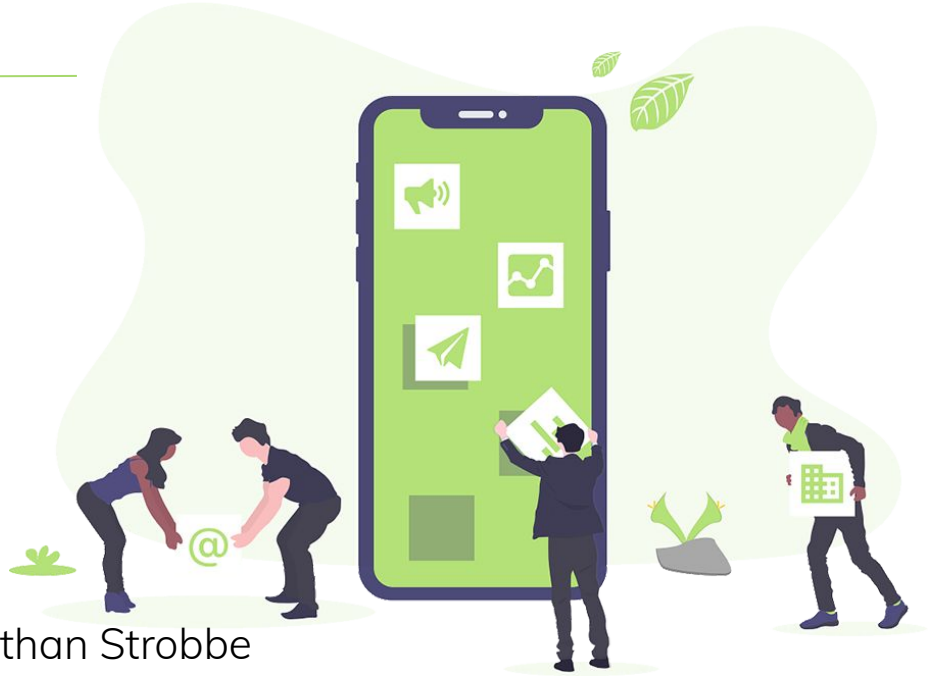
Architectures Logicielles

Site e-banking CréditRama



Équipe D

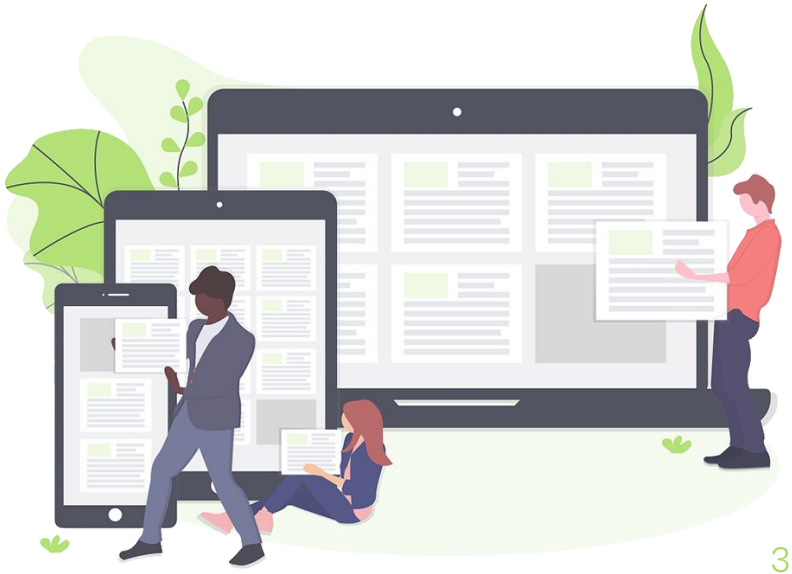
Théo Foray - Alexis Gardin - Laura Lopez - Nathan Strobbe

