Capstone LOI

Use this form to submit your LOI for Capstone Projects

Wallet Address

9rxVXJ12mBoRrtegEjjs7KBkHuYaDJ5YjrbY4ZEuyQPy

Name

Francis Ohilebo

Github Link

https://github.com/francis-codex

Your Capstone Project Name/Idea

Payclip

Brief Project Description

"Payclip" - A one-click USDC payment solution on Solana that simplifies crypto transactions for everyone.

This web3 application allows users to send and receive USDC instantly through shareable payment links, similar to PayPal or Cash App, but powered by Solana's blockchain.

The key feature is its accessibility - recipients don't need an existing crypto wallet to receive funds, making it perfect for crypto beginners.

The reason I am building/choosing this project is...

Problem & Motivation

1. Current Pain Points:
- Traditional bank transfers are slow
- High international transfer fees (3-5% average)
- Complex crypto onboarding scares newcomers
- Most crypto payment solutions require both parties to have wallets

2. Market Opportunity:
- Growing demand for instant, borderless payments
- Large untapped market of non-crypto users
- Solana's advantages aren't fully utilized in payment solutions
- Existing solutions are often too complex for average users

3. Personal Mission:
I'm building this project to bridge the gap between traditional finance and crypto, making instant payments accessible to everyone. While crypto offers amazing

benefits, the entry barrier remains too high for many. By creating a simple, one-click solution, I'm helping solve real-world payment problems while onboarding new users to web3."

4. Why Now:
- Growing mainstream crypto adoption
- Solana's maturity as a platform
- Rising demand for low-fee alternatives
- Increasing frustration with traditional banking delays

This project represents a practical solution to real-world problems while potentially driving crypto adoption through ease of use.

Go-to-Market Strategy

GMT plan for Payclip.

1. Initial Targe Market
- Start with young professionals (25-35)
- Focus on freelancers/digital nomads
- Target crypto-curious users who find current solutions complex

2. Growth Strategy

Phase 1: Launch (1-3 months)
- Beta release with limited features
- Focus on core user experience
- Free transactions for early adopters
- Build community through social media
- Collect user feedback and testimonials

Phase 2: Growth (3-6 months)
- Feature rollout based on user feedback
- Partnership with freelance platforms
- Referral program launch

Phase 3: Scale (6+ months)
- Regional expansion
- Business account features
- Integration with e-commerce platforms
- Mobile app launch
- Enterprise partnerships

3. Marketing Channels
- Crypto Twitter/X communities
- LinkedIn (for freelancers/professionals)
- Influencer partnerships

4. Revenue Model
- Free for basic transfers
- Premium features for power users
- Small fee for instant fiat off-ramp
- Business account subscriptions
- API access for integrations

5. Key Performance Indicators (KPIs)

- User acquisition cost
- Transaction volume
- Active users (daily/monthly)
- User retention rate
- Customer support satisfaction

6. Competitive Edge
- One-click simplicity
- No-wallet-needed feature
- Lower fees than traditional services
- Instant settlements
- User-friendly interface

Tech: The technologies I will use in my build are...

Frontend:
Next.js/React: For fast, SEO-friendly web app
TailwindCSS: For responsive styling
TypeScript: For type safety
Wallet Adapters/Tiplink: For crypto wallet connections
Web3 Libraries: @solana/web3.js, @solana/wallet-adapter

Backend:
Node.js/Express: Main server framework
TypeScript: For type safety
Solana Web3.js: For blockchain interactions

Smart Contract:
Rust: For Solana programs
Anchor: Framework for Solana development
SPL Token: For USDC handling

Database:
PostgreSQL: Main database
Prisma: ORM for database operations

DevOps/Infrastructure:
Vercel: Hosting
Docker: Containerization
GitHub Actions: CI/CD

Testing Tools:
Solana Test Validator: Local blockchain testing

Security:
Web3Auth: For wallet creation
SSL/TLS: For encryption
Rate Limiting: For API protection

Method: Programming language for SC / Testing

Rust/Anchor Framework

Front End Plans (Ok if you need help here)

i will build it out myself, it would not need much, because I want a slick UX

Frontend Plans:
Next.js/React: For fast, SEO-friendly web app
TailwindCSS: For responsive styling
TypeScript: For type safety
Wallet Adapters/Tiplink: For crypto wallet connections
Web3 Libraries: @solana/web3.js, @solana/wallet-adapter

Initials