

# Webhook System Integrator Rapport

Guide can be found at this [URL](#).

## 1. Started the webhook\_receiver and hosted it on localtunnel.

Started the webhook receiver main.py file at this [URL](#). And then I opened the project up to be accessed using this command:

```
'lt -l 127.0.0.1 --port 8001 --subdomain integrator-webhook'
```

Which gave access to the webhook\_receiver project on the URL:

“https://integrator-webhook.loca.lt”

## 2. Ran the script to register webhooks.

In the script code I set the ‘exposee\_base\_url’ = “https://424e-85-82-70-165.ngrok-free.app”

And I set the ‘integrator\_base\_url’ = “https://integrator-webhook.loca.lt”

The script code can be found on this [URL](#).

When running the script code this was printed to the terminal:

```
1  import requests
2
3      You, yesterday • Started getting ready to integrate with Olivers...
4  exposee_base_url = "https://424e-85-82-70-165.ngrok-free.app"
5  integrator_base_url = "https://integrator-webhook.loca.lt"
6
7  # Subscribe the consumer server to webhook events
8  subscriptions = [
9      {"event": "payment_received", "url": f"{integrator_base_url}/webhook"},
10     {"event": "payment_processed", "url": f"{integrator_base_url}/webhook"},
11     {"event": "invoice_processing", "url": f"{integrator_base_url}/webhook"},
12     {"event": "invoice_completed", "url": f"{integrator_base_url}/webhook"}
13 ]
14
15 def register_webhooks():
16     for subscription in subscriptions:
17         response = requests.post(f"{exposee_base_url}/register", json=subscription)
18         if response.ok: # Check if the response status code is 200-299
19             print(f"Subscription response: {response.status_code}, {response.json()}")
20         else:
21             print(f"Failed to register webhook: {response.status_code}, {response.text}")
22
23
24
25 if __name__ == "__main__":
26     register_webhooks()
27
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS GITLENS

Marcus@DESKTOP-AJ2RGJ0 MINGW64 ~/Desktop/KEA/KEA\_Studie/System Integration/Opgaver/mandatory/12a.\_Expose\_and\_integrate\_with\_a\_webhook\_system/webhook\_receiver (main)

```
$ poetry run python webhook_script.py
Subscription response: 200, {'message': 'Webhook registered successfully', 'url': 'https://integrator-webhook.loca.lt/webhook', 'event': 'payment_received'}
Subscription response: 200, {'message': 'Webhook registered successfully', 'url': 'https://integrator-webhook.loca.lt/webhook', 'event': 'payment_processed'}
Subscription response: 200, {'message': 'Webhook registered successfully', 'url': 'https://integrator-webhook.loca.lt/webhook', 'event': 'invoice_processing'}
Subscription response: 200, {'message': 'Webhook registered successfully', 'url': 'https://integrator-webhook.loca.lt/webhook', 'event': 'invoice_completed'}
```

I then used Postman to register a nonexistent Webhook URL, which can be seen here:

HTTP 12a\_Exposee\_and\_integrate\_with\_a\_webhook\_system / Oliver exposee Webhook system / Register Webhook - Invalid URL

POST {{base\_url}}/register

Params Authorization Headers (10) Body ● Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▾

```
1 {
2   "event": "payment_received",
3   "url": "https://this-needs-to-fail.lt/webhook"
4 }
```

Body Cookies Headers (5) Test Results ↻

{ JSON ▾ ▶ Preview 🔗 Visualize ▾

```
1 {
2   "message": "Webhook registered successfully",
3   "url": "https://this-needs-to-fail.lt/webhook",
4   "event": "payment_received"
5 }
```

### 3. Pinging the registered webhooks.

Used Postman to call the “/ping” endpoint to ping all the registered webhooks, and I get this in Postman, where you can see which webhooks was a success and which was not:

12a\_Exposee\_and\_integrate\_with\_a\_webhook\_system / Oliver exposee Webhook system / Ping All Webhooks

GET {{base\_url}}/ping Send

Params Authorization Headers (7) Body Scripts Settings Co

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (5) Test Results ↻

{ JSON ▾ ▶ Preview 🔗 Visualize ▾ 200 OK · 4.55 s · 1.28 KB · Save Response

