

TIKTOK HACKATHON 2023

TIKIWALLET

No Internet? No problem.

Problem Statement 4:

Digital Wallets

- Building trustworthy, secure and reliable e-wallets. How can new technologies in digital wallets enable new trustworthy user experiences?
- Wallets are no longer used to simply store value. They are a medium for every type of payment, promote cross-border transactions, drive engagement and promote loyalty. While functioning as a driver of financial connectivity, the wallet revolution needs to continue to build substantial trust amongst consumers, in order to invest their hard earned cash while keeping their personal information secure. Such trust is quickly lost if stability or security or compliance is compromised.



Our Features



Authentication

- To reduce fraud and phishing using 2FA method through OTP SMS verification
- Authentication and session management is carried out using JSON Web Tokens.



Topups and Withdrawals

- Utilising Stripe's API to securely add funds from the user's bank to the wallet and to securely retract funds from the user's wallet to the user's bank

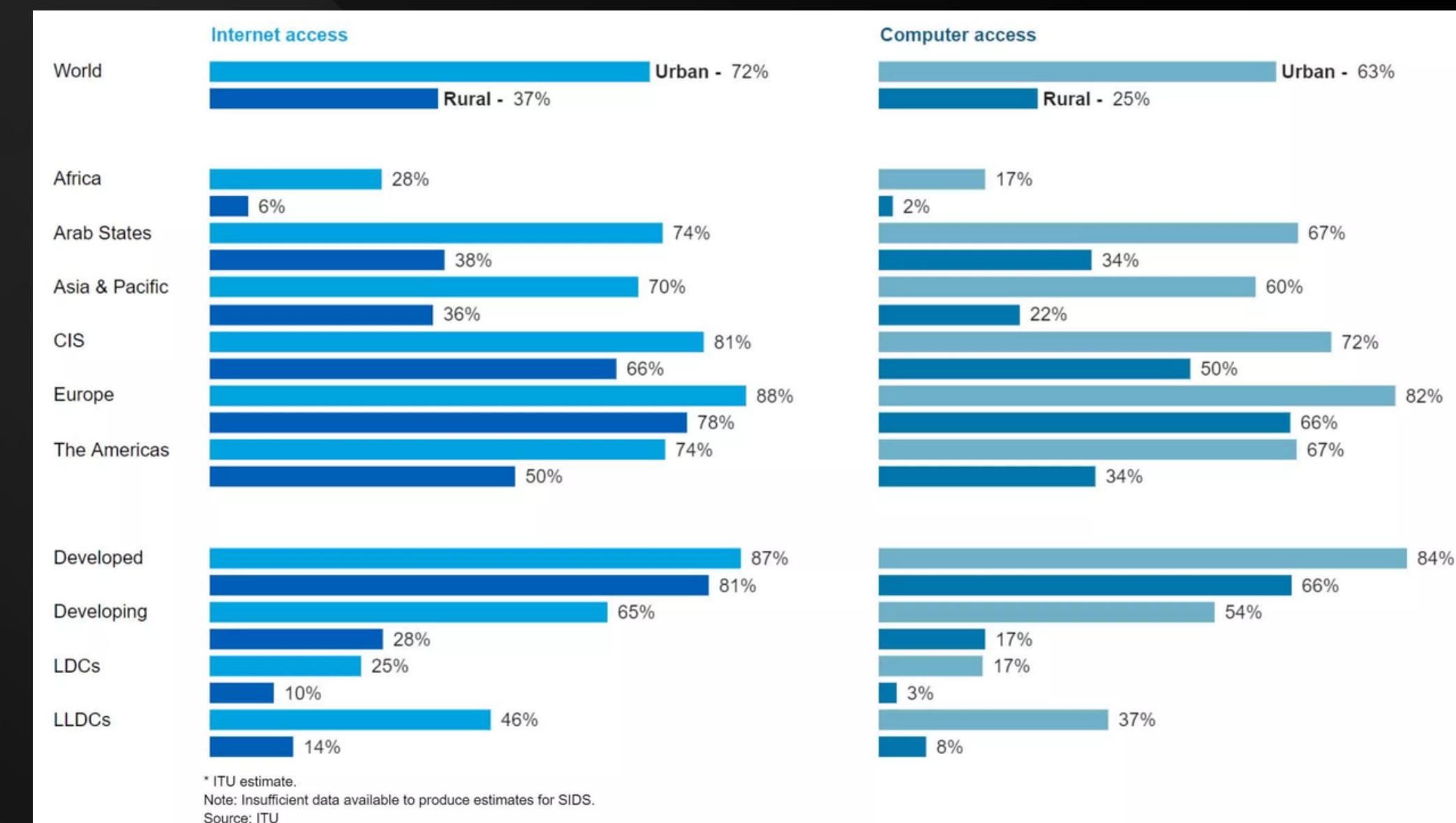


Online/Offline Wallets

- Transfer of funds between users over the internet (*online wallet*)
- Transfer of funds between users without internet access to extend payment services in areas with lack of proper internet connection. (*offline wallet*)

