Board Proposal: Progressive Rollout Strategy Using Apigee

Prepared by: Vamshi Krishna Veggalam

# Problem Statement

As our partner integrations grow and new features are rolled out at a faster pace, we face a growing need to:  
- Introduce new functionality safely without impacting existing partner traffic  
- Allow specific partners to test new features early (by choice)  
- Provide a clean rollback path if issues arise  
- Minimize full-scale releases and manual validations for every change  
However, our current setup lacks a consistent and scalable mechanism to support this.

# Current Limitation

- We rely on full deployment rollouts without safe traffic segregation  
- There's no way to expose new versions only to select partners unless we clone entire apps or proxies  
- Partners who wish to opt-in for early testing have no structured way to do so  
- Rollbacks require code-level reversions or emergency releases, increasing downtime and risk

# Goal

Implement a controlled canary release mechanism using Apigee request headers + developer app attributes so that:  
- Only pre-approved partners can access new features  
- Partners can control when to test via headers  
- Synchrony can route traffic safely to either main or new logic  
- We reuse the same Apigee proxy — no duplication needed