

Projeto Le Menu

Para a comunidade acadêmica da UNIFESSPA que precisa de acesso prático às informações do restaurante, o Le Menu é uma plataforma web que facilita consulta de cardápio, comunicação e feedback, diferente de processos manuais, nosso sistema centraliza todas as informações em um local acessível 24/7.

Sequenciador



Figura 1: Sequenciador

Product backlog

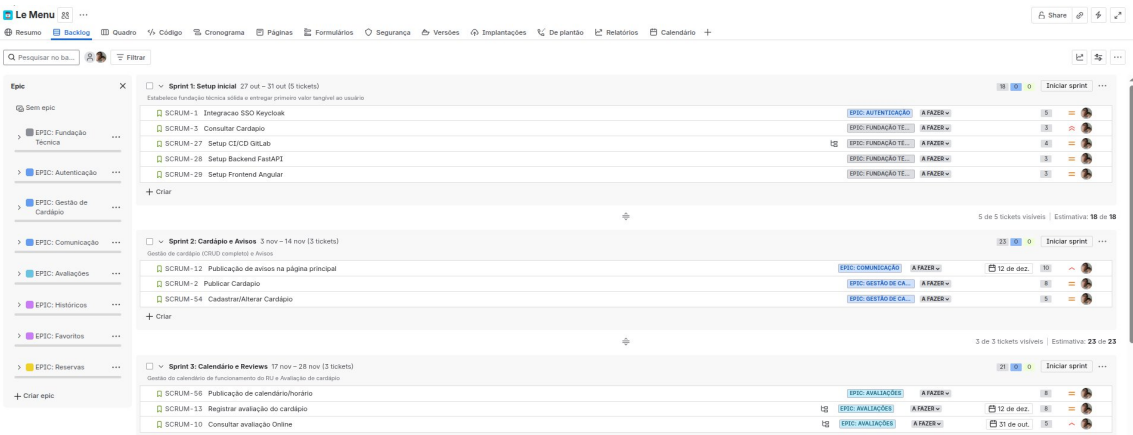


Figura 2: Backlog do MVP



Figura 3: Backlog dos incrementos

Detalhamento do Sprint 1

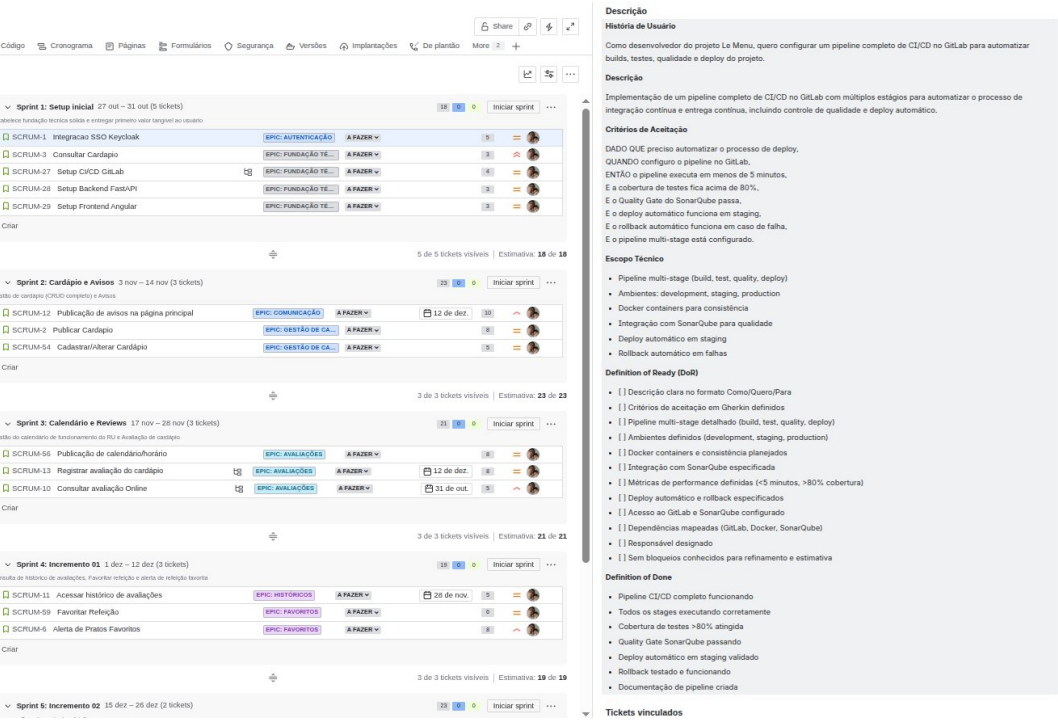