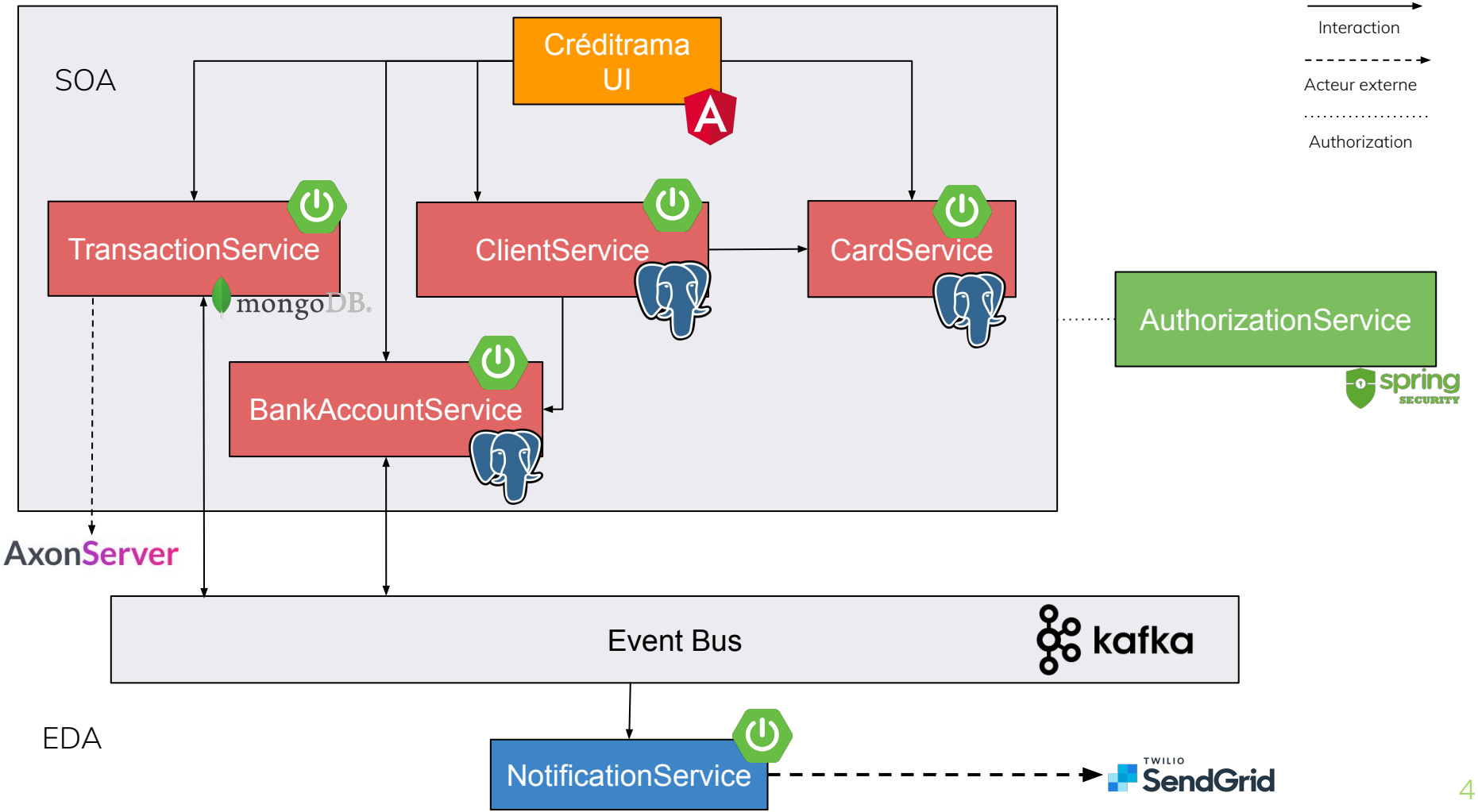
Nos défis

Code de vérification

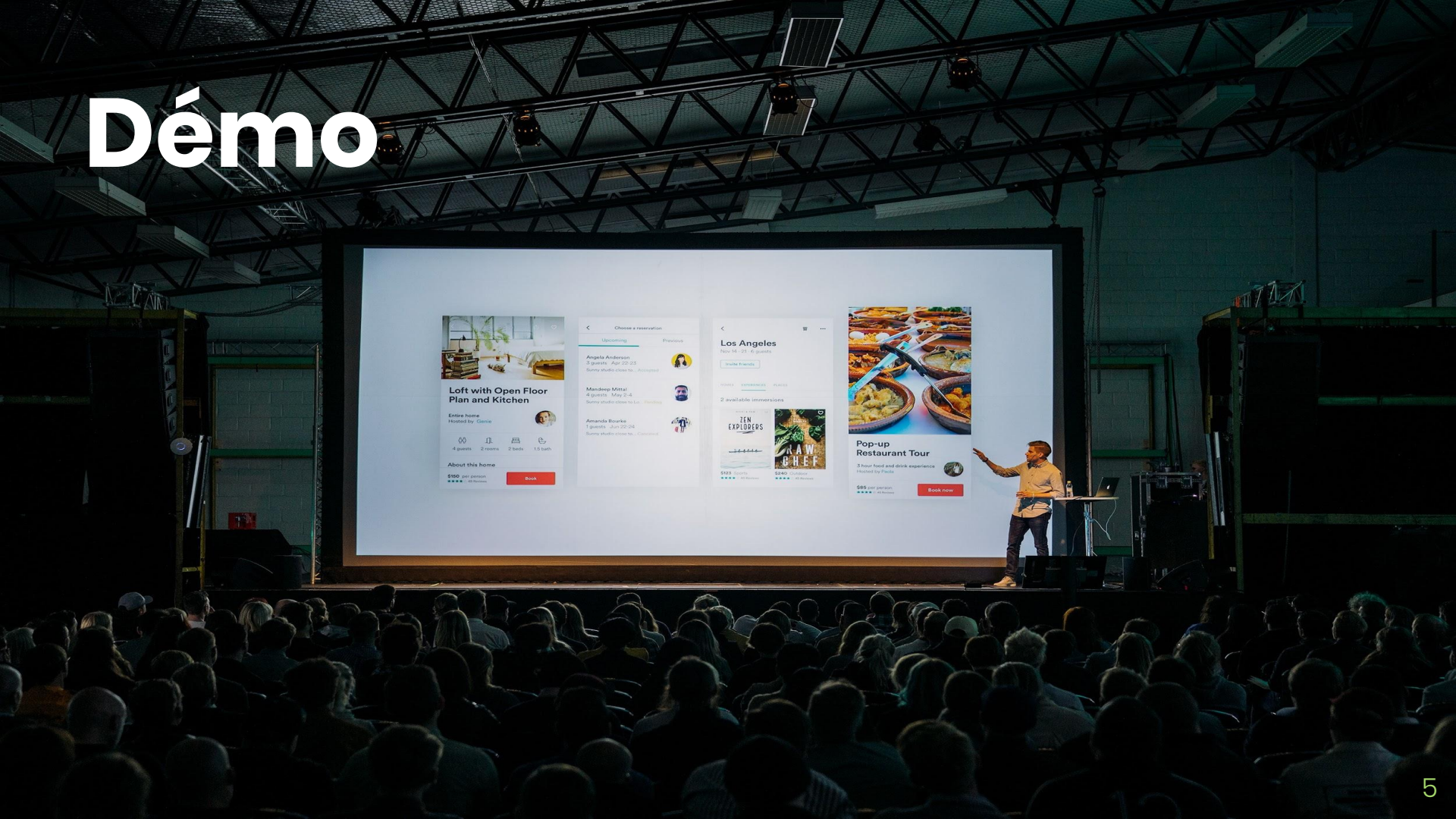
