

Alunos: Breno Assis e Higor Nogueira

## 1. Análise Organizacional

### Descrição

Empresa: "Stylo Model"

Ramo: Varejo de vestuário e acessórios (moda jovem/casual).

Porte: Pequena empresa (Microempresa).

Missão: "Oferecer moda acessível e um atendimento personalizado à comunidade local, criando uma experiência de compra amigável e próxima."

Estrutura Organizacional:

- Nível 1 (Proprietário/Gerente): Acumula funções estratégicas (compras, finanças) e táticas.
- Nível 2 (Vendedores/Caixa): 1 ou 2 funcionários que executam as operações de venda, atendimento e organização da loja.

### Identificação dos principais processos de negócio e problemas informacionais

Gestão de Vendas: Atender o cliente, registrar os itens, aplicar descontos, receber o pagamento e fechar o caixa.

- Controle Manual: Uso de caderno, calculadora ou planilha básica.
- Erros no Caixa: Dificuldade em fechar o caixa no fim do dia; erros de troco.
- Sem Métricas: Impossível saber o ticket médio ou os horários de pico de forma confiável

Gestão de Estoque: Receber mercadorias, cadastrar produtos, etiquetar, armazenar e dar baixa no momento da venda.

- Furo de Estoque: Vender um produto que não existe mais (muito comum).
- Contagem Lenta: Necessidade de fechar a loja para fazer balanço (contagem manual).
- Estoque Parado: Não saber quais produtos estão encalhados há meses.

Gestão de Compras: Identificar quais produtos precisam ser repostos e negociar com fornecedores.

- Decisão "no tato": Comprar baseado em "achismo" e não em dados.
- Risco de Ruptura: Só perceber que um item campeão de vendas acabou quando o cliente pede.

Gestão Financeira: Controlar o fluxo de caixa, contas a pagar (fornecedores, aluguel) e contas a receber (vendas "no fiado").

- "Caderninho do Fiado": Alto risco de inadimplência e dificuldade de cobrança.
- Sem Visão de Lucro: Não saber o lucro líquido real da loja, apenas o faturamento bruto.

**Gestão de Clientes:** Manter um cadastro (muitas vezes em agenda ou WhatsApp) e avisar sobre promoções.

- Falta de Histórico: Não saber quem são os melhores clientes ou o que eles compraram no passado.
- Marketing Ineficiente: Dificuldade em segmentar clientes para promoções (ex: "avisar só quem compra calça jeans").

### **Justificativa da necessidade do sistema**

A loja opera em um modo de gestão reativo e baseado na intuição, o que é insustentável para o crescimento. A falta de integração entre Vendas e Estoque gera perda de vendas e insatisfação do cliente. O controle financeiro manual em cadernos ou planilhas é frágil, propenso a erros e não fornece uma visão clara da saúde financeira do negócio.

Um sistema de informação é justificado pela necessidade urgente de:

1. Automatizar processos operacionais (vendas, baixa de estoque).
2. Centralizar as informações (saber exatamente o que tem, o que vendeu e quanto lucrou).
3. Fornecer dados confiáveis para a tomada de decisão (comprar melhor e gerir o caixa).

## **2. Papel do Sistema de Informação**

### **Classificação do sistema**

1. Sistema de Processamento de Transações (SPT) / Sistema Transacional (SIT): Esta é a sua função base. Ele deve ser excelente em registrar as transações diárias de forma rápida e confiável: vendas (PDV - Ponto de Venda), entradas de estoque, pagamentos e recebimentos.
2. Sistema de Gestão Comercial (ou "Mini-ERP"): Por integrar os dados do SPT (Vendas) com outros módulos essenciais como Estoque e Financeiro, ele funciona como um pequeno Sistema de Gestão Integrada (ERP) focado nas necessidades do varejo.
3. Com Módulo CRM (Opcional): Se o sistema incluir o cadastro detalhado de clientes e o histórico de compras para ações de marketing, ele também incorpora funcionalidades de CRM (Customer Relationship Management).

### **Níveis organizacionais e funções que ele apoia**

O sistema proposto deve apoiar todos os três níveis da organização:

Nível Operacional (O Dia-a-Dia):

- Função: Vendedores e Caixa.
- Apoio do Sistema: Módulo de Ponto de Venda (PDV) ágil para registrar vendas, consultar preços, verificar disponibilidade de tamanhos/cores no estoque e fechar o caixa.

