

# ACADEMIA

R A F A   O U T S Y S T E M S



# MÓDULO

Transição para Reactive

Aula 01

Introdução & Conceitos Básicos



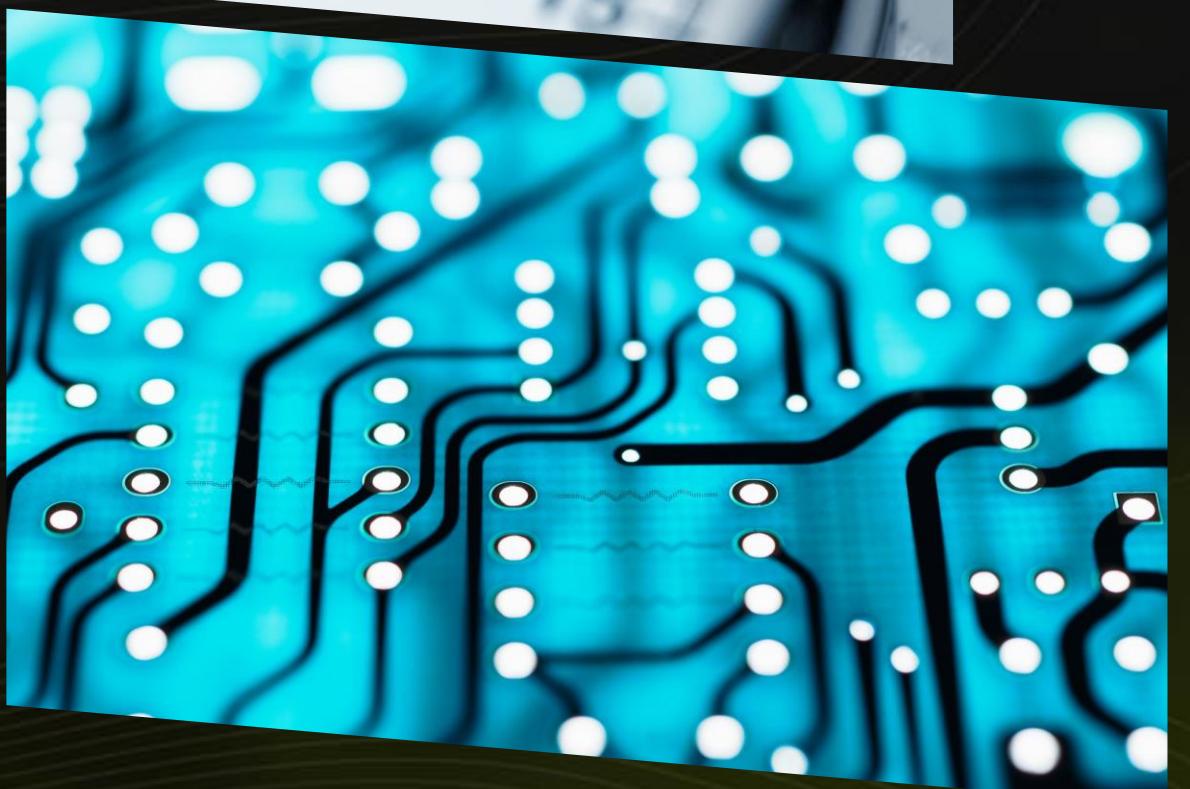
# Caio Santana

- OutSystems MVP
- 30+ componentes na Forge
- Arrastando bolinhas desde 2015

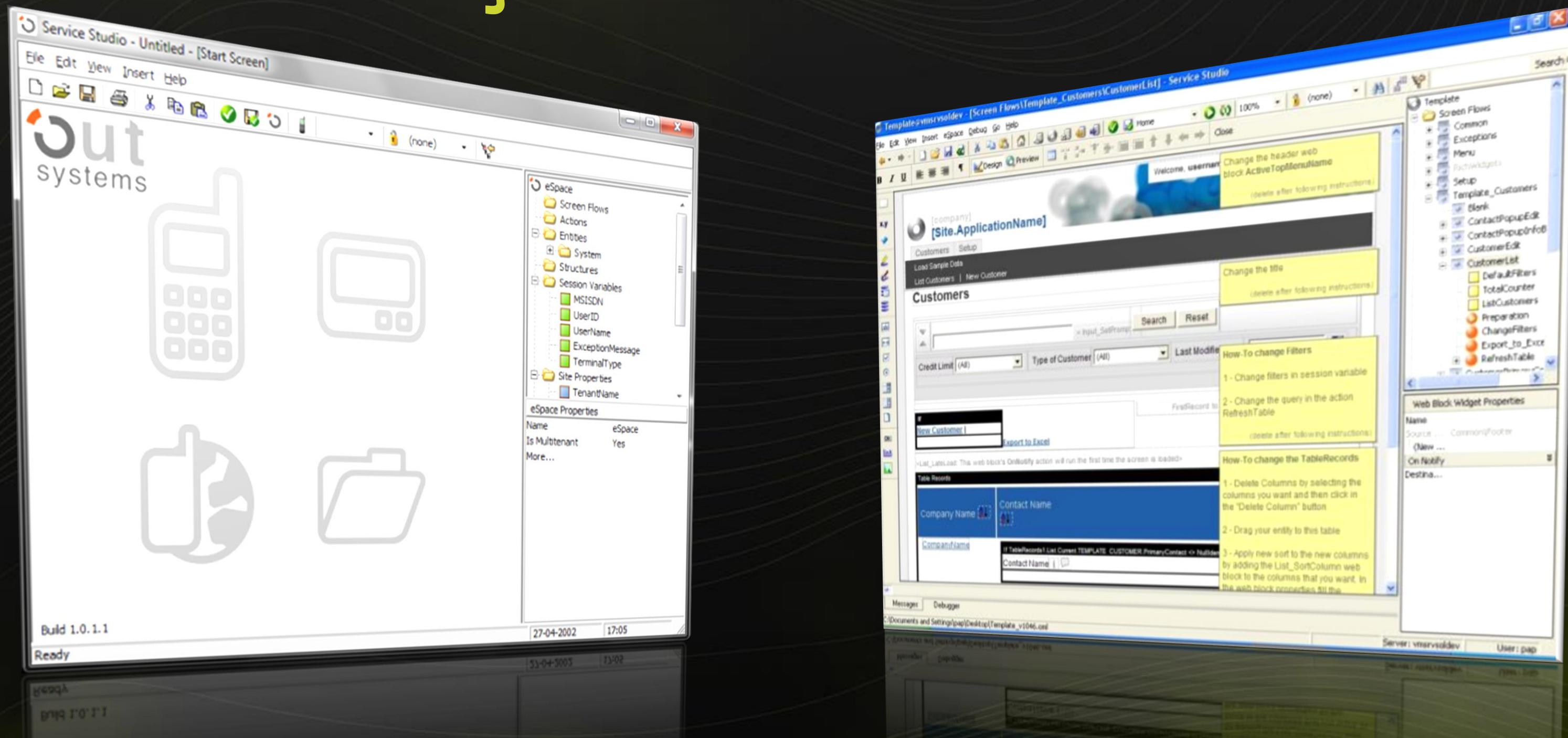
# Traditional Web

# O que é “Traditional”?

- Uma das Stacks mais antigas
- Fruto da evolução da plataforma desde o início (2001)
- Recebeu melhorias de alto impacto a cada 1-2 anos
- Antecede tecnologias modernas (Mobile/Reactive)
- Continuará recebendo suporte
  - ... mas não receberá novas features

