

Relatório do Sistema EasyClinic - Sprint 02

Papéis e responsabilidades dessa semana

- Product Owner (PO): Guilherme Alves - Responsável por direcionar requisitos, priorizar a construção do backlog, realizar a validação de telas e garantir a coerência do produto.
- Scrum Master (SM): Nicolas Damasceno - Responsável por garantir que a equipe siga as práticas do Scrum, organizar artefatos e realizar o registro das cerimônias da sprint.
- Time de Desenvolvimento (Devs): Francisco do Nascimento, Renato Paiva e Hilton Tavares - Responsáveis pelos protótipos, estruturação das telas, produção de código HTML e revisões do backlog.

Refinamento de histórias de usuário, Critérios de aceitação e cenários de uso

1. CADASTRAR PACIENTES

HISTÓRIA DE USUÁRIO

ITEM	DESCRIÇÃO
História	Como recepcionista, quero cadastrar novos pacientes informando nome, CPF, telefone e endereço, para manter um controle sobre os atendimentos realizados pela clínica.
Prioridade	Alta

CRITÉRIOS DE ACEITAÇÃO

Nº	CRITÉRIO
1	O sistema deve permitir o preenchimento de nome, CPF,

	telefone e endereço
2	CPF e telefone devem ser validados (não podem ser inválidos)
3	Após o cadastro, o sistema deve exibir a mensagem de “Paciente salvo!”
4	Caso haja erro no preenchimento dos campos, o sistema deve exibir “É preciso que o formato corresponda ao exigido.”
5	Caso fique algum campo sem preencher, o sistema deve exibir “Preencha este campo”

CENÁRIOS DE USO

CENÁRIO	DESCRIÇÃO
C1 - Cadastro bem-sucedido	Dado que o atendente está na tela de cadastro, Quando ele preenche nome, CPF, telefone e endereço e clicar em “Salvar”, Então o sistema deve salvar o paciente e exibir a mensagem “Paciente salvo!”
C2 - CPF inválido	Dado que o atendente está na tela de cadastro, Quando ele tenta salvar um cliente com CPF inválido, Então o sistema deve exibir a mensagem “É preciso que o formato corresponda ao exigido.”
C3 - Telefone inválido	Dado que o atendente está na tela de cadastro, Quando ele tenta salvar um cliente com telefone inválido, Então o sistema deve exibir a mensagem “É preciso que o formato

	corresponda ao exigido.”
--	--------------------------

2. AGENDAR CONSULTAS

HISTÓRIA DE USUÁRIO

ITEM	DESCRIÇÃO
História	Como recepcionista, quero agendar consultas vinculando paciente, médico, data e horário, para organizar a agenda diária da clínica de forma eficiente e evitar conflitos de atendimento
Prioridade	Alta

CRITÉRIOS DE ACEITAÇÃO

Nº	CRITÉRIO
1	O sistema deve permitir selecionar pacientes e médicos cadastrados
2	A data e horário devem ser válidos e não podem estar vazios
3	Após o agendamento, o sistema deve exibir a mensagem de “Agendamento realizado com sucesso!”

CENÁRIOS DE USO

CENÁRIO	DESCRIÇÃO
C1 - Agendamento bem-sucedido	Dado que o atendente está na tela de agendamento, Quando selecionar paciente, médico, data e horário e clicar em “Confirmar

	Agendamento”, Então o sistema deve registrar o agendamento e exibir a mensagem “Agendamento realizado com sucesso!”
C2 - Data anterior ao dia atual	Dado que o atendente está na tela de agendamento, Quando ele tenta salvar registrar um agendamento com uma data passada, Então o sistema deve exibir a mensagem “Data inválida!”
C3 - Campos vazios	Dado que o atendente está na tela de agendamento, Quando ele tenta confirmar sem preencher todos os campos, então o sistema exibe “Preencha todos os campos.”

3. GERENCIAR USUÁRIOS

HISTÓRIA DE USUÁRIO

ITEM	DESCRIÇÃO
História	Como administrador, quero cadastrar, editar e excluir usuários internos que possuem nome,email/login,função e senha, para controlar o acesso ao sistema
<i>Prioridade</i>	<i>Alta</i>

CRITÉRIOS DE ACEITAÇÃO

Nº	CRITÉRIO
1	O sistema deve permitir incluir nome, e-mail/login,função e senha do usuário

2	A edição deve atualizar imediatamente a tabela
3	A exclusão deve solicitar confirmação
4	Após criar, deve exibir “Usuário criado com sucesso!”

