

Les 17 — Huiswerk

Webhook + Resend email + retry + APIS.md

Deadline: Voor Les 18 — Supabase Auth + RLS
Inleveren: GitHub repo + APIS.md in root
Tijd: ~2 uur

Doel

Bouw je Stripe-demo uit naar productie-flow: betaling triggert webhook -> database update -> welkomstmail. Plus retry-logic en eindopdracht-reflectie.

A

Webhook handler met signature verify (verplicht)

Stripe CLI installeren + lokaal luisteren:

```
brew install stripe/stripe-cli/stripe
stripe login
stripe listen --forward-to localhost:3000/api/webhook/stripe
```

Webhook endpoint

```
import { headers } from "next/headers";
import Stripe from "stripe";

const stripe = new Stripe(process.env.STRIPE_SECRET_KEY!);

export async function POST(req: Request) {
  const body = await req.text();
  const sig = (await headers()).get("stripe-signature")!;
  let event: Stripe.Event;
  try {
    event = stripe.webhooks.constructEvent(
      body, sig, process.env.STRIPE_WEBHOOK_SECRET!
    );
  } catch (err: any) {
    return new Response(err.message, { status: 400 });
  }
  console.log("Event:", event.type);
  return Response.json({ received: true });
}
```

Eisen:

- Webhook endpoint werkt
- Signature verification actief
- checkout.session.completed event wordt herkend
- Console toont event-data

B Resend welkomstmail (verplicht)

```
pnpm add resend  
# RESEND_API_KEY in .env.local
```

```
import { Resend } from "resend";  
const resend = new Resend(process.env.RESEND_API_KEY!);  
  
// In webhook handler, na event.type === "checkout.session.completed":  
await resend.emails.send({  
  from: "Premium <onboarding@resend.dev>",  
  to: session.customer_email!,  
  subject: "Welkom bij Premium!",  
  html: "<h1>Welkom!</h1><p>Bedankt voor je aanmelding.</p>",  
});
```

Eisen:

- Email wordt verstuurd bij succesvolle betaling
- Email komt aan (check je inbox)
- Resend dashboard toont email in logs

C Retry logic (verplicht)

```
export async function retry<T>(  
  fn: () => Promise<T>,  
  attempts = 3,  
  baseDelayMs = 1000  
) : Promise<T> {  
  for (let i = 0; i < attempts; i++) {  
    try { return await fn(); }  
    catch (err) {  
      if (i === attempts - 1) throw err;  
      const delay = baseDelayMs * Math.pow(2, i);  
      console.log(`Retry ${i + 1}/${attempts} na ${delay}ms`);  
      await new Promise(r => setTimeout(r, delay));  
    }  
  }  
  throw new Error("unreachable");  
}
```

Eisen:

- retry() helper bestaat en werkt
- Wordt gebruikt op minimaal 1 externe call

-
- Log toont retries gebeuren
 - In APIS.md: 1 voorbeeld van failed call + retry succes

D APIS.md (verplicht)

In repo-root. Vijf secties:

1. **Welke APIs jouw app gebruikt** — tabel met auth + cost
2. **Webhook setup** — Stripe CLI + endpoint code + verify screenshot
3. **Voorbeeld successful flow** — end-to-end met screenshots
4. **Failed call met retry** — voorbeeld log van retry succes
5. **Eindopdracht-toepassing** — welke API ga je gebruiken? Webhook? Email-flows? (min 200 woorden)

E Bonus (optioneel)

- OAuth login via Auth.js (GitHub)
- Idempotency check in webhook (event.id in Supabase)
- Rate limiting met Upstash op /api/checkout
- Refund flow — charge.refunded event handler

F Beoordeling (10 pt — voldoende 6+)

Criterium	Punten
A — Webhook + signature verify	3
B — Resend email bij betaling	2
C — Retry helper + bewijs	2
D — APIS.md compleet (5 secties)	3
Totaal	10

Veelvoorkomende valkuilen

- Signature verify faalt: gebruik req.text() niet req.json()
- Email komt niet aan: check spam, Resend logs
- Stripe CLI 401: stripe login opnieuw
- Webhook secret verkeerd: nieuwe secret per `stripe listen` sessie