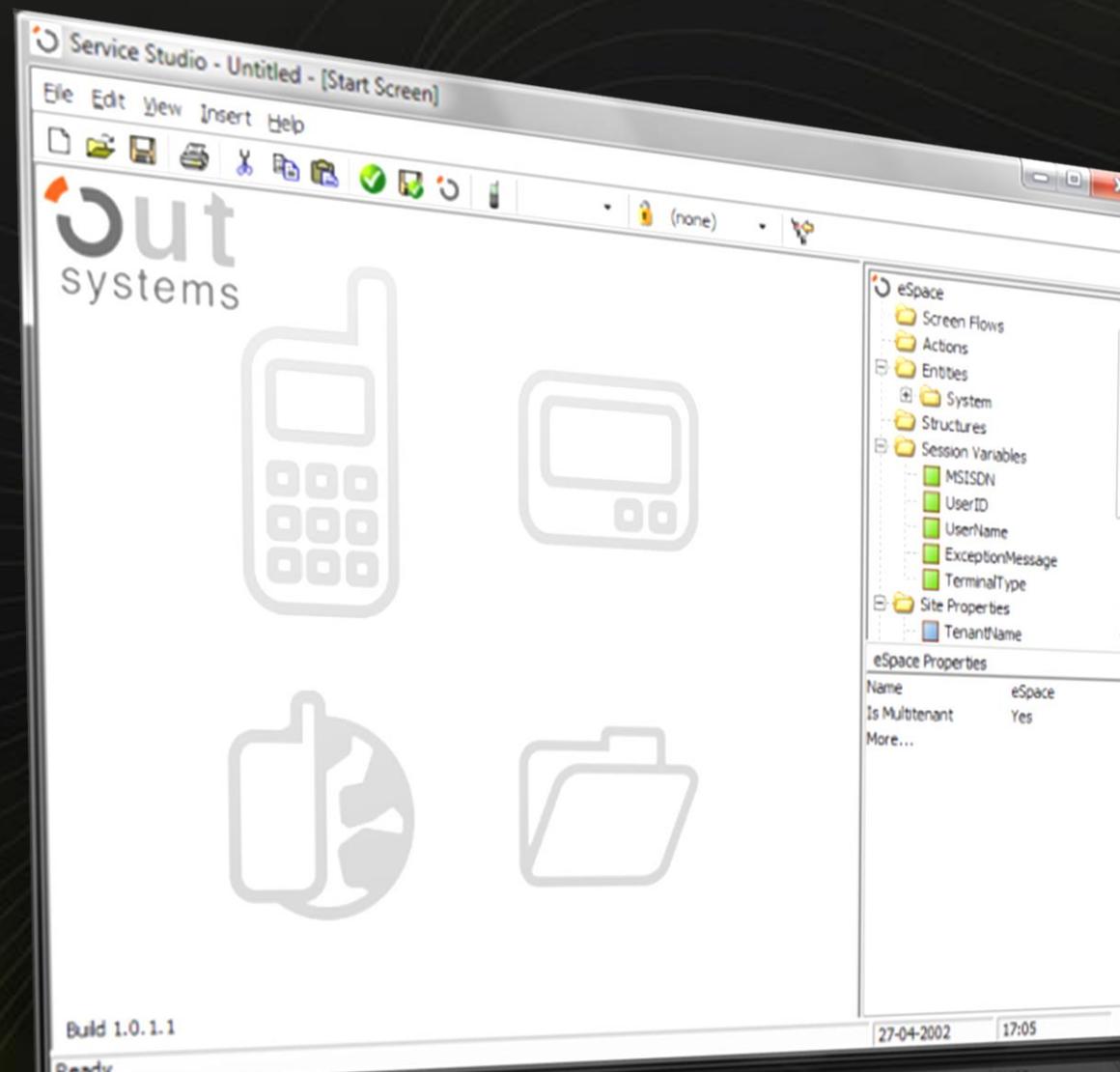


# Evolução da Plataforma



<https://www.outsystems.com/blog/posts/not-new-product-timeline/>

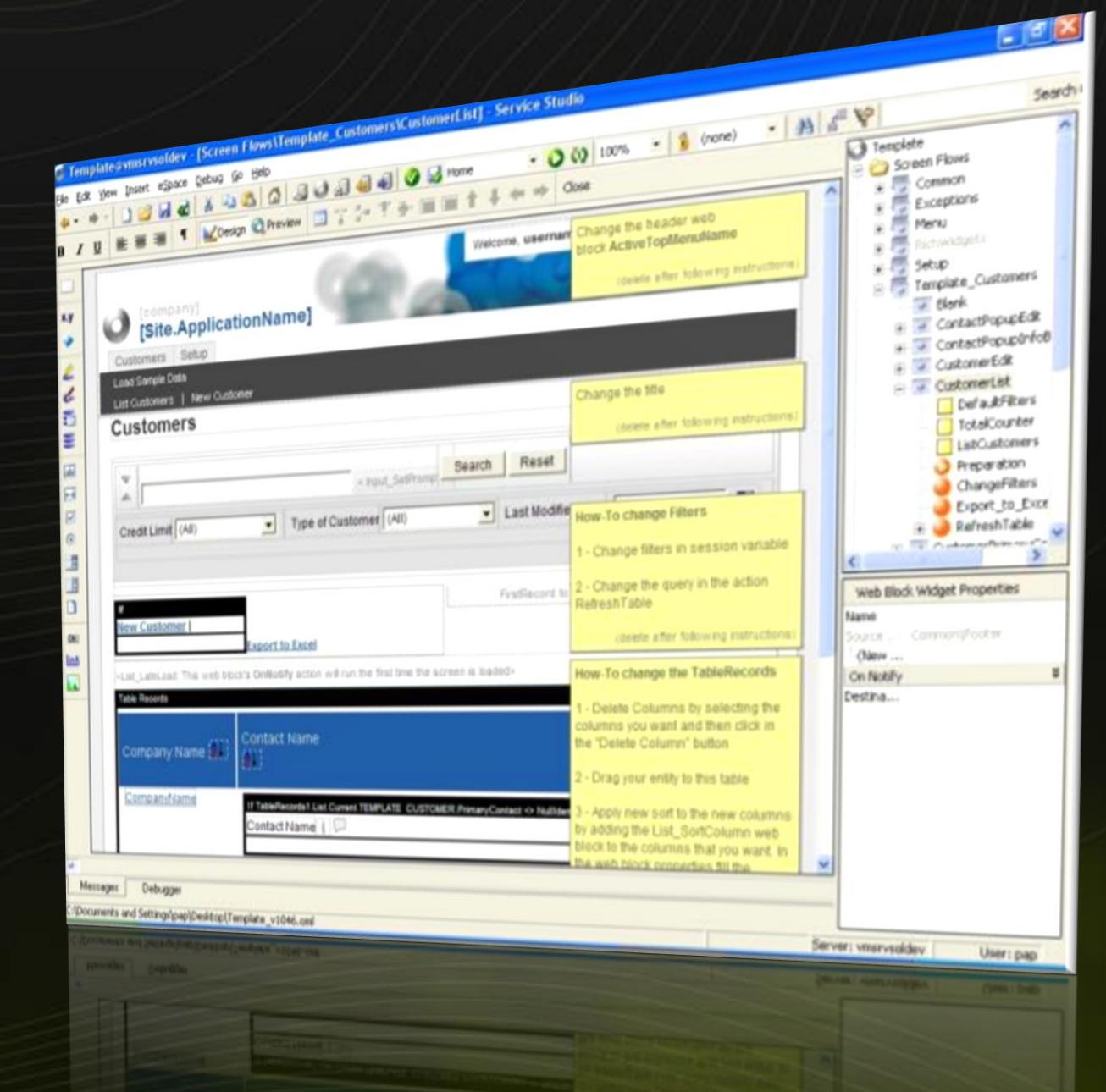