Erreurs sur l’écriture des transactions

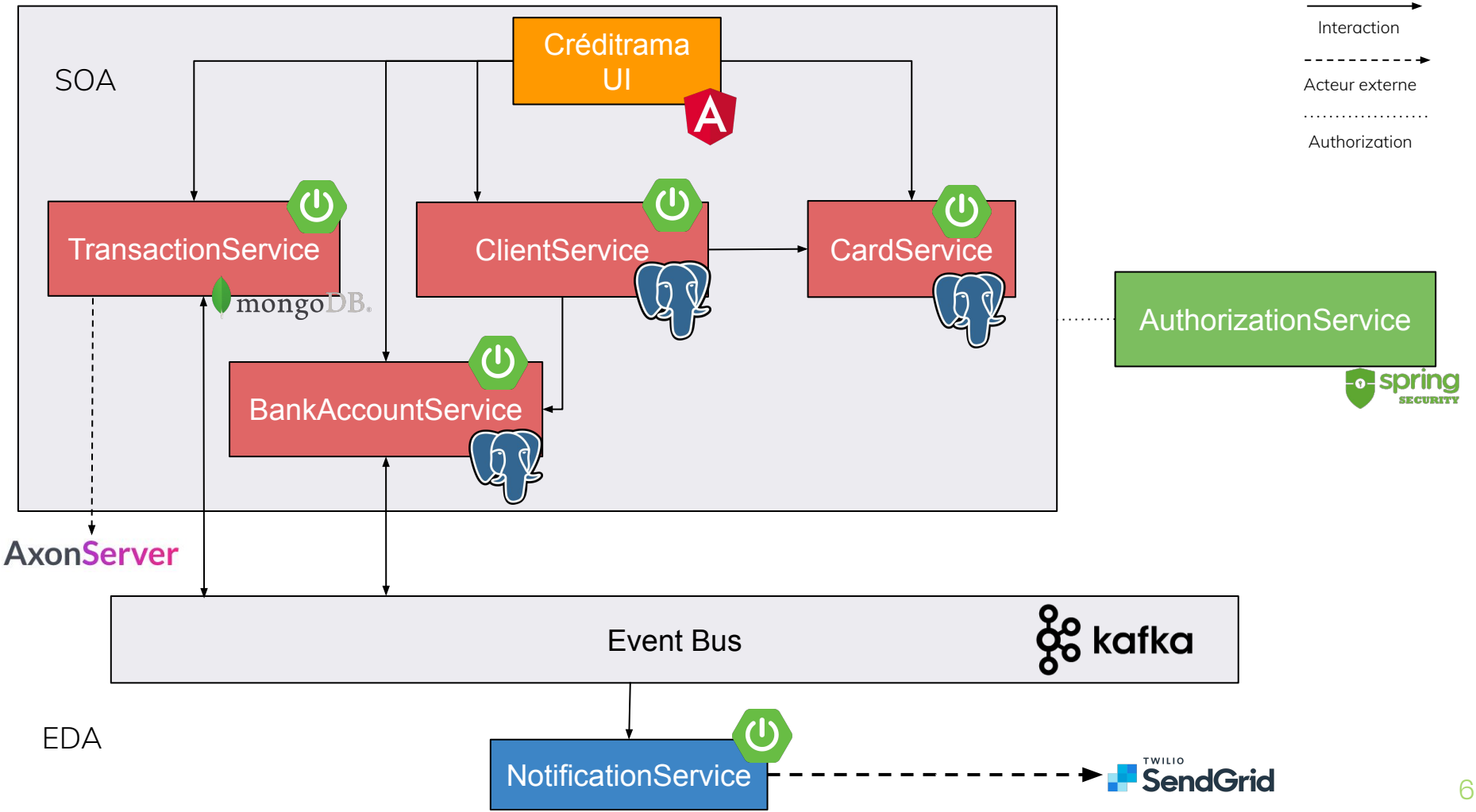
Présentation de l'architecture



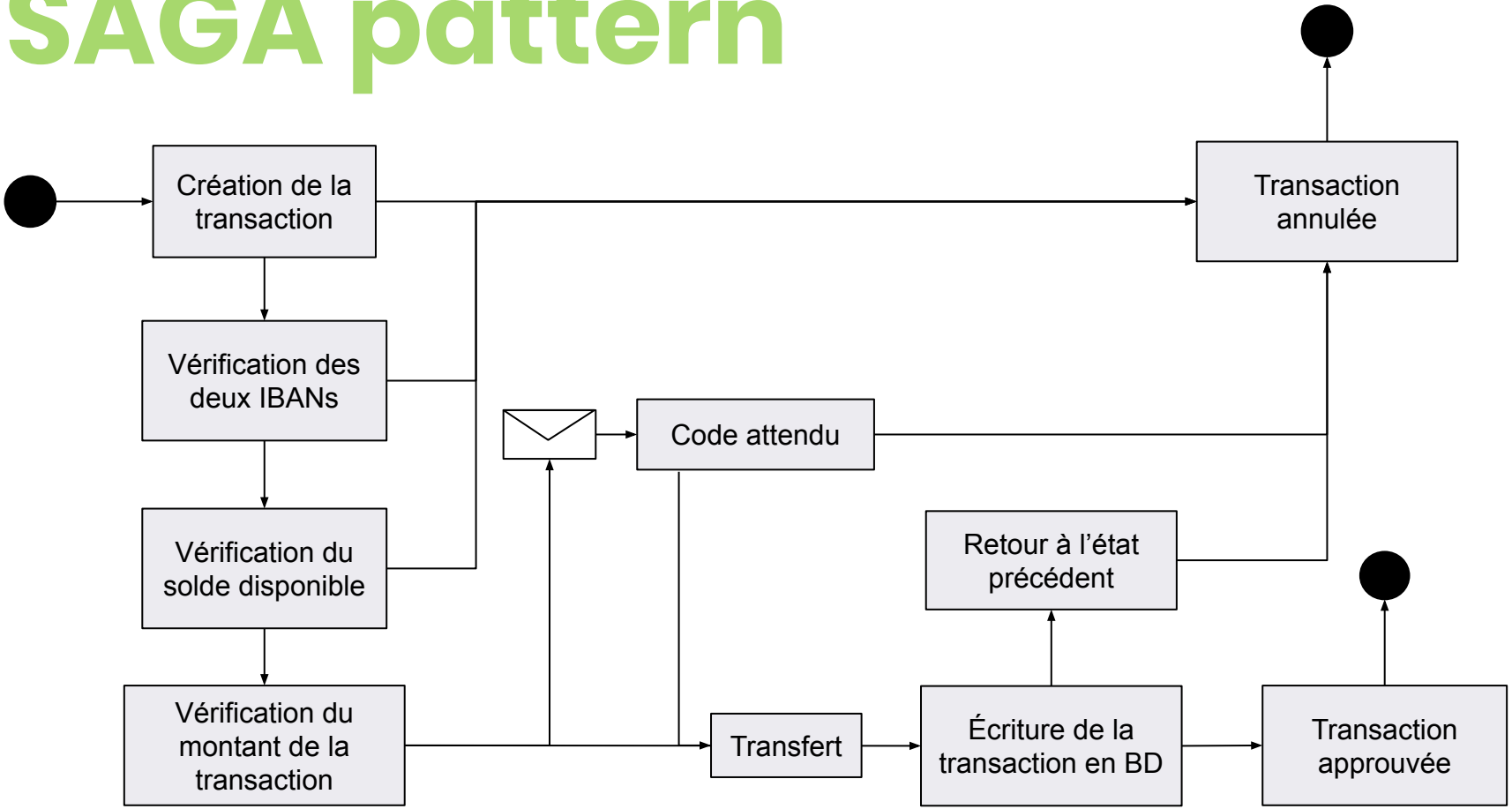

SOA
Créditrama UI
TransactionService
mongoDB.
ClientService
CardService
BankAccountService
AuthorizationService
spring SECURITY
AxonServer
Event Bus
kafka
EDA
NotificationService
TWILIO SendGrid
Interaction
Acteur externe
Authorization

