

# Product Backlog

Projeto / MVP - Engenharia de Software

quatro MVP

Resumo

Cronograma

Backlog

Sprints ativos

Calendário

Relatórios

Lista

Formulários

Metas

Todos os tickets

Componentes

Código

Versões

Tickets arquivados

Páginas

Atalhos

Pesquisar no back...

Versão

Epic

Filtros rápidos

+ Criar

Epic

Sem epic

> Acesso ao Aplicativo

> Gerenciamento de Compromissos

+ Criar epic

quatro Sprint 1

Adicionar datas (2 tickets)

MVP-3 Login de Usuário

MVP-10 [Backend] - Desenvolvimento da API de login

MVP-11 [Android] - Desenvolvimento da tela de login

MVP-12 [iOS] - Desenvolvimento da tela de login

MVP-2 Cadastro de Usuário

MVP-13 [Backend] - Desenvolvimento API de cadastro

MVP-14 [Android] - Desenvolvimento de tela de cadastro

MVP-15 [iOS] - Desenvolvimento de tela de cadastro

+ Criar

quatro Sprint 2

Adicionar datas (2 tickets)

MVP-6 Criação de tela inicial

MVP-22 [Backend] - Desenvolver API para retornar compromissos e dados do usuário

MVP-23 [Android] - Desenvolver telas iniciais

MVP-24 [iOS] - Desenvolver telas iniciais

MVP-7 Criação de Compromisso

MVP-25 [Backend] - Desenvolver API para cadastro de compromisso

MVP-26 [Android] - Desenvolver telas para cadastro de compromisso

MVP-27 [iOS] - Desenvolver telas para cadastro de compromisso

+ Criar

quatro Sprint 3

Adicionar datas (1 ticket)

MVP-8 Notificação de Lembrete de Compromisso

MVP-28 [Backend] - Desenvolver API para envio de Push Notification

MVP-29 [Android] - Desenvolver telas de notificação

MVP-30 [Android] - Desenvolver telas de notificação

+ Criar

quatro Sprint 4

Adicionar datas (2 tickets)