Nível Tático (O Gerenciamento):

- Função: Proprietário/Gerente.
- Apoio do Sistema: Módulos de Relatórios Gerenciais. O gerente usará o sistema para responder perguntas como:
  - "Qual o produto mais vendido este mês?" (Relatório de Curva ABC de Produtos).
  - "Preciso pagar quais fornecedores esta semana?" (Relatório de Contas a Pagar).
  - "Quanto tenho em estoque da nova coleção?" (Relatório de Posição de Estoque).
  - "Quem está me devendo?" (Relatório de Contas a Receber / "Fiado").

Nível Estratégico (O Futuro):

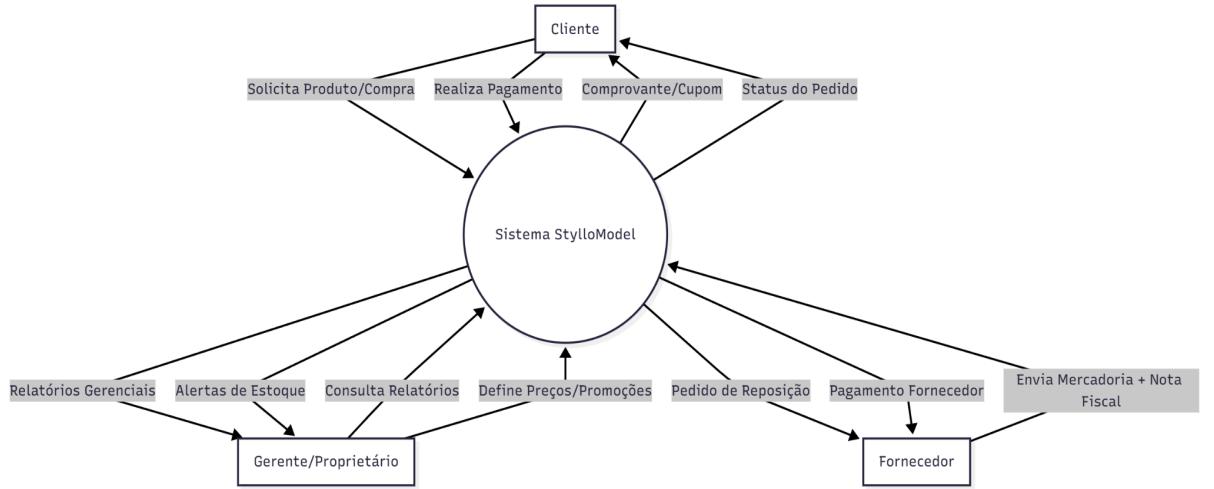
- Função: Proprietário (em seu papel de estrategista).
- Apoio do Sistema: Módulos de Análise e BI (Business Intelligence), mesmo que simples. O proprietário usará os dados históricos para decidir:
  - "Qual estação do ano dá mais lucro?"
  - "Vale a pena abrir uma segunda loja?"
  - "Quais marcas/fornecedores me dão maior margem de lucro?"

