

Project description and technical proposal request prompt by the user:

In KSA everybody pays using Stripe. Almost no one there uses bank transfers or PayPal. As a digital nomad I travel everywhere, hence I mostly use PayPal when I'm traveling abroad, otherwise I use the bank account when I'm in my hometown Egypt.

I need an AI agent that routes payments from stripe to my PayPal or bank account depending on my location.

Technical Proposal

Project Title: Intelligent Payment Routing System

Freelancer: Khalid Salim

Client: Ahmed Fouda

Date: 07-09-2025

1. Executive Summary

This proposal outlines a technical solution to automate the routing of Stripe payments to either a PayPal account or a local bank account, depending on the client's current location. The system will use a lightweight AI-based agent or rules engine to determine the optimal payout method, improving payment flexibility, minimizing fees, and adapting to the client's nomadic lifestyle. The solution ensures seamless integration with Stripe's payout system, providing a robust, secure, and location-aware financial automation tool.

2. Introduction and Background

The client operates in multiple countries, primarily receiving payments via **Stripe**, with **PayPal** and **bank accounts** serving as the preferred payout options depending on geographic location. Currently, there's no streamlined method to automatically redirect incoming Stripe payments based on the client's location.

Challenges:

- Stripe does not natively support automated routing to multiple payout destinations.
 - Manual transfers can lead to delays, higher fees, and potential currency exchange losses.
 - The client requires a flexible, intelligent payment pipeline that adapts in real time.
-

3. Proposed Solution & Add Value

Overview

I propose to build a **location-aware payment routing agent** that:

- Monitors incoming payments via Stripe.
- Detects the user's current location (through IP, GPS API, or manual override).
- Automatically triggers payouts to either PayPal or a connected bank account.

Architecture & Flow:

1. **Stripe Webhook Listener:** Listens for successful payment events.
2. **Location Detection Module:** Uses an API (e.g., IPStack or GeoIP) or allows manual input for greater control.
3. **Decision Engine:** Determines payout method based on rules:
 - If in Egypt → Route to Bank.
 - If abroad → Route to PayPal.
4. **Payout Executor:**
 - Triggers PayPal Payout API or sends bank transfer via a third-party processor (e.g., Wise, Stripe Connect with Custom Accounts).
5. **Dashboard Interface (Optional):** A web portal to view payment logs, manage payout rules, and manually override location if needed.

Value Added:

- **Saves time and manual effort.**
- **Reduces fees** by automating optimal route decisions.
- **Improves cash flow** consistency.
- Scalable for future integrations (e.g., crypto, Wise, Revolut, etc.).

4. Timeline and Deliverables

Week	Milestone	Deliverables
1	Requirements Finalization & System Architecture	- Final specs document- API access confirmation
2	Stripe Webhook Integration & Location Detection Module	- Live webhook endpoint- Working location module
3	Payment Routing Logic & API Integration (PayPal, Bank/Transfer API)	- Working routing prototype- Automated payout test
4	Dashboard (Optional) & Final Testing	- Admin dashboard (if included)- Logs, error handling
5	Deployment & Handover	- Production deployment- Documentation & training

5. Resources

Tools & Technologies:

- **Languages:** Node.js / Python
 - **Frameworks:** Express / Flask (for backend), React (for dashboard)
 - **APIs:** Stripe API, PayPal Payouts API, IPStack/GeoIP
 - **Hosting:** AWS / Vercel / Heroku (based on budget)
 - **Security:** HTTPS, API key management, OAuth2 for PayPal
-

6. Technical Specifications

- **Stripe Integration:** Webhook-based architecture for real-time payment monitoring.
 - **Location Module:** API-based location detection, fallback manual mode.
 - **Routing Rules Engine:** JSON-based config for easy logic modification.
 - **Payout System:**
 - **PayPal:** Via PayPal Payouts API.
 - **Bank:** Through bank API if available (e.g., Wise or a fintech API). Stripe does not natively transfer to external bank accounts without a payout account.
 - **Error Handling:** Retry logic, fallback notification via email or Telegram.
 - **Data Security:** Encrypted storage of keys, 2FA for dashboard access.
-

7. Terms and Conditions

- **Project Duration:** 4–5 weeks (based on timely feedback).
 - **Payment Terms:** Milestone-based payments via Upwork.
 - **Revisions:** Up to 2 rounds of revisions per phase.
 - **Data Confidentiality:** All financial and personal data will be handled securely.
 - **Post-Deployment Support:** 2 weeks of bug fixes included. Optional maintenance plan available.
-

8. Conclusion and Call to Action

This solution addresses your core need for payment flexibility in a global lifestyle. By automating Stripe-to-PayPal/Bank transfers based on your location, you gain control, reduce fees, and simplify operations.