Démo


Loft with Open Floor Plan and Kitchen
Entire home
Hosted by Genie
4 guests
2 rooms
2 beds
1.5 bath
About this home
$150 per person
Book
Choose a reservation
Upcoming
Previous
Angela Anderson
3 guests Apr 22-23
Mandeep Mittal
4 guests May 2-4
Amanda Bourke
1 guests Jun 22-24
Los Angeles
Nov 14 - 21 · 6 guests
Invite friends
2 available immersions
ZEN EXPLORERS
RAW CHEF
$123
$240
Pop-up Restaurant Tour
3 hour food and drink experience
Hosted by Paola
$85 per person
Book now


SOA
Créditrama UI
TransactionService
mongoDB.
ClientService
CardService
BankAccountService
AuthorizationService
spring SECURITY
AxonServer
Event Bus
kafka
EDA
NotificationService
TWILIO SendGrid
Interaction
Acteur externe
Authorization

SAGA pattern



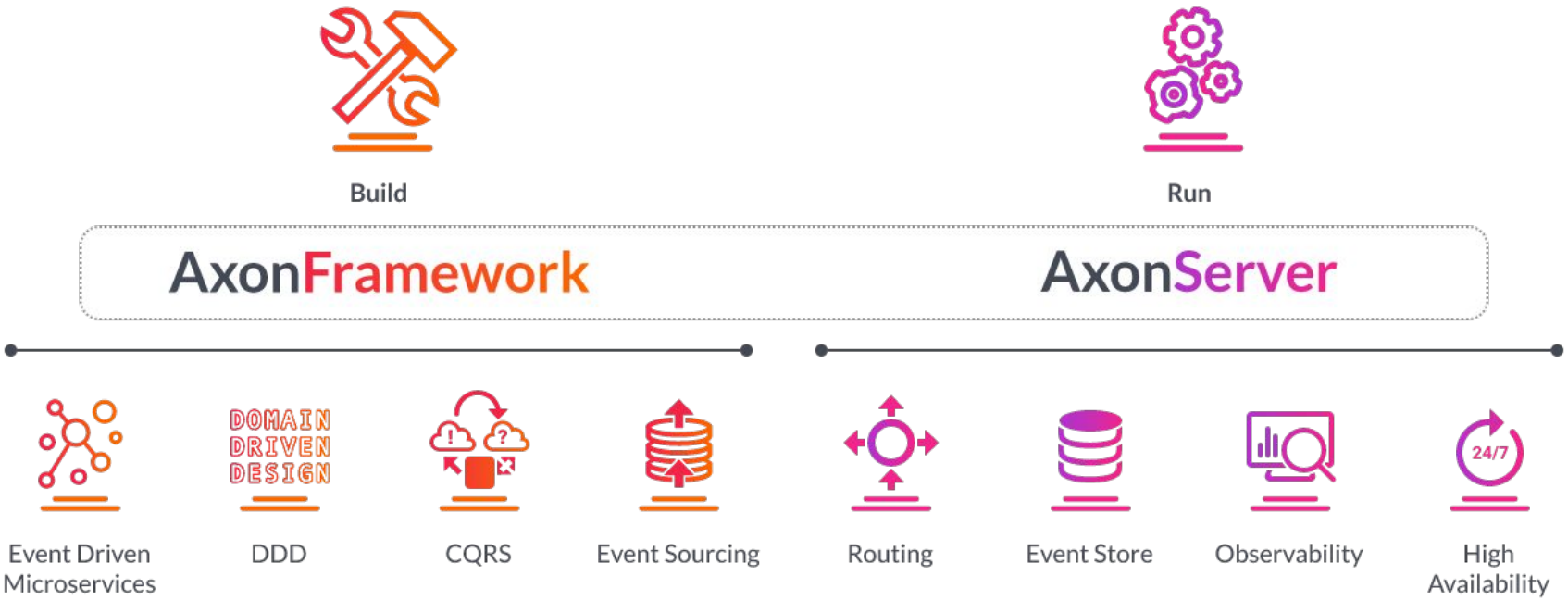
SAGA pattern

Aggregat	Actions	Événement déclenché	Actions de compensation
Transaction	createTransaction()	TransactionCreatedEvent	
Transaction	checkValidity()	TransactionCheckedEvent	rejectTransaction()
Transaction	verifyAmount()	CodeNeededEvent	
Transaction	confirmCode()	CodeConfirmedEvent	
BankAccount	makeTransfer()	TransferDoneEvent	
Transaction	storeTransaction()	TransactionStoredEvent	reverseTransfer()
Transaction	approveTransaction()	TransactionApprovedEvent	