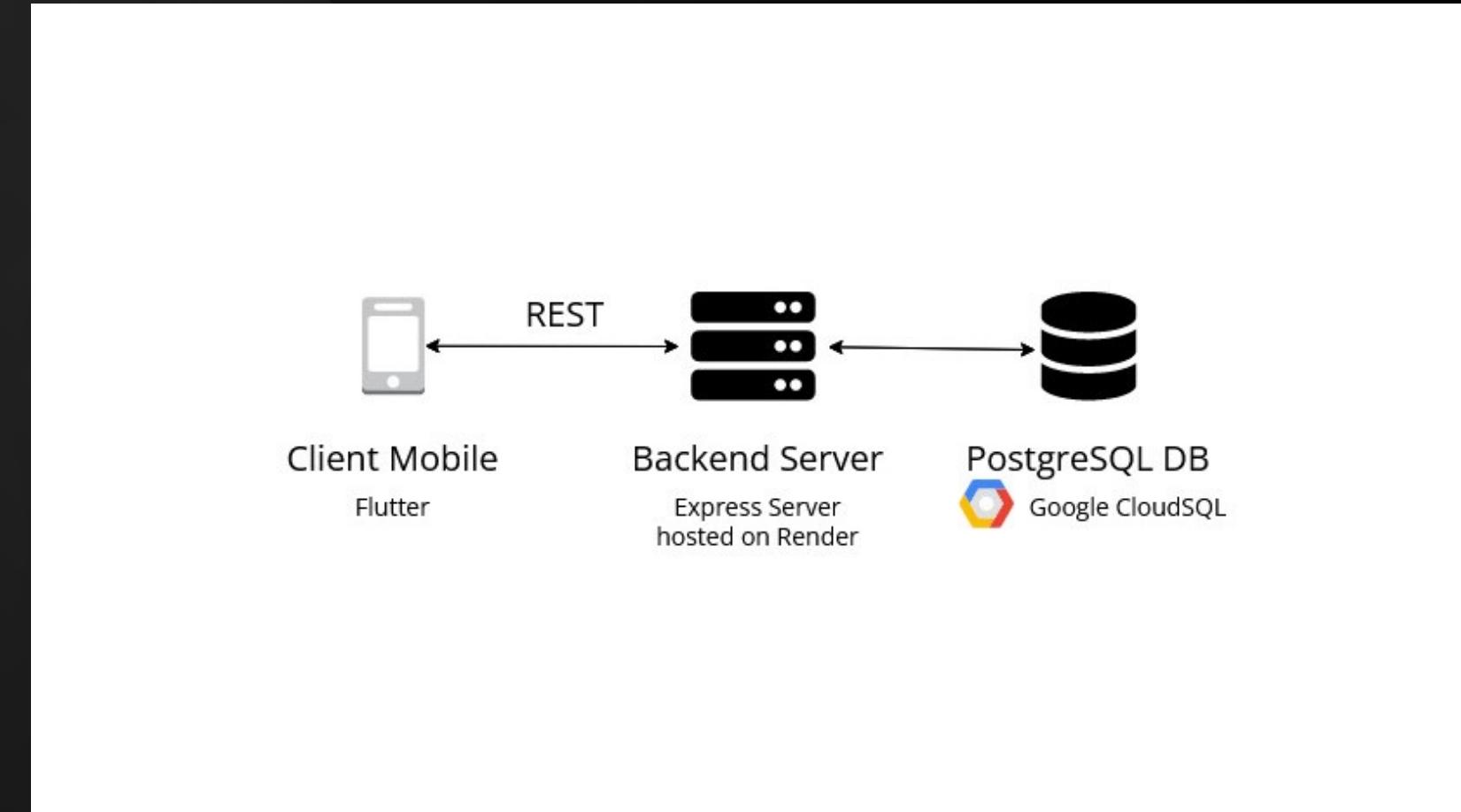
Financial Inclusion

- “Nearly 40% of rural homes globally do not have access to internet: ITU”
- In less developed countries & communities, people struggle to find constant & consistent internet connection. With a lack of stable internet access, it will be difficult to use payment services offered by regular e-wallets.
- Thus, in order to extend TikiWallet’s services to these underserved areas & communities, we decide to implement offline wallets that can be used without the internet but rather via WiFi-Direct.
- WiFi-Direct Uses device and service discovery protocols, allowing devices to search & discover nearby WiFi-enabled devices without the need of an internet connection.



TECH STACK & ARCHITECTURE

- Client:
 - Flutter
- Backend:
 - Express/NodeJS hosted on Render
 - PostgreSQL hosted on Google CloudSQL
 - Prisma ORM
- APIs used:
 - Stripe
 - Twilio
- Other Tools used:
 - Git/Github
 - Trello
 - Miro Whiteboard
 - Postman
 - VSCode



Roadmap for Scalability and Availability



01

Assessment & Planning

- **Define objectives & metrics** – what level of traffic can app handle? Acceptable downtime?
- **Understand current state of the app** – assess infrastructure, tech stack, architecture. Identify bottlenecks and failure points