# Evolução da Plataforma



Hub Edition  
(2002)

↓  
Agile Platform  
(2008)

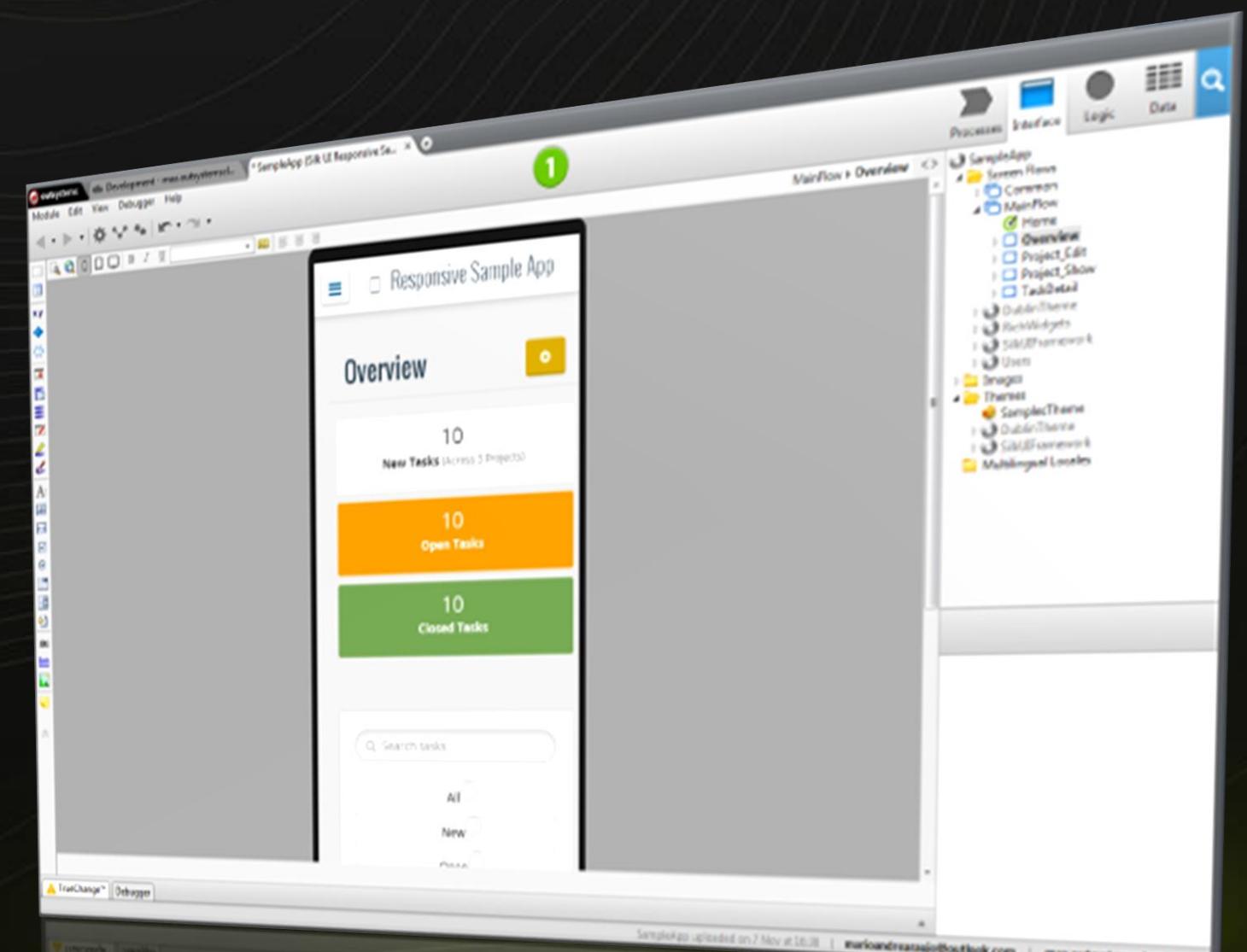
↓  
OutSystems Platform  
(2012)



