



# aws SUMMIT

INDIA | MAY 25, 2023

FS1007

# Powering multiple financial products from card transactions to wealth at Niyo with AWS

Prince Kumar  
Engineering Manager  
Niyo Solutions Inc.

Sathish Hariharan (he / him)  
Senior Solutions Architect  
AWS India



# Agenda

- Introduction to neo banking
- Architecture evolution at Niyo
- Architecture overview
- Building on AWS
- Key takeaways

# Neo banking and architecture evolution at Niyo

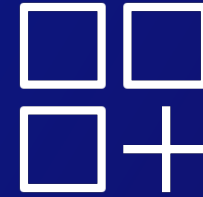
# Customers want personalized and seamless interactions with their bank



On-demand customer service via their preferred channel



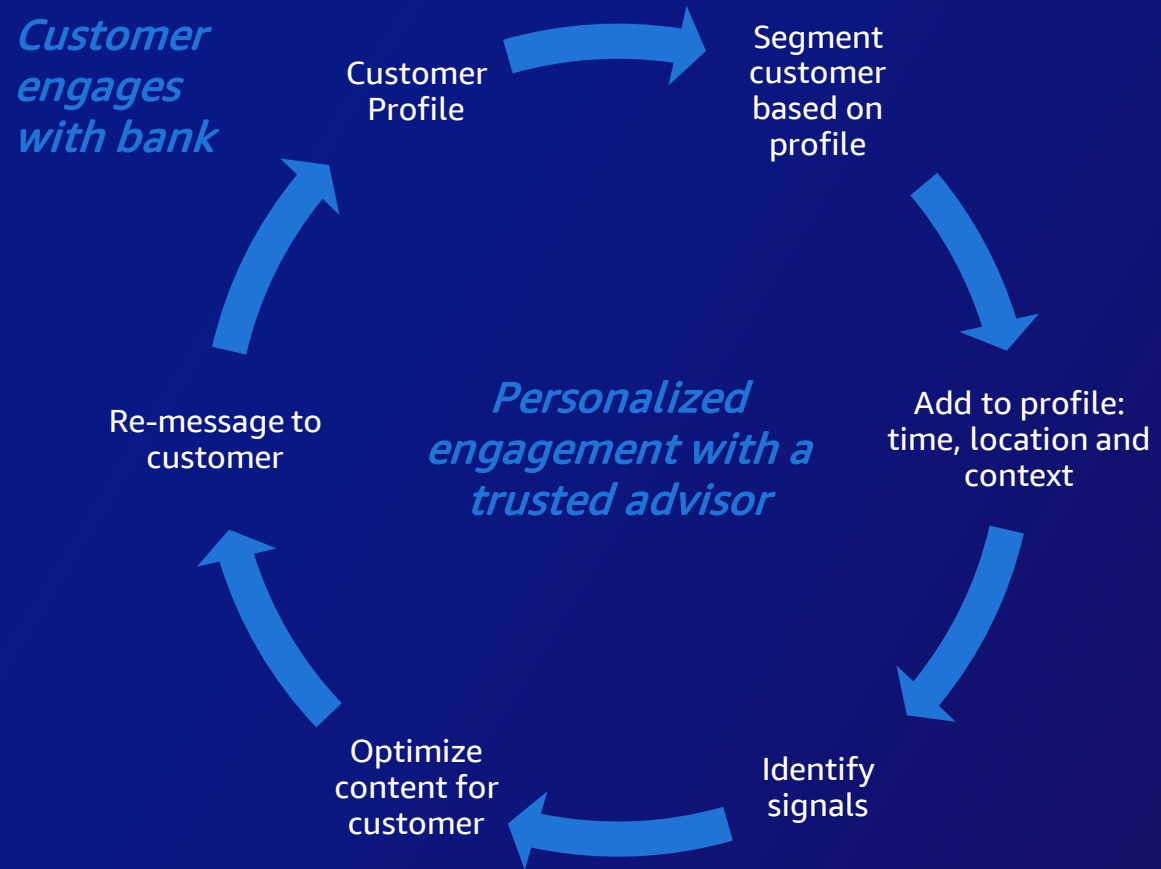
Targeted product recommendations and customized direct marketing



Consistent omni-channel experiences

# Successful engagement requires deeper customer insights

By building a **positive feedback loop** where customer insights are gathered and used to inform and send more **personalized messages**, banks can engage customers based on **their needs, interests, and preferences**.

