

Rapor

Açık kaynak ödevi



Aisan Kheiri
170420995

Soru: Belirleyeceğiniz bir kaynağı REST API Server ile sunmak için OpenAPI dökümanı tasarlayıp, bu dökümana uygun olarak belirleyeceğiniz bir programlama dili ile implemente ediniz. Açık kaynak kodlu yazılımlar/projeler kullanabilirsiniz. Create, Read, Update, Delete ve List metodlarını sağlamanız yeterlidir. Veritabanı kullanmanız zorunlu değildir, ödevin amacı OpenAPI tasarımını sağlamanız ve uygulamanın buna uyumlu geliştirilmiş olmasıdır. Docker veya Docker compose ile container imajı üretip dockerhub sitesindeki public proje alanına yükleyiniz. Projenizi Github sayfanızda yeni bir repoya yükleyip linkini ödev altında text dosyaya yazıp iletmeniz yeterlidir. Oluşturacağınız repoda muhakkak açık kaynak kodlu projelere özgü tanımlayıcı dosyalar bulunmalıdır (readme, license, notice, code of conduct vs. md dosyaları)

Cevap:

kitap kaynağı üzerinden bir OpenAPI spesifikasyonunu gerçekleştirdim. Bu örnek, kitapların CRUD operasyonlarını ve List metodunu uygulayan basit bir RESTful API'yi tanımlamaktadır.

Openapi.yaml dosyası

```
openapi: 3.0.0
info:
  title: Kitap API
  version: 1.0.0
  description: CRUD operasyonlarını ve List metodunu sağlayan Kitap API'si

paths:
  /books:
    get:
      summary: Tüm kitapları listele
      operationId: listBooks
      responses:
        '200':
```

description: Başarılı yanıt

content:

application/json:

schema:

type: array

items:

\$ref: '#/components/schemas/Book'

post:

summary: Yeni bir kitap oluştur

operationId: createBook

requestBody:

required: true

content:

application/json:

schema:

\$ref: '#/components/schemas/Book'

responses:

'201':

description: Başarılı yanıt

content:

application/json:

schema:

\$ref: '#/components/schemas/Book'

/books/{id}:

get:

summary: Belirli bir kitabı getir

operationId: getBookById

parameters:

- name: id

in: path

required: true

schema:

type: string

responses:

'200':
 description: Başarılı yanıt
 content:
 application/json:
 schema:
 \$ref: '#/components/schemas/Book'

'404':
 description: Kitap bulunamadı

put:
 summary: Belirli bir kitabı güncelle
 operationId: updateBook
 parameters:
 - name: id
 in: path
 required: true
 schema:
 type: string
 requestBody:
 required: true
 content:
 application/json:
 schema:
 \$ref: '#/components/schemas/Book'

responses:
 '200':
 description: Başarılı yanıt
 content:
 application/json:
 schema:
 \$ref: '#/components/schemas/Book'

'404':
 description: Kitap bulunamadı

delete:
 summary: Belirli bir kitabı sil
 operationId: deleteBook
 parameters:

```
- name: id
  in: path
  required: true
  schema:
    type: string
responses:
  '204':
    description: Başarılı yanıt
  '404':
    description: Kitap bulunamadı
```

```
components:
  schemas:
    Book:
      type: object
      properties:
        id:
          type: string
        title:
          type: string
        author:
          type: string
        year:
          type: integer
```

Python dilinde OpenAPI spesifikasyonuna uygun bir API oluşturdum.