MVP-9 Edição e Exclusão de Compromisso

MVP-19 [Backend] - Desenvolver APIs para alteração de compromissos

MVP-20 [Android] - Desenvolver telas para edição e exclusão de compromissos

MVP-21 [iOS] - Desenvolver telas para edição e exclusão de compromissos

MVP-4 Recuperação de Senha

MVP-16 [Backend] - Desenvolver API para recuperação de senha

MVP-17 [Android] - Desenvolver telas de recuperação de senha

MVP-18 [iOS] - Desenvolver telas de recuperação de senha

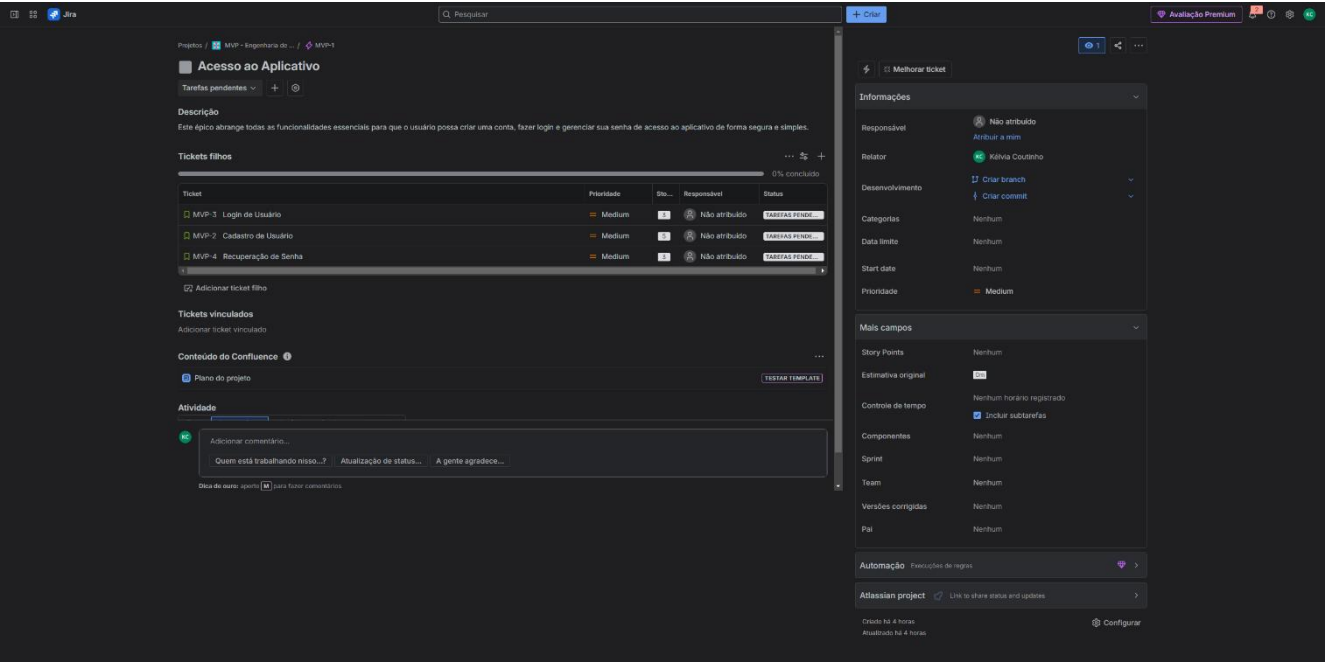
+ Criar

Backlog (0 ticket)

