

Assignment 1: Setup and Test Report

1. Introduction

This report documents the setup process and testing results for Assignment 1. The goal is to ensure the environment is correctly configured and the API is functioning as expected.

2. Environment Setup

- **Operating System:** Linux
- **Programming Language:** Go
- **Project Structure:**
 - API 0.1/
 - cmd/api/main.go
 - internal/api/handlers/
 - internal/api/middleware/
 - internal/api/repository/
 - internal/api/server/
 - internal/api/service/

Steps:

1. Cloned the repository from the provided source.
2. Installed Go, mingw32-base, mingw32-gcc-g++ packages and verified the version.
3. Navigated to the project directory and ran `go mod tidy` to install dependencies.

```
● muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ go version
go version go1.22.5 linux/amd64
● muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ gcc --version
gcc (Ubuntu 10.5.0-1ubuntu1~22.04.2) 10.5.0
Copyright (C) 2020 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
● muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ g++ --version
g++ (Ubuntu 10.5.0-1ubuntu1~22.04.2) 10.5.0
Copyright (C) 2020 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
○ muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$
```

3. Running the API

- Navigated to `API 0.1/cmd/api/`.
- Ran the API server using:

```
go run main.go
```

- Confirmed the server started successfully and was accessible at the expected endpoint.

```
muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ go run main.go
2025/12/10 23:12:42 main.go:61: Starting server on :8080...
^C2025/12/10 23:16:05 server.go:41: Gracefully shutting down server...
2025/12/10 23:16:05 main.go:67: Server gracefully shutdown complete.
muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ go env -w CGO_ENABLED=1
muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ go build
muditha@lap:~/IntelligentDevices/Assignments/1st Assignment/API 0.1/cmd/api$ go run main.go
2025/12/10 23:16:39 main.go:61: Starting server on :8080...
```

4. Testing

- API Test with Thunder Client

The screenshot shows the Thunder Client interface. On the left, a POST request is configured to `http://127.0.0.1:8080/data`. The 'Headers' tab is active, showing 'Content-Type' set to 'application/json'. On the right, the 'Response' tab shows a successful status of 201 Created, with a size of 145 Bytes and a time of 27 ms. The response body is a JSON object:

```
{
  "id": 13,
  "device_id": "device1",
  "device_name": "device1",
  "value": 1,
  "type": "type1",
  "date_time": "2021-01-01T00:00:00Z",
  "description": "description1"
}
```

Setup and Test Report.md 2, M

New Request X

GET

http://127.0.0.1:8080/data?page=0

Send

Query

Headers 1

Auth 1

Body 1

Tests

Pre Run

JSON

XML

Text

Form

Form-encode

GraphQL

Binary

JSON Content

Format

```
1 {
2   "type": "none"
3 }
```

Status: 200 OK

Size: 1.67 KB

Time: 5 ms

Response

Headers 5

Cookies

Results

Docs

{}

```
1 [
2   {
3     "id": 3,
4     "device_id": "ML123 ",
5     "device_name": "Lämpötilan mittauspiste",
6     "value": 22.23,
7     "type": "XXXX",
8     "date_time": "2024-04-22T15:17:15Z",
9     "description": ""
10  },
11  {
12    "id": 4,
13    "device_id": "ML123 ",
14    "device_name": "Lämpötilan mittauspiste",
15    "value": 22.23,
16    "type": "XXXX",
17    "date_time": "2024-04-22T15:17:15Z",
18    "description": ""
19  },
20  {
21    "id": 5,
22    "device_id": "ML123 ",
23    "device_name": "Lämpötilan mittauspiste",
24    "value": 22.23,
25    "type": "XXXX",
26    "date_time": "2024-04-22T15:17:15Z",
27    "description": ""
28  },
29  {
```

Setup and Test Report.md 2, M

New Request X

GET

http://127.0.0.1:8080/data/4

Send

Query

Headers 1

Auth 1

Body 1

Tests

Pre Run

JSON

XML

Text

Form

Form-encode

GraphQL

Binary

JSON Content

Format

```
1 {
2   "type": "none"
3 }
```

Status: 200 OK

Size: 152 Bytes

Time: 3 ms

Response

Headers 5

Cookies

Results

Docs

{}

```
1 {
2   "id": 4,
3   "device_id": "ML123 ",
4   "device_name": "Lämpötilan mittauspiste",
5   "value": 22.23,
6   "type": "XXXX",
7   "date_time": "2024-04-22T15:17:15Z",
8   "description": ""
9 }
```

Setup and Test Report.md 2, M

New Request X

1.pdf

PUT

http://127.0.0.1:8080/data

Send

Query

Headers 1

Auth 1

Body 1

Tests

Pre Run

JSON

XML

Text

Form

Form-encode

GraphQL

Binary

JSON Content

Format

```
1 {
2   "id": 4,
3   "device_id": "DEVICE001",
4   "device_name": "Updated Sensor",
5   "value": 26.0,
6   "type": "TEMP",
7   "date_time": "2024-12-11T11:00:00Z",
8   "description": "Updated description"
9 }
```

Status: 200 OK

Size: 160 Bytes

Time: 8 ms

Response

Headers 5

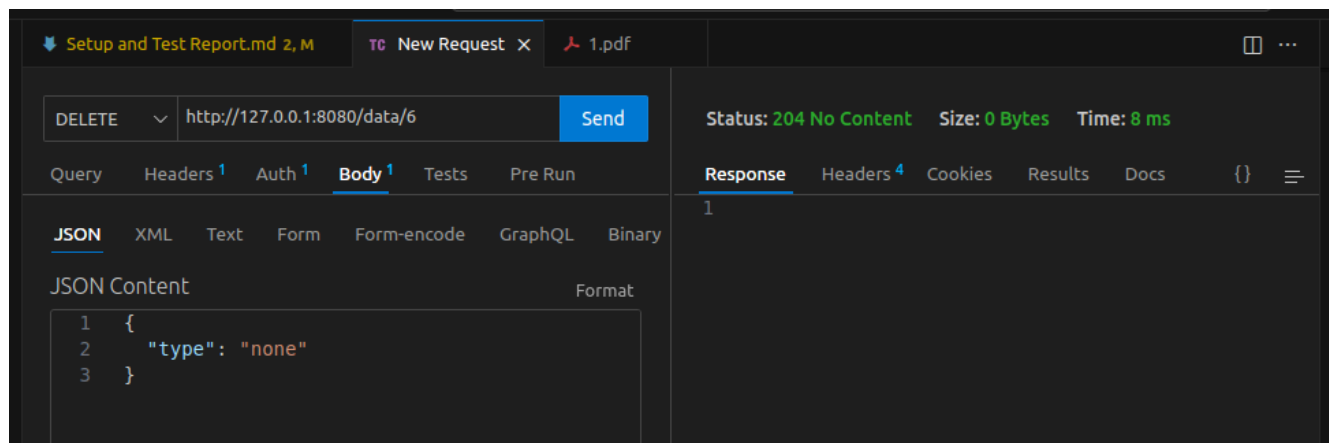
Cookies

Results

Docs

{}

```
1 {
2   "id": 4,
3   "device_id": "DEVICE001",
4   "device_name": "Updated Sensor",
5   "value": 26,
6   "type": "TEMP",
7   "date_time": "2024-12-11T11:00:00Z",
8   "description": "Updated description"
9 }
```



5. Issues Encountered

- No major issues encountered during setup or testing.

6. Conclusion

The environment was set up successfully, and all tests passed. The API is ready for further development and integration.