### **Benefícios esperados para a empresa**

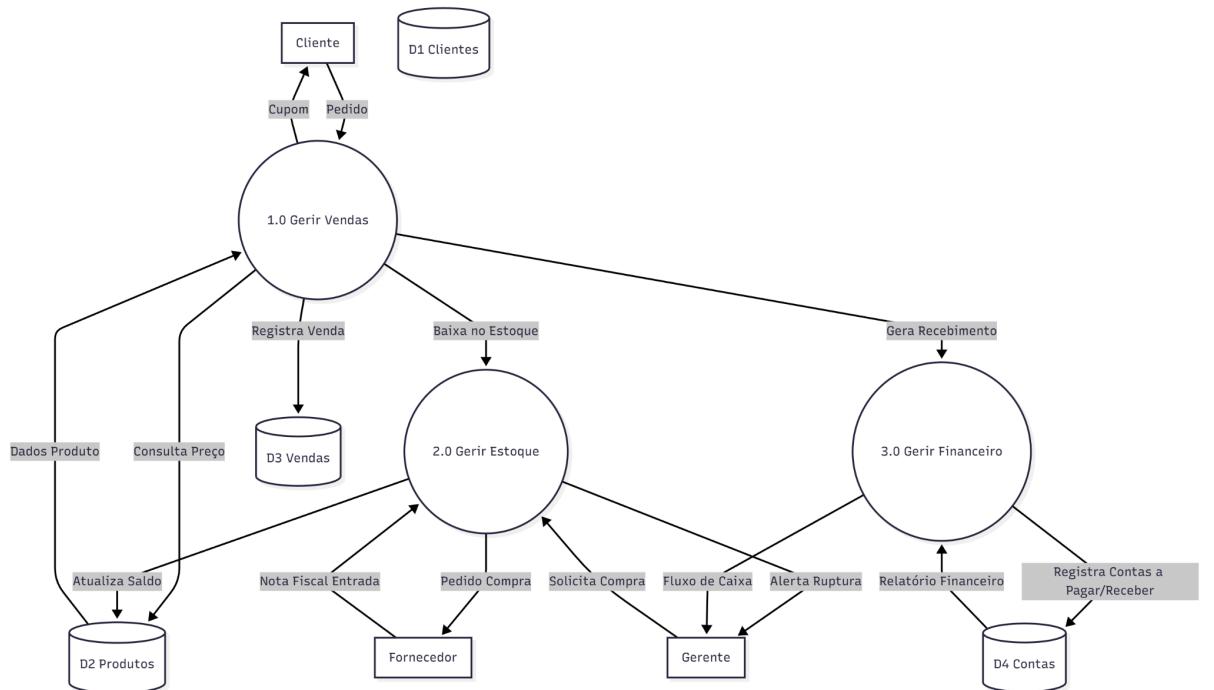
1. Redução de Perdas: Controle de estoque em tempo real, diminuindo perdas por "furo" de estoque ou produtos encalhados.
2. Confiabilidade Financeira: Visão clara do fluxo de caixa, controle rigoroso do "fiado" (contas a receber) e das contas a pagar.
3. Aumento da Eficiência Operacional: Agilidade no caixa (PDV rápido), automatização da baixa de estoque e facilidade para fazer o balanço (contagem de estoque).
4. Tomada de Decisão Baseada em Dados: O proprietário deixará de comprar "no tato" e passará a comprar com base em relatórios de vendas, melhorando o mix de produtos.
5. Melhoria no Atendimento: Capacidade de consultar o estoque rapidamente (ex: "tem esse vestido no tamanho P?") e conhecer o histórico do cliente.
6. Aumento da Lucratividade: Como resultado da redução de perdas e do aumento da eficiência nas vendas e compras.

### **3. Diagrama de Fluxo de Dados (DFD)**

- DFD Nível 0 (Diagrama de Contexto): Representa o sistema "Stylo Model" como um processo único central, interagindo com as entidades externas: Cliente (que gera a demanda), Gerente (que consome informações para decisão) e Fornecedor (que alimenta o estoque). O foco aqui é visualizar as fronteiras do sistema



- DFD Nível 1 (Diagrama de Processos): Decompõe o sistema central em seus três macroprocessos principais: Gestão de Vendas, Controle de Estoque e Gestão Financeira, evidenciando os depósitos de dados (tabelas do banco de dados) compartilhados entre eles.



## **Identificação das Entradas, Processamento e Saídas**

Realizar Venda:

- Entradas: Código de barras do produto, quantidade, forma de pagamento, CPF do Cliente (opcional).
- Processamento: Validar existência do produto, recuperar preço unitário, calcular subtotais e totais, aplicar descontos (se houver), efetuar baixa automática no estoque.
- Saídas: Cupom Não Fiscal/Recibo, registro de venda no banco de dados, atualização do saldo de estoque.

Gerenciar Estoque:

- Entradas: Nota Fiscal do Fornecedor, dados do produto (tamanho, cor, marca), preço de custo e margem de lucro desejada.
- Processamento: Verificar se o produto já existe no sistema (decidir entre novo cadastro ou atualização), somar a quantidade recebida ao saldo atual, calcular o preço de venda sugerido.
- Saídas: Produto cadastrado ou atualizado no sistema, geração de etiquetas de código de barras, registro no histórico de entradas.