Figura 4: Sprint 1: Integração SSO Keycloak

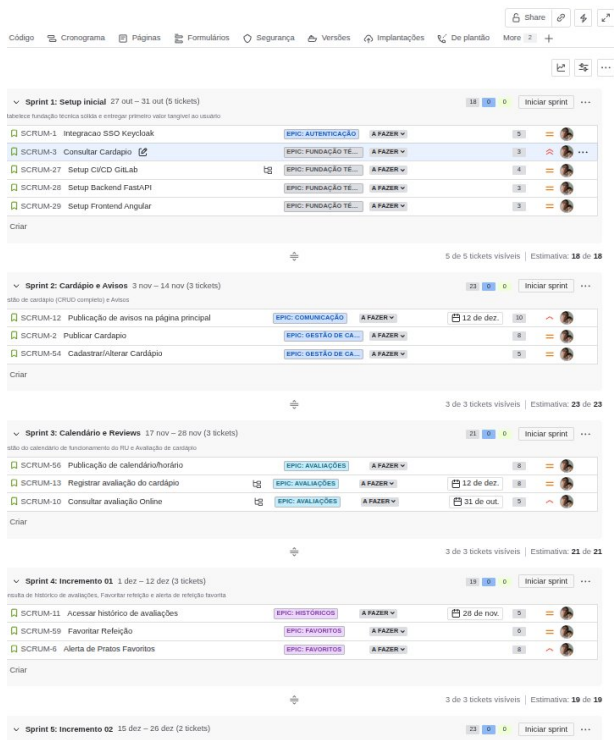


Figura 5: Sprint 1: Consultar Cardápio

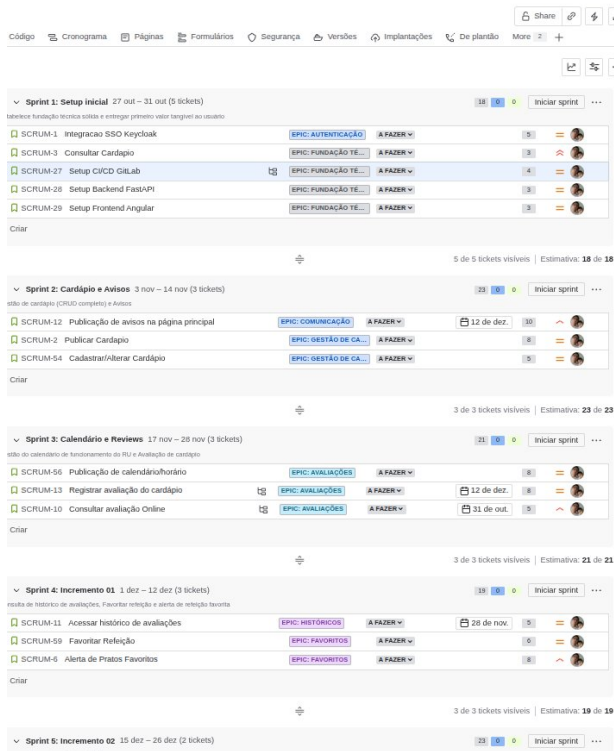
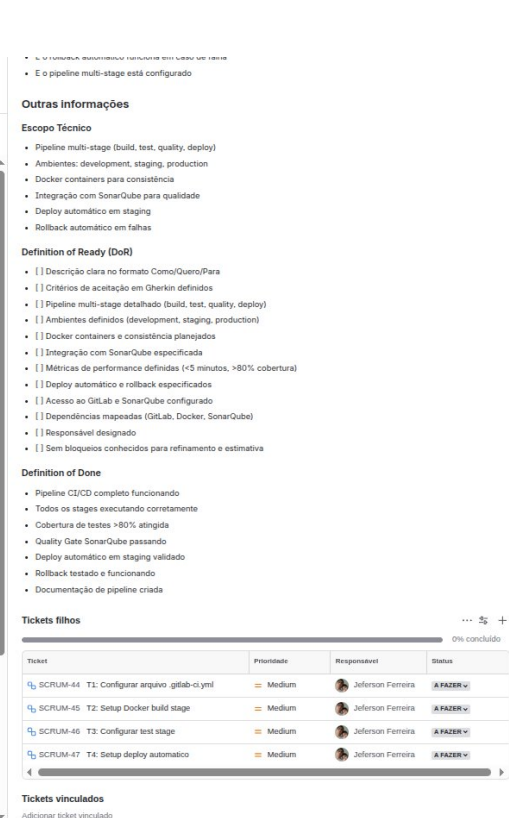
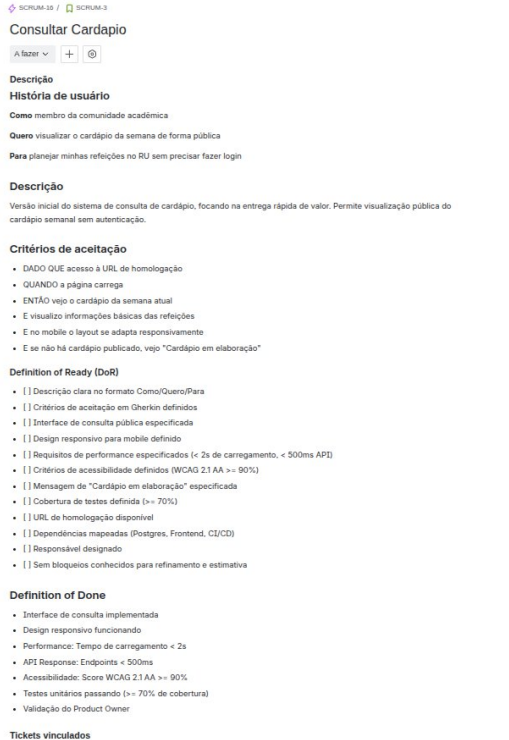