<https://www.outsystems.com/blog/posts/not-new-product-timeline/>

# Evolução da Plataforma

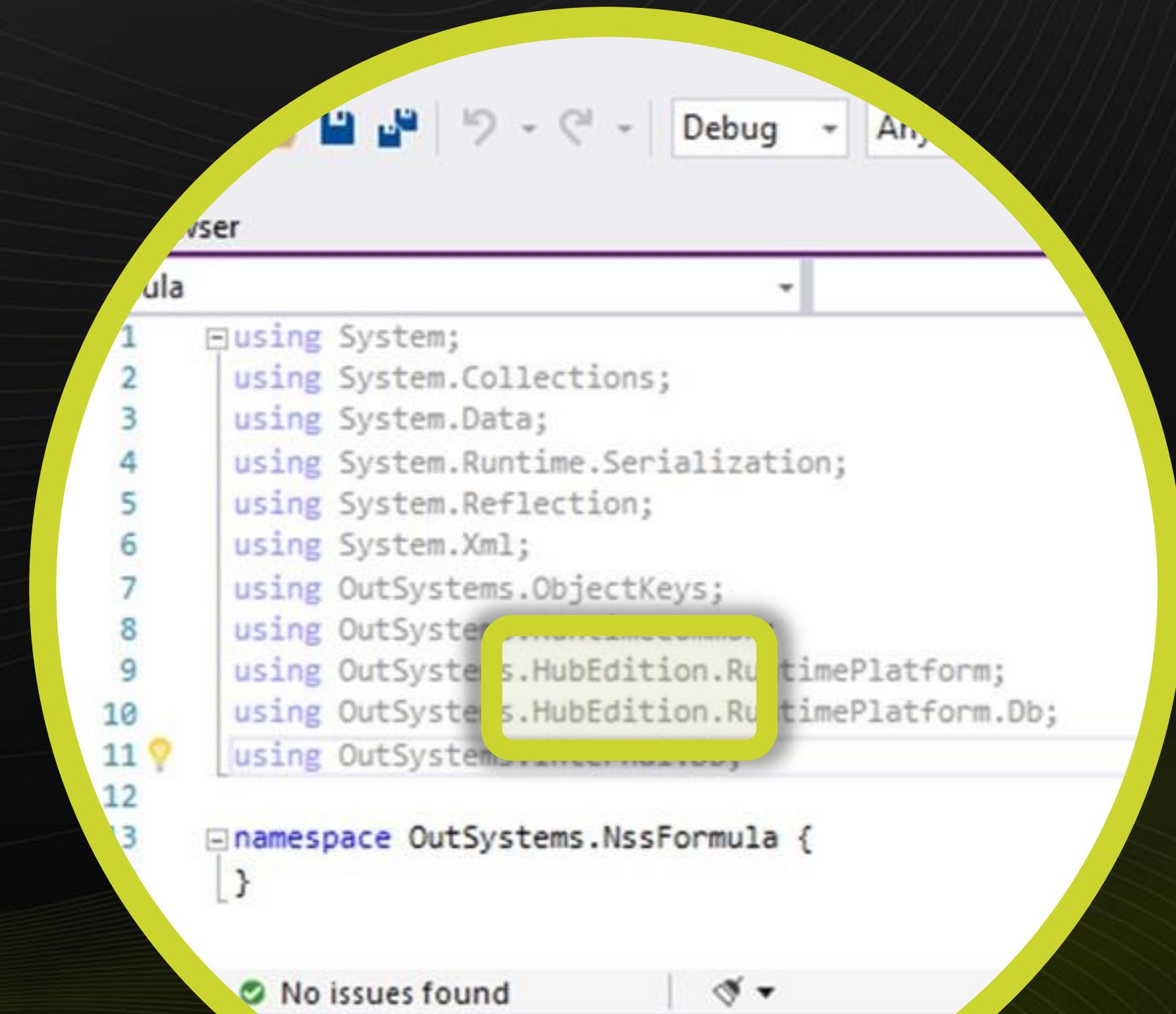
Mobile Apps  
(2016)



# Fato Curioso

Até hoje podemos ver o legado deixado pelo Hub Edition.

Basta olhar as referências das extensões (XIFs) em .NET.



A screenshot of a .NET code editor showing a C# file named 'Fórmula.cs'. The code includes several 'using' statements:

```
1  using System;
2  using System.Collections;
3  using System.Data;
4  using System.Runtime.Serialization;
5  using System.Reflection;
6  using System.Xml;
7  using OutSystems.ObjectKeys;
8  using OutSystems.HubEdition.RuntimePlatform;
9  using OutSystems.HubEdition.RuntimePlatform.Db;
10 using OutSystems.HubEdition.RuntimePlatform.NssFormula;
11
12
13 namespace OutSystems.NssFormula {
```

The line 'using OutSystems.HubEdition.RuntimePlatform.NssFormula;' is highlighted with a yellow rectangular box.