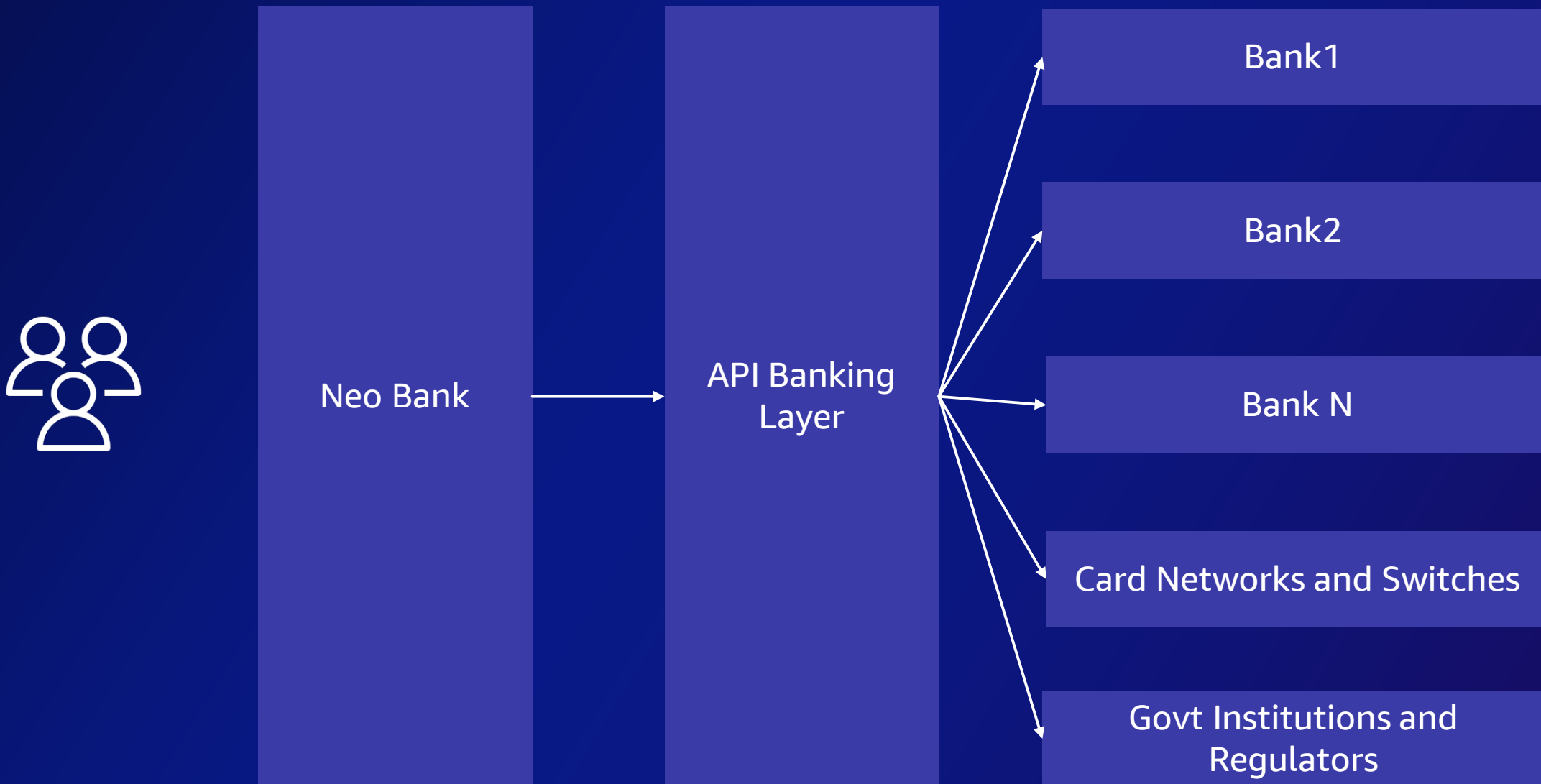


# How digital banking works?

- **Full-Stack (Licensed) Digital banks**
- **Autonomous unit of traditional banks**
- **Front-End Only Neo-banks**

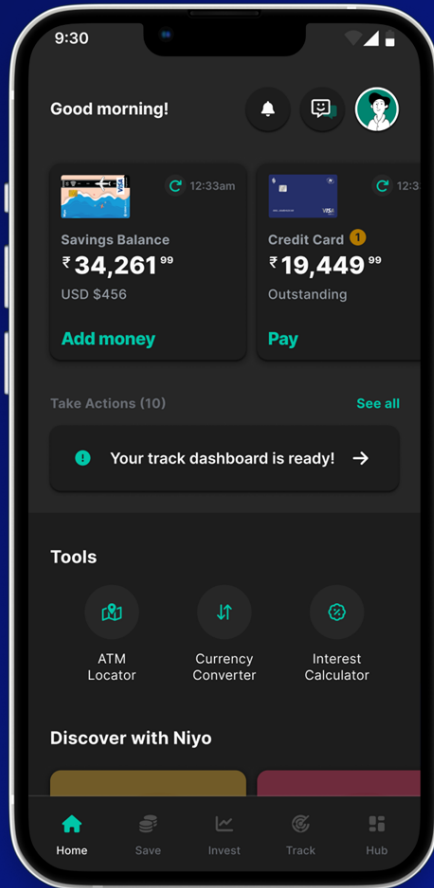


# Neo banking model





# Neo banking at Niyo



## All-in-one Financial App



5 Mn+ Users



Among top 3 digital banks in India



1st ever Savings + Wealth A/c

Forex

Payments

Savings

Investing

Credit

Commerce



## 6 Partner banks

Vast experience of bank integration with industry leaders makes us a preferred neo banking partner