Figura 6: Sprint 1: Setup CI/CD GitLab



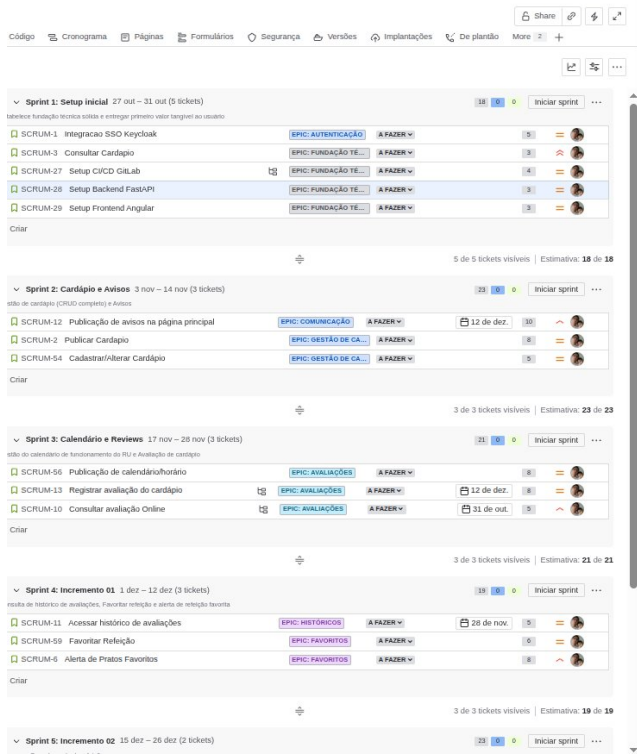


Figura 7: Sprint 1: Setup Backend FastAPI

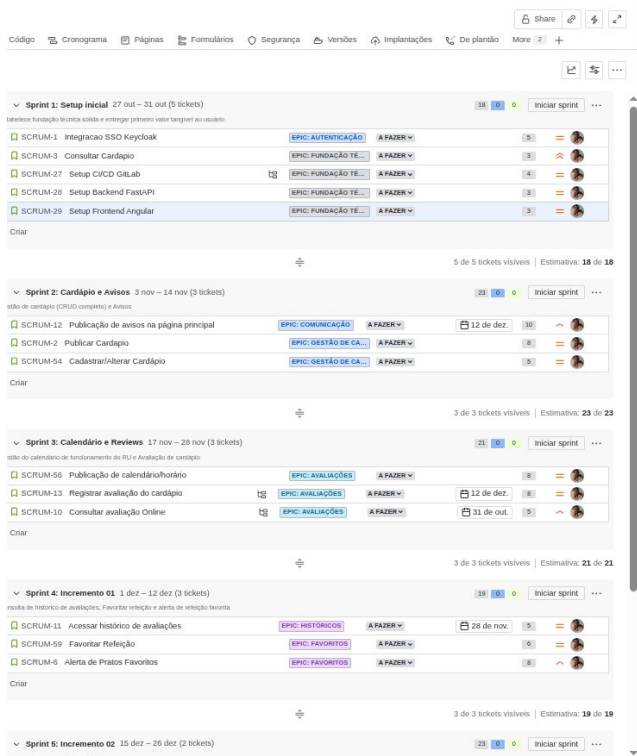


Figura 8: Sprint 1: Setup Frontend Angular

Critérios de Aceitação

- DADO QUE preciso iniciar o desenvolvimento do backend
- QUANDO configuro o ambiente de desenvolvimento
- ENTÃO tenho:
 - FastAPI funcionando com Python 3.13+
 - PostgreSQL conectado via SQLAlchemy
 - Alembic configurado para migrations
 - Poetry gerenciando dependências
 - Estrutura de pastas organizada
 - Autenticação JWT preparada para Keycloak

Outras Informações

Arquitetura Técnica

- FastAPI + Python 3.13+
- SQLAlchemy ORM + Alembic migrations
- Banco de dados PostgreSQL
- Pydantic para validação
- Poetry para dependências

Estrutura de Pastas

- app/core/ - Configurações e utilitários
- app/api/v1/ - Endpoints da API
- app/models/ - Modelos SQLAlchemy
- app/services/ - Lógica de negócio

Definition of Ready (DoR)

- [] Descrição clara no formato Como/Quero/Para
- [] Critérios de aceitação em Gherkin definidos
- [] Stack tecnológico especificado (FastAPI, Python 3.13+, PostgreSQL)
- [] Estrutura de pastas detalhada
- [] SQLAlchemy ORM e Alembic migrations planejados
- [] Pydantic para validação especificado
- [] Poetry para gerenciamento de dependências definido
- [] Autenticação JWT preparada para Keycloak
- [] Configurações e utilitários planejados
- [] Dependências mapeadas (PostgreSQL, Poetry, FastAPI)
- [] Responsável designado
- [] Sem bloqueios conhecidos para refinamento e estimativa

Definition of Done

- Estrutura base do backend implementada
- FastAPI funcionando com endpoints básicos
- PostgreSQL conectado com SQLAlchemy
- Alembic migrations configurado
- Poetry dependencies gerenciadas
- Pydantic validation funcionando
- JWT auth preparado para Keycloak
- Testes básicos implementados

Ticket

SCRUM-10 / SCRUM-29

Setup Frontend Angular

A fazer + @

Descrição

Resumo
Configuração da estrutura base do frontend Angular para o projeto Le Menu. O objetivo é ter uma fundação sólida e escalável para o desenvolvimento da interface.