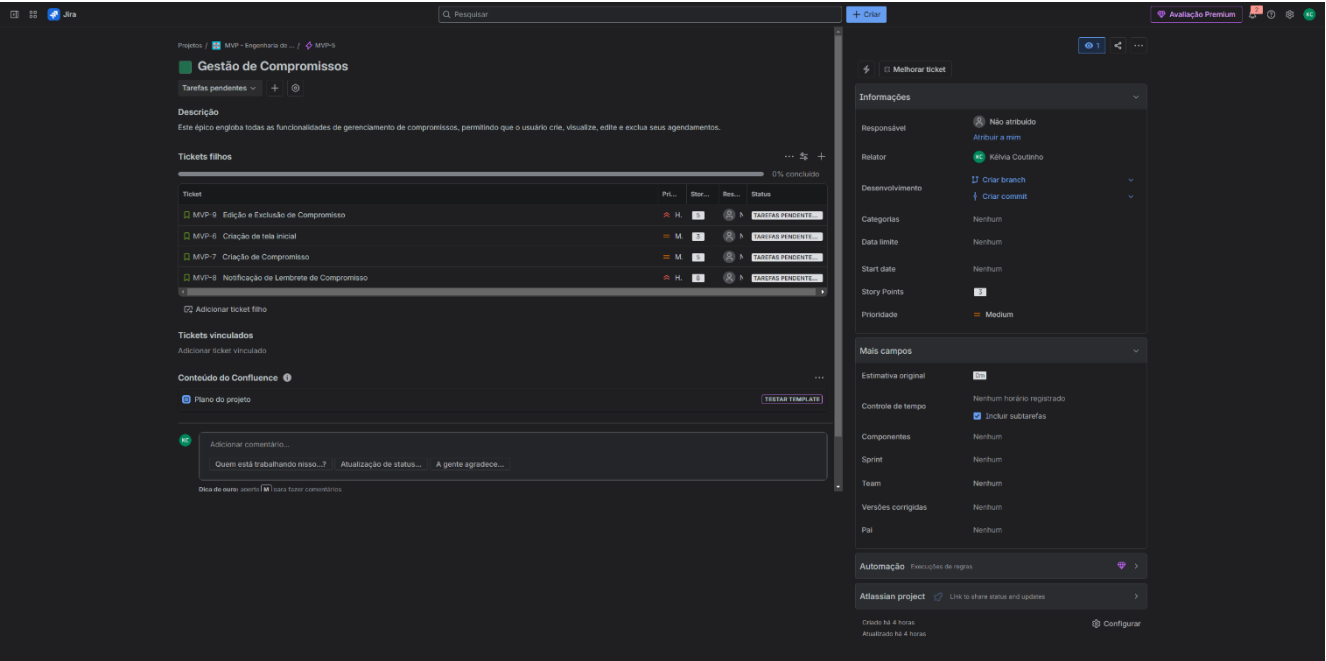
Seu backlog está vazio

+ Criar

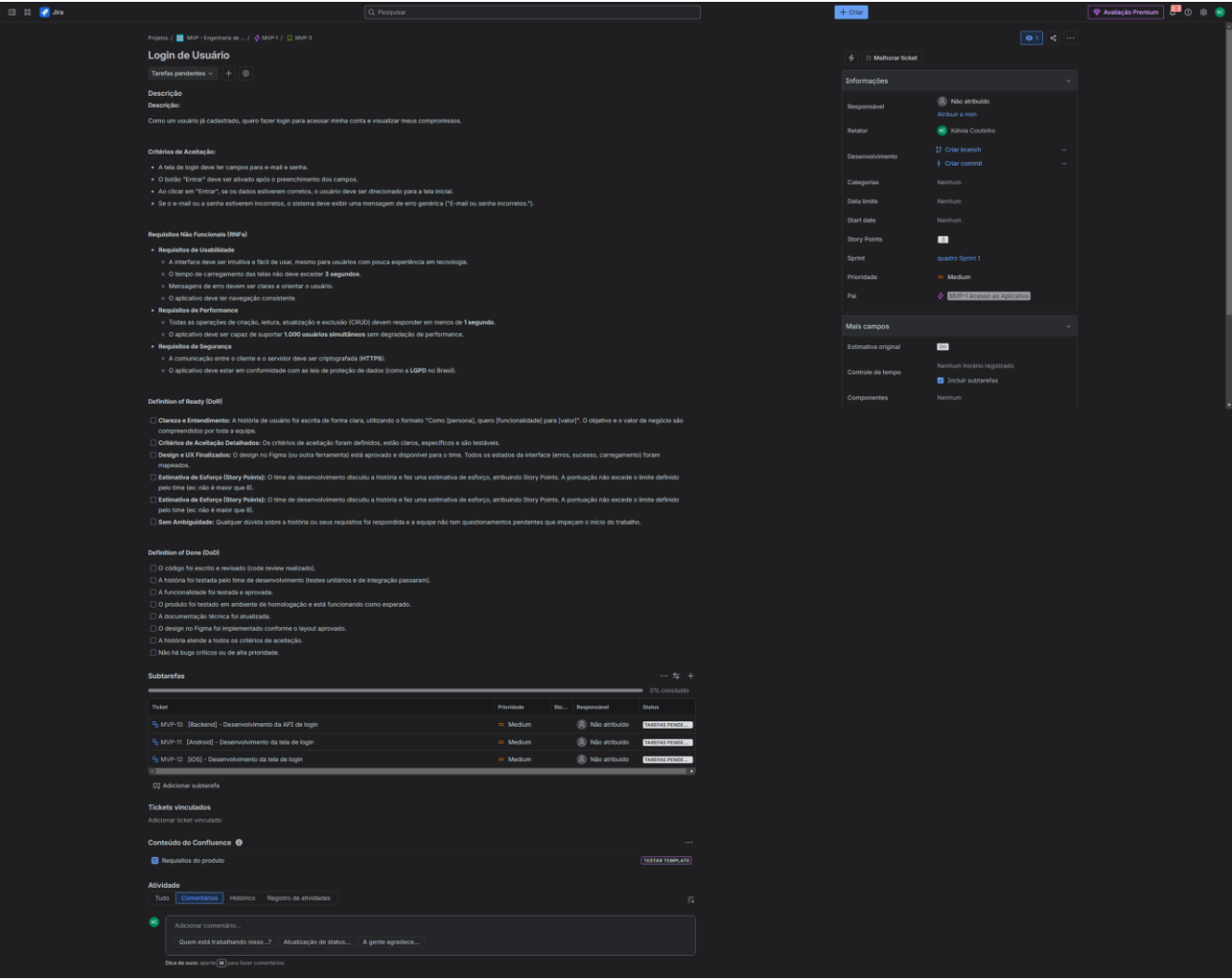
1. Acesso ao aplicativo



