

Task 2

Observation – Running Without Docker

1. Execution Environment:

- The FastAPI application was executed using:

```
C:\Users\Srivatsan\Documents\LexTrack AI\Task 2\chatbot-mini-app>uvicorn main:app --reload
INFO: Will watch for changes in these directories: ['C:\\Users\\Srivatsan\\Documents\\LexTrack AI\\Task 2\\chatbot-mini-app']
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: Started reloader process [29528] using StatReload
INFO: Started server process [31936]
INFO: Waiting for application startup.
INFO: Application startup complete.
```

- This ran the app directly from the system-installed Python environment on Windows 10.
- Uvicorn Logs:**
 - Server started successfully at: **http://127.0.0.1:8000**
 - Live reloading enabled using StatReload
 - No startup errors were observed.

2. Swagger UI Interface:

- Swagger UI was accessible at: <http://127.0.0.1:8000/docs>
- The API endpoint /generate appeared correctly with method POST.
- The input schema expected a JSON object with a single field:

FastAPI 0.10.0 OAS 3.0

default

POST /generate Generate Response

Parameters

No parameters

Request body *required* application/json

Edit Value | Schema

```
{
  "message": "string"
}
```

Execute Clear

3. Response

Input:

Request body *required* application/json

Edit Value | Schema

```
{
  "message": "string"
}
```

Execute Clear

Server returned HTTP 200 OK with output:

Code

Details

200

Response body

```
{
  "response": "I'm a simple chatbot. Ask me anything!"
}
```

Copy

Download

Response headers

```
content-length: 53
content-type: application/json
date: Wed, 18 Jun 2025 00:58:38 GMT
server: uvicorn
```

Responses

- The response was quick and matched expected chatbot behavior.

Validation Error Scenario

- When malformed input was submitted (e.g., missing or incorrect fields), FastAPI returned:
 - HTTP Status: 422 (Unprocessable Entity)
 - Error body included:

422

Validation Error

No links

Media type

application/json

Example Value

Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

Conclusion

- The app executed smoothly in the local environment.
- FastAPI automatically handled:
 - API route exposure
 - Swagger documentation
 - Input validation and error handling

The chatbot logic responded as designed, and the /generate endpoint functioned as expected.

Observation – Running With Docker

Docker Build Process

- The application was containerized using the following command:

```
Microsoft Windows [Version 10.0.26100.4351]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Arijvatsan\Documents>Track AI\Task 2\chatbot-mini-app>docker build -t chatbot-mini-app .
[+] Building 1.3s (11/11) FINISHED
=> [internal] load build definition from Dockerfile 0.0s
=> transferring dockerfile: 239B 0.0s
=> [internal] load metadata for docker.io/library/python:3.10-slim 0.9s
=> [auth] library/python:pull token for registry-l.docker.io 0.0s
=> [internal] load dockerignore 0.0s
=> transferring context: 2B 0.0s
=> [internal] load build context 0.0s
=> transferring context: 377B 0.0s
=> [1/5] FROM docker.io/library/python:3.10-slim@sha256:63d724ef64585eeb6e82385e9aabcbeabf5f7cae2c2dcedb1da9511 0.1s
=> resolve docker.io/library/python:3.10-slim@sha256:63d724ef64585eeb6e82385e9aabcbeabf5f7cae2c2dcedb1da9511 0.0s
=> CACHED [2/5] WORKDIR /app 0.0s
=> CACHED [3/5] COPY requirements.txt . 0.0s
=> CACHED [4/5] RUN pip install --no-cache-dir -r requirements.txt 0.0s
=> CACHED [5/5] COPY . . 0.0s
=> exporting to image 0.1s
=> exporting layers 0.0s
=> exporting manifest sha256:494d95e82d761d27f61612a985ab613d3b569ebc794a4879d52d1688def0cc63 0.0s
=> exporting config sha256:657826ad88c3779563e5980749e71794977aad39c5465fc32d241ed165aa32 0.0s
=> exporting attestation manifest sha256:4f288401ec965b57acd5ad7881242acde61e3265f2f8ee9c186df658ab2e9758 0.0s
=> exporting manifest list sha256:2d572385c1b9c2d7a1d977eaf77ac34d32f271b33157076ad6042f6b79dc24 0.0s
=> naming to docker.io/library/chatbot-mini-app:latest 0.0s
=> unpacking to docker.io/library/chatbot-mini-app:latest 0.0s
```

- The Docker build completed successfully:
 - Based on python:3.10-slim image.