Contexto

Início do desenvolvimento do frontend utilizando Angular 18.3+ com TypeScript. A configuração inclui design system, roteamento, guards de autenticação, interceptors HTTP e integração com APIs do backend.

Critérios de Aceitação

- DADO QUE preciso iniciar o desenvolvimento do frontend
- QUANDO configuro o ambiente Angular
- ENTÃO tenho:
 - Angular 18.3+ funcionando com TypeScript
 - Angular Material CD configurado
 - Tailwind CSS configurado e integrado com o angular
 - RxJS configurado para programação reativa
 - Roteamento e navegação implementados
 - Guards de autenticação preparados
 - Interceptors HTTP configurados
 - Error handling global ativo
 - Loading states implementados
 - Design responsivo (mobile-first) funcionando

Outras Informações

Stack Técnica

- Angular 18.3+ com TypeScript
- Angular Material CDK para UX
- Tailwind CSS para UI
- RxJS para programação reativa
- PWA capabilities
- Responsive design (mobile-first)

Funcionalidades Base

- Roteamento e navegação
- Guards de autenticação
- Interceptors HTTP

Observação: Não adicionamos capturas de telas de todos os sprints devido o tamanho final do documento gerado. Todos os PDF exportados do projeto Le Menu estarão no mesmo repositório deste arquivo.

Cronograma

O cronograma do projeto foi estruturado em cinco Sprints. O primeira, com duração de uma semana, é dedicado ao setup inicial e às configurações de CI/CD. Ainda assim, já haverá entrega de valor, disponibilizando a página inicial do sistema com o cardápio da semana de implantação, carregado diretamente no banco de dados.

Os sprints seguintes foram planejados com duração de duas semanas cada. O MVP será entregue no terceiro Sprint, restando mais dois incrementos posteriores. Durante todo o período de implementação e incrementos, serão coletados feedbacks de usuários e stakeholders para orientar o planejamento da segunda fase do projeto, que abrangerá integrações com outros sistemas, além de melhorias de desempenho e usabilidade.

A terceira e última fase terá como foco a integração com o sistema de controle de entrada do restaurante, permitindo o gerenciamento automático de pontos e saldos por meio da leitura do QR Code nas catracas do restaurante.

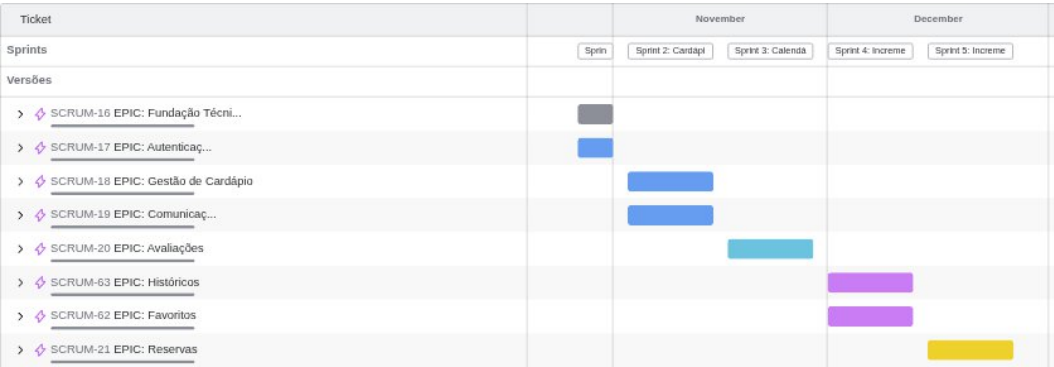


Figura 9: Cronograma do projeto com seis sprints