CENÁRIOS DE USO

CENÁRIO	DESCRIÇÃO
C1 - Cadastro de usuário bem-sucedido	Dado que o administrador está na tela de usuários, Quando selecionar nome, email/login, função e senha e clicar em “Criar Usuário”, Então o sistema deve registrar o usuário e exibir a mensagem “Usuário criado com sucesso!”
C2 - Edição confirmada	Dado que o administrador está na tela de usuários, Quando ele clica em “Editar” e confirma a modificação de dados, Então o sistema deve modificar o usuário dentro da tabela e exibir a mensagem “ <i>Usuário modificado com sucesso!</i> ”
C3 - Exclusão confirmada	Dado que o administrador está na tela de usuários, Quando ele clica em “Excluir” e confirmar, Então o sistema deve remover o usuário dentro da tabela e exibir a mensagem “Usuário removido com sucesso!”

4. VISUALIZAR AGENDA MÉDICA DIÁRIA

HISTÓRIA DE USUÁRIO

ITEM	DESCRIÇÃO
História	Como médico, quero visualizar a agenda diária de consultas com horários, pacientes e status da consulta, para acompanhar melhor o fluxo de atendimentos
Prioridade	Alta

CRITÉRIOS DE ACEITAÇÃO

Nº	CRITÉRIO
1	O sistema deve exibir uma agenda com horário, nome do paciente e status da consulta
2	Deve permitir visualizar consultas do dia atual
3	Deve atualizar automaticamente conforme novos agendamentos
4	Caso não haja consultas no dia, mostrar "Nenhuma consulta agendada"

CENÁRIOS DE USO

CENÁRIO	DESCRIÇÃO
C1 - Visualização normal	O médico acessa a agenda médica e vê todos os horários do dia
C2 - Dia sem consultas	Caso não haja registros, aparece a mensagem nenhuma consulta agendada
C3 - Atualização pelo atendente	Novos cadastros e consultas realizadas são refletidos na tela de agenda

5. VISUALIZAR DADOS GERAIS (TELA INICIAL)

HISTÓRIA DE USUÁRIO

ITEM	DESCRIÇÃO
História	Como gestor, quero visualizar rapidamente indicadores gerais como o total de consultas do dia, médicos ativos, cancelamentos, etc, para acompanhar a movimentação diária da clínica
Prioridade	Média

CRITÉRIOS DE ACEITAÇÃO

Nº	CRITÉRIO
1	O painel do sistema deve ter cartões com valores numéricos de consultas, pacientes, médicos e cancelamentos
2	Precisa apresentar uma lista de atividades recentes
3	Deve exibir gráfico semanal de consultas
4	Deve ser responsivo e de leitura clara em dispositivos menores

CENÁRIOS DE USO

CENÁRIO	DESCRIÇÃO
C1 - Visualização normal	Cartões numéricos, gráficos e as atividade
C2 - Atualização pelo administrador	Novos cadastros e consultas realizadas são refletidos na tela inicial

Roteiros de criação de interfaces

História 1 - Interface de cadastro de paciente

1) Fixar o escopo

História de usuário: *Como recepcionista, quero cadastrar novos pacientes informando nome, CPF, telefone e endereço, para manter um controle sobre os atendimentos realizados pela clínica.*

Extração:

- **Ator:** recepcionista
- **Ação:** cadastrar paciente
- **Benefício:** manter controle atualizado dos atendimentos

Objetivo: permitir o cadastro válido de um paciente e confirmar o sucesso do registro

2) Mapear requisitos visuais a partir dos critérios de aceitação

Critérios de Aceitação:

1. Permitir cadastrar nome, CPF, telefone e endereço -> campos de texto.
2. Validar CPF e telefone -> mensagem de erro.
3. Exibir mensagem de sucesso -> banner verde.
4. Lista atualizada automaticamente -> atualização visual.

3) Derivar fluxos de interface a partir dos cenários

Cenário 1 (sucesso): preencher todos os campos -> clicar em salvar -> ver mensagem de sucesso -> retorna a lista

Cenário 2 (CPF inválido): erro inline.

Cenário 3 (Telefone inválido): erro inline.