If this proposal aligns with your vision, I'd be happy to proceed with a short call to finalize details and kick off development.

Let's make your payments work for you—no matter where you are.

Financial Proposal

Project Title: Intelligent Payment Routing System

Freelancer: Khalid Salim

Client: Ahmed Fouda

Date: 07-09-2025

1. Value & Benefits of the Proposed Solution

As a digital nomad, your income depends on smooth, timely, and cost-efficient access to funds. This solution automates the routing of payments from **Stripe** to either **PayPal** or a **local bank account**, based on your **real-time location**. Here's how it benefits you:

- ✔ **No manual transfers** — the system handles payouts automatically.
- ✔ **Save on transaction fees** — choose the most cost-efficient route per location.
- ✔ **Cash flow consistency** — no delays in accessing your funds, wherever you are.
- ✔ **Built for scale** — future-ready for additional payment methods (e.g., Wise, Revolut, crypto).
- ✔ **Security-first design** — uses secure APIs and encryption for your financial data.

This isn't just code — it's a **personal financial automation assistant** designed to match your lifestyle.

2. Tailored Cost Breakdown

Component	Hours Est.	Rate (USD/hr)	Subtotal
Requirements Analysis & Planning	6	\$17.5	\$105
Stripe Webhook + Listener Integration	10	\$17.5	\$175
Location Detection Module	6	\$17.5	\$105
Payout Logic & Routing Engine	8	\$17.5	\$140
PayPal Payouts API Integration	8	\$17.5	\$140
Bank Routing (via Wise or equivalent)	6	\$17.5	\$105
Logging & Notification System	4	\$17.5	\$70
Optional Admin Dashboard (React-based)	10	\$17.5	\$175

Component	Hours Est.	Rate (USD/hr)	Subtotal
Testing, QA & Final Optimization	6	\$17.5	\$105
Deployment, Documentation & Handover	4	\$17.5	\$70
Total Estimated Hours	68		
Total Estimated Project Cost			\$1,190

Without the Dashboard: \$1,015

With the Dashboard: \$1,190

3. Transparency and Flexibility

What's Included:

- Full backend setup with real-time routing
- Stripe + PayPal + Bank integration
- Location-based rule engine
- Error handling and basic logging
- Optional web dashboard for control and insights
- 2 weeks of post-launch support (bug fixes and assistance)

Potential Variables (if applicable):

- Third-party API charges (e.g., IPStack for geolocation, Wise transfer fees)
- Extra hours for advanced dashboard features (filters, analytics, notifications)
- Future add-ons or currency/localization features (billed separately)

All scope adjustments will be **discussed and approved before any work begins**.

4. Payment Terms & Milestones







Milestone	Amount (USD)	Due
Milestone 1 – Architecture + Stripe Setup	\$400	End of Week 1
Milestone 2 – Logic & API Integrations	\$600	End of Week 2

Milestone	Amount (USD)	Due
Milestone 3 – Dashboard & QA	\$700	End of Week 4
Milestone 4 – Final Handover & Support	\$680	Week 5 (Final Phase)

All payments are processed securely via **Upwork Milestones**.

5. Terms and Conditions

To protect both you and me as your developer, the following terms apply:

-  **Confidentiality:** Your financial data and personal information will be securely handled and never shared.
-  **Ownership:** You retain full rights to the codebase and solution upon final payment.
-  **Revisions:** Includes up to **2 revisions per milestone** to ensure alignment.
-  **Bug Fixes:** Covered for **2 weeks after final delivery**.
-  **Scope Changes:** Any feature not defined at project start will be discussed and billed separately.
-  **Cancellation:** Either party may cancel before Milestone 2; payment will be due only for completed work.

Next Steps

If the proposal meets your expectations, I'm ready to begin **immediately** upon milestone approval. Please confirm:

1. Do you want the optional dashboard?
2. Do you prefer geolocation via IP or want a manual override option as well?

Once confirmed, I'll send the Upwork contract and lock in the timeline.

Let's build a solution that travels as freely as you do.

Looking forward to working together,

Khalid
