



UNIVERSIDADE FEDERAL DO CEARÁ – CAMPUS QUIXADÁ
CURSO DE ENGENHARIA DE COMPUTAÇÃO | ENGENHARIA
DE SOFTWARE | SISTEMAS DE INFORMAÇÃO
PROJETO DE EXTENSÃO EM ENGENHARIA DE SOFTWARE

NAVALHA: SISTEMA DE GERENCIAMENTO ALABAMA BARBERS

Arthur Lelis Uchoa

Julio Emanuel Pereira da Silva

Mateus Valentim Vieira Furtado

Victor Farias da Silva

Rodrigo Rodrigues de Sousa (**fora da disciplina**)

Relatório Boas Práticas em Engenharia de Software



Histórico

Data	Versão	Autor	Descrição
31/05/2026	0	Mateus Valentim	Sprint 1
30/05/2026	0.1	Mateus Valentim	Sprint 2



Observações:

Data	Versão	Autor	Observação
31/05/2026	0	Mateus Valentim	Entrega Sprint 1: 31/05
31/05/2026	0	Mateus Valentim	Entrega Sprint 2: 14/06
31/05/2026	0	Mateus Valentim	Entrega MVP: 28/06



Identificação

Link do Repositório: [Github](#)

Link da Board: [ClickUp](#)

Google Drive com Documentos e Evidências:

 Extensão - Alabama Barbers

Observação: Os tópicos com “*” são *exatamente os tópicos solicitados no documento no moodle, os que não tem esse simbolo são exclusivamente para guiar a equipe*

Sprint 1

1 - Objetivo (Descrição Geral) *

A Sprint 1 teve como objetivo implementar as funcionalidades básicas necessárias para o gerenciamento inicial do sistema da barbearia, estabelecendo a estrutura do sistema, configuração dos ambientes de desenvolvimento e implementação dos primeiros módulos de cadastro, porém com dados ainda mockados.

2 - Atividades de Planejamento e Documentação

Antes do início da implementação do sistema, a equipe dedicou-se à elaboração de diversos documentos de Engenharia de Software com o objetivo de reduzir riscos, alinhar requisitos com o cliente e definir a estrutura do projeto.

Foram produzidos os seguintes artefatos:

- Documento de Arquitetura de Software;
- Documento de Requisitos;
- Planejamento de Releases e Sprints;
- Modelagem inicial da solução;
- Organização do Backlog do Produto;
- Estruturação do Board no ClickUp;
- Definição da stack tecnológica do projeto.

A elaboração desses documentos permitiu que a equipe tivesse uma visão clara dos objetivos do sistema antes do início da codificação, reduzindo retrabalho e facilitando a divisão das tarefas entre os integrantes.

2.1 - Engenharia de Software Antes da Implementação

Uma das principais boas práticas adotadas pela equipe foi a realização de atividades de engenharia e planejamento antes do desenvolvimento do código-fonte. Foram elaborados documentos de requisitos, arquitetura, planejamento de releases e definição de processos, permitindo uma melhor compreensão do problema e da solução proposta.

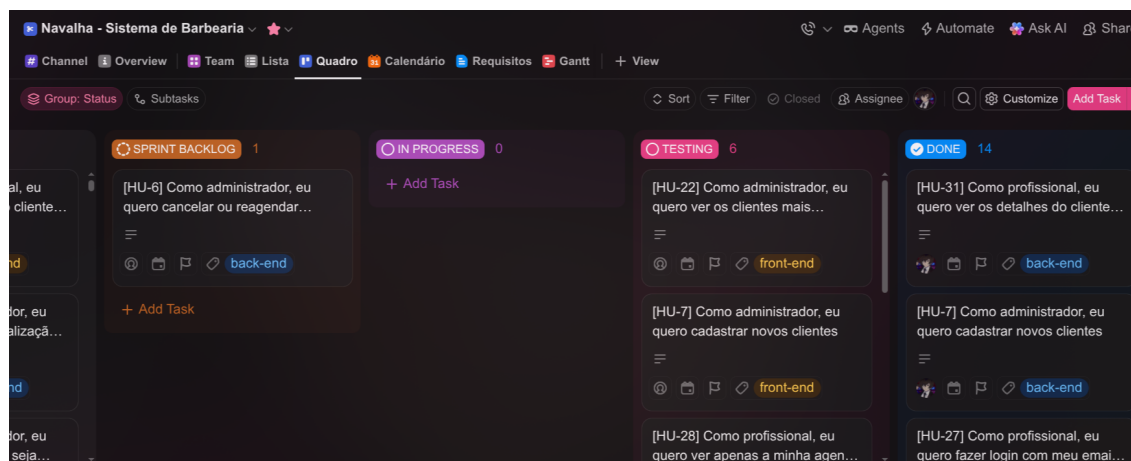
Essa abordagem contribuiu para:

