week with no description provided. It was closed by "Prachi26" 2 minutes ago. To the right, there are sections for Assignees (No one - Assign yourself), Labels (No labels), Projects (No projects), Milestone (No milestone), and Relationships.

## Create a new issue

A POST request is being made to <http://localhost:8080/issues>. The request body contains the JSON payload: {"title": "New Bug found", "body": "Create a Issue"}. The response status is 201 Created, with a response time of 960 ms and a response size of 432 B. The response body shows the created issue details:

```
1 {  
2   "number": 6,  
3   "html_url": "https://github.com/Prachi26/issues-gw-sandbox/issues/6",  
4   "state": "open",  
5   "title": "New Bug found",  
6   "body": "Create a Issue",  
7   "labels": [],  
8   "created_at": "2025-09-26T23:23:51Z",  
9 }
```

# HW #2 - GitHub Service

Get by passing issue no

The screenshot shows the Postman application interface. At the top, there's a navigation bar with 'space', 'New', 'Import', and other options. Below it is a search bar for collections. The main area shows a collection named 'New Collection'. A specific API endpoint is selected: `http://localhost:8080/issues?state=open&per_page=10`. The method is set to 'GET'. The 'Body' tab is active, showing a JSON response:

```
1 {  
2   "number": 1,  
3   "html_url": "https://github.com/Prachii26/issues-gw-sandbox/issues/1",  
4   "state": "closed",  
5   "title": "Testing Issue",  
6   "body": null,  
7   "labels": [],  
8   "created_at": "2025-09-18T07:03:15Z",  
}
```

The response status is 200 OK, with a duration of 576 ms and a size of 452 B. The response body is displayed in JSON format.

## Reusable schemas

- Defined once under components/schemas and reused across multiple endpoints:
- Issue : reused for GET, POST, PATCH issue responses.
- Comment : reused for comment creation.
- CreateIssue, UpdateIssue, CreateComment : reused for request bodies.
- Error : reused in all error responses.
- Event : reused for /events output.
- Label : reused inside Issue.labels.
- Also reusable parameters: IssueNumber, State, Labels, Page, PerPage.

# HW #2 - GitHub Service

## Unit Tests

```
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw> pytest --nologcapture
C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw\venv\Lib\site-packages\fastapi\applications.py:4495: DeprecationWarning: on_event is deprecated, use lifespan event handlers instead.

  Read more about it in the
  [FastAPI docs for Lifespan Events](https://fastapi.tiangolo.com/advanced/events/).

  return self.router.on_event(event_type)

tests/test_error_mapping.py::test_create_issue_unauthorized_maps_to_401
tests/test_error_mapping.py::test_get_issue_not_found_maps_to_404
tests/test_pagination.py::test_list_issue_forwards_pagination_headers
tests/test_validation.py::test_create_issue_requires_title
tests/test_webhook.py::test_webhook_valid_signature
tests/test_webhook.py::test_webhook_invalid_signature
C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw\venv\Lib\site-packages\httpx\_client.py:1437: DeprecationWarning: The 'app' shortcut is now deprecated. Use the explicit style 'transport=ASGITTransport(app=...)'
  warnings.warn(message, DeprecationWarning)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
2 passed, 8 warnings in 4.45s
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw>
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw>
```

## Integration Test

```
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw> pytest --nologcapture
C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw\venv\Lib\site-packages\fastapi\applications.py:4495: DeprecationWarning: on_event is deprecated, use lifespan event handlers instead.

  Read more about it in the
  [FastAPI docs for Lifespan Events](https://fastapi.tiangolo.com/advanced/events/).

  return self.router.on_event(event_type)

tests/test_integration.github.py::test_full_issue_lifecycle
tests/test_integration.github.py::test_webhook_local_signature
C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw\venv\Lib\site-packages\httpx\_client.py:1437: DeprecationWarning: The 'app' shortcut is now deprecated. Use the explicit style 'transport=ASGITTransport(app=...)'
  warnings.warn(message, DeprecationWarning)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
2 passed, 4 warnings in 6.17s
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw>
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw>
```

## Coverage

```
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw> coverage run -m pytest
-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
-- Warning: 1 warning in 8.78s
(venv) PS C:\Users\sarth\Desktop\Prachi\Code\github-issues-gw> coverage report -m
Name           Stmts   Miss  Cover   Missing
-----      -----
src__init__.py      0     0   100%
srcconfig.py       18     5    72%  25-32
srcgithub_client.py  18     6    67%  44-46, 54, 63, 83
srcstorage.py        3     0   100%
srcmodels.py        20     3    85%  11-12, 15
srcpagination.py    20     0   100%
srcroutes_init.py   20     0   100%
srcroutes_issues.py 63    23    63%  26-29, 48-51, 63-65, 77-82, 90, 93-98
srcroutes_webhook.py 36     6    83%  18, 41-42, 46, 61-62
srcstorage.py       20     8    60%  20-22, 44-46, 50-60
testscomfetest.py   22     1    95%  14
testserrormapping.py 24     0   100%
testsintegration.github.py 35     0   100%
testspagination.py   19     0   100%
testsvalidation.py   9     0   100%
testswebhook.py      16     0   100%
-----      -----
TOTAL            419    55    87%
```

# HW #2 - GitHub Service

## Exception Handling - Reliability

The screenshot shows the Postman application interface. In the top navigation bar, there are links for Home, Workspaces (with a dropdown menu), API Network, and a search bar labeled "Search Postman". On the far right, there are buttons for "Invite", "Upgrade", and a profile icon.

The main workspace is titled "My Workspace". It contains a collection named "New Collection". A recent item in the collection is a PATCH request to "http://localhost:8080/issues/10".

The request details are as follows:

- Method: PATCH
- URL: http://localhost:8080/issues/10
- Body type: raw (selected)
- Body content:

```
1  {"state": "closed"}
```

The response status is 404 Not Found, with a duration of 447 ms and a size of 308 B. The response body is displayed in JSON format:

```
1  {
2    "detail": {
3      "error": "NotFound",
4      "message": "Issue 10 not found",
5      "details": {
6        "github_status": 404,
7        "github_message": "Not Found"
8      }
9    }
10 }
```