# Tipos de Aplicação



Traditional Web  
(2001)



Incompatível



Mobile  
(2016)



Compatível



Reactive  
(2019)

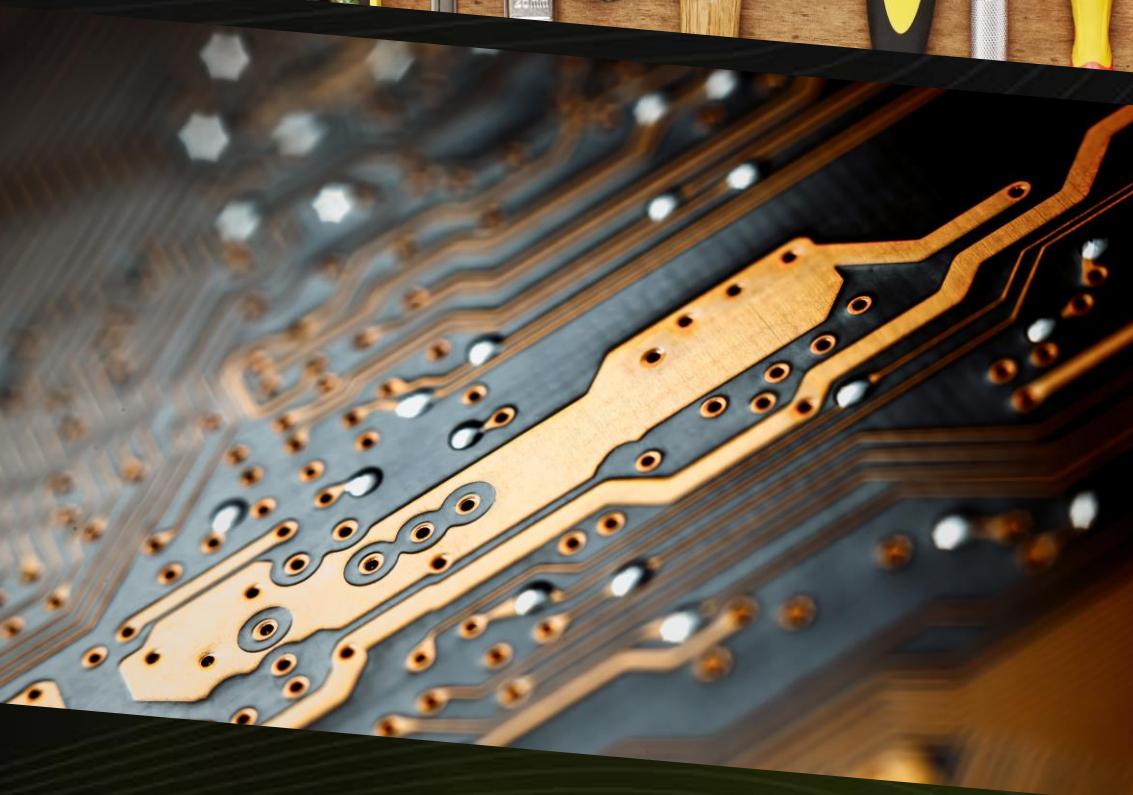
Framework Proprietário

Cordova

Local Storage

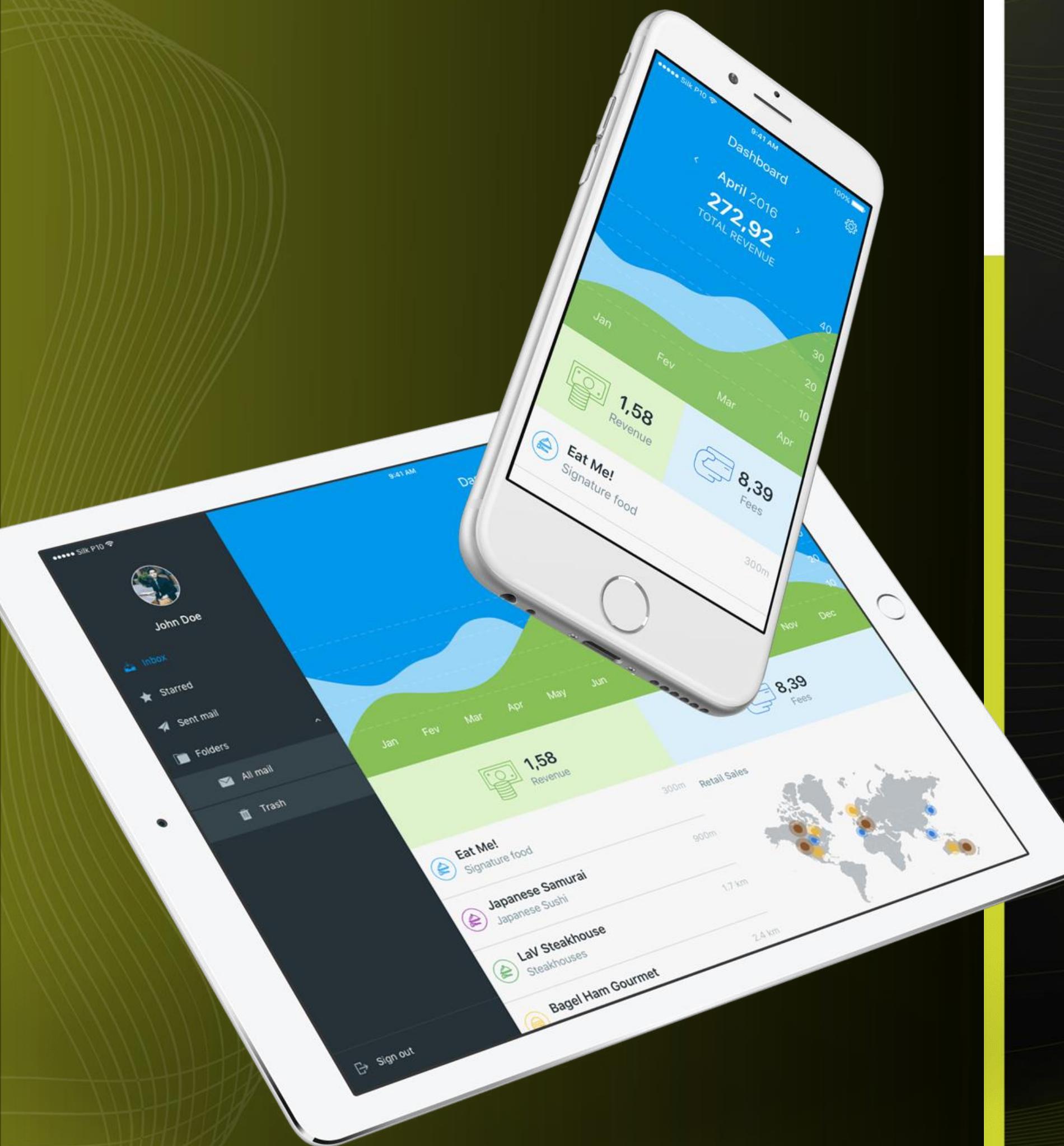
# Por que Migrar?