- Maior alinhamento entre os integrantes da equipe;
- Melhor entendimento das necessidades do cliente;
- Redução de inconsistências durante o desenvolvimento;
- Organização das funcionalidades em histórias de usuário;
- Planejamento adequado das Sprints;

- Definição prévia da arquitetura e das tecnologias utilizadas.

3 - Status de Desenvolvimento

O acompanhamento das atividades foi realizado por meio do ClickUp, utilizando um fluxo de trabalho baseado em Scrum com colunas de Sprint Backlog, In Progress, Testing e Done. Utilizando ordem de prioridade para decidir as sprints e fluxo do projeto, juntamente com a divisão de tarefas pré-definidas no app.



[ClickUp Board](#)



4 - Itens Entregues *

3.1 - Backend

Gestão de Clientes

- Cadastro de clientes;
- Consulta de clientes;
- Atualização de clientes;
- Remoção de clientes.

Gestão de Profissionais

- Cadastro de barbeiros/profissionais;
- Consulta de profissionais;
- Atualização de profissionais;
- Exclusão de profissionais.

Estrutura Inicial da API

- Configuração do Spring Boot pelo Spring Initializer;
- Integração com PostgreSQL;
- Criação das entidades principais;
- Implementação dos repositórios JPA.

3.2 - Frontend

Interfaces Desenvolvidas

- Tela inicial do sistema;
- Tela de listagem de clientes;
- Tela de cadastro de clientes;
- Tela de listagem de profissionais;
- Componentes reutilizáveis da interface.



4 - Versionamento de Código

O projeto utiliza Git e GitHub para controle de versão, nos quais foram adotados commits padronizados utilizando Conventional Commits:

Type	Description
feat	Introduce a new feature to the codebase
fix	Fix a bug in the codebase
docs	Create/update documentation
style	Feature and updates related to styling
refactor	Refactor a specific section of the codebase
test	Add or update code related to testing
chore	Regular code maintenance

feat(core): estrutura dtos, mappers de entidade e componentes de validacao
Arthur32p committed 12 hours ago

fix(exception): centraliza tratamento de erros e remove ambiguidades de runtime
Arthur32p committed 12 hours ago

feat(infra): define entidades do modelo de dados e interfaces de repositório
Arthur32p committed 12 hours ago

Commits on May 31, 2026

fix: visual bugs
vistomia committed yesterday · ✓ 1 / 1

Merge branch 'main' of github.com:techpratics/navalha
vistomia committed yesterday · ✗ 0 / 1

feat: screens
vistomia committed yesterday

feat: estrutura inicial do frontend
DevJulioEmanuel committed 4 days ago · ✓ 1 / 1

Commits on May 26, 2026

feat: Update project metadata and dependencies in pom.xml
mattsu014 authored last week · ✓ 1 / 1

5. Problemas e Dificuldades *

Nenhuma, a Sprint 1 seguiu um bom fluxo, às vezes atrapalhado pela semana de prova da equipe mas foi bem organizado

6. Dados Utilizados *

Os dados foram mockados nessa primeira sprint, porém a partir da sprint 2 o sistema de banco de dados já será feito e implementado (possivelmente)