## Licenses

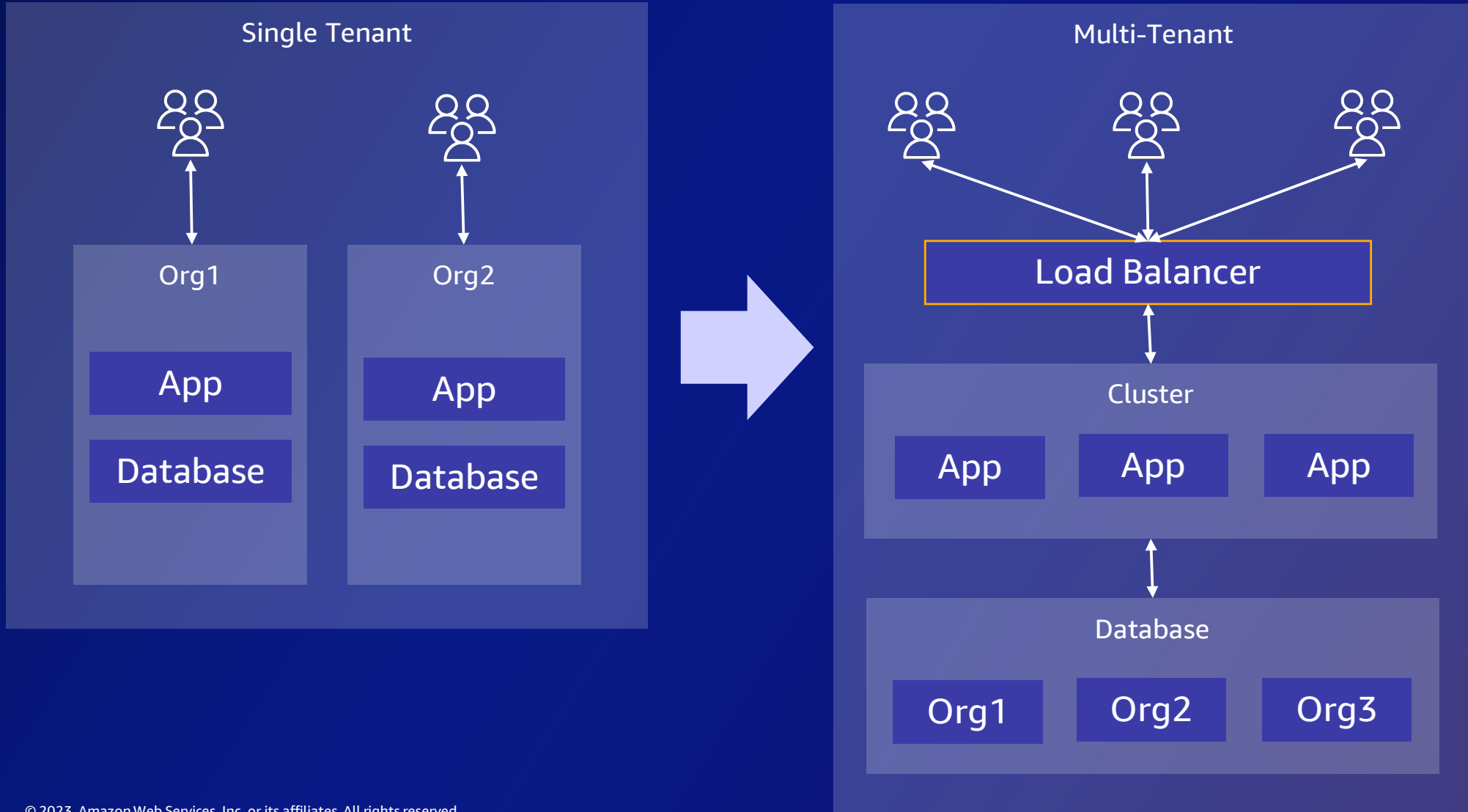


SEBI - RIA license  
For distribution of direct mutual funds



