

DOCUMENTO DE ESPECIFICAÇÃO ÁGIL

Sistema: Leon's Cupcake - E-commerce Mobile de Cupcakes Gourmet

Versão: 2.5

Data: 17 de Novembro de 2025

Equipe: Desenvolvimento Leon's Cupcake

Metodologia: Scrum + TDD (Test-Driven Development)

Framework: SAFe (Scaled Agile Framework) adaptado

SUMÁRIO

1. Introdução
2. Visão do Produto
3. Metodologia Ágil
4. Stack Tecnológico
5. Épicos e Features
6. Product Backlog
7. Histórias de Usuário
8. Definition of Ready (DoR)
9. Definition of Done (DoD)
10. Planejamento de Sprints
11. Critérios de Aceitação
12. Métricas e KPIs

1. INTRODUÇÃO

1.1 Propósito do Documento

Este documento apresenta a especificação ágil completa do projeto **Leon's Cupcake**, sistema de e-commerce mobile para venda de cupcakes gourmet, contemplando:

- Histórias de usuário priorizadas
- Product backlog organizado
- Critérios de aceitação detalhados
- Planejamento de sprints
- Definições de pronto (DoR e DoD)

1.2 Escopo

O sistema atende três perfis principais:

- **Cliente:** Navegação, compra, pagamento e acompanhamento
- **Entregador:** Gestão de entregas e atualização de status
- **Administrador:** Gestão completa do sistema

1.3 Público-Alvo

- Product Owner
- Scrum Master
- Time de Desenvolvimento
- Stakeholders
- QA Team

2. VISÃO DO PRODUTO

2.1 Declaração de Visão

"Proporcionar uma experiência digital completa e intuitiva para compra de cupcakes gourmet, conectando clientes a produtos de qualidade através de uma plataforma mobile segura, rápida e acessível."

2.2 Objetivos do Produto

1. **Acessibilidade:** Interface intuitiva e responsiva
2. **Segurança:** Autenticação JWT e pagamentos seguros
3. **Performance:** Carregamento < 2s
4. **Escalabilidade:** Arquitetura preparada para crescimento
5. **Disponibilidade:** 99.9% de uptime

2.3 Proposta de Valor

Para Clientes:

- ☒ Catálogo completo com 40+ sabores
- ☒ Processo de compra simplificado
- ☒ Múltiplas formas de pagamento
- ☒ Rastreamento em tempo real
- ☒ Notificações automáticas

Para Administradores:

- ☒ Dashboard completo de vendas
- ☒ Gestão de estoque automatizada
- ☒ Relatórios detalhados
- ☒ Controle total de usuários

Para Entregadores:

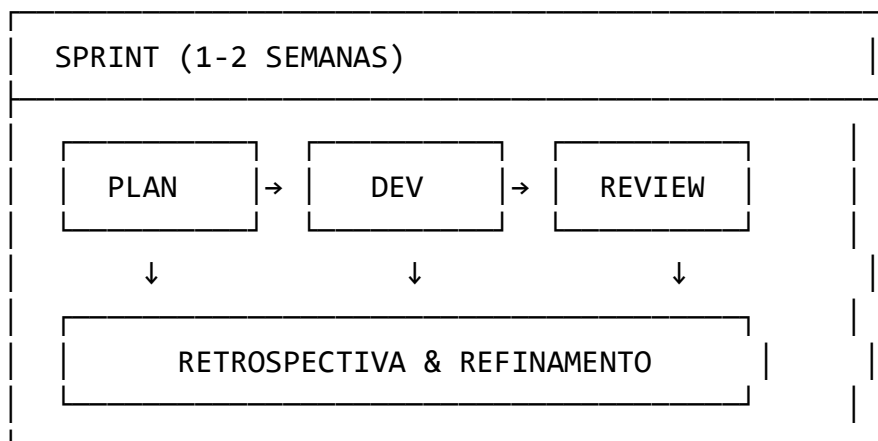
- ☒ Lista de entregas organizada
- ☒ Atualização fácil de status
- ☒ Histórico de entregas realizadas

3. METODOLOGIA ÁGIL

3.1 Framework Adotado

Scrum com adaptações para time pequeno (Solo Scrum quando aplicável)

3.2 Ciclo de Desenvolvimento






3.3 Cerimônias Scrum

Cerimônia	Duração	Frequência	Participantes
Sprint Planning	2-4h	Início de Sprint	Todo o time
Daily Standup	15min	Diária	Time dev
Sprint Review	1-2h	Fim de Sprint	Time + Stakeholders
Retrospectiva	1h	Fim de Sprint	Todo o time
Backlog Refinement	1h	Semanal	PO + Time