4) Definir a anatomia da tela (layout)

- **Título:** 'Cadastrar Paciente'
- **Campos:** Nome, CPF(com máscara), telefone(com máscara) e endereço
- **Botões:** Salvar
- **Mensagens:** Sucesso / erro

5) Preparar estados da UI

- Estado normal
- Estado com campos preenchidos
- Estado com erro (CPF/telefone)
- Estado de sucesso ("Paciente Salvo!")
- Estado de carregamento (ao salvar)

6) Checklist UI ↔ Critérios de aceitação

CRITÉRIO DE AVALIAÇÃO	COMPONENTE / TELA	VALIDAÇÃO VISUAL
Validar CPF	Campo CPF(com máscara e erro)	Erro inline ao digitar inválido
Mensagem de sucesso	Banner verde no final	Após clicar salvar com dados válidos
Exibir paciente na lista	Página lista de clientes	Novo item visível no topo

7) ROTEIRO DE TESTE

1. Abrir a tela “Cadastrar Paciente”.
2. Preencher nome.
3. Digitar um CPF inválido -> ver mensagem de erro.
4. Corrigir CPF
5. Digitar um telefone inválido -> ver mensagem de erro.
6. Corrigir telefone e preencher endereço.
7. Clicar em Salvar -> ver banner de sucesso.
8. Verificar item na Lista de Pacientes.

História 2 - Interface de agendamento de consultas

1) Fixar o escopo

História de usuário: *Como recepcionista, quero agendar consultas vinculando paciente, médico, data e horário, para organizar a agenda diária da clínica de forma eficiente e evitar conflitos de atendimento*

Extração:

- **Ator:** recepcionista
- **Ação:** agendar consulta
- **Benefício:** manter a agenda organizada e sem horários duplicados

Objetivo: permitir o agendamento válido de uma consulta e confirmar o sucesso do registro

2) Mapear requisitos visuais a partir dos critérios de aceitação

Critérios de Aceitação:

1. Permitir selecionar paciente, médico e horário -> selects de texto
2. Permitir selecionar horário -> select de time

2. Permitir selecionar data -> campo de date
3. Validar horário e disponibilidade -> mensagem de erro
4. Exibir mensagem de sucesso -> banner verde
5. Lista atualizada automaticamente -> atualização visual

3) Derivar fluxos de interface a partir dos cenários

Cenário 1 (sucesso): preencher todos os selects e campos -> clicar em salvar -> ver mensagem de sucesso -> retorna a lista

Cenário 2 (conflito de horário): erro inline.

Cenário 3 (campos vazios): exibe aviso.

4) Definir a anatomia da tela (layout)

- **Título:** 'Agendar consulta'
- **Campos:** Paciente (select), médico (select), data(input date) e horário (select)
- **Botões:** Confirmar
- **Mensagens:** Sucesso / erro

5) Preparar estados da UI

- Estado normal
- Estado com campos preenchidos
- Estado com erro (data/horário)
- Estado de sucesso ("Agendamento realizado com sucesso!")
- Estado de carregamento (ao salvar)

6) Checklist UI ↔ Critérios de aceitação

CRITÉRIO DE AVALIAÇÃO	COMPONENTE / TELA	VALIDAÇÃO VISUAL
Validar data	Campo data	Erro inline ao informar

		inválida
Mensagem de sucesso	Banner verde no final	Após clicar salvar com dados válidos
Exibir consulta na lista	Página lista de consultas	Novo item visível no topo

7) ROTEIRO DE TESTE

1. Abrir a tela “Agendar Consulta”.
2. Selecionar paciente e médico.
3. Informar uma data inválida -> ver mensagem de erro.
4. Corrigir data e selecionar horário.
5. Clicar em Confirmar -> ver banner de sucesso.
6. Verificar item na Lista de consultas.

História 3 - Interface de gerenciamento de usuários

1) Fixar o escopo

História de usuário: *Como administrador, quero cadastrar, editar e excluir usuários internos que possuem nome, email/login, função e senha para controlar o acesso ao sistema*

Extração:

- **Ator:** administrador
- **Ação:** gerenciar usuários
- **Benefício:** garantir controle e segurança do sistema

Objetivo: permitir com que o administrador realize o CRUD completo dos usuários.