- Grandes melhorias em performance e UX
- Compatibilidade Reactive vs. Mobile
- Não existe migração automatizada, pois:
  - Os paradigmas são muito diferentes
  - Seria perdida uma grande oportunidade para refinamento da aplicação (otimizações de performance e UX)



# Com Grandes Poderes...

- Vêm novos conceitos para aprender!
- Patterns UX-First
- Interfaces/blocos direcionados a eventos
- Contexto Cliente vs. Servidor
- Conceitos de segurança
- “Cadê o Preparation?”



# Estratégias de Migração

## Big-Bang

- Grande entrega do app completo e “virada de a chave”
- Pouco ou nenhum feedback do usuário
- Pouca aderência com metodologias ágeis

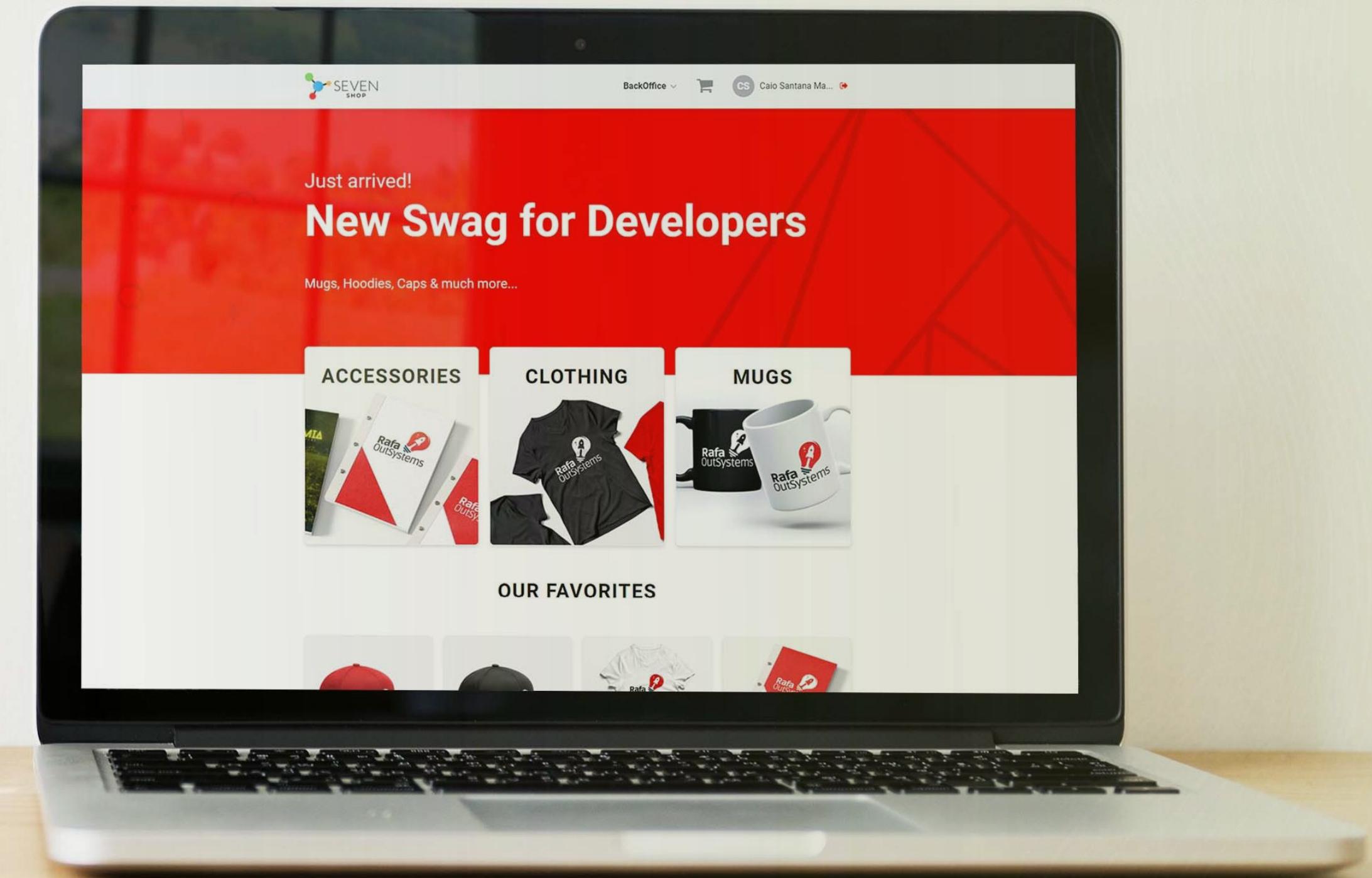
## Faseada

- Entregas constantes de pequenas partes da aplicação
- Ideal para metodologias ágeis
- Melhor ROI

