Full-Stack Assignment

Title: Event → Queue → Push (Redis • Webhook • Firebase)

Timebox: 1 working day

Objective

Build a tiny pipeline that accepts third-party **webhooks**, queues jobs in **Redis**, and sends **Firebase Cloud Messaging (FCM)** notifications. Include a minimal Next.js dashboard to inspect and replay events.

Scope & Requirements

1) Webhook Ingest (API)

- Endpoint: POST /api/webhook/order.created

- Verify **HMAC-SHA256** signature header X-Signature over the **raw body** using WEBHOOK_SECRET.

- Enforce X-Timestamp (reject if older than **5 minutes**).

- **Idempotency**: use event_id to ignore duplicates for **24h**.

- Enqueue to **Redis** (orders-stream via Streams, or a List if you prefer).

- Respond fast (**≤300 ms**) after enqueue.

- **Rate limit**: 10 requests / 10s per IP (429 on overflow).

2) Worker

- Consume events from Redis (Streams + Consumer Group preferred).

- Enrich: map userId → device token from Redis key fcm:token:<userId>.

- Send **FCM** via Firebase Admin:

  - Title: New Order

  - Body: Order <order_id> placed by <userId>

  - Target: device token (fallback to topic orders if none).

- Retries with exponential backoff (e.g., 1s → 4s → 10s); after 3 failures, move to **DLQ** (orders-dlq).

3) Mini Dashboard (Next.js)

- Route: /admin/events

- Show last 20 events with status: queued | processing | sent | failed.

- **Replay** button to requeue a failed event.

- Search by event_id.

- Health endpoints:

  - GET /api/health → { ok: true, redis: "up" }

  - GET /api/metrics → counters: received, deduped, sent, failed, dlq

4) Security & Ops

- Use **raw** request body for HMAC (no stringify/parse before hashing).

- .env for secrets: WEBHOOK_SECRET, Firebase creds, Redis URL.

- FCM_DRY_RUN=true should **log** instead of sending real pushes.

- No heavy queue libs (Bull/BullMQ). Implement minimal logic yourself.

Sample Payload & Headers

```
{
 "event_id": "evt_12345",
 "type": "order.created",
```

  "data": { "order_id": "ord_987", "userId": "user_42", "amount": 1599 }
}

Headers:

X-Signature: <hex(hmac_sha256(WEBHOOK_SECRET, raw_body))>
X-Timestamp: <unix seconds>

Deliverables

1. **GitHub repo** with:

   ○ /README.md — setup, run, env vars, **ASCII diagram** of flow.

   ○ API routes (/api/webhook/*, /api/health, /api/metrics).

   ○ Worker script (/worker.ts) with run instructions (npm run worker).

   ○ Next.js admin page /admin/events with Replay + search.

2. **Postman collection or cURL** examples:

   ○ Valid signed webhook

   ○ Duplicate webhook (shows dedupe)

   ○ Replay failed event

3. **Tests (Jest)** for:

   ○ Signature verification (good/bad)

   ○ Timestamp window

   ○ Idempotency (duplicate event_id)

   ○ Retry backoff calculation

Evaluation Rubric

- **Security & Correctness:** raw-body HMAC, timestamp, idempotency.

- **Queue Design:** Redis use, retries, DLQ, consumer logic.

- **Code Quality:** types, structure, small functions, error handling.

- **Observability & UX:** logs, metrics, clear dashboard & replay.

- **Scope Discipline:** delivers the required parts within a day.

Bonus (optional, nice-to-have)

- Redis **Streams Consumer Group** with XACK/XCLAIM.

- Docker + Compose.

- GitHub Actions for tests on PR.