2) Mapear requisitos visuais a partir dos critérios de aceitação

Critérios de Aceitação:

1. Permitir cadastrar nome e email -> campos de texto
2. Permitir selecionar permissão -> select de texto
3. Permitir cadastrar senha -> campo password
4. Permitir editar e excluir usuários existentes -> botão de ação e confirmação
5. Exibir mensagem de sucesso -> banner verde
6. Lista atualizada automaticamente -> atualização visual

3) Derivar fluxos de interface a partir dos cenários

Cenário 1 (sucesso): preencher todos os selects e campos -> clicar em criar usuário -> ver mensagem de sucesso -> visualizar na lista

Cenário 2 (edição): clicar em editar -> preencher todos os selects e campos -> clicar em confirmar -> ver mensagem de sucesso -> visualizar na lista

Cenário 3 (exclusão): clicar em excluir -> clicar em confirmar -> ver mensagem de sucesso -> visualizar na lista

4) Definir a anatomia da tela (layout)

- **Título:** 'Gerenciar usuários'
- **Campos:** Nome, email(com máscara), função (select),senha
- **Botões:** Criar, editar e excluir
- **Mensagens:** Sucesso / erro

5) Preparar estados da UI

- Estado normal
- Estado com campos preenchidos
- Estado com erro (email)
- Estado de sucesso ("Usuário criado com sucesso!")
- Estado de carregamento (ao criar)

- Estado de edição
- Estado de confirmação de exclusão

6) Checklist UI ↔ Critérios de aceitação

CRITÉRIO DE AVALIAÇÃO	COMPONENTE / TELA	VALIDAÇÃO VISUAL
Validar email	Campo email	Erro inline ao digitar inválido
Editar usuário	Botão editar	Substituir dados na tabela
Excluir usuário	Botão excluir	Remover linha da tabela
Mensagem de sucesso	Banner verde no final	Após clicar criar com dados válidos
Exibir usuário na lista	Lista de usuários	Novo item visível no topo

7) ROTEIRO DE TESTE

1. Abrir a tela “Acessar Usuários”.
2. Selecionar nome.
3. Informar um email inválido -> ver mensagem de erro.
4. Corrigir email e selecionar função e senha
5. Clicar em Criar -> ver banner de sucesso.
6. Lista atualizada automaticamente -> atualização visual.
7. Selecionar editar.
8. Selecionar nome.
9. Informar um email inválido -> ver mensagem de erro.
10. Corrigir email e selecionar função e senha.
11. Clicar em confirmar -> ver banner de sucesso.

12. Lista atualizada automaticamente -> atualização visual.
13. Selecionar excluir.
14. Clicar em confirmar -> ver banner de sucesso.
15. Lista atualizada automaticamente -> atualização visual.

História 4 - Interface de agenda médica diária

1) Fixar o escopo

História de usuário: *Como médico, quero visualizar a agenda diária de consultas com horários, pacientes e status da consulta, para acompanhar melhor o fluxo de atendimentos*

Extração:

- **Ator:** médico
- **Ação:** visualizar agenda diária
- **Benefício:** acompanhar consultas e organizar melhor o fluxo de atendimentos.

Objetivo: Permitir a visualização clara da agenda do dia, exibindo horários, pacientes e status da consulta para facilitar o controle operacional

2) Mapear requisitos visuais a partir dos critérios de aceitação

Critérios de Aceitação:

1. Exibir a agenda diária com pacientes e status da consulta organizados por horário -> grade de cards
2. Grade atualizada automaticamente -> atualização visual.
3. Exibir nenhuma consulta agendada quando não houverem consultas -> ver banner de grade vazia

3) Derivar fluxos de interface a partir dos cenários

Cenário 1 (agenda com consultas): carregar lista -> usuário visualiza horários -> ver consulta com paciente/horário/status

Cenário 2 (agenda vazia): ver mensagem de nenhuma consulta agendada

Cenário 3 (atualização): novo agendamento salvo -> agenda atualiza -> novo item aparece automaticamente.

4) Definir a anatomia da tela (layout)

- **Título:** 'Agenda Médica - Hoje'
- **Botões:** Hoje
- **Mensagens:** Vazia

5) Preparar estados da UI

- Estado normal
- Estado vazio
- Estado carregamento(quando atualiza uma nova consulta)

6) Checklist UI ↔ Critérios de aceitação

CRITÉRIO DE AVALIAÇÃO	COMPONENTE / TELA	VALIDAÇÃO VISUAL
Exibir agenda diária	Lista de consultas por horário	Horários ordenados de forma crescente
Atualizar automaticamente	Agenda médica	Novo item visível no topo
Exibir mensagem quando vazia	Agenda médica	Mensagem cinza centralizada “nenhuma consulta agendada para hoje”

7) ROTEIRO DE TESTE

1. Abrir a tela “Agenda Médica”.
2. Verificar se a data atual aparece no topo.
3. Confirmar se todas as consultas estão organizadas por horário.
4. Verificar se cada item mostra: horário, paciente e status.
5. Agendar uma nova consulta → Retornar à Agenda Médica → verificar se a nova consulta aparece.
6. Apagar todas as consultas → Recarregar → ver se aparece “Nenhuma consulta agendada para hoje”.