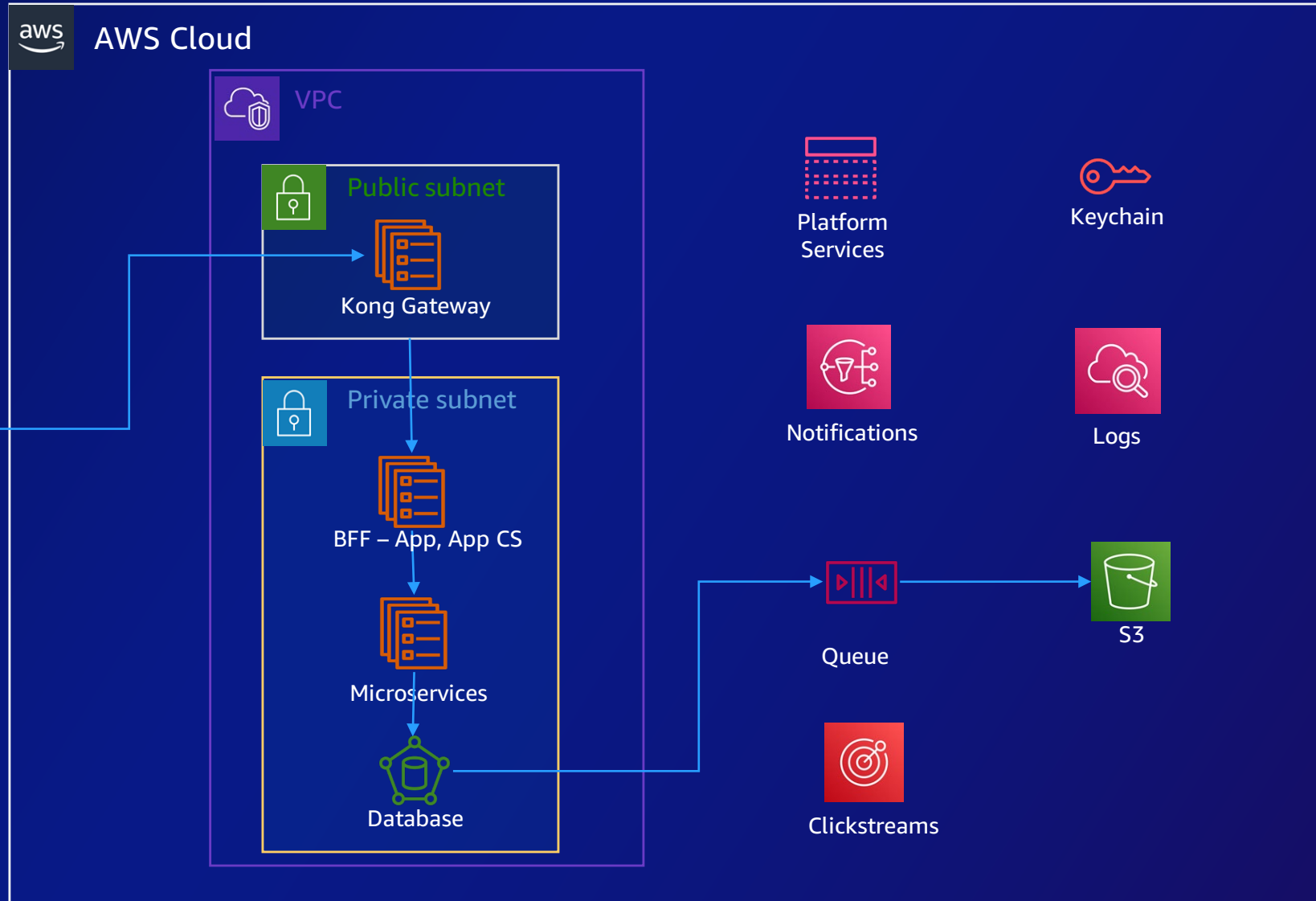
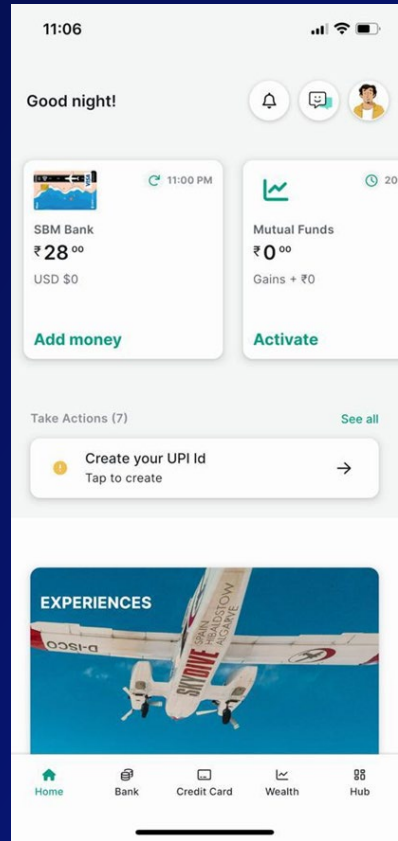
IRDA - Insurance  
Corporate Agency  
For insurance distribution