```
1 {
2   "successful_webhooks_count": 4,
3   "successful_webhooks": [
4     {
5       "event": "payment_received",
6       "url": "https://integrator-webhook.local.it/webhook",
7       "payload": {
8         "message": "Webhook received successfully"
9       },
10      "status_code": 200
11    },
12    {
13      "event": "payment_processed",
14      "url": "https://integrator-webhook.local.it/webhook",
15      "payload": {
16        "message": "Webhook received successfully"
17      },
18      "status_code": 200
19    },
20    {
21      "event": "invoice_processing",
22      "url": "https://integrator-webhook.local.it/webhook",
23      "payload": {
24        "message": "Webhook received successfully"
25      },
26      "status_code": 200
27    },
28    {
29      "event": "invoice_completed",
30      "url": "https://integrator-webhook.local.it/webhook",
31      "payload": {
32        "message": "Webhook received successfully"
33      },
34      "status_code": 200
35    }
36  ],
37   "failed_webhooks_count": 1,
38   "failed_webhooks": [
39     {
40       "event": "payment_received",
41       "url": "https://this-needs-to-fail.lt/webhook",
42       "error": "HTTPConnectionPool(host='this-needs-to-fail.lt', port=443): Max retries exceeded with url: /webhook (Caused by NameResolutionError(\"curllib3.connection.HTTPSConnection object at 0x00000619116E5090: Failed to resolve 'this-needs-to-fail.lt' ([Errno 1100])",
43       "getaddrinfo failed()\")",
44       "status_code": null
45     }
46  ],
47   "message": "Pinged all registered webhooks successfully."
48 }
```

And I got this in the terminal running the webhook\_receiver project:

```
1 from fastapi.middleware.cors import CORSMiddleware
2 from fastapi import FastAPI, Request, Query
3 from fastapi.templating import Jinja2Templates
4 from fastapi.responses import HTMLResponse
5 from fastapi.staticfiles import StaticFiles
6 from typing import Dict, List, Optional
7 from datetime import datetime, timezone
8
9 app = FastAPI()
10
11 CORS_SETTINGS = {
12     "allow_origins": ["*"], # Allow all origins
13     "allow_credentials": True,
14     "allow_methods": ["*"], # Allow all HTTP methods
15     "allow_headers": ["*"], # Allow all headers
16 }
17
18 # Add CORS middleware to the application
19 app.add_middleware(CORSMiddleware, **CORS_SETTINGS)
20
21 # Set up templates for the frontend
22 templates = Jinja2Templates(directory="templates")
23 app.mount("/static", StaticFiles(directory="static"), name="static")
24
25 # In-memory database
26 received_events: List[Dict] = [] # To store all received webhook events
27
28 @app.post("/webhook")
29 async def webhook_endpoint(request: Request):
30     """
31     A simple webhook endpoint that logs the received payload and stores them in the in-memory database.
32     """
33     payload: Dict = await request.json()
34     print(f"Received webhook payload: {payload}")
35
36     event_type = payload.get("event", "unknown")
37     event_data = payload.get("data", {})
38     event = {
39         "event_type": event_type,
40         "data": event_data,
41         "received_at": datetime.now(timezone.utc).isoformat()
42     }
43     received_events.append(event)
44
45     print(f"Webhook received: {event}")
46
47     return {"message": "Webhook received successfully"}
48
49
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

INFO: Application startup complete.

Received webhook payload: {'event': 'payment\_received', 'data': {'message': 'Ping from Webhook Service'}}

Webhook received: {'event\_type': 'payment\_received', 'data': {'message': 'Ping from Webhook Service'}, 'received\_at': '2025-05-08T08:21:36.955168+00:00'}

Webhook received: {'event\_type': 'payment\_received', 'data': {'message': 'Ping from Webhook Service'}, 'received\_at': '2025-05-08T08:21:36.955168+00:00'}

INFO: 85.82.70.165:0 - "POST /webhook HTTP/1.1" 200 OK

Received webhook payload: {'event': 'payment\_processed', 'data': {'message': 'Ping from Webhook Service'}}

Webhook received: {'event\_type': 'payment\_processed', 'data': {'message': 'Ping from Webhook Service'}, 'received\_at': '2025-05-08T08:21:37.923578+00:00'}

INFO: 85.82.70.165:0 - "POST /webhook HTTP/1.1" 200 OK

Received webhook payload: {'event': 'invoice\_processing', 'data': {'message': 'Ping from Webhook Service'}}