3.4 TDD (Test-Driven Development)

Ciclo Red-Green-Refactor:

1.  RED: Escrever teste que falha
2.  GREEN: Implementar código mínimo para passar
3.  REFACTOR: Refatorar mantendo testes passando

Cobertura de Testes:

- Unitários: > 80%
- Integração: > 70%
- E2E: Fluxos críticos

4. STACK TECNOLÓGICO

4.1 Frontend

Tecnologia	Versão	Propósito
Angular	17+	Framework principal
Ionic	7+	UI Components mobile
TypeScript	5+	Linguagem
RxJS	7+	Programação reativa
Capacitor	5+	Acesso nativo mobile

4.2 Backend

Tecnologia	Versão	Propósito
------------	--------	-----------

Python	3.9+	Linguagem
Flask	3.0+	Web framework
SQLAlchemy	2.0+	ORM
Flask-JWT-Extended	4.5+	Autenticação
Bcrypt	4.0+	Hash de senhas

4.3 Banco de Dados

Tecnologia	Versão	Propósito
MySQL	8.0+	Banco relacional
Redis	7+	Cache (futuro)

4.4 DevOps & Infraestrutura

Ferramenta	Propósito
Git/GitHub	Controle de versão
Docker	Containerização
GitHub Actions	CI/CD
Pytest	Testes Python
Jasmine/Karma	Testes Angular

4.5 Ferramentas de Gestão

Ferramenta	Propósito
Jira/Trello	Gestão de backlog
Figma	Design UI/UX
Draw.io	Diagramas UML
MindMeister	Mapas mentais
Lucidchart	Modelagem

5. ÉPICOS E FEATURES

ÉPICO 1: Gestão de Usuários e Autenticação

Features:

- F1.1: Cadastro de usuários
- F1.2: Login e logout
- F1.3: Recuperação de senha
- F1.4: Perfil de usuário
- F1.5: Gerenciamento de endereços

Valor de Negócio: ALTO

Prioridade: P0 (Crítico)

ÉPICO 2: Catálogo e Produtos

Features:

- F2.1: Listagem de produtos
- F2.2: Detalhes do produto
- F2.3: Busca e filtros
- F2.4: Categorização
- F2.5: Gerenciamento de estoque (Admin)

Valor de Negócio: ALTO

Prioridade: P0 (Crítico)

ÉPICO 3: Carrinho e Checkout

Features:

- F3.1: Adicionar ao carrinho
- F3.2: Gerenciar carrinho
- F3.3: Processo de checkout
- F3.4: Cálculo de valores
- F3.5: Validação de estoque

Valor de Negócio: ALTO

Prioridade: P0 (Crítico)

ÉPICO 4: Pagamentos

Features:

- F4.1: Integração Mercado Pago
- F4.2: Pagamento PIX
- F4.3: Pagamento Cartão
- F4.4: Confirmação de pagamento
- F4.5: Tratamento de falhas

Valor de Negócio: ALTO

Prioridade: P0 (Crítico)

ÉPICO 5: Pedidos e Acompanhamento

Features:

- F5.1: Criação de pedidos
- F5.2: Listagem de pedidos
- F5.3: Detalhes do pedido
- F5.4: Status do pedido
- F5.5: Cancelamento (quando permitido)

Valor de Negócio: ALTO

Prioridade: P0 (Crítico)

ÉPICO 6: Entregas

Features:

- F6.1: Atribuição de entregador
- F6.2: Lista de entregas (entregador)
- F6.3: Atualização de status
- F6.4: Confirmação de entrega
- F6.5: Histórico de entregas

Valor de Negócio: MÉDIO

Prioridade: P1 (Alto)

ÉPICO 7: Administração

Features:

- F7.1: Dashboard administrativo
- F7.2: CRUD de produtos
- F7.3: Gerenciamento de usuários
- F7.4: Relatórios de vendas
- F7.5: Estatísticas

Valor de Negócio: MÉDIO

Prioridade: P1 (Alto)

ÉPICO 8: Notificações

Features:

- F8.1: Notificações push
- F8.2: Notificações de status
- F8.3: Configurações de notificação

Valor de Negócio: BAIXO

Prioridade: P2 (Médio)

