Hi there,

Thanks for applying for our Python Developer role.

To move forward, please complete this short questionnaire + mini task and submit today.

Deadline: 23:59 PKT (UTC+5) — 14 January 2026 (that's 20:59 CET — 14 January 2026)

Reply to this email with your answers (copy/paste and keep the numbering). You may include links (GitHub, docs, Loom).

---

1) Quick Info (required)

1. Full name: Mehdia Humais

2. Location + time zone: Pakistan + Time zone in Pakistan (GMT+5)

3. English level (A2/B1/B2/C1/C2): B2

4. Available full-time (6–8 hours/day)? ☑ Yes ☐ No

5. Start date: Immediately

---

2) Core Skills (tick all that apply)
A) Python & Backend

☑ Python (scripting/automation)

☑ Python (FastAPI / Flask / Django)

☑ Async Python (asyncio / websockets / aiohttp)

☐ Background jobs (Celery / RQ / cron)

☑ Testing (pytest)

☑ Logging/monitoring (structured logs, Sentry, etc.)

B) Databases

☑ PostgreSQL

☐ MySQL/MariaDB

☐ SQLite

☑ MongoDB

☑ Redis

☐ Schema design (relations, indexes)

☐ Migrations (Alembic / Django migrations)

☐ ORM (SQLAlchemy / Django ORM)

C) APIs & Webhooks

☑ REST API integration

☑ Webhooks (receive/verify/retry/idempotency)

☑ OAuth2 (Google, etc.)

☐ Rate limits & backoff handling

☐ Queue-based processing (Kafka/RabbitMQ/Redis queues)

D) Provider Integrations (experience)

Tick what you've integrated before (via API/webhooks):

☑ Telegram Bots (Bot API)

☑ Telegram MTProto (advanced)

☑ Notion API

☑ Time Doctor API

☑ Google APIs (Drive/Sheets/Gmail/Calendar)

☑ OpenAI API

☑ Stripe / PayPal

☑ Web scraping (legal/ethical)

☑ Other: _____

E) Infrastructure / DevOps (nice-to-have)

☐ Linux server administration

☐ Nginx/Apache reverse proxy

☑ Docker / docker-compose

☑ CI/CD (GitHub Actions/GitLab CI)

☑ Virtual Machines (Proxmox / VMware / VirtualBox / KVM)

☐ VDS hardening (firewall, SSH keys, fail2ban)

☑ Secrets management (env vars, vault-style approach)

☐ Backups & restore tests

F) PHP (optional)

☑ PHP (basic)

☑ PHP (Laravel / frameworks)

☐ Maintaining legacy PHP apps

---

3) Practical Experience (short answers)

6. Describe 2–3 automation/bot projects you built (what it did, stack, scale).

7. Have you built systems that ingest webhooks + write to DB + trigger workflows?   ☑ Yes
    ☐ No

   If yes: describe briefly.

8. What's your approach to reliability? (retries, idempotency keys, deduplication, monitoring)

9. Have you worked with Telegram bots? ☐ Yes ☑No

○ Ifyes: what features? (commands, inline keyboards, payments, webhooks, groups, etc.)

10. Areyoucomfortable maintaining a VDS running multiple VMs (security + uptime)? ☑ Yes ☐ No

   ○ Ifyes: which tooling (Proxmox, Docker, systemd, etc.)?

---

4) Availability & Output

11.In an 8-hour day, what's a realistic output for you?

- ☑ Small bugfixes + integrations (2–5)

- ☑ One medium feature end-to-end (1)

- ☑ Mainly maintenance/on-call + 1–2 improvements

12.Preferred working style:

- ☑ Daily standup + daily commits

- ☑ Task-based delivery with end-of-day report

- ☐ Both

---

5) Mini Task (required)

Please choose ONE of the options below and submit today:

Option A — Webhook Receiver + DB

Build a small service (FastAPI or Flask) that:

- Receives a webhook POST /webhook

- Validates a shared secret header (e.g., X-Signature)

- Stores the payload in a database (SQLite/Postgres)

- Implements idempotency (do not store duplicates)

- Returns a clean JSON response

    Include: README + how to run locally.

Option B — Telegram Bot + Provider API

Build a Telegram bot that:

- Accepts a command like /note <text>

- Writes the text to a database

- (Optional bonus) pushes it to Notion via API

    Include: README + env var setup (no secrets in code).

Deliverable: GitHub repo link OR zipped code + README + brief explanation of architecture.

---

6) Portfolio / Links

13. GitHub / GitLab: https://github.com/MehdiaHumais/

14. Any live projects or code samples: https://britsyncaiacademy.online/

15. LinkedIn (optional):

---

7) Final Confirmation

16. Confirm you can work full-time (6–8 hours/day) for the next 3+ months. ☑ Yes ☐ No

17. Confirm you can submit this questionnaire + mini task today by the deadline above. ☑ Yes ☐ No

18.Also we can pay you 40k-50k PKR. Are you interested ☑ Yes ☐ No

---

If your submission is complete and on time, we'll reply with next steps.

Thank you,

7. Have you built systems that ingest webhooks + write to DB + trigger workflows?
☑ Yes
If yes: describe briefly:
"Yes, I developed an AI Call Agent using Twilio and ElevenLabs. The system ingests real-time webhooks from Twilio
track call status, writes conversation logs and user intent to a Supabase (PostgreSQL) database, and triggers
downstream workflows (like voice generation or follow-up tasks) based on the caller's input."

8. What's your approach to reliability? (retries, idempotency keys, deduplication, monitoring)
Retries: I implement exponential backoff for failed webhook deliveries or API calls (e.g., using
Python's tenacity library) to handle temporary outages without overloading the server.
Idempotency Keys: I use unique identifiers (like X-Request-ID or Twilio's CallSid) to cache
processed events. Before processing, I check if the key exists in Redis or a Supabase table to
prevent double-execution.
Deduplication: I enforce unique constraints on database columns for critical data (like transaction
IDs) to ensure that even if a retry happens, duplicate records are rejected at the database level.
Monitoring: I use structured logging to track webhook statuses and set up alerts (via Sentry or simple
Slack notifications) for high failure rates or 5xx errors.

Option A — Webhook Receiver + DB
https://github.com/Ali-Hamas/webhook-mini-task