Controle Financeiro:

- Entradas: Registro das vendas realizadas (diário), contas a pagar (fornecedores, despesas fixas), contas a receber (vendas a prazo/fiado).
- Processamento: Somar as entradas de receita, subtrair as saídas de despesas, calcular o saldo operacional do dia (fechamento de caixa).
- Saídas: Relatório de Fluxo de Caixa, Demonstrativo de Resultado (DRE) simplificado, alertas de contas a vencer.

## **Fontes de Dados, Usuários e Subsistemas**

Fontes de Dados

As informações que alimentam o sistema provêm de três fontes primárias:

1. Clientes: Fornecem dados pessoais para cadastro e fidelização, além da demanda de compra.
2. Fornecedores: Fornecem dados dos produtos (catálogo, preços de custo, prazos de entrega).
3. Gestão Interna: O gerente define parâmetros como margem de lucro, limites de desconto e cadastro de usuários do sistema.

Usuários do Sistema

- Vendedor/Caixa (Nível Operacional):
  - Acesso: Restrito às operações de venda, consulta de preços e consulta de estoque. Não tem permissão para alterar estoques manualmente ou ver o lucro da empresa.
- Gerente/Administrador (Nível Tático/Estratégico):
  - Acesso: Total (Full Access). Responsável por cadastrar produtos, dar entrada em notas, visualizar relatórios financeiros e gerenciar permissões de outros usuários.

## Subsistemas Interligados

O sistema é estruturado em módulos integrados, onde a saída de um serve de entrada para o outro, garantindo a consistência dos dados:

1. Subsistema de Vendas (Front-end/PDV): Interface ágil para o atendimento ao cliente.
2. Subsistema de Estoque (Back-end/Logística): Responsável pela guarda e conferência física e lógica dos itens.
3. Subsistema Financeiro (Back-end/Adm): Consolida as informações monetárias geradas pelas vendas e compras.
4. Subsistema de Relatórios (BI Básico): Módulo transversal que extrai dados de todos os outros subsistemas para gerar gráficos e indicadores de desempenho para a gerência.

## 4. Protótipo

O protótipo foi desenvolvido utilizando HTML, CSS e JavaScript para validar a interface e a experiência do usuário. O projeto abrange a navegação principal, incluindo a página institucional e de contato, demonstrando a identidade visual da marca "Stylo Model". O código-fonte completo do front-end e o protótipo navegável estão disponíveis para consulta no repositório oficial do projeto (link na Seção 7).

## 5. Casos de Teste e Implementação com JUnit

```

Test Results ×
Tests:StyloModeljar:1.0-SNAPSHOT (Unit) ×
  Tests passed: 100,00 %
  All 10 tests passed. (0,118 s)

Output - Test (ProdutoTest) ×
  --- compiler:3.11.0:testCompile (default-testCompile) @ StyloModel ---
  Nothing to compile - all classes are up to date

  --- surefire:3.2.5:test (default-cli) @ StyloModel ---
  Using auto detected provider org.apache.maven.surefire.junitplatform.JUnitPlatformProvider

  -----
  T E S T S
  -----
  Running teste.stylomodel.ProdutoTest
  Tests run: 10, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.118 s -- in teste.stylomodel.ProdutoTest
  Results:
  Tests run: 10, Failures: 0, Errors: 0, Skipped: 0

  -----
  BUILD SUCCESS
  -----
  Total time: 3.746 s
  Finished at: 2025-11-19T20:39:01-03:00
  -----

```

## **Contribuição dos Testes para a Qualidade e Confiabilidade**

A implementação da suíte de testes unitários com JUnit 5 desempenha um papel fundamental na garantia da qualidade do software desenvolvido para a Styllo Model, elevando a confiabilidade do sistema em três pilares principais:

1. Validação Automática das Regras de Negócio Os testes funcionam como "guardiões" da lógica do sistema. Ao implementar casos como testVendaSemEstoque e testPrecoNegativo, garantimos matematicamente que o sistema jamais permitirá operações financeiramente perigosas, como vender produtos que não existem fisicamente ou cadastrar itens com preços inválidos. Isso previne prejuízos operacionais diretos para a loja.
2. Segurança na Manutenção (Testes de Regressão) Um dos maiores riscos no desenvolvimento de software é que uma correção em uma parte do código quebre outra funcionalidade que já funcionava (regressão). Com os 10 testes automatizados, qualquer alteração futura na classe Produto pode ser validada instantaneamente. Se uma mudança na lógica de desconto quebrar o cálculo de vendas, o teste testCalculoDesconto falhará imediatamente, alertando o desenvolvedor antes que o erro chegue ao cliente.
3. Confiabilidade dos Dados (Integridade) Para um sistema Transacional (SPT) e de Gestão Comercial, a precisão dos dados é crítica. Os testes testVendaSucesso e testReposicaoEstoque asseguram que as operações de soma e subtração no banco de dados (simulado pelos atributos da classe) ocorram com precisão exata, garantindo que o relatório de estoque final corresponda à realidade física da loja.