6. PRODUCT BACKLOG

6.1 Backlog Priorizado

ID	História de Usuário	Épico	Story Points	Prioridade	Sprint
HU01	Cadastro e Login	E1	3	P0	1
HU02	Visualizar Catálogo	E2	5	P0	1
HU03	Adicionar ao Carrinho	E3	3	P0	1
HU04	Efetuar Pagamento	E4	5	P0	2
HU05	Acompanhar Status	E5	3	P0	2
HU06	Avaliar Pedido	E5	2	P2	4
HU07	Notificações Push	E8	2	P2	4
HU08	Recomendações ML	E2	5	P2	5
HU09	Filtrar Produtos	E2	2	P1	3
HU10	Cancelar Pedido	E5	3	P1	3
HU11	Gerenciar Produtos	E7	5	P1	2
HU12	Relatórios	E7	3	P1	3
HU13	Cupons Desconto	E3	2	P2	5
HU14	Custos Ingredientes	E7	3	P2	4

HU15	Pipeline CI/CD	-	5	P0	1
HU16	Monitoramento	-	3	P1	3
HU17	Rollback	-	3	P2	4
HU18	Treinar Modelo ML	E2	5	P2	5
HU19	Monitorar Drift	E2	3	P2	5
HU20	Segurança LGPD	-	8	P0	1-2

6.2 Legenda de Prioridades

- **P0:** Crítico - Bloqueante para MVP
- **P1:** Alto - Importante para release
- **P2:** Médio - Desejável mas não essencial
- **P3:** Baixo - Pode ser postergado

6.3 Story Points (Fibonacci)

- **1 ponto:** 1-2 horas
- **2 pontos:** 2-4 horas
- **3 pontos:** 1 dia
- **5 pontos:** 2-3 dias
- **8 pontos:** 1 semana
- **13 pontos:** Épico (quebrar)

7. HISTÓRIAS DE USUÁRIO

HU01 - Cadastro e Login do Cliente

Como cliente

Quero criar uma conta e fazer login

Para acessar o aplicativo e realizar compras

Critérios de Aceitação:

- CA01: E-mail deve ser único no sistema
- CA02: Senha deve ter mínimo 6 caracteres
- CA03: Senha armazenada com hash bcrypt
- CA04: Token JWT gerado no login bem-sucedido
- CA05: Mensagem de erro clara em caso de falha

- CA06: Redirecionamento automático após login

Tarefas Técnicas:

1. Criar modelo de usuário no banco (2h)
2. Implementar API de cadastro `/api/auth/register` (2h)
3. Implementar API de login `/api/auth/login` (2h)
4. Criar tela de cadastro no frontend (3h)
5. Criar tela de login no frontend (2h)
6. Implementar AuthGuard e interceptor (2h)
7. Testes unitários backend (1h)
8. Testes E2E fluxo completo (1h)

DoD Específico:

- Cobertura de teste > 80%
- Validações de formulário implementadas
- Documentação API atualizada

Story Points: 3**Prioridade: P0****Sprint: 1**

HU02 - Visualizar Catálogo de Cupcakes

Como cliente

Quero visualizar o catálogo com fotos, nomes e preços

Para escolher os produtos que desejo comprar

Critérios de Aceitação:

- CA01: Exibir no mínimo 40 sabores
- CA02: Cada produto deve mostrar: foto, nome, descrição, preço
- CA03: Tempo de carregamento < 2s
- CA04: Grid responsivo (2 colunas mobile, 3-4 desktop)
- CA05: Pull-to-refresh implementado
- CA06: Imagem placeholder para produtos sem foto

Tarefas Técnicas:

1. Criar modelo Produto e Categoria (2h)
2. Popular banco com 40+ produtos (1h)
3. Implementar endpoint `/api/produtos` (2h)
4. Criar componente de listagem (3h)
5. Otimizar carregamento de imagens (2h)

6. Implementar pull-to-refresh (1h)
7. Testes de performance (1h)
8. Testes unitários (1h)

DoD Específico:

- Lighthouse Performance > 90
- Imagens otimizadas (WebP)
- Lazy loading implementado

Story Points: 5

Prioridade: P0

Sprint: 1

HU03 - Adicionar Cupcakes ao Carrinho

Como cliente

Quero adicionar produtos ao carrinho

Para visualizar o total e finalizar compra posteriormente

Critérios de Aceitação:

- CA01: Badge no ícone do carrinho mostra quantidade
- CA02: Total calculado automaticamente
- CA03: Carrinho persiste entre sessões (localStorage)
- CA04: Botões +/- para ajustar quantidade
- CA05: Remover item com confirmação
- CA06: Mensagem "Produto adicionado" (toast)

Tarefas Técnicas:

1. Criar serviço de carrinho (CarrinhoService) (2h)
2. Implementar localStorage para persistência (1h)
3. Criar tela de carrinho (3h)
4. Adicionar badge no ícone (1h)
5. Implementar cálculos de total (1h)
6. Toast de confirmação (1h)
7. Testes unitários (1h)

Story Points: 3

Prioridade: P0

Sprint: 1

HU04 - Efetuar Pagamento via PIX ou Cartão

Como cliente

Quero realizar pagamento do pedido

Para confirmar minha compra

Critérios de Aceitação:

- CA01: Integração com API Mercado Pago
- CA02: Opções: PIX e Cartão
- CA03: QR Code gerado para PIX
- CA04: Validação de dados do cartão
- CA05: Webhook para confirmação assíncrona
- CA06: Tratamento de erros de pagamento
- CA07: Logs de transações

Tarefas Técnicas:

1. Configurar credenciais Mercado Pago (1h)
2. Implementar endpoint de pagamento (4h)
3. Criar webhook de confirmação (2h)
4. Implementar tela de pagamento (3h)
5. Integrar QR Code para PIX (2h)
6. Tratamento de erros (2h)
7. Logs e auditoria (1h)
8. Testes de integração (2h)

Story Points: 5

Prioridade: P0

Sprint: 2

HU05 - Acompanhar Status do Pedido

Como cliente

Quero ver o status do meu pedido em tempo real

Para saber quando vai chegar

Critérios de Aceitação:

- CA01: 5 status: Pendente, Em Preparo, A Caminho, Entregue, Cancelado
- CA02: Timeline visual com progresso
- CA03: Notificação ao mudar status
- CA04: Detalhes do pedido visíveis

- CA05: Informações do entregador (quando atribuído)

Tarefas Técnicas:

1. Criar endpoint /api/pedidos/meus-pedidos (2h)
2. Implementar tela de pedidos (3h)
3. Component de timeline de status (2h)
4. Integrar com notificações (2h)
5. Testes (1h)

Story Points: 3**Prioridade: P0****Sprint: 2**

8. DEFINITION OF READY (DoR)

Uma história está pronta para ser trabalhada quando:

☒ Checklist DoR

- História escrita no formato: Como/Quero/Para
- Critérios de aceitação definidos e testáveis
- Épico associado identificado
- Story points estimados pelo time
- Dependências técnicas identificadas
- Mockups/wireframes disponíveis (se UI)
- Requisitos não-funcionais especificados
- Tarefas técnicas decompostas
- Aceita pelo Product Owner
- Priorizada no backlog
- Time entende a história completamente

9. DEFINITION OF DONE (DoD)

Uma história está concluída quando:

☒ Checklist DoD

Código:

- Código desenvolvido e commitado
- Code review aprovado por pelo menos 1 dev
- Padrões de código seguidos (linting)
- Sem warnings ou erros no build
- Código refatorado (sem débitos técnicos)

Testes:

- Testes unitários escritos (TDD)
- Cobertura de teste > 80%
- Testes de integração passando
- Testes E2E nos fluxos críticos
- Testes manuais de QA realizados

Documentação:

- README atualizado
- Documentação de API atualizada
- Comentários em código complexo
- Changelog atualizado

Deploy:

- Merge na branch develop
- Deploy em ambiente de staging
- Smoke tests passando
- Aceito pelo Product Owner
- Demo apresentada na Sprint Review

10. PLANEJAMENTO DE SPRINTS

Sprint 1 (Semanas 1-2) - MVP Foundation

Objetivo: Estabelecer base autenticação, catálogo e carrinho

Histórias:

- HU01: Cadastro e Login (3pts)
- HU02: Visualizar Catálogo (5pts)
- HU03: Adicionar ao Carrinho (3pts)
- HU15: Pipeline CI/CD (5pts)
- HU20: Segurança básica (4pts de 8pts)

Total: 20 pontos

Capacidade do Time: 20-25 pontos

Risk Rating: Médio

Sprint 2 (Semanas 3-4) - Pedidos e Pagamento

Objetivo: Completar fluxo de compra

Histórias:

- HU04: Efetuar Pagamento (5pts)
- HU05: Acompanhar Pedido (3pts)
- HU11: Gerenciar Produtos (5pts)
- HU20: Segurança LGPD (4pts restantes)

Total: 17 pontos

Capacidade: 20-25 pontos

Risk Rating: Alto (integração pagamento)

Sprint 3 (Semanas 5-6) - Gestão e Otimização

Objetivo: Features administrativas e melhorias

Histórias:

- HU09: Filtrar Produtos (2pts)
- HU10: Cancelar Pedido (3pts)
- HU12: Relatórios (3pts)
- HU16: Monitoramento (3pts)

Total: 11 pontos

Capacidade: 20-25 pontos

Risk Rating: Baixo

Sprint 4 (Semanas 7-8) - Experiência do Usuário

Objetivo: Melhorias de UX e features secundárias

Histórias:

- HU06: Avaliar Pedido (2pts)
- HU07: Notificações Push (2pts)
- HU14: Custos Ingredientes (3pts)
- HU17: Rollback Seguro (3pts)

Total: 10 pontos

Capacidade: 20-25 pontos

Risk Rating: Baixo

Sprint 5 (Semanas 9-10) - Inteligência e Inovação

Objetivo: Features avançadas com ML

Histórias:

- HU08: Recomendações ML (5pts)
- HU13: Cupons Desconto (2pts)
- HU18: Treinar Modelo (5pts)
- HU19: Monitorar Drift (3pts)

Total: 15 pontos

Capacidade: 20-25 pontos

Risk Rating: Alto (ML)

11. CRITÉRIOS DE ACEITAÇÃO

Template de Critérios

Cada história deve ter critérios:

Formato Given-When-Then:

DADO QUE [contexto/pré-condição]

QUANDO [ação executada]

ENTÃO [resultado esperado]

Exemplo:

DADO QUE sou um cliente autenticado

QUANDO adiciono um cupcake ao carrinho

ENTÃO o badge do carrinho é atualizado com +1

E uma mensagem de confirmação é exibida

E o produto aparece na lista do carrinho

Checklist de Validação

Critérios devem ser:

- **Específicos:** Sem ambiguidade
- **Testáveis:** Podem ser automatizados
- **Mensuráveis:** Resultado objetivo
- **Alcançáveis:** Tecnicamente viável
- **Relevantes:** Agregam valor ao usuário

12. MÉTRICAS E KPIs

12.1 Métricas de Desenvolvimento

Métrica	Meta	Medição
Velocity	20-25 pts/sprint	Por sprint
Cobertura de Testes	> 80%	Contínua
Code Quality	> 85 (SonarQube)	Por commit
Build Time	< 5 min	Por build
Deploy Frequency	2x/semana	Semanal

12.2 Métricas de Qualidade

Métrica	Meta	Medição
Bug Rate	< 5 bugs/sprint	Por sprint
Mean Time to Repair	< 4h	Por incidente
Test Pass Rate	> 95%	Contínua
Technical Debt Ratio	< 5%	Mensal

12.3 Métricas de Produto

Métrica	Meta	Medição
Sprint Goal Success	100%	Por sprint
Story Completion	> 90%	Por sprint
Reopened Stories	< 10%	Por sprint
Stakeholder Satisfaction	> 8/10	Após review

12.4 Métricas de Negócio

Métrica	Meta	Medição
Time to Market	10 semanas	Fim do projeto
ROI	> 200%	6 meses pós-lançamento
User Adoption	> 500 usuários/mês	Mensal
Conversion Rate	> 3%	Mensal

GLOSSÁRIO

Backlog: Lista priorizada de funcionalidades

DoD: Definition of Done (Definição de Pronto)

DoR: Definition of Ready (Definição de Preparado)

Épico: Grande funcionalidade dividida em histórias

- Feature:** Funcionalidade menor que um épico
- Sprint:** Ciclo de desenvolvimento (1-2 semanas)
- Story Points:** Unidade de medida de esforço
- Velocity:** Quantidade de story points entregues por sprint

CONTROLE DE VERSÕES

Versão	Data	Autor	Alterações
1.0	11/11/2025	Time Dev	Versão inicial
2.0	15/11/2025	PO	Refinamento backlog
2.5	17/11/2025	Scrum Master	Ajustes finais

- Elaborado por:** Equipe Leon's Cupcake
- Aprovado por:** Product Owner
- Data de Aprovação:** 17/11/2025
- Próxima Revisão:** Início de cada Sprint