História 5 - Interface do painel inicial

1) Fixar o escopo

História de usuário: *Como gestor, quero visualizar rapidamente indicadores gerais como o total de consultas do dia, médicos ativos, cancelamentos, etc, para acompanhar a movimentação diária da clínica*

Extração:

- **Ator:** gestor
- **Ação:** visualizar indicadores
- **Benefício:** facilita a tomada de decisões rápidas.

Objetivo: Exibir indicadores de desempenho e atividades da clínica de forma clara

2) Mapear requisitos visuais a partir dos critérios de aceitação

Critérios de Aceitação:

1. Exibir cartões de indicadores como consultas do dia, pacientes cadastrados, etc

2. Exibir atividades recentes e visão rápida -> lista vertical
3. Exibir próximas consultas -> tabela
4. Exibir gráfico semanal -> SVG

3) Derivar fluxos de interface a partir dos cenários

Cenário 1 (visualização normal): carregar métricas como cartões numéricos, gráficos e tabelas -> usuário visualiza o panorama geral do clínica

Cenário 2 (atualização): enviar dados ao sistema -> página inicial é recarregada -> cartões numéricos, gráficos e tabelas atualizam -> usuário visualiza o panorama geral atualizado da clínica

4) Definir a anatomia da tela (layout)

- **Título:** 'Página inicial'
- **Mensagem:** Vazia

5) Preparar estados da UI

- Estado normal
- Estado vazio
- Estado carregamento (quando atualizam as métricas)

6) Checklist UI ↔ Critérios de aceitação

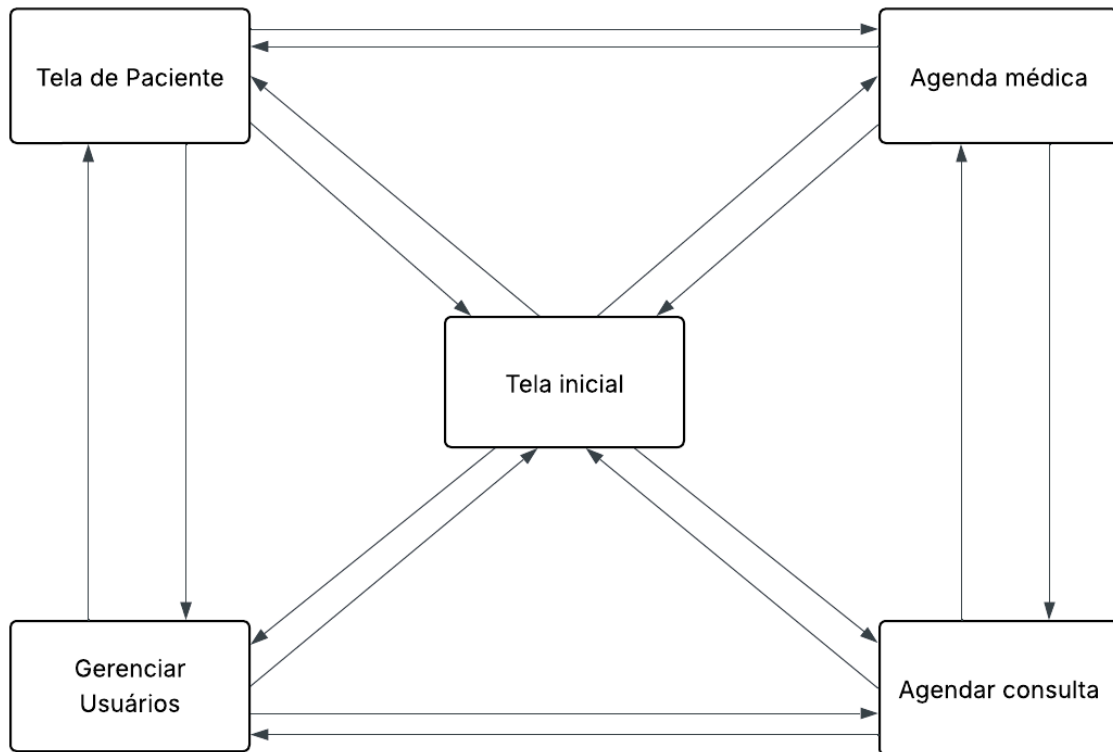
CRITÉRIO DE AVALIAÇÃO	COMPONENTE / TELA	VALIDAÇÃO VISUAL
Mostrar métricas	Cards	Números atualizados
Exibir consultas	Tabela	Lista de itens
Exibir atividades	Painel lateral	Lista atualizada