App.py dosya içeriği:

```
from flask import Flask, jsonify, request

app = Flask(__name__)

# Kitap verilerini depolamak için geçici bir liste kullanıyoruz

books = []

@app.route('/')

def home():
```

```
    return 'Merhaba benim adım Aisan Kheiri (170420995) ve bu açık kaynak ödevidir! Kitaplara erişim için ./book kullanarak erişim sağlayabilirsin.'
```

```
# Tüm kitapları listele
```

```
@app.route('/books', methods=['GET'])
```

```
def list_books():
```

```
    return jsonify(books)
```

```
# Yeni bir kitap oluştur
```

```
@app.route('/books', methods=['POST'])
```

```
def create_book():
```

```
    new_book = request.json
```

```
    books.append(new_book)
```

```
    return jsonify(new_book), 201
```

```
# Belirli bir kitabı getir
```

```
@app.route('/books/<string:id>', methods=['GET'])
```

```
def get_book_by_id(id):
```

```
    for book in books:
```

```
        if book['id'] == id:
```

```
            return jsonify(book)
```

```
    return jsonify({'message': 'Kitap bulunamadı'}), 404
```

```
# Belirli bir kitabı güncelle
```

```
@app.route('/books/<string:id>', methods=['PUT'])
```

```
def update_book(id):
```

```
    for book in books:
```

```
        if book['id'] == id:
```

```
            book.update(request.json)
```

```
            return jsonify(book)
```

```
    return jsonify({'message': 'Kitap bulunamadı'}), 404
```

```

# Belirli bir kitabı sil

@app.route('/books/<string:id>', methods=['DELETE'])

