Get Perk’d Supabase Trigger Logic – Phase 3 Automation

This document outlines the Supabase PostgreSQL trigger logic and functions to automate notifications within the Get Perk’d system. These triggers enable event-based messaging for students, merchants, and admins based on user interactions and milestones.

# 🎯 Student Notification Triggers

Trigger: new\_deal\_insert  
- Event: A new deal is added to the database  
- Action: Notify all students within the associated campus/city  
- Delivery: Push alert + optional email  
  
Trigger: reward\_tier\_update  
- Event: A student’s reward tier changes (e.g., Bronze → Silver)  
- Action: Notify the student of their milestone and next goal  
- Delivery: Push + badge UI refresh  
  
Trigger: deal\_redeemed  
- Event: A student redeems a deal  
- Action: Log redemption, check for streaks or milestones  
- Delivery: Milestone push + reward tracker update  
  
Trigger: saved\_deal\_expiry  
- Event: Saved deal reaches expiry window (e.g., 2 hours left)  
- Action: Notify student

# 🏪 Merchant Notification Triggers

Trigger: redemption\_log\_insert  
- Event: A redemption is recorded  
- Action: Notify the associated merchant  
- Delivery: Email via SendGrid  
  
Trigger: deal\_count\_check  
- Event: Daily check (CRON) finds 0 active deals  
- Action: Notify merchant to re-engage  
  
Trigger: redemption\_threshold  
- Event: A merchant hits 10/50/100 redemptions  
- Action: Notify merchant of milestone

# 🛠️ Admin Notification Triggers

Trigger: merchant\_inactivity  
- Event: 7 days pass with no redemptions  
- Action: Notify admin for outreach  
  
Trigger: student\_signup\_milestone  
- Event: 1000, 5000, etc. student signups  
- Action: Notify admin and log  
  
Trigger: abuse\_report  
- Event: Student flags abuse or redemption failure  
- Action: Notify admin immediately

# 🔧 Supabase Notes

- Use Postgres TRIGGERS + FUNCTIONS for low-latency alerts  
- Log all notifications in a notifications table with columns:  
 (recipient\_id, role, type, content, status, timestamp)  
- Use Supabase Edge Functions for advanced logic (SendGrid/Twilio integration)  
- Use CRON jobs via Supabase for scheduled checks