# Architecture evolution at Niyo

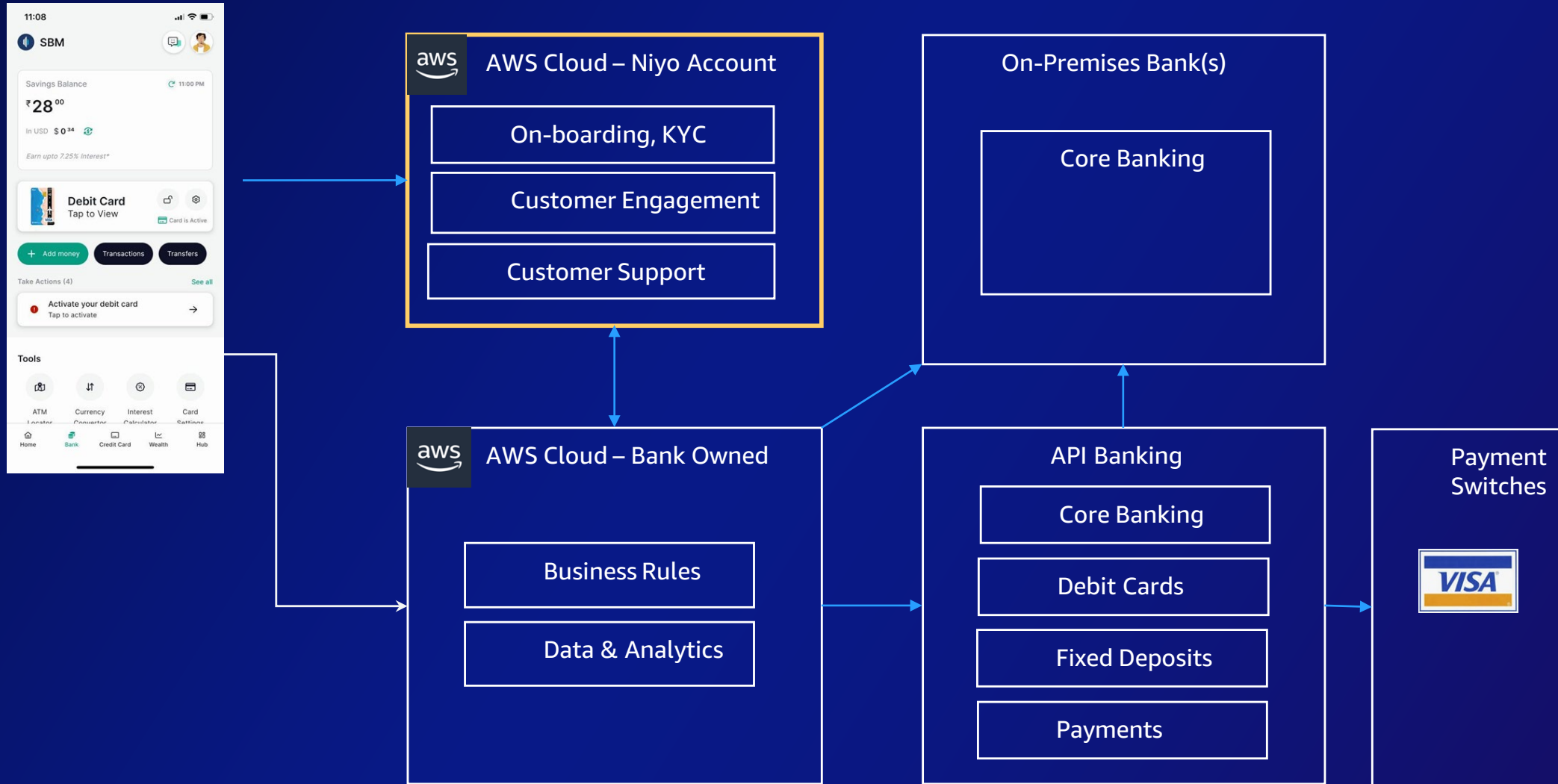


# Architecture overview

# Architecture – backend for frontend

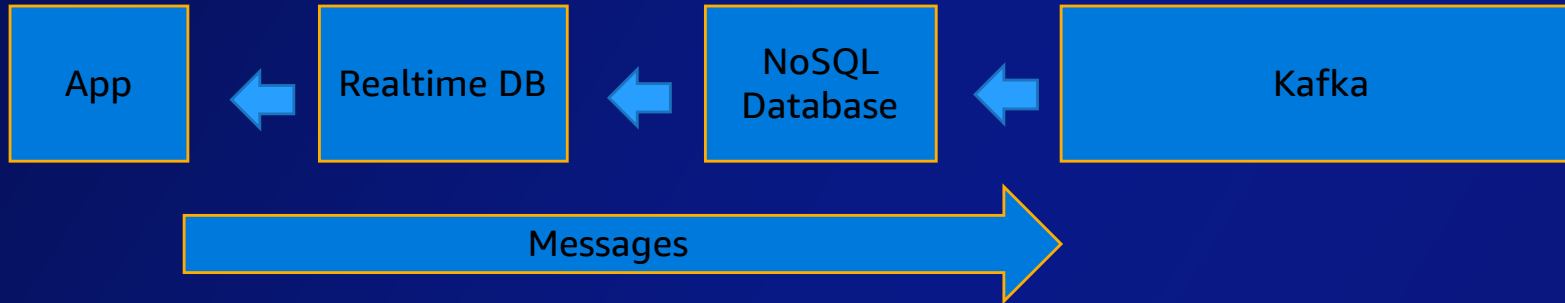


# Business architecture

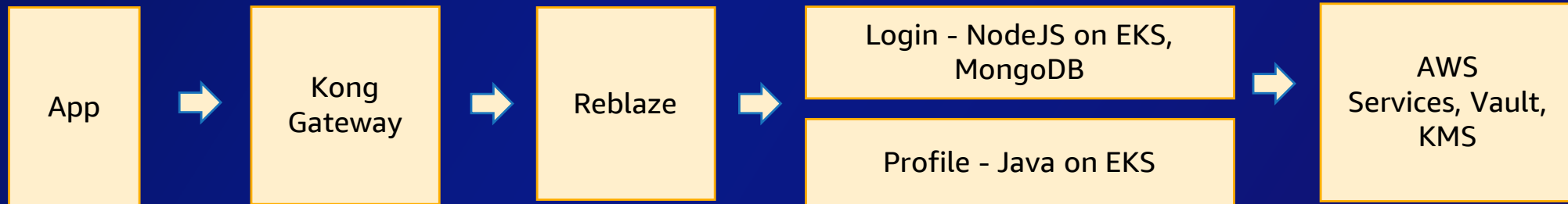


# Front-to-back flow

## Reactive Flow for Mobile updates



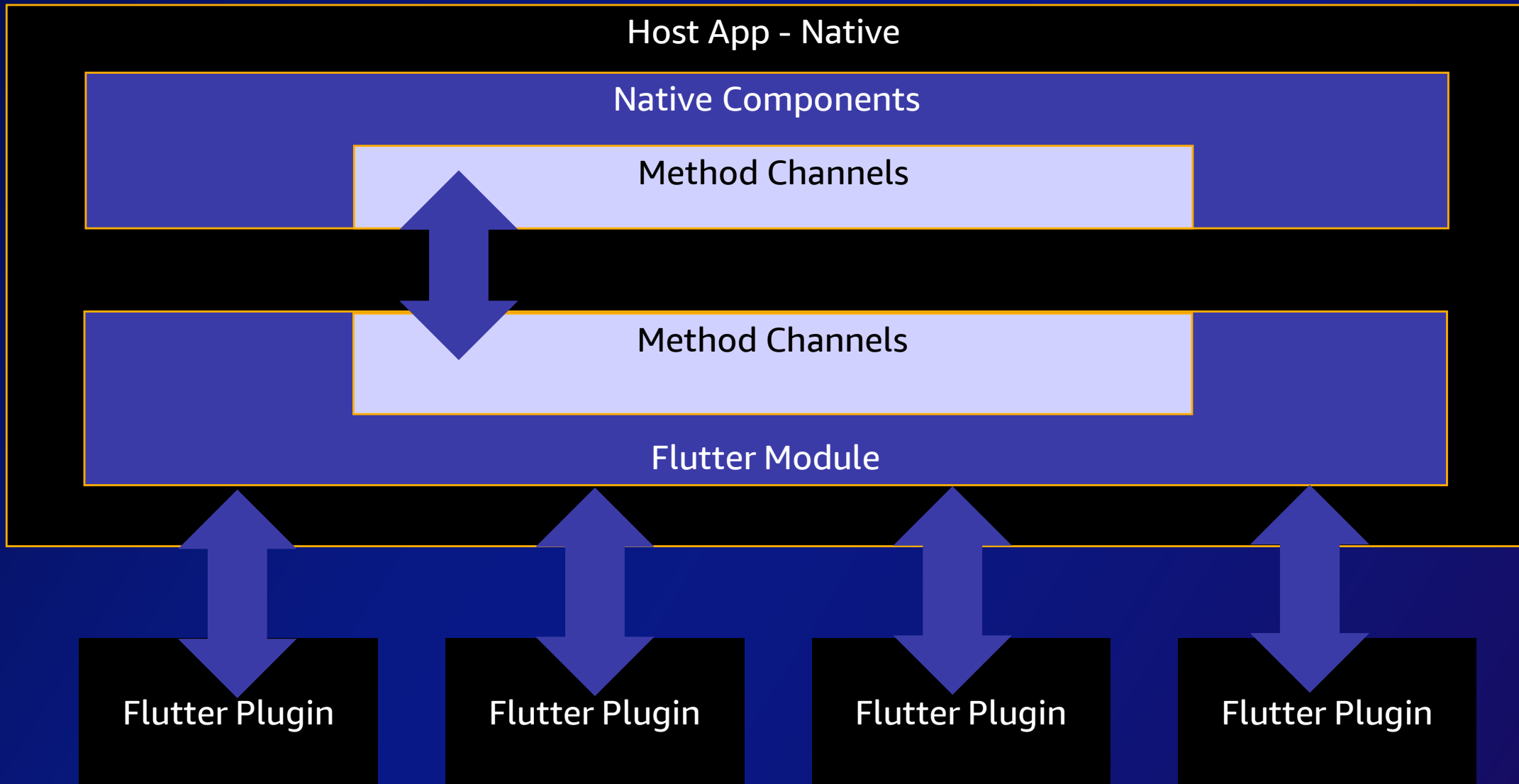