# SevenShop

## Projeto Migração

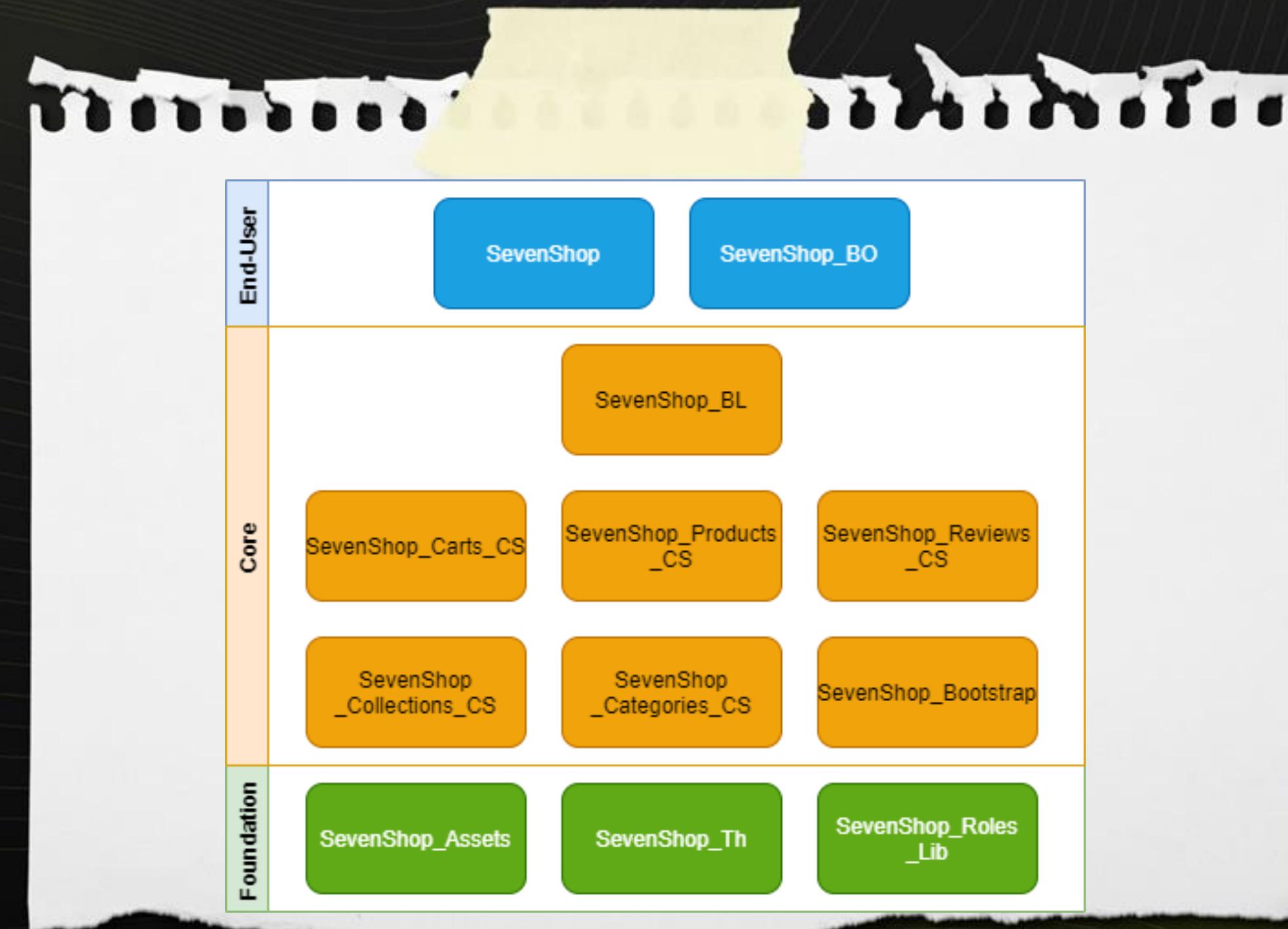
- Aplicação de demonstração do tipo e-commerce em Traditional Web
- Modelado seguindo Architecture Canvas
- Front-Office + Back-Office
- 12 módulos