Webhook received: {'event\_type': 'invoice\_processing', 'data': {'message': 'Ping from Webhook Service'}, 'received\_at': '2025-05-08T08:21:38.797720+00:00'}

INFO: 85.82.70.165:0 - "POST /webhook HTTP/1.1" 200 OK

Received webhook payload: {'event': 'invoice\_completed', 'data': {'message': 'Ping from Webhook Service'}}

Webhook received: {'event\_type': 'invoice\_completed', 'data': {'message': 'Ping from Webhook Service'}, 'received\_at': '2025-05-08T08:21:39.694107+00:00'}

INFO: 85.82.70.165:0 - "POST /webhook HTTP/1.1" 200 OK

In the HTML site I made for seeing the webhook was being triggered, on the URL:  
“https://integrator-webhook.loca.lt/received-events”

You could see that the registered events had been triggered:

## Webhook Dashboard

All Events

Payment Received

Payment Processed

Invoice Processing

Invoice Completed

### Received Webhook Events

Event Type	Data	Received At
payment_received	{"message": "Ping from Webhook Service"}	2025-05-08T08:21:36.955168+00:00
payment_processed	{"message": "Ping from Webhook Service"}	2025-05-08T08:21:37.923578+00:00
invoice_processing	{"message": "Ping from Webhook Service"}	2025-05-08T08:21:38.797720+00:00
invoice_completed	{"message": "Ping from Webhook Service"}	2025-05-08T08:21:39.694107+00:00

## 4. Unregister webhooks.

I then used the unregister endpoint to unregister the nonexistent webhook endpoint, which can be seen here:

12a\_Exposee\_and\_integrate\_with\_a\_webhook\_system / Oliver exposee Webhook system / Unregister Invalid Webhook

POST {{base\_url}}/unregister

Params Authorization Headers (10) Body Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON**

```
1 {
2   "event": "payment_received",
3   "url": "https://this-needs-to-fail.it/webhook"
4 }
```

Body Cookies Headers (4) Test Results

204 No Content

Key	Value
Content-Type	application/json
Date	Thu, 08 May 2025 08:27:10 GMT
Ngrok-Agent-ips	85.82.70.165
Server	uvicorn

I then pinged the registered webhooks using Postman and I got only the successful ones:

12a\_Exposee\_and\_integrate\_with\_a\_webhook\_system / Oliver exposee Webhook system / Ping All Webhooks

GET {{base\_url}}/ping

Params Authorization Headers (7) Body Scripts Settings

Body Cookies Headers (5) Test Results

{ } JSON Preview Visualize

```
1 {
2   "successful_webhooks_count": 4,
3   "successful_webhooks": [
4     {
5       "event": "payment_received",
6       "url": "https://integrator-webhook.local.it/webhook",
7       "payload": {
8         "message": "Webhook received successfully"
9       },
10      "status_code": 200
11    },
12    {
13      "event": "payment_processed",
14      "url": "https://integrator-webhook.local.it/webhook",
15      "payload": {
16        "message": "Webhook received successfully"
17      },
18      "status_code": 200
19    },
20    {
21      "event": "invoice_processing",
22      "url": "https://integrator-webhook.local.it/webhook",
23      "payload": {
24        "message": "Webhook received successfully"
25      },
26      "status_code": 200
27    },
28    {
29      "event": "invoice_completed",
30      "url": "https://integrator-webhook.local.it/webhook",
31      "payload": {
32        "message": "Webhook received successfully"
33      },
34      "status_code": 200
35    }
36  ],
37   "failed_webhooks_count": 0,
38   "failed_webhooks": [],
39   "message": "Pinged all registered webhooks successfully."
40 }
```

I then unregistered all the webhooks using Postman and tried to ping them all using Postman again, and I got this:

12a\_Exposee\_and\_integrate\_with\_a\_webhook\_system / Oliver exposee Webhook system / Ping All Webhooks

GET {{base\_url}}/ping

Params Authorization Headers (7) Body Scripts Settings

Body Cookies Headers (5) Test Results

{ } JSON Preview Visualize

```
1 {
2   "successful_webhooks_count": 0,
3   "successful_webhooks": [],
4   "failed_webhooks_count": 0,
5   "failed_webhooks": [],
6   "message": "Pinged all registered webhooks successfully."
7 }
```