- Installed fastapi and uvicorn via requirements.txt.
- Image tagged as: chatbot-mini-app:latest.

Docker Container Execution:

- The app was launched using:

```
C:\Users\Srivatsan\Documents\LexTrack AI\Task 2\chatbot-mini-app>docker run -p 8000:8000 chatbot-mini-app
INFO: Started server process [1]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
INFO: 172.17.0.1:50638 - "GET /docs HTTP/1.1" 200 OK
INFO: 172.17.0.1:50638 - "GET /openapi.json HTTP/1.1" 200 OK
INFO: 172.17.0.1:50626 - "POST /generate HTTP/1.1" 200 OK
INFO: 172.17.0.1:59244 - "GET /docs HTTP/1.1" 200 OK
INFO: 172.17.0.1:59244 - "GET /openapi.json HTTP/1.1" 200 OK
INFO: 172.17.0.1:59234 - "POST /generate HTTP/1.1" 200 OK
```

- Output logs confirm successful container start:
 - Server running at <http://0.0.0.0:8000>
 - Swagger UI and endpoints served on port 8000 mapped to host.
- **Swagger UI Inside Docker:**
 - Accessed at: <http://localhost:8000/docs>
 - The FastAPI Swagger interface correctly rendered inside Docker.
 - /generate POST endpoint was visible and functional.

Successful Response via Docker

- **Input JSON:**

FastAPI 0.100.0 **QAS 3.1**

default

POST /generate Generate Response

Parameters

No parameters

Request body *required* application/json

Edit Value Schema

```
{
  "message": "string"
}
```

Execute Clear

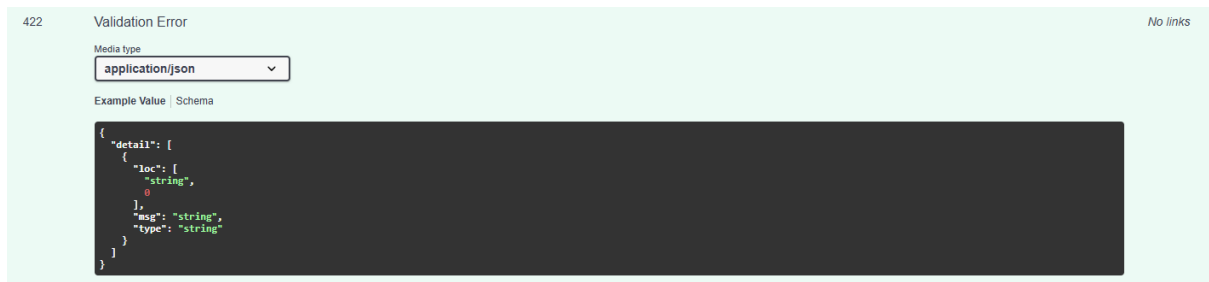
- **Server response:**

Responses		
Code	Description	Links
200	Successful Response	No links
	Media type	
	application/json	
	Controls Accept header	
	Example Value Schema	
	"string"	
422	Validation Error	No links

- HTTP status: 200 O
- Response headers showed:
 - application/json as content-type
 - Uvicorn running inside the container

Validation Error Handling

- When required input was invalid (e.g., missing message field), server returned:
 - HTTP 422 Unprocessable Entity
 - Proper error body with detail, loc, msg, and type
 - Confirms that FastAPI's request validation also works as expected inside Docker



Conclusion

- The Docker container behaved identically to the local version.
- Swagger UI, API functionality, request handling, and validation were all consistent.
- The main difference: Docker isolated the runtime environment, making it portable and deployment-ready.