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 <u>URL</u>. 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:

```
import requests

You, yesterday * Started getting ready to integrate with Olivers.

**expose_base_unl = "https://date=85-82-78-165.ngnok_free.app"

integrator_base_unl = "https://date=85-82-78-165.ngnok_free.app"

**subscriptions = [

**event*: "payment_neceived", "unl": f"(integrator_base_unl)/webhook"),

**event*: "payment_neceived", "unl": f"(integrator_base_unl)/webhook"),

**event*: "payment_neceived", "unl": f"(integrator_base_unl)/webhook"),

**event*: "payment_neceived", "unl": f"(integrator_base_unl)/webhook"),

**event*: "payment_neceived", "unl": f*(integrator_base_unl)/webhook"),

**event*: "invoice_completed", "unl": f*(integrator_base_unl)/webhook"),

**event*: "invoice_completed", "unl": f*(integrator_base_unl)/webhook"),

**event*: "invoice_completed", "unl": f*(integrator_base_unl)/webhook"),

**ferouse.end*: f*(integrator_base_unl)/webhook"),

**event*: "invoice_nocessing", "unl": f*(integrator_base_unl)/webhook"),

**ferouse.end*: f*(integrator_base_unl)/webhook",

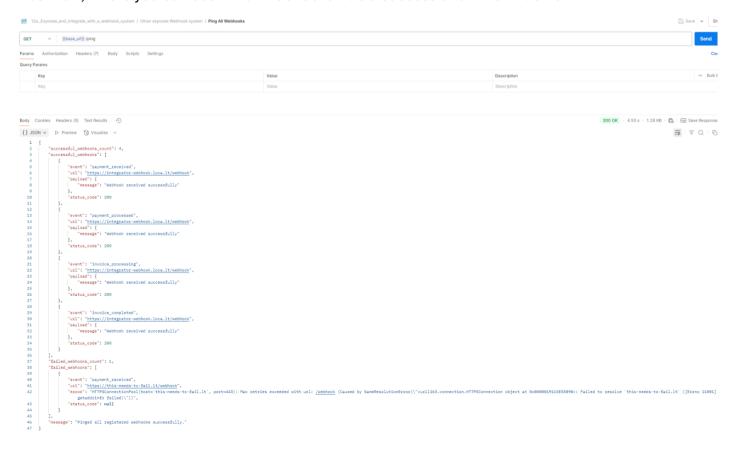
**ferouse.end*:
```

I then used Postman to register a none existent Webhook URL, which can be seen here:

12a. Exposee and integrate with a webhook system / Oliver exposee Webhook system / Register Webhook - Invalid URL POST {{base_url}} /register Authorization Headers (10) Params Body • Scripts Settings O none O form-data O x-www-form-urlencoded O raw O binary O GraphQL JSON ∨ 1 "event": "payment_received", 2 3 "url": "https://this-needs-to-fail.lt/webhook" 4 Body Cookies Headers (5) Test Results {} JSON ✓ ▷ Preview 🍪 Visualize ✓ 1 "message": "Webhook registered successfully", 2 "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:



And I got this in the terminal running the webhook_receiver project:

```
from fastapi.middleware.cors import CORSMiddleware
         from fastapi import FastAPI, Request, Query
         from fastapi.templating import Jinja2Templates
         from fastapi.staticfiles import StaticFiles
         from typing import Dict, List, Optional
         from datetime import datetime, timezone
        app = FastAPI()
         CORS_SETTINGS = {
               "allow_origins": ["*"], # Allow all origins
               "allow_methods": ["*"], # Allow all HTTP methods
"allow_headers": ["*"], # Allow all headers
         app.add_middleware(CORSMiddleware, **CORS_SETTINGS)
         templates = Jinja2Templates(directory="templates")
         app.mount("/static", StaticFiles(directory="static"), name="static")
         received events: List[Dict] = [] # To store all received webhook events
         @app.post("/webhook")
         async def webhook_endpoint(request: Request):
               A simple webhook endpoint that logs the received payload and stores them in the in-memory database.
              payload: Dict = await request.json()
              print(f"Received webhook payload: {payload}")
              event_type = payload.get("event", "unknown")
               event_data = payload.get("data", {})
               event = {
                     "event_type": event_type,
                    "data": event_data,
                     "received_at": datetime.now(timezone.utc).isoformat()
               received events.append(event)
               print(f"Webhook received: {event}")
               return {"message": "Webhook received successfully"}
             OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
            Application startup complete.
Webhook received: {'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
"POST /webhook HTTP/1.1" 200 Ok
            85.82.70.165:0 -
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

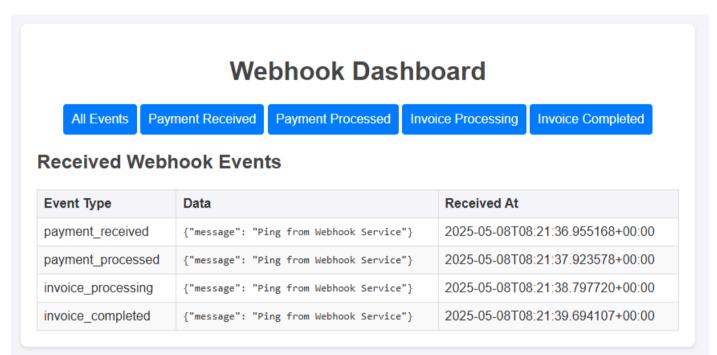
Received webhook payload: {'event': 'invoice_completed', 'data': {'message': 'Ping from Webhook Service'}}

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
```

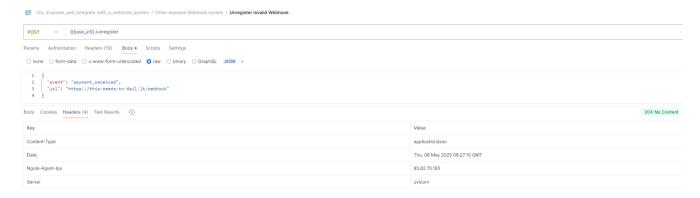
In the HTML site I made for seeing the webhook was being triggered, on the URL: "https://integrator-webhook.loca.lt/received-events"

You could see that the registered events had been triggered:



4. Unregister webhooks.

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



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
Params Authorization Headers (7) Body Scripts Settings
Body Cookies Headers (5) Test Results 4
{} JSON V Preview 🖔 Visualize V
               "successful_webhooks_count": 4,
"successful_webhooks": [
                          "event": "payment_received",
"url": "https://integrator-webhook.loca.lt/webhook",
"payload": {
"message": "Webhook received successfully"
                           },
"status_code": 200
                           "event": "payment_processed",
"url": "https://integrator-webhook.loca.lt/webhook",
"payload": {
                                  "message": "Webhook received successfully"
                          "event": "invoice_processing",
"url": "https://integrator-webhook.loca.lt/webhook",
"payload": {
                                  "message": "Webhook received successfully"
                             r,
'status_code": 200
                          "event": "invoice_completed",
"url": "https://integrator-webhook.loca.lt/webhook",
"payload": {
                                  "message": "Webhook received successfully"
                           },
"status_code": 200
               ],
"failed_webhooks_count": 0,
"failed_webhooks": [],
"message": "Pinged all registered webhooks successfully."
```

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