## Request-Response Flow



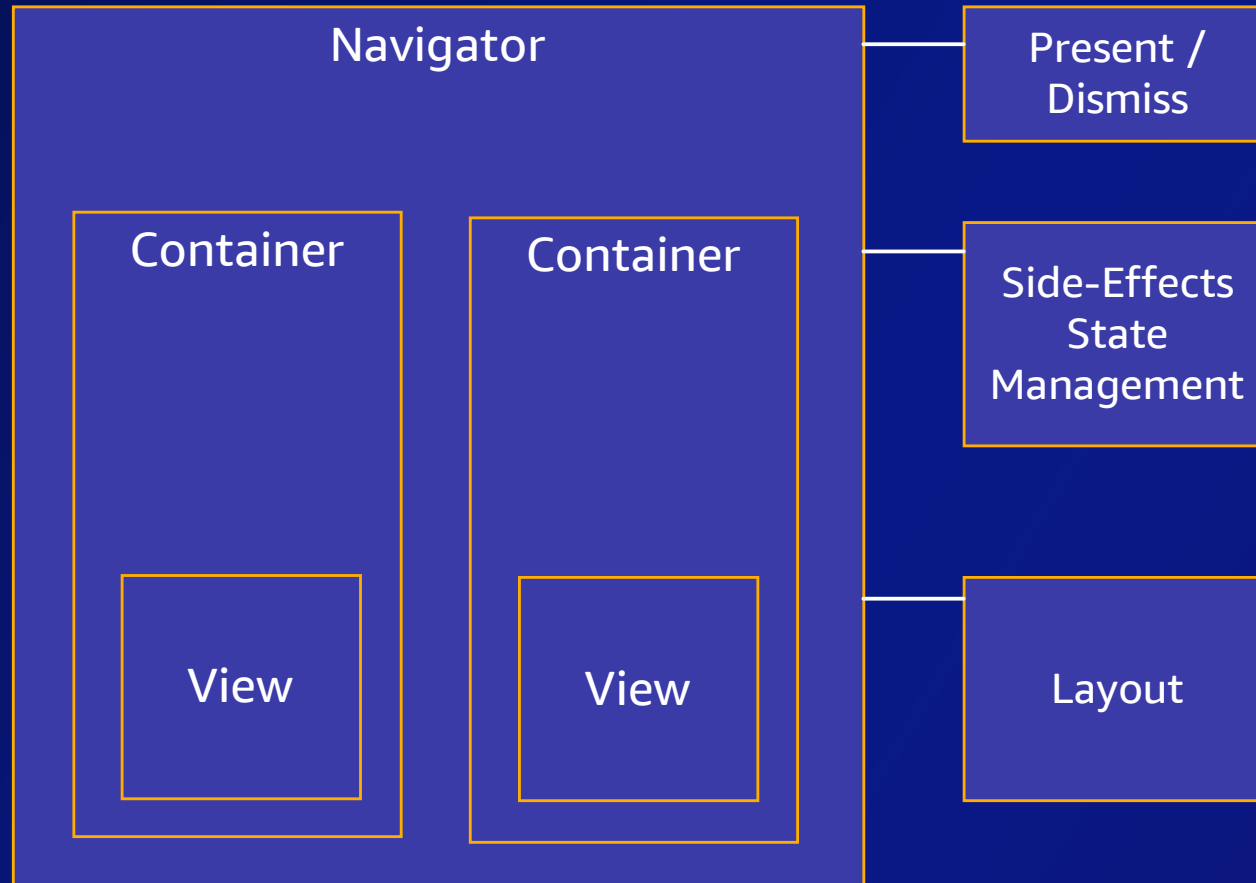
## Clickstream Analytics Flow



# Front-end reactive plugin based architecture



# Reactive mobile frontend



- **Navigators**

- ✓ Ordering
- ✓ Commands
- ✓ Data transfer

- **Containers**

- ✓ View state
- ✓ Side Effects

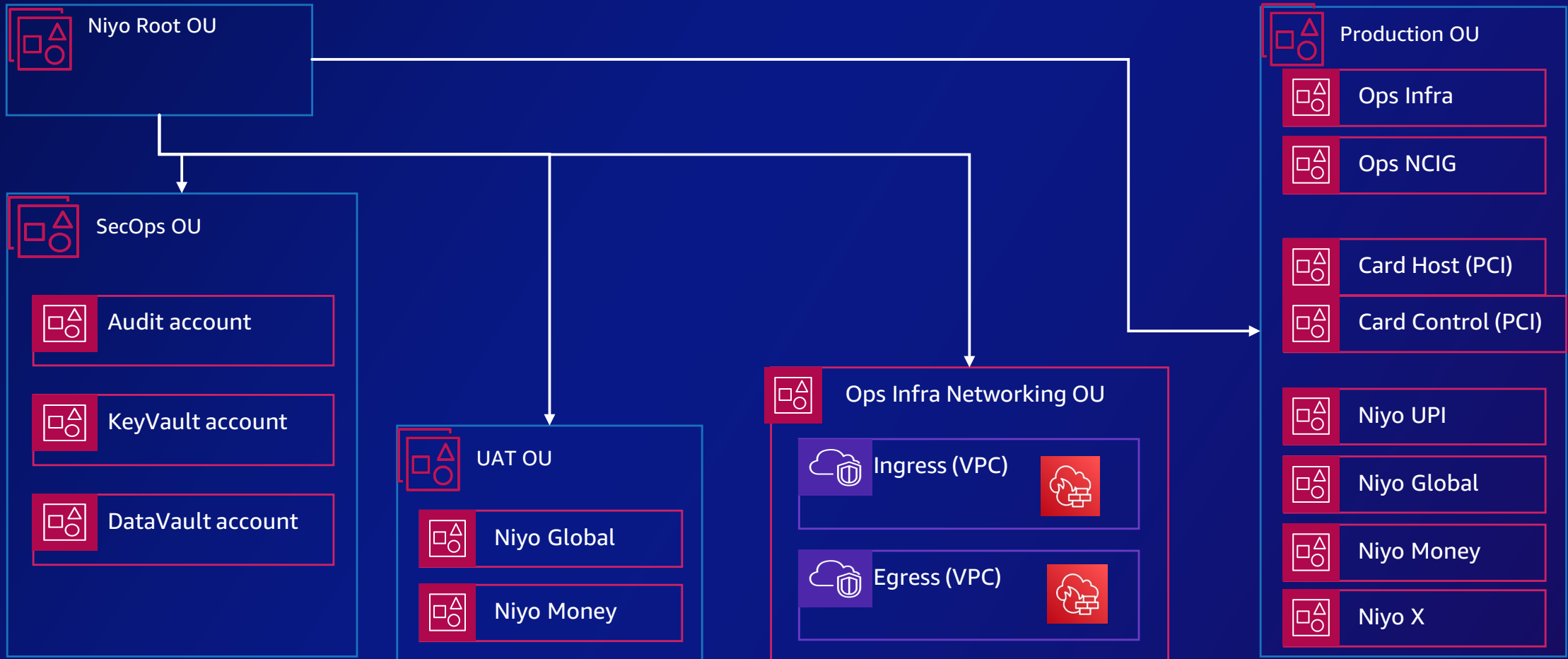
- **Views**

- ✓ Layout
- ✓ Relaying of interactions



# Building, Performing CI/CD and testing on AWS

# Using AWS organizations



# Continuous integration

Platform	Reload	Test Stability	Testability	Score
Flutter	Medium	Medium-High	High	Medium-High
Reactive Native	Low	Medium	High	Medium
Kotlin Native	Medium	Low	Low	Low