- **Conduct Analysis** – analyse user patterns, peak usage times, estimate costs and budgets



02

Architecture & Design

- **Microservices Architecture** – implement auth services, transaction services
- **Load Balancing** – distribute traffic across multiple servers
- **Database scaling** – hybrid approach by partitioning data into 2 DBs: PSQL & mongoDB for accounts.
- **Caching Strategies** – use Redis to reduce database load & improve response times, allow rate limiting & throttling strategies



03

Infrastructure & Deployment

- **Cloud adoptions** – deploy onto AWS EKS
- **Auto-scaling** – configure auto-scaling policies, automatically adjust resources based on traffic. Utilise serverless computing
- **Data replication & backup** – implement across regions for disaster recovery & backup



04

Monitoring & Optimisation

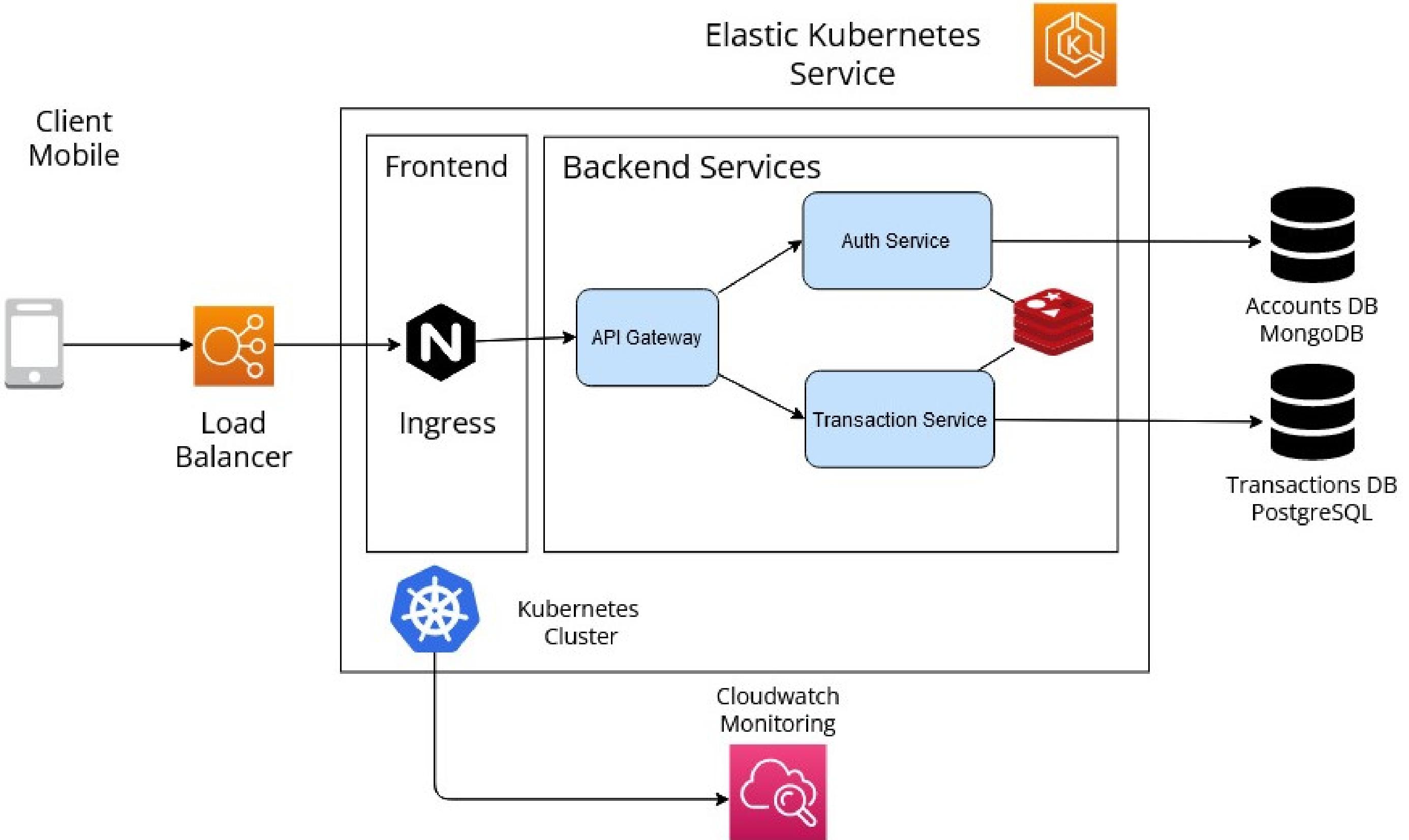
- **Real-time monitoring** – AWS cloudwatch to track application health & performance. Configure alarms for exceeding thresholds
- **Performance Testing** – Regularly conduct load testing, identify bottlenecks
- **Disaster Recovery Plan** – data replication for databases, automate cluster & disaster recovery services using AWS



