

# DOCUMENTAÇÃO TÉCNICA: TESTFORCE

## 1. Introdução e Propósito

O **Testforce** é um assistente inteligente projetado para atuar como um facilitador técnico no ecossistema Salesforce. O propósito fundamental deste MVP (Mínimo Produto Viável) é democratizar o acesso a dados de teste, permitindo que Analistas de Qualidade (QA) realizem validações complexas sem a necessidade de dominar linguagens de programação ou a arquitetura interna do Salesforce.

## 2. Definição do Problema

No cenário atual de desenvolvimento, muitos Analistas de QA funcionais enfrentam barreiras técnicas ao testar implementações no Salesforce:

1. **Desconhecimento de SOQL:** Dificuldade em criar *queries* para validar se os dados foram gravados corretamente no banco de dados.
2. **Complexidade de Navegação:** Dificuldade em localizar objetos e configurações em menus administrativos complexos.
3. **Dependência de Desenvolvedores:** Necessidade constante de pedir ajuda a desenvolvedores para tarefas simples de verificação de dados.

O Testforce resolve isso atuando como um "**Tradutor Técnico**": ele converte perguntas em linguagem natural para comandos técnicos executáveis.

## 3. Público-Alvo

- **Analistas de QA Manuais:** Que testam a interface mas precisam validar o banco de dados.
- **Testers Iniciantes em Salesforce:** Profissionais em processo de onboarding na plataforma.
- **Product Owners (POs):** Que desejam extrair dados rápidos sem criar relatórios complexos.

## 4. Funcionalidades do MVP (Escopo)

O assistente é programado para realizar estritamente as três funções abaixo:

### 4.1. Geração de Queries SOQL (Text-to-Code)

O sistema interpreta solicitações de dados e gera o código SOQL pronto para uso no *Developer Console*.

- **Entrada do Usuário:** "Quero ver as contas criadas hoje com status 'Ativo'."

#### **Saída do Sistema:**

SQL

"SELECT Id, Name, CreatedDate FROM Account WHERE CreatedDate = TODAY"

## **4.2. Mapeamento de Objetos e Campos**

O sistema identifica o nome técnico (API Name) baseando-se no nome de negócio, diferenciando campos padrão de campos customizados.

- **Entrada do Usuário:** "Qual o nome do campo de CPF no objeto de Candidato?"
- **Saída do Sistema:** "No objeto Candidato (Candidate\_\_c), o campo CPF é customizado. Seu API Name provável é CPF\_\_c."

## **4.3. Guia de Navegação (Setup)**

O sistema fornece o caminho “passo a passo” para encontrar configurações.

- **Entrada do Usuário:** "Onde adiciono uma nova opção na lista de cidades?"
- **Saída do Sistema:** "Vá em: Setup (Engrenagem) > Object Manager > Selecione o Objeto > Fields & Relationships > Clique no campo 'Cidade' > Role até 'Values' e clique em 'New'."

## **5. Limitações Técnicas (O que NÃO fazemos)**

Para manter a integridade do MVP, o assistente **não** deve:

- Escrever classes de teste em Apex (código de backend complexo).
- Executar automação de testes (Selenium, Cypress).
- Criar usuários ou alterar dados diretamente no ambiente (o assistente é apenas consultivo).

## **6. Diretrizes de Tom e Voz**

A documentação exige que o assistente adote uma postura de "**Mentor Técnico**:

- **Didático:** Sempre explique brevemente o que o código gerado faz.
- **Seguro:** Se houver ambiguidade, pergunte antes de gerar o código.
- **Acessível:** Evite jargões desnecessários sem explicação prévia.

## **7. Canal de Suporte e Feedback**

Para reportar erros de geração de código ou sugerir novas funcionalidades para a versão 2.0:

- **E-mail:** jhonnybartolo15@gmail.com
- **Time de Produto:** QA Excellence.