SAGA pattern

	SAGA chorégraphié	SAGA orchestré
Avantages	Communication événementielle Séparation des responsabilités	Simplifie maintenance et compréhension Un service chargé du cycle de vie
Inconvénients	Complexité du suivi des échanges Dépendances cycliques	Peu de communications interservices “God class tendency”

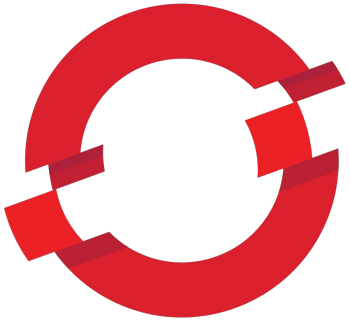
SAGA pattern



Environnement

TEKTON

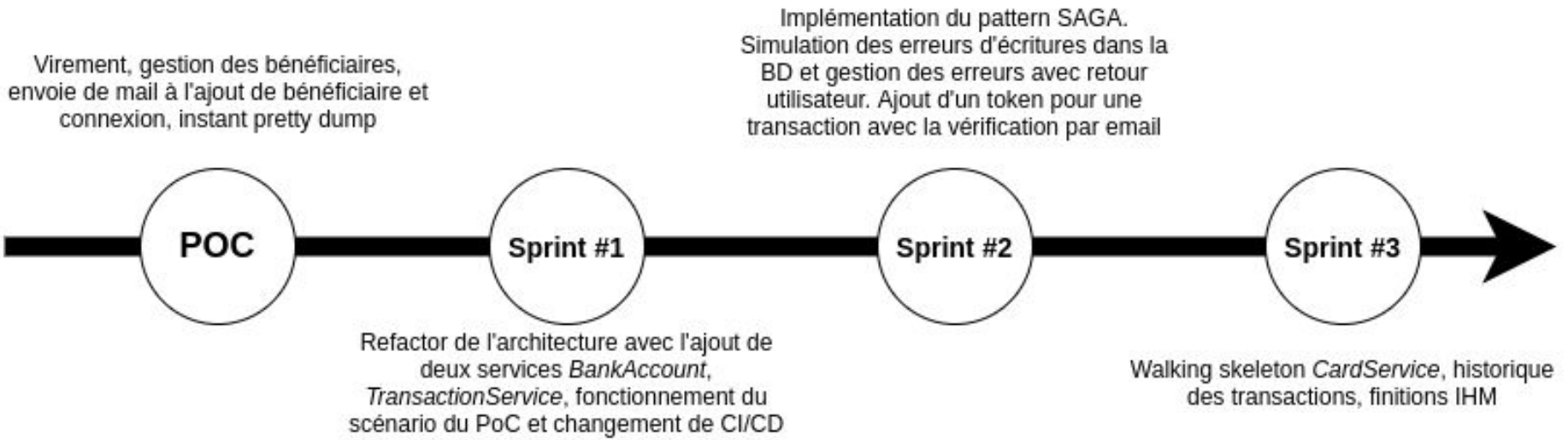
POSTMAN



OPENSHIFT

Prise de recul

Roadmap


Virement, gestion des bénéficiaires, envoie de mail à l'ajout de bénéficiaire et connexion, instant pretty dump
POC
Refactor de l'architecture avec l'ajout de deux services *BankAccount*, *TransactionService*, fonctionnement du scénario du PoC et changement de CI/CD
Sprint #1
Implémentation du pattern SAGA. Simulation des erreurs d'écritures dans la BD et gestion des erreurs avec retour utilisateur. Ajout d'un token pour une transaction avec la vérification par email
Sprint #2
Walking skeleton *CardService*, historique des transactions, finitions IHM
Sprint #3

Merci de votre écoute