7. Conclusão

A Sprint 1 permitiu estabelecer a base tecnológica do sistema Navalha, com a implementação base do Frontend e a criação dos principais CRUDs do Backend, juntamente com a definição da arquitetura do projeto. De forma geral os objetivos planejados foram atingidos, possibilitando a continuidade do desenvolvimento nas próximas Sprints. Em suma, a sprint 1 visava a criação do modelo base do projeto

Sprint 2

1 - Objetivo *

A Sprint 2 teve como objetivo expandir as funcionalidades administrativas e operacionais do sistema Navalha, implementando recursos de autenticação, gerenciamento avançado de agendas, configuração do estabelecimento e controle de atendimentos. Além disso, foram iniciados os módulos relacionados a planos de assinatura e relatórios gerenciais, aproximando o sistema da versão MVP planejada

2 - Atividades de Planejamento e Documentação

Durante a Sprint 2, a equipe manteve as práticas de planejamento e acompanhamento definidas na Sprint anterior. As histórias de usuário foram priorizadas de acordo com o valor de negócio e dependências técnicas, garantindo uma evolução contínua do produto.

As atividades foram organizadas, **inclusive melhoradas**, no ClickUp, permitindo o acompanhamento do progresso das tarefas, distribuição de responsabilidades e monitoramento do fluxo de desenvolvimento.

2.1 - Engenharia de Software Antes da Implementação

A equipe continuou aplicando as práticas definidas no início do projeto, incluindo:

- Utilização de histórias de usuário para definição dos requisitos;
- Planejamento incremental através de Sprints;
- Controle das atividades por meio do ClickUp;
- Versionamento de código utilizando Git e GitHub;
- Padronização dos commits com Conventional Commits;
- Separação das responsabilidades entre Frontend e Backend;
- Integração contínua entre as equipes de desenvolvimento.

Essas práticas contribuíram para manter a organização do projeto e facilitar a evolução das funcionalidades implementadas.

3 - Funcionalidades Implementadas

3.1 - Backend

Segurança e controle de acesso

Foi implementado um sistema completo de autenticação para garantir que apenas usuários autorizados possam acessar funcionalidades restritas da plataforma.

Agora, cada usuário realiza login para acessar o sistema e recebe permissões de acordo com seu perfil. Algumas funcionalidades permanecem públicas, como o cadastro de clientes, a consulta de horários disponíveis e a visualização das informações da empresa, enquanto as demais áreas exigem autenticação.

Perfis de usuário

O sistema passou a trabalhar com três perfis de acesso:

- Cliente;
- Profissional;
- Administrador.

Cada perfil possui permissões específicas de acordo com suas responsabilidades dentro da plataforma. Dessa forma, administradores têm acesso às funcionalidades de gerenciamento, profissionais podem administrar sua agenda e disponibilidade, e clientes utilizam os recursos de agendamento.

Histórico de alterações

Foi adicionada uma funcionalidade de auditoria que registra automaticamente informações sobre a criação e atualização dos principais dados do sistema.

Com isso, é possível identificar quando um registro foi criado ou alterado e qual usuário realizou a ação, aumentando a rastreabilidade e o controle das informações.

Configurações da empresa

Foi criado um módulo dedicado ao gerenciamento das informações da barbearia.

Nele é possível cadastrar e atualizar dados como:

- Nome da empresa;
- Endereço;
- Telefone para contato;
- Horários de funcionamento para cada dia da semana.

Essas informações ficam disponíveis para consulta no sistema e podem ser administradas pelos usuários responsáveis pela gestão.

Gerenciamento do banco de dados

Foi implementado um mecanismo de controle da estrutura do banco de dados, permitindo que a criação e atualização das tabelas aconteçam de forma organizada e padronizada.

Essa melhoria garante maior segurança durante a evolução do sistema e facilita a instalação do ambiente em diferentes máquinas.

Documentação da API

Foi disponibilizada uma documentação interativa da API utilizada pelo sistema.