Em suma, o uso do JUnit transforma a verificação de qualidade de um processo manual e sujeito a falhas humanas para um processo automático, repetível e auditável, entregando à Styllo Model um software robusto e confiável.

## **6. Segurança e Governança da Informação**

Considerando que a *Styllo Model* lida com dados sensíveis (informações de clientes, preços de custo e fluxo de caixa), a segurança da informação é tratada como um requisito não funcional crítico. Abaixo, detalhamos os riscos identificados e as medidas de governança implementadas, alinhadas aos pilares da segurança da informação (Confidencialidade, Integridade, Disponibilidade, Autenticidade e Não Repúdio).

### **Identificação de Riscos e Ameaças**

Para o ambiente da microempresa, foram mapeados os seguintes riscos prioritários:

1. Acesso Indevido (Privilégio Excessivo): Risco de um usuário de nível operacional (ex: Vendedor) acessar módulos gerenciais, visualizando o lucro da empresa ou alterando preços sem autorização.
2. Perda de Dados (Falha ou Sinistro): Risco de falha no hardware (computador queimar) ou ataque de *ransomware*, resultando na perda total do histórico de vendas e contas a receber.

3. **Fraudes Internas (Manipulação de Dados):** Risco de alteração de registros de vendas ou exclusão de transações para desvio de valores.
4. **Vazamento de Dados de Clientes:** Exposição de dados pessoais (LGPD), como telefones e endereços, por falta de proteção no banco de dados.

## **Medidas de Segurança Propostas**

Para mitigar os riscos acima, o sistema implementa as seguintes políticas e mecanismos técnicos:

### A. Controle de Acesso e Autenticação

- Medida: Implementação de tela de login obrigatória com validação de credenciais (usuário e senha) e divisão de perfis de acesso (*Role-Based Access Control*).
  - Perfil Vendedor: Acesso apenas ao PDV e consulta de estoque.
  - Perfil Gerente: Acesso total (Relatórios, Cadastro de Produtos, Usuários).
- Princípios Atendidos: Confidencialidade (garante que informação só seja acessível a quem tem autorização) e Autenticidade (garante que o usuário é quem diz ser).

### B. Criptografia de Senhas

- Medida: As senhas dos usuários não são armazenadas em texto plano no banco de dados. Utiliza-se um algoritmo de *hash* seguro (ex: BCrypt ou SHA-256) antes da persistência.
- Princípios Atendidos: Confidencialidade (mesmo se o banco for invadido, as senhas reais não são expostas).

### C. Logs de Auditoria (Trilha de Auditoria)

- Medida: O sistema registra automaticamente ações críticas, como exclusão de vendas, aplicação de descontos acima de um limite ou alterações manuais de estoque. O log grava: *ID do Usuário + Ação + Data/Hora*.
- Princípios Atendidos: Não Repúdio (o usuário não pode negar que realizou a ação) e Integridade (monitoramento contra alterações indevidas).

### D. Rotina de Backup

- Medida: Script automatizado para gerar cópia de segurança do banco de dados (dump SQL) ao final do expediente, salvo em diretório local e sincronizado com nuvem (Google Drive/OneDrive corporativo).
- Princípios Atendidos: Disponibilidade (garante que os dados estejam acessíveis e recuperáveis mesmo após falhas críticas).

## **Riscos x Soluções**

A relação entre os riscos identificados, as medidas técnicas adotadas e os pilares de segurança atendidos está descrita abaixo:

- **Risco:** Acesso indevido a informações estratégicas (ex: visualização do lucro por vendedores).