7) ROTEIRO DE TESTE

1. Abrir a tela "Página inicial".
2. Conferir números dos cards.
3. Verificar tabela de consultas.

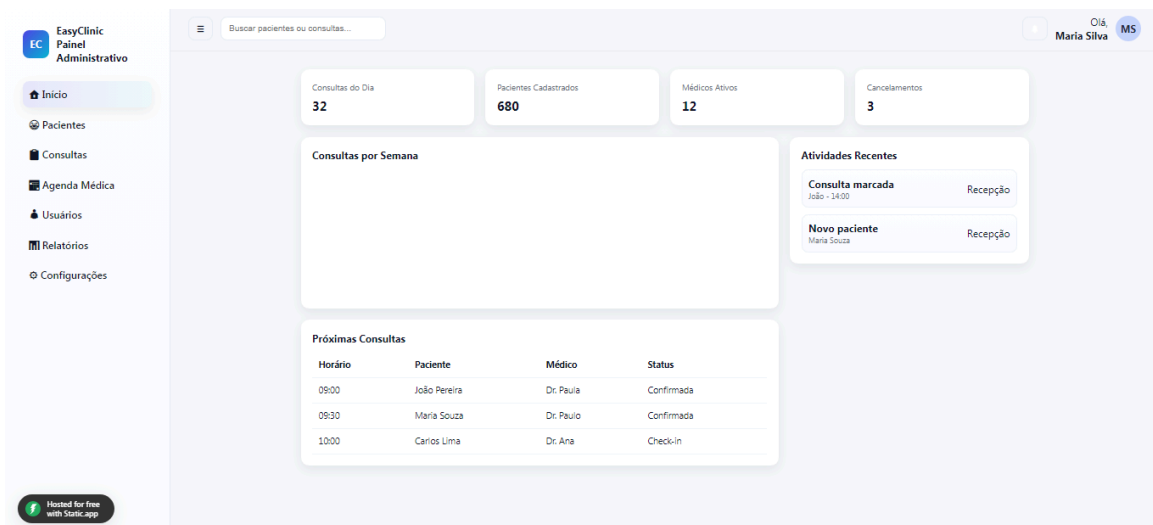
4. Analisar painel de atividades.