Essa funcionalidade facilita o entendimento das integrações realizadas pelo frontend e auxilia nos testes e na manutenção das funcionalidades existentes.

Execução simplificada do sistema

Foi adicionada uma configuração que permite iniciar todos os serviços necessários para o funcionamento da aplicação de forma automatizada.

Com isso, o processo de configuração do ambiente ficou mais simples, rápido e padronizado, facilitando a execução do projeto em diferentes ambientes de desenvolvimento.

3.1 - Frontend

Camada de serviços

Para melhorar a organização do sistema, foi criada uma estrutura responsável por centralizar a comunicação com a API. Com isso, funcionalidades como autenticação, gerenciamento de clientes, profissionais, serviços e agendamentos passaram a utilizar uma padronização nas requisições, tornando o sistema mais seguro, organizado e fácil de manter.

Fluxo de agendamento do cliente

O processo de agendamento foi reformulado para oferecer uma experiência mais simples e próxima da realidade. Enquanto na Sprint 1 os dados eram apenas simulados, na Sprint 2 o fluxo passou a utilizar informações reais da API.

O novo processo acontece em quatro etapas:

1. Escolha do serviço desejado;
2. Seleção da data e horário disponíveis;
3. Confirmação dos dados do agendamento;
4. Exibição da tela de sucesso.

Além disso, os horários apresentados ao cliente agora são calculados automaticamente de acordo com a disponibilidade dos profissionais e a duração do serviço escolhido.

Controle de acesso

Foi implementado um sistema de proteção de rotas para garantir que cada usuário acesse apenas as funcionalidades compatíveis com seu perfil. Usuários autenticados são direcionados para sua área específica, enquanto páginas restritas ficam protegidas contra acessos indevidos.

Novas funcionalidades para administradores

Agenda Geral

Foi criada uma nova tela de agenda para administradores, permitindo visualizar todos os agendamentos em formatos diário, semanal e mensal. Também é possível criar novos agendamentos diretamente pelo sistema, realizar reagendamentos e acompanhar a ocupação da agenda.

Agenda por profissional

Os administradores agora podem visualizar os compromissos de cada profissional individualmente, facilitando o acompanhamento da rotina de trabalho e da disponibilidade da equipe.

Novas funcionalidades para profissionais

Configuração de disponibilidade

Os profissionais passaram a contar com uma área dedicada para definir seus horários de atendimento. Nela é possível:

- Ativar ou desativar dias da semana;
- Definir horários de início e término do expediente;
- Replicar a mesma configuração para vários dias, agilizando o cadastro da disponibilidade.

Recursos de gerenciamento de agendamentos

Foram adicionadas novas ferramentas para facilitar o controle dos atendimentos:

- Visualização detalhada dos agendamentos;
- Confirmação, conclusão ou cancelamento de atendimentos;
- Reagendamento de horários;
- Cadastro rápido de novos agendamentos realizados presencialmente.

Gerenciamento de serviços dos profissionais

Foi implementada uma funcionalidade que permite vincular ou remover serviços de cada profissional, possibilitando controlar quais atendimentos cada membro da equipe está apto a realizar.

5. Problemas e Dificuldades *

A Sprint 2 saiu funcional como planejado pela equipe, porém não tivemos tempo de corrigir determinados bugs, como interação de paginas no frontend, porém esses problemas podem ser corrigidos até a sprint final

6. Dados Utilizados *

O sistema de banco de dados foi implementado com sucesso nessa sprint

7. Conclusão *

A Sprint 2 representou uma transição do estado de protótipo para o de produto: o sistema saiu de um conjunto de telas com dados simulados para uma aplicação integrada, com autenticação real baseada em JWT, controle de acesso por papel de usuário, persistência confiável via Flyway e uma camada de serviços estruturada no frontend. O calendário administrativo e a página de disponibilidade do profissional completaram os fluxos centrais do sistema, permitindo que todas as personas (cliente, profissional e administrador) operem com dados reais de ponta a ponta.