- Solução: Implementação de Controle de Acesso Baseado em Papéis (RBAC).
  - Pilar de Segurança: Confidencialidade.
- Risco: Vazamento de credenciais em caso de invasão ao Banco de Dados.
  - Solução: Armazenamento de senhas com Criptografia (Hash).
  - Pilar de Segurança: Confidencialidade.
- Risco: Perda total de dados por falha de hardware ou sinistro.
  - Solução: Execução de Rotina de Backup sincronizada em nuvem.
  - Pilar de Segurança: Disponibilidade.
- Risco: Negação de autoria em ações críticas (ex: exclusão de uma venda).
  - Solução: Registro automático de Logs de Auditoria (quem, quando e o que fez).
  - Pilar de Segurança: Não Repúdio e Integridade.
- Risco: Alteração não autorizada de cadastros (ex: mudança de preços).
  - Solução: Exigência de autenticação válida para permissão de edição.
  - Pilar de Segurança: Integridade e Autenticidade.

## 7. Versionamento e Repositório (Git)

Para garantir a integridade do código e o histórico de evolução do projeto, foi utilizado o sistema de versionamento Git. O repositório contém a estrutura completa do projeto, dividida em código-fonte, documentação e protótipos.

- Link do Repositório Público: <https://github.com/brenoassisX/stylomodel-project>

## 8. Conclusão e Integração

O desenvolvimento do sistema para a *Stylo Model* permitiu a aplicação prática dos conceitos teóricos abordados nas disciplinas de Sistemas de Informação e Engenharia de Software, resultando em uma solução tecnológica alinhada às necessidades reais de uma microempresa de varejo.

### Resolução dos Problemas Informacionais

O sistema proposto atinge seu objetivo principal ao transformar o modelo de gestão da empresa, anteriormente baseado em processos manuais e intuição, para um modelo informatizado e orientado a dados.

- Operacional: A automação do Ponto de Venda (PDV) elimina os erros de cálculo e agiliza o atendimento, resolvendo o gargalo das filas e do fechamento de caixa.
- Tático: A integração automática entre Vendas e Estoque mitiga o problema crítico de "furo de estoque" e compras desnecessárias, pois o sistema garante que o saldo físico corresponda ao saldo lógico.
- Estratégico: A geração de relatórios financeiros e de vendas substitui o "achismo" por métricas concretas, permitindo que o proprietário tome decisões de compra e investimento com base na lucratividade real e no comportamento do consumidor.

### Integração entre Sistemas de Informação e Engenharia de Software

Este projeto evidenciou que a tecnologia não é um fim em si mesma, mas um meio para agregar valor ao negócio.

- A disciplina de Sistemas de Informação foi fundamental para a etapa de análise, permitindo compreender a estrutura organizacional, mapear os fluxos de dados (DFDs) e definir o que precisava ser construído para apoiar os níveis operacional, tático e estratégico.
- A disciplina de Engenharia de Software forneceu o ferramental técnico para definir como construir a solução com qualidade. A utilização de práticas como Versionamento de Código (Git), Testes Unitários (JUnit) e levantamento de Requisitos Funcionais garantiu que o software entregue fosse robusto, manutenível e auditável.

A integração dessas duas áreas permitiu entregar não apenas um código que funciona, mas um Sistema de Informação que resolve problemas de negócio.

## Aprendizados e Limitações

Durante o ciclo de desenvolvimento, consolidamos aprendizados importantes:

- Importância dos Testes: A implementação de casos de teste com JUnit demonstrou como a prevenção de falhas na etapa de desenvolvimento economiza tempo e garante a integridade das regras de negócio (ex: não permitir venda com estoque negativo).
- Colaboração: O uso do Git forçou a organização do trabalho em equipe, simulando um ambiente real de desenvolvimento colaborativo.

Limitações e Trabalhos Futuros: Como limitações do projeto atual, destacamos que o sistema foi desenvolvido como um protótipo focado no *back-office* (Desktop). Para evoluções futuras, sugere-se:

1. Migração para uma arquitetura Web ou Mobile, permitindo que o gerente acesse relatórios remotamente.
2. Implementação de um módulo de Nota Fiscal Eletrônica (NFC-e) para conformidade fiscal real.
3. Integração com meios de pagamento digitais (API de PIX e Cartões).

Em suma, o projeto da Styllo Model cumpre os requisitos acadêmicos e apresenta um MVP (Minimum Viable Product) funcional, capaz de iniciar o processo de transformação digital da empresa.