Fluxo de navegação das telas

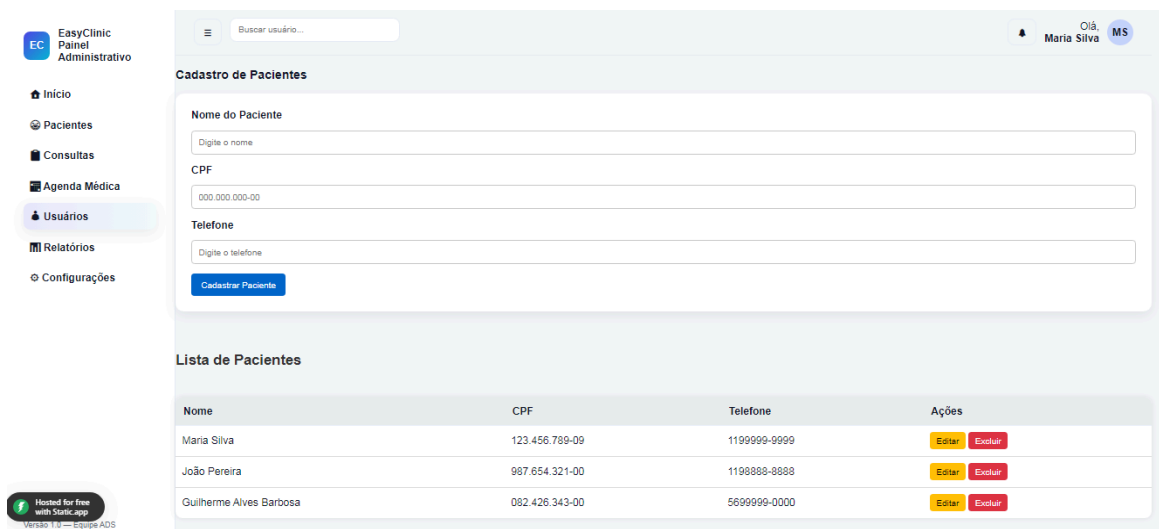


Imagens das interfaces

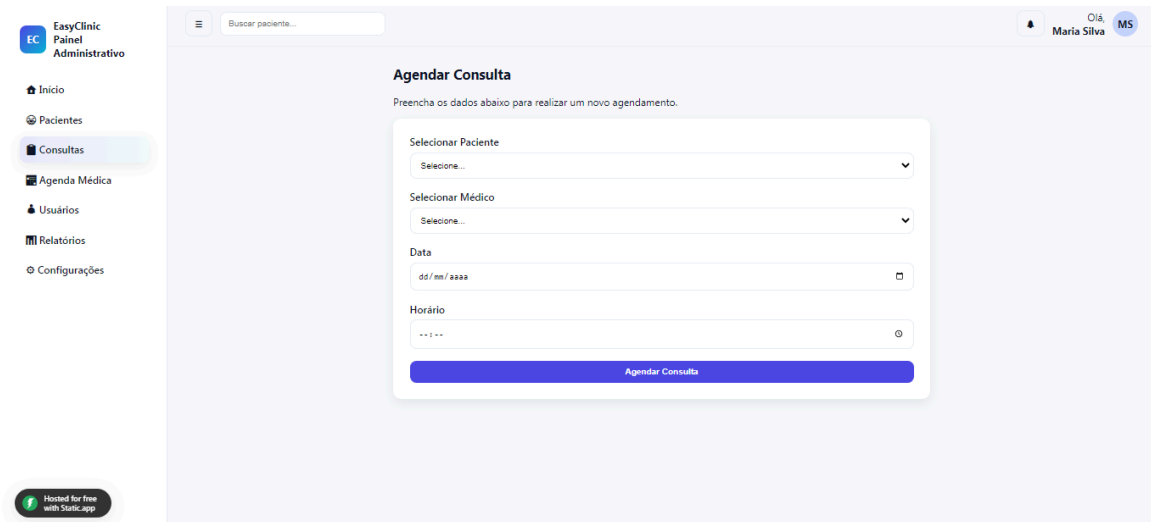
Tela inicial:



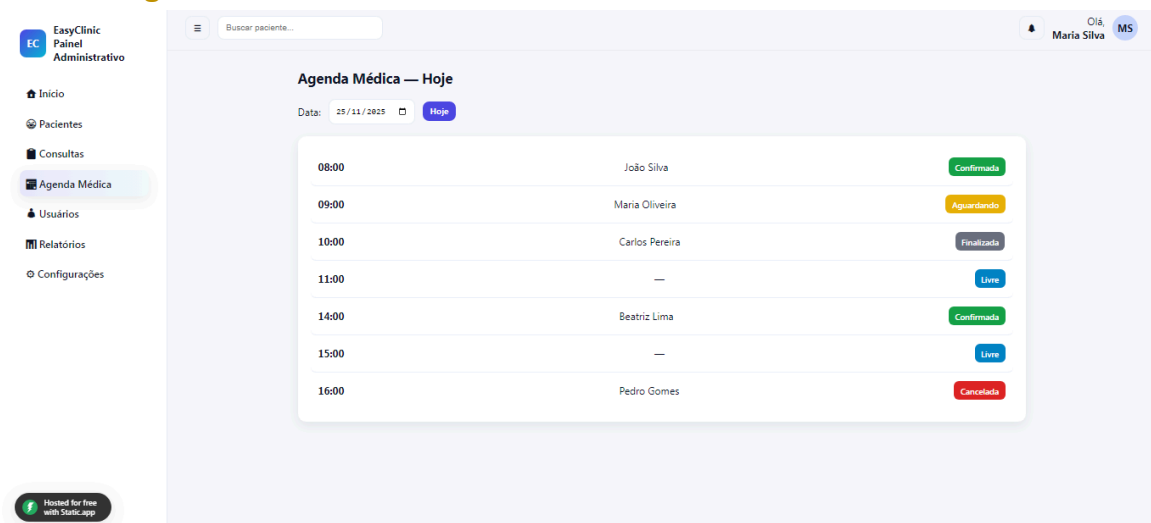
Tela de pacientes:



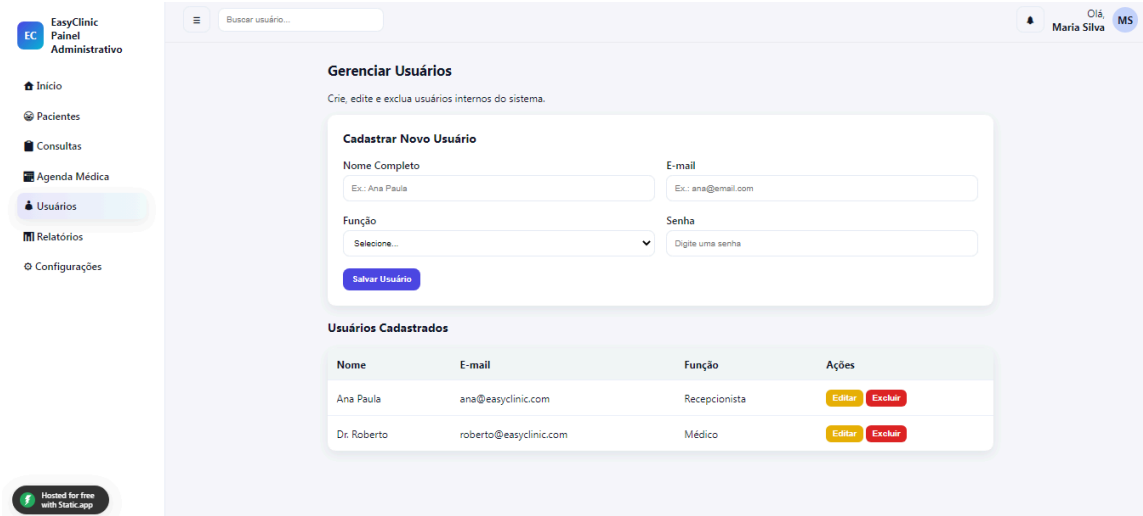
Tela de agendar consultas:



Tela de agenda médica:



Tela de usuários:



Registro das Cerimônias da sprint

Sprint Planning (24/11/2025)

Formato: Conversa pelo Whatsapp

Duração: 1 hora

Resumo das decisões:

- Leituras do documento de guia dessa sprint para definir as entregas obrigatórias.
- Seleção das histórias prioritárias que seriam refinadas a partir do backlog da sprint 01.
- Organização do Trello com as novas colunas da sprint 02 de backlogs e reuniões.
- Distribuição dos papéis.

Daily Meeting (25/11/2025)

Formato: Reunião pelo discord

Duração: 30 minutos

Avanços informados:

- Refinamento inicial das histórias de usuário escolhidas como prioritárias.
- Início da construção da nova estrutura para o documento de entrega da sprint no Google Documentos.
- Organização das tarefas no Trello com responsáveis definidos.
- Início de produção do slide.

Daily Meeting (26/11/2025)

Formato: Reunião pelo discord

Duração: 30 minutos

Progresso:

- Histórias de usuários finalizadas com os critérios de aceitação e cenários de uso.
- Construção dos fluxos de interface com base nos cenários.
- Início de criação dos roteiros e das telas das interfaces com base neles.
- Criação de pasta do sprint 02 no Github e implementação do slide finalizado nela.
- Melhorias no HTML para garantir a interligação entre as telas.

Daily Meeting (27/11/2025)

Formato: Reunião presencial

Duração: 40 minutos

Encerramentos da Sprint:

- Conclusão dos roteiros de teste e checklist das interfaces.
- Finalização do fluxo de navegação completo do EasyClinic.

- HTML já concluído e pronto para submissão após uma revisão.
- Documento da sprint 02 já finalizado com a estrutura adequada.

Sprint Review (28/11/2025)

Formato: Reunião pelo Discord

Duração: 30 minutos

A equipe apresentou:

- Interfaces do HTML do sprint 02 finalizadas.
- Documento da sprint 02 praticamente concluído.
- Um repositório do Github com organização adequada e com o documento da sprint e o HTML
- Quadro do Trello com o fluxo da sprint

Feedback:

- As interfaces estão bem mais consistentes que a apresentada na sprint 01
- Focar em armazenamento de dados e validações de dados na próxima sprint

Retrospectiva (28/11/2025)

Formato: Reunião pelo Discord

Duração: 10 minutos

Pontos positivos:

- Refinamento claro das histórias e cenários
- Boa distribuição das tarefas entre os membros
- HTML bem construído e organizado

Pontos a melhorar:

- Dividir o trabalho do documento mais cedo para evitar correrias
- Antecipar os prazos de commits no Github

Ações para a próxima Sprint:

- Iniciar a prototipação antes da metade do sprint
- Adotar padrões fixos de commits no Github e de adição de quadros no Trello