- Release cadence is not affected by the individual release cycles
- Its easy to maintain the individual plugins
- Sanity check done for the individual plugins

# Testing mobile applications

- Our engineering culture strongly encourages test automation
- Built-in testing infrastructure
- Each module has individual responsibilities so testing the capabilities is easier
- UI and view abstraction is encompassed by the framework
- View crash and soft crashes are logged and suppressed gracefully

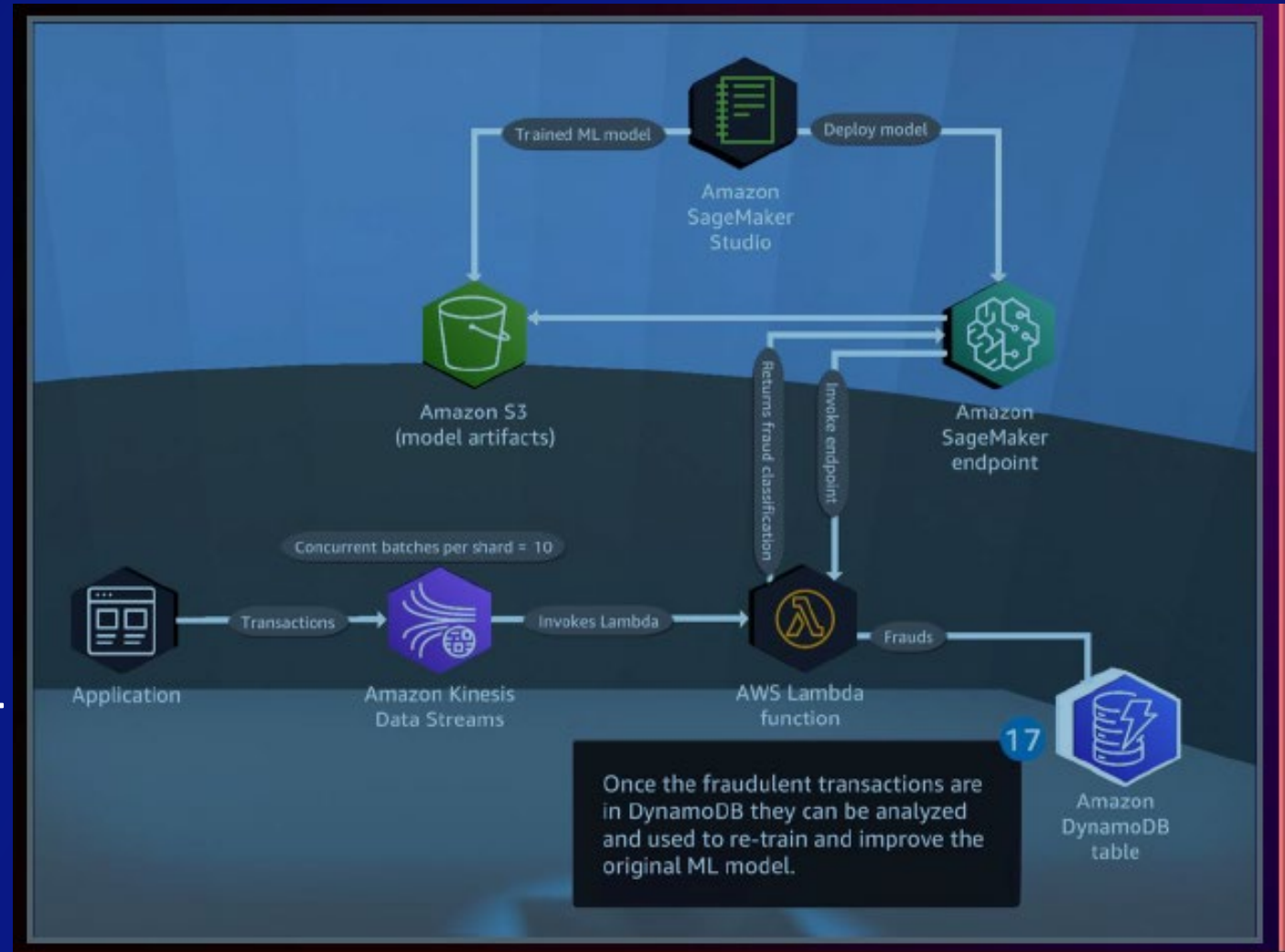
# Key takeaways

# Best practices at Niyo

- Plugin based Front-end architecture
- Microservices
- Separation of workloads organizations
- Continuous integration
- Automated testing

# AWS industry quest – Financial services

- Study real –world financial services architectures
- Enable “Risk free” experimentation
- Get hand-on with guided deployments for 20 FSI use-cases



# Thank you!

Prince Kumar  
Engineering Manager  
Niyo Solutions Inc.

Sathish Hariharan  
Senior Solutions Architect  
AWS India



Please complete the  
session survey