05

Scaling & Growth

- **Scalability Testing** – Conduct scalability tests to ensure increased loads can be handled, especially at peak times
- **User Feedback** – continuously gather feedback & adjust strategies based on needs & expectations
- **Global expansion** – Expand the app to new geographic regions, demonstrate financial inclusion



A brief overview of the scalability and availability plan

Our team



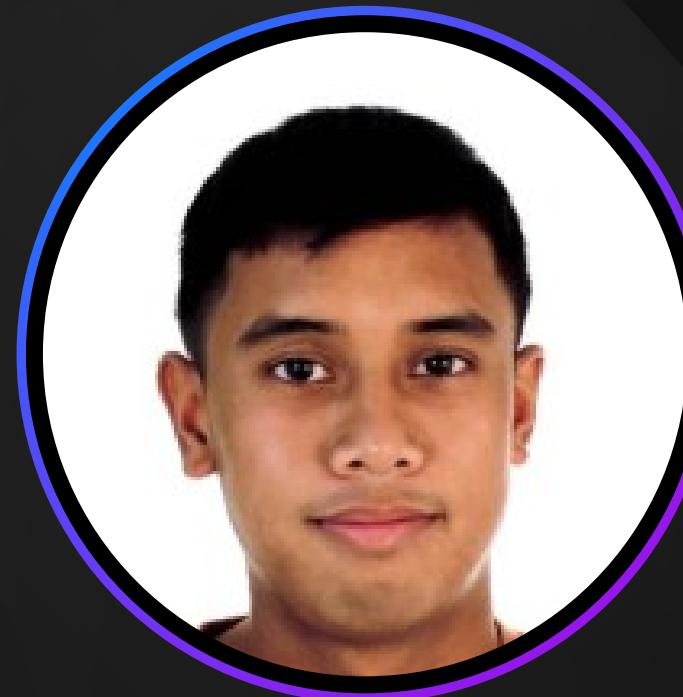
Ryan Lim
Project Lead

NUS
Information Systems



Ryan Ong
Frontend

NUS
Computer Science



Jabir
Frontend

NUS
Business Analytics



Benjamin
Backend

NUS
Computer Engineering



Yu Teng
Backend

NTU
Computer Science and Economics
Double Degree

THANK YOU

For watching this presentation