# Demonstração

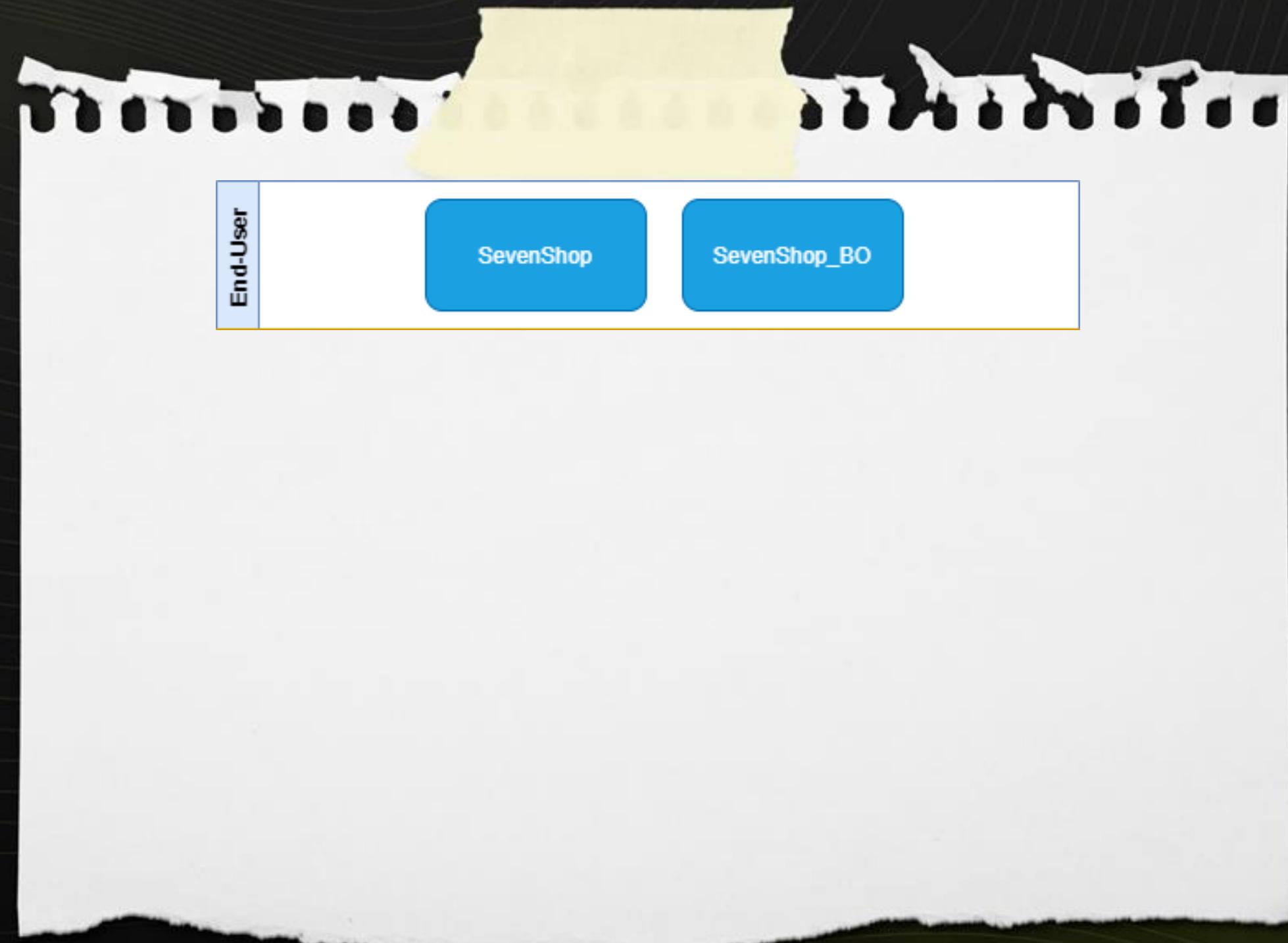
# Plano de Ataque

1. Diagramar arquitetura



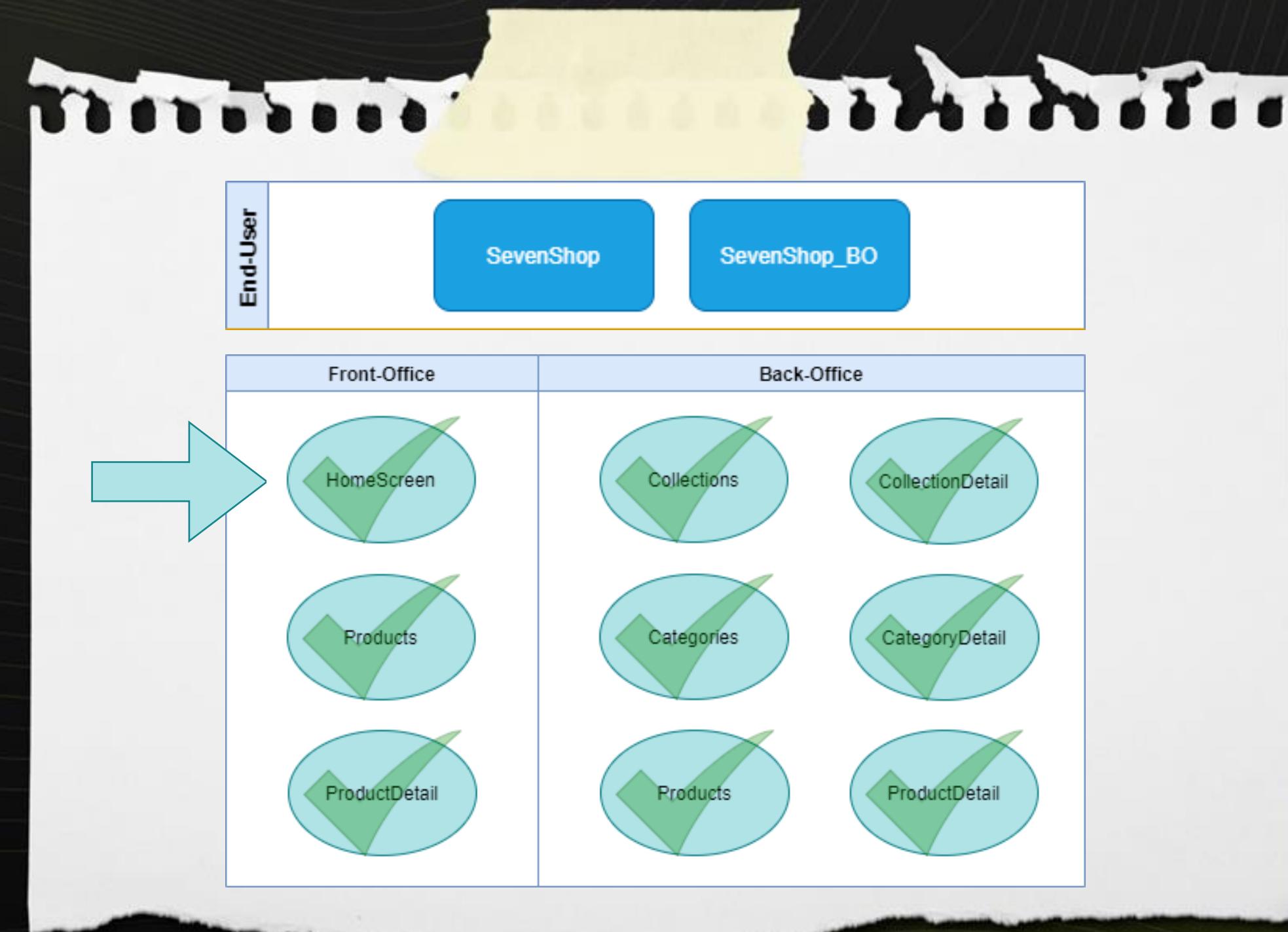
# Plano de Ataque

1. Diagramar arquitetura
2. Atuar à partir de módulos End-User



# Plano de Ataque

1. Diagramar arquitetura
2. Atuar à partir de módulos End-User
3. Escolher uma das telas Traditional Web para migrar (estratégia faseada)
4. Re-criar tela em Reactive usando técnicas para integrar App Antigo ⇔ App Novo
5. Repetir o processo até migrar toda a aplicação



# Na Próxima Aula

- Reestruturação dos módulos do SevenShop Traditional Web
- Preparação para migração
- Desenvolvimento inicial do SevenShop Reactive



# RAFA PRÓXIMA

ACADEMIA  
RAFA OUTSYSTEMS