def delete_book(id):

    for book in books:

        if book['id'] == id:

            books.remove(book)

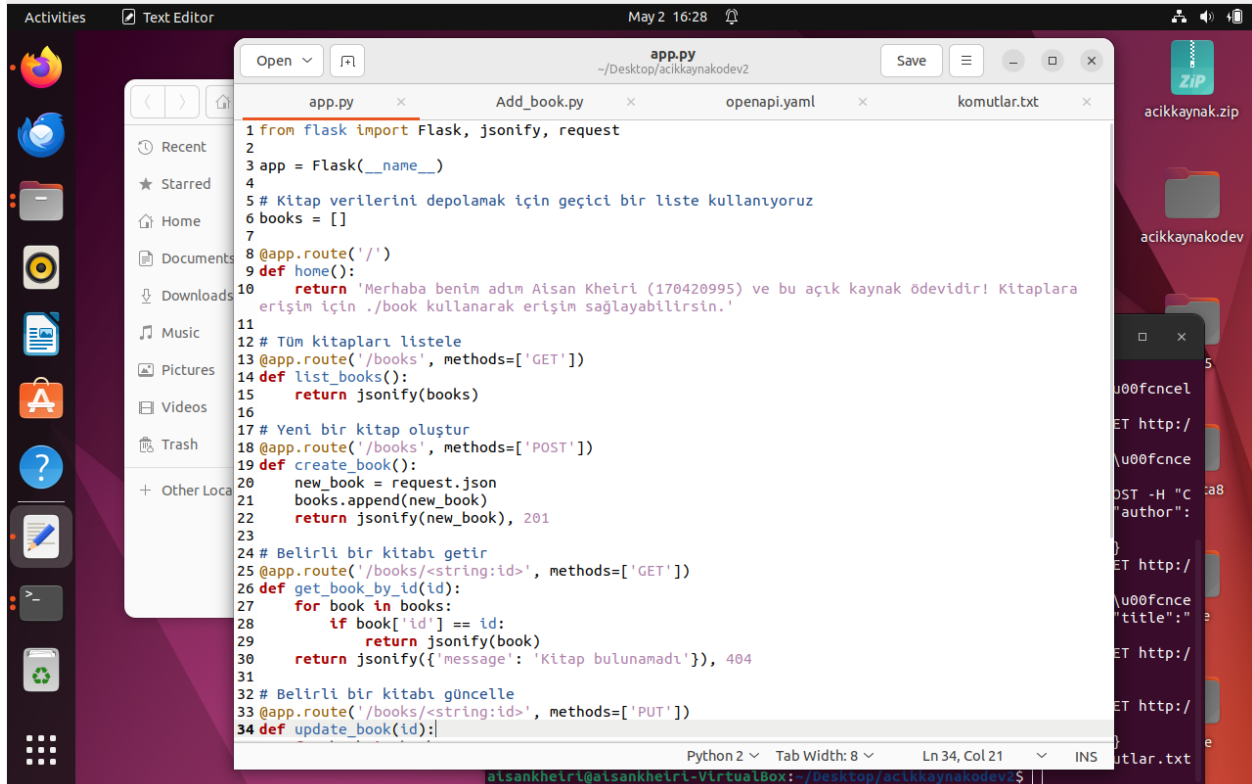
            return ", 204

    return jsonify({'message': 'Kitap bulunamadı'}), 404

if __name__ == '__main__':

    app.run()

```



```

app.py
~/Desktop/acikkaynakodev2

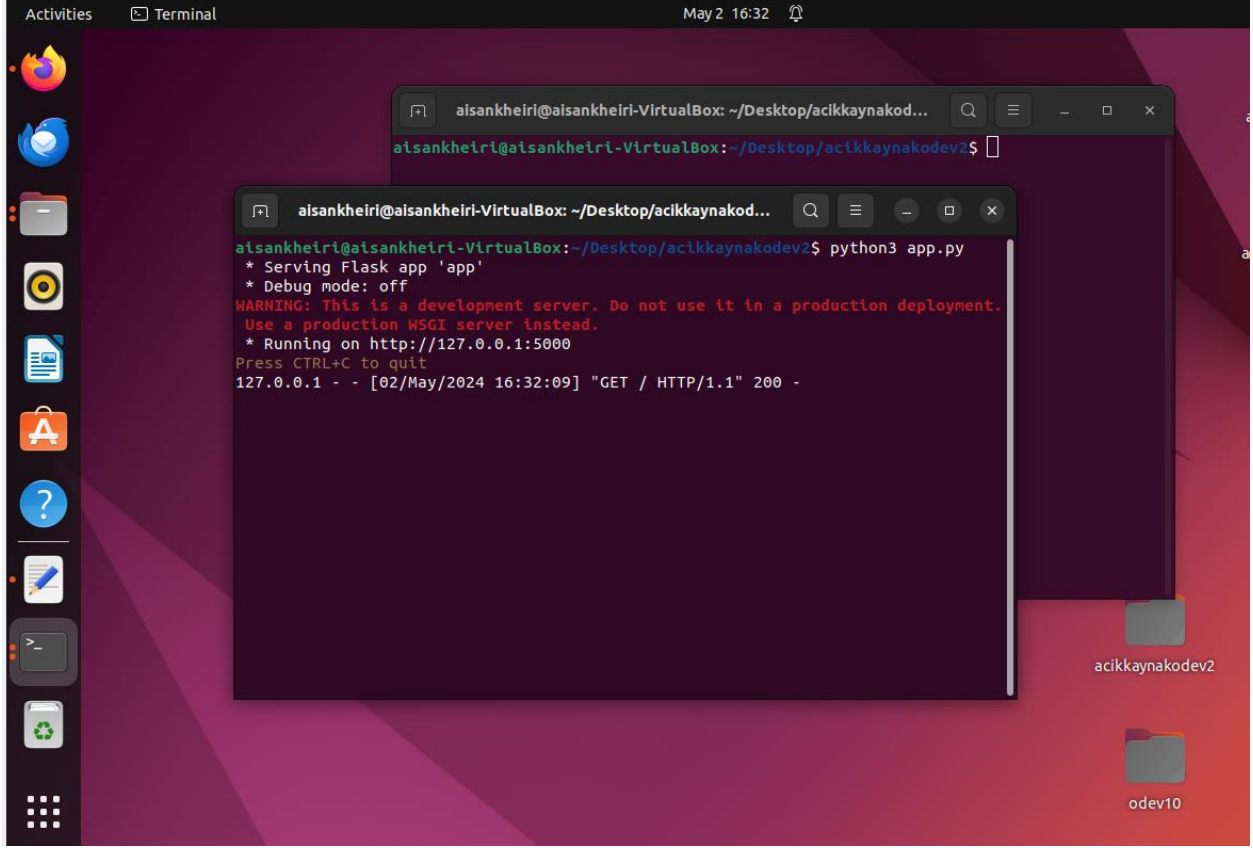
1 from flask import Flask, jsonify, request
2
3 app = Flask(__name__)
4
5 # Kitap verilerini depolamak için geçici bir liste kullanıyoruz
6 books = []
7
8 @app.route('/')
9 def home():
10     return 'Merhaba benim adım Aisankheiri (170420995) ve bu açık kaynak ödevidir! Kitaplara erişim için ./book kullanarak erişim sağlayabilirsiniz.'
11
12 # Tüm kitapları listele
13 @app.route('/books', methods=['GET'])
14 def list_books():
15     return jsonify(books)
16
17 # Yeni bir kitap oluştur
18 @app.route('/books', methods=['POST'])
19 def create_book():
20     new_book = request.json
21     books.append(new_book)
22     return jsonify(new_book), 201
23
24 # Belirli bir kitabı getir
25 @app.route('/books/<string:id>', methods=['GET'])
26 def get_book_by_id(id):
27     for book in books:
28         if book['id'] == id:
29             return jsonify(book)
30     return jsonify({'message': 'Kitap bulunamadı'}), 404
31
32 # Belirli bir kitabı güncelle
33 @app.route('/books/<string:id>', methods=['PUT'])
34 def update_book(id):

```

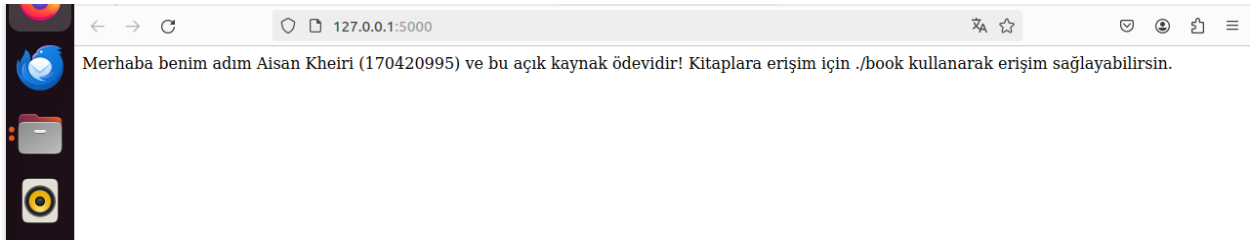
Görsel 1: app.py dosya içeriği

Flask çerçevesini kullanarak basit bir Kitap API'si oluşturdum. API rotaları, /books üzerinde CRUD operasyonlarını ve List metodunu gerçekleştirir. Kitap verileri geçici olarak books adlı bir liste içinde saklanır.

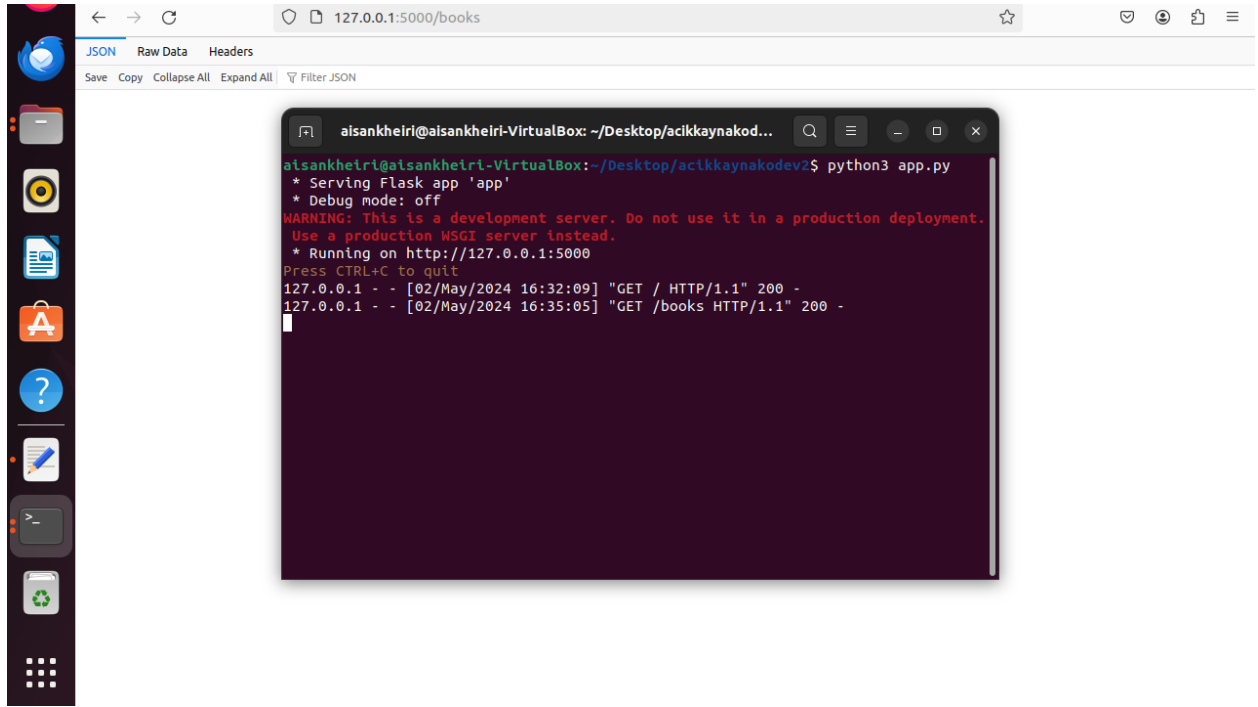
App.py dosyayı çalıştırdığımda elde edilenler:



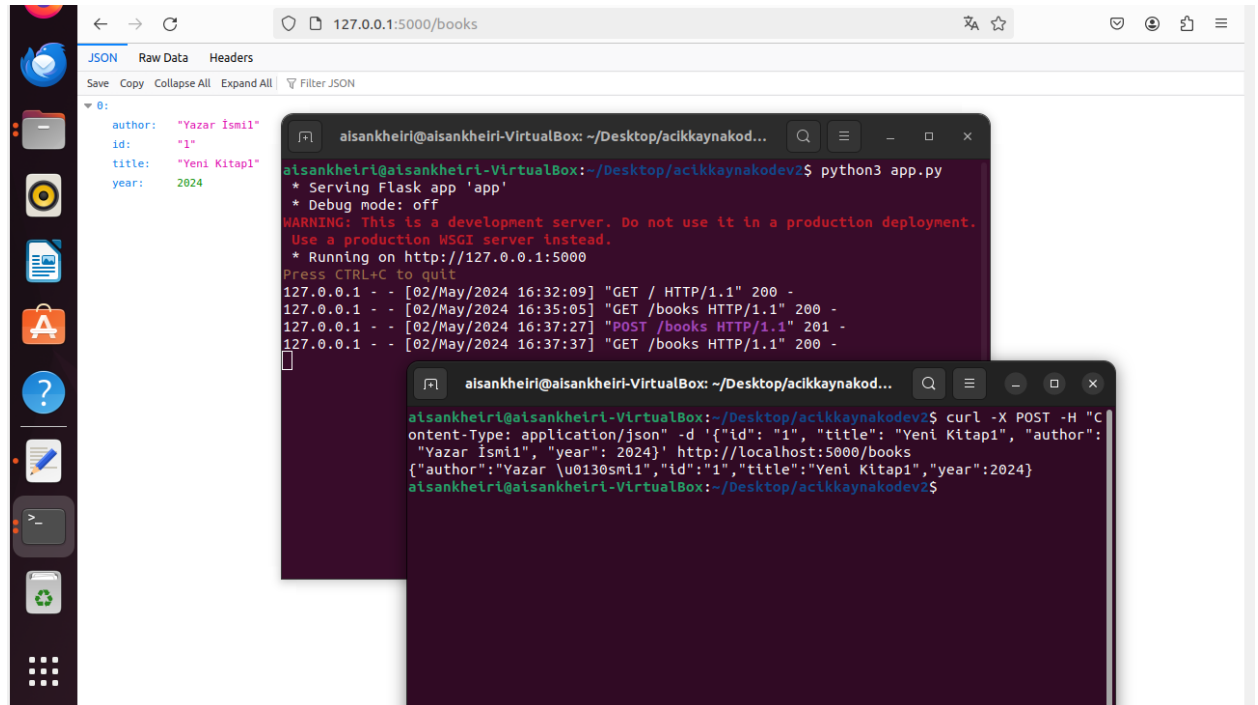
Görsel 2: Terminal ve 200 OK cevabı



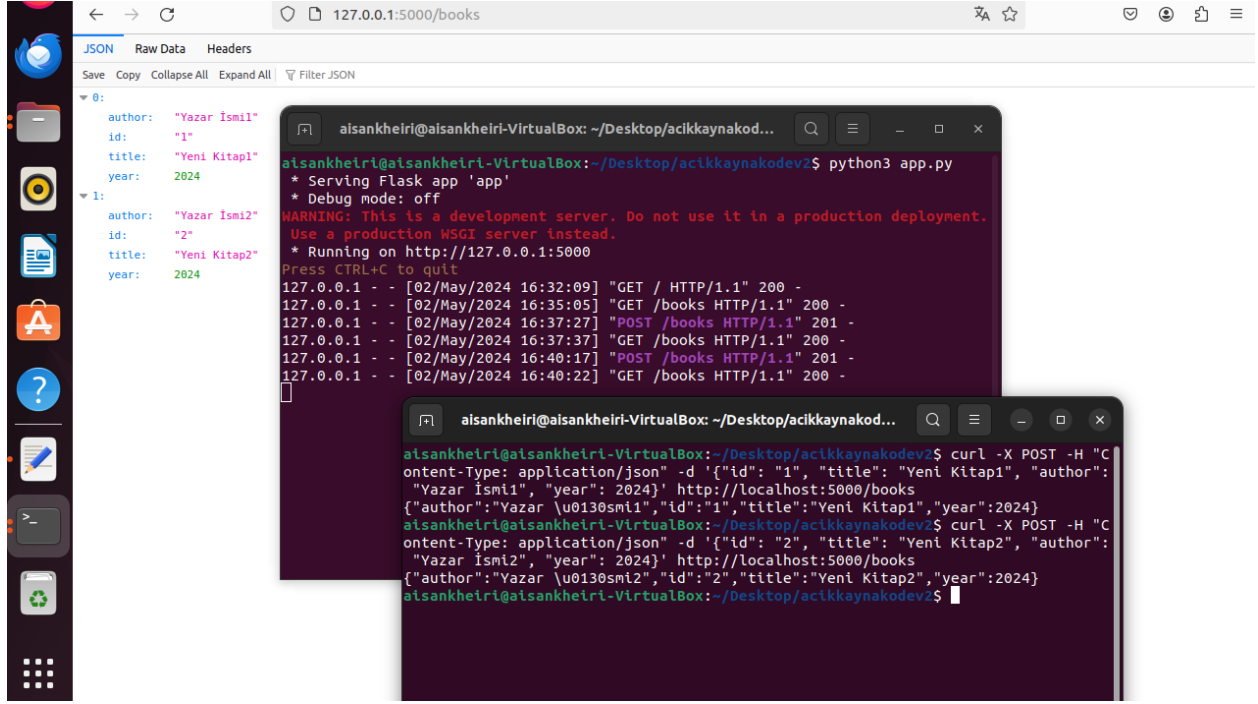
Görsel 3: Oluşan sonucu 127.0.0.1:5000



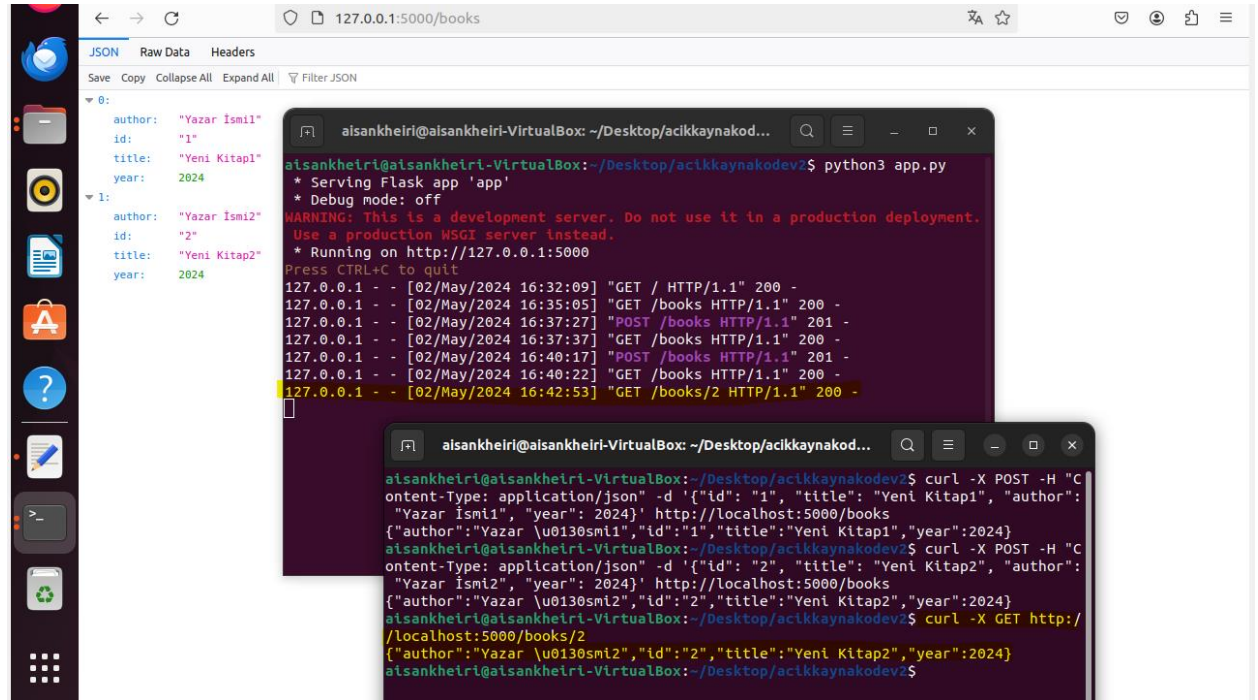
Görsel 4: 127.0.0.1:5000/books (GET)



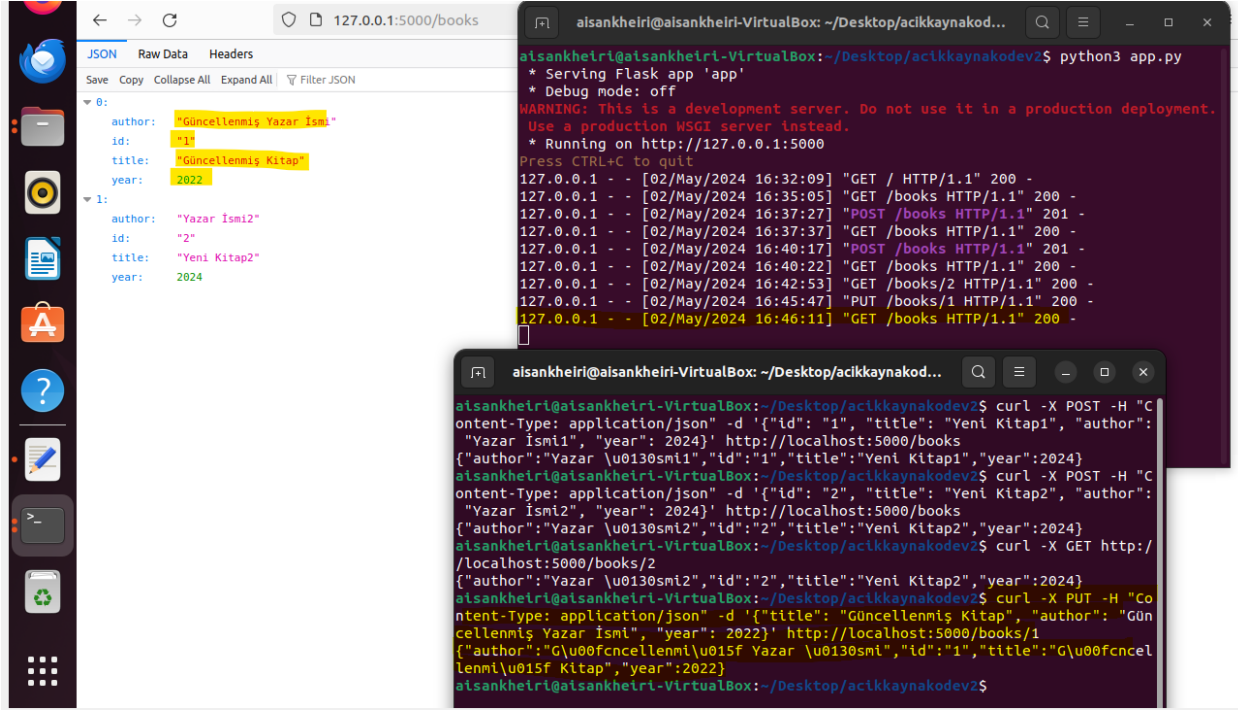
Görsel 5: 127.0.0.1:5000/books ekleme işlemi yapıldı (POST)



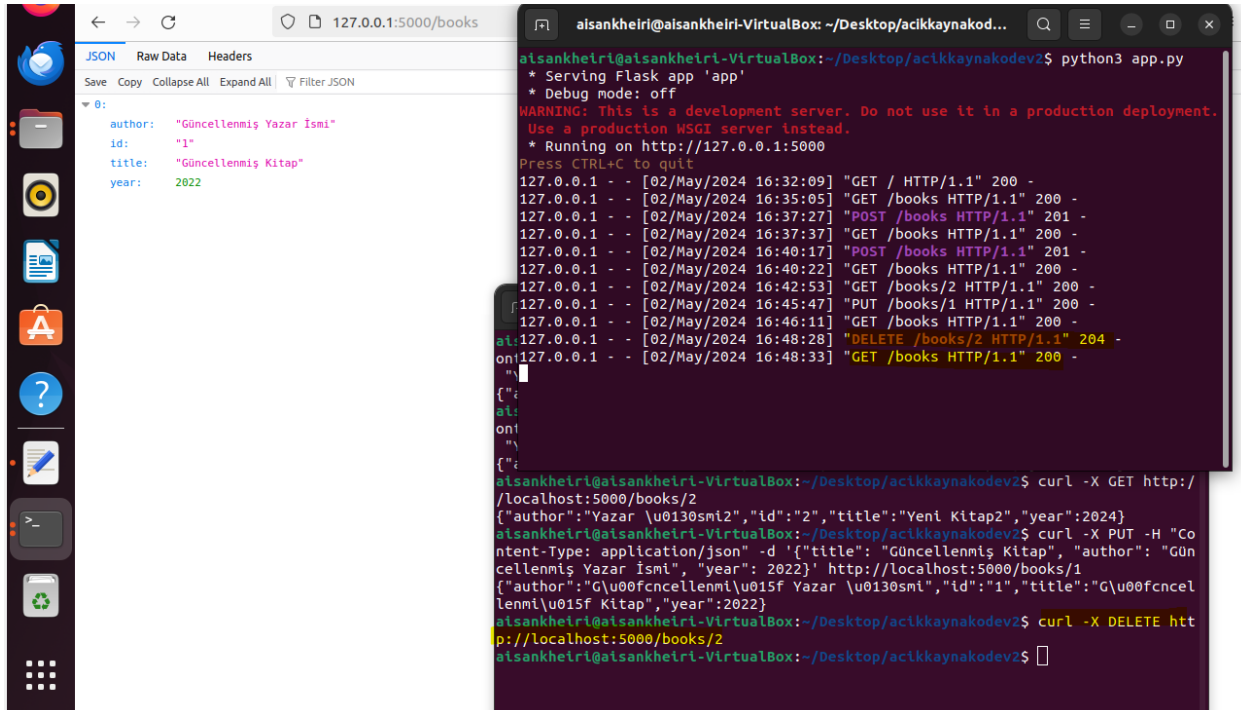
Görsel 6: 127.0.0.1:5000/books ikinci ekleme işlemi yapıldı (POST)



Görsel 7: Belirli bir kitabı almak için GET /books/{id}



Görsel 8: Belirli bir kitabı güncellemek için PUT /books/{id}



Görsel 9: Belirli bir kitabı silmek için DELETE /books/{id}

Docker huba yüklemek için önce dockerfile oluşturmalıyız.

Dockerfile dosya içeriği

Docker imajın temel alınacak taban imajı

FROM python:3.10.12

Uygulamanın çalışacağı çalışma dizisi

WORKDIR /home/aisankheiri/Desktop/acikkaynakodev2

Uygulama bağımlılıkların kopyası

COPY requirements.txt .

Bağımlılıkları yükle

RUN pip install --no-cache-dir -r requirements.tx

Uygulama dosyalarını kopyala

COPY . .

Konteynerin hangi portu dinleyeceğini belirle

EXPOSE 5000

Uygulamayı başlatmak için komutu belirle

CMD ["python", "app.py"]

```
aisankheiri@aisankheiri-VirtualBox:~/Desktop/acikkaynakodev2$ sudo docker build -t aisankheiri/actik_kaynak_odev_5mayis .
[+] Building 7.2s (11/11) FINISHED
      docker:default
=> [internal] load build definition from Dockerfile                                0.0s
=> => transferring dockerfile: 527B                                              0.0s
=> [internal] load metadata for docker.io/library/python:3.10.12                1.1s
=> [auth] library/python:pull token for registry-1.docker.io                   0.0s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                  0.0s
=> CACHED [1/5] FROM docker.io/library/python:3.10.12@sha256:bac3a0e0d16        0.0s
=> [internal] load build context                                                0.1s
=> => transferring context: 5.46kB                                              0.0s
=> [2/5] WORKDIR /home/aisankheiri/Desktop/acikkaynakodev2                    0.1s
=> [3/5] COPY requirements.txt .                                                0.1s
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt                    5.2s
=> [5/5] COPY . .                                                              0.2s
=> exporting to image                                                           0.3s
=> => exporting layers                                                         0.3s
=> => writing image sha256:2339f2f8fd5909466731783c06d6568919ff6e5bb6f8b      0.0s
=> => naming to docker.io/aisankheiri/actik_kaynak_odev_5mayis                0.0s

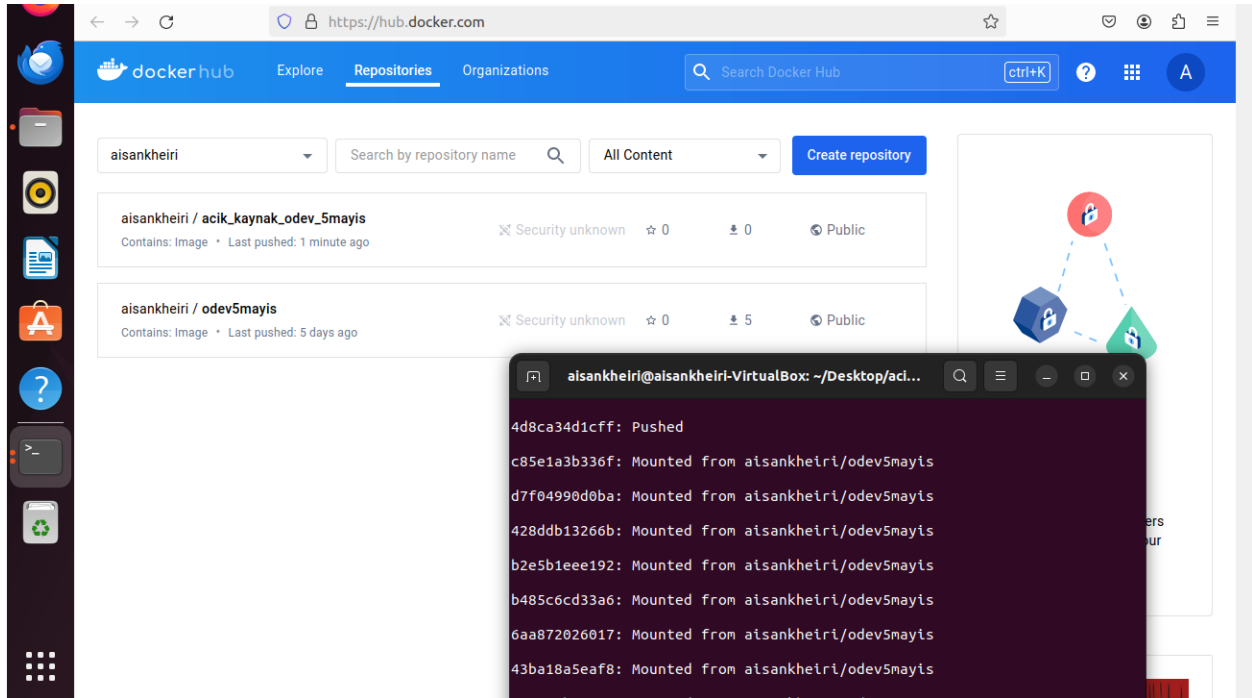
What's Next?
View a summary of image vulnerabilities and recommendations -> docker scout quickview
aisankheiri@aisankheiri-VirtualBox:~/Desktop/acikkaynakodev2$
```

Görsel 10: Docker imajı oluşturma

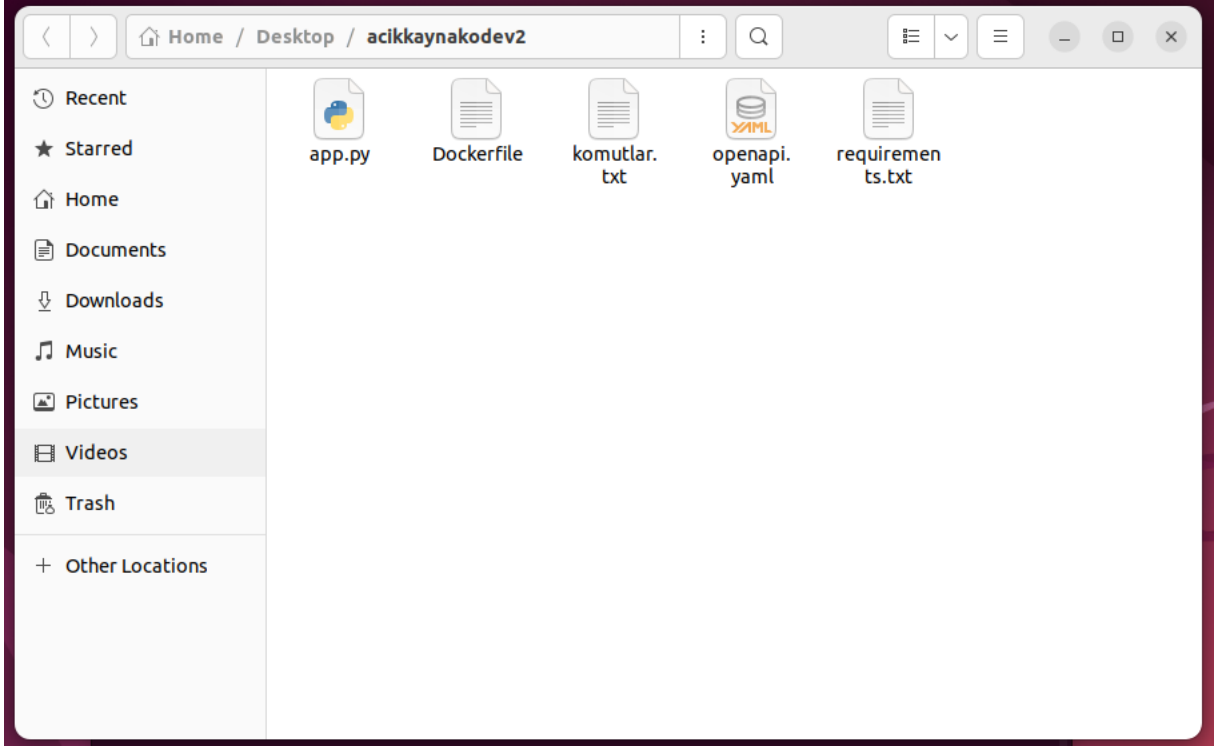
```
Activities Terminal May 2 17:31
aisankheiri@aisankheiri-VirtualBox: ~/Desktop/acikkaynakodev2
n%2Fdocker.sock/_ping": dial unix /var/run/docker.sock: connect: permission denied
aisankheiri@aisankheiri-VirtualBox:~/Desktop/acikkaynakodev2$ sudo docker build -t aisankheiri/acik_kaynak_odev_5mayis .
[+] Building 7.2s (11/11) FINISHED
=> [internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 527B 0.0s
=> [internal] load metadata for docker.io/library/python:3.10.12 1.1s
=> [auth] library/python:pull token for registry-1.docker.io 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> CACHED [1/5] FROM docker.io/library/python:3.10.12@sha256:bac3a0e0d16 0.0s
=> [internal] load build context 0.1s
=> => transferring context: 5.46kB 0.0s
=> [2/5] WORKDIR /home/aisankheiri/Desktop/acikkaynakodev2 0.1s
=> [3/5] COPY requirements.txt . 0.1s
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt 5.2s
=> [5/5] COPY . . 0.2s
=> exporting to image 0.3s
=> => exporting layers 0.3s
=> => writing image sha256:2339f2f8fd5909466731783c06d6568919ff6e5bb6f8b 0.0s
=> => naming to docker.io/aisankheiri/acik_kaynak_odev_5mayis 0.0s

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview
aisankheiri@aisankheiri-VirtualBox:~/Desktop/acikkaynakodev2$ sudo docker push aisankheiri/acik_kaynak_odev_5mayis
Using default tag: latest
The push refers to repository [docker.io/aisankheiri/acik_kaynak_odev_5mayis]
8aefb13c0468: Pushed
4fd82c31550: Pushed
267ff93e0fd5: Pushed
4d8ca34d1cff: Pushed
c85e1a3b336f: Mounted from aisankheiri/odev5mayis
d7f0499d0ba: Mounted from aisankheiri/odev5mayis
428ddb13266b: Mounted from aisankheiri/odev5mayis
b2e5b1eee192: Mounted from aisankheiri/odev5mayis
b485c6cd33a6: Mounted from aisankheiri/odev5mayis
6aa872026017: Mounted from aisankheiri/odev5mayis
43ba18a5eaf8: Mounted from aisankheiri/odev5mayis
ff61a9b258e5: Mounted from aisankheiri/odev5mayis
latest: digest: sha256:3807cc1e6996b1914e36603fb1635f18e9e0c2510abe337f62ea0e81c55af4e6 size: 2840
aisankheiri@aisankheiri-VirtualBox:~/Desktop/acikkaynakodev2$
```

Görsel 11: Docker imajını push etme



Görsel 12: Docker imajını docker hub da görüntülemek



Görsel 13: Ödev dosyaları

```
C:\Windows\System32\cmd.exe
C:\Users\Aisan\Desktop\acikkaynakodev2>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   README.md

C:\Users\Aisan\Desktop\acikkaynakodev2>git commit -m "README dosyasi eklendi"
[master 48f7b2a] README dosyasi eklendi
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

C:\Users\Aisan\Desktop\acikkaynakodev2>git branch -M main

C:\Users\Aisan\Desktop\acikkaynakodev2>git remote add origin https://github.com/aisankheiri/homework-5-may-open-source-code.git

C:\Users\Aisan\Desktop\acikkaynakodev2>git push -u origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (10/10), 2.35 KiB | 2.35 MiB/s, done.
Total 10 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/aisankheiri/homework-5-may-open-source-code.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

C:\Users\Aisan\Desktop\acikkaynakodev2>
```

Görsel 14: Git işlemleri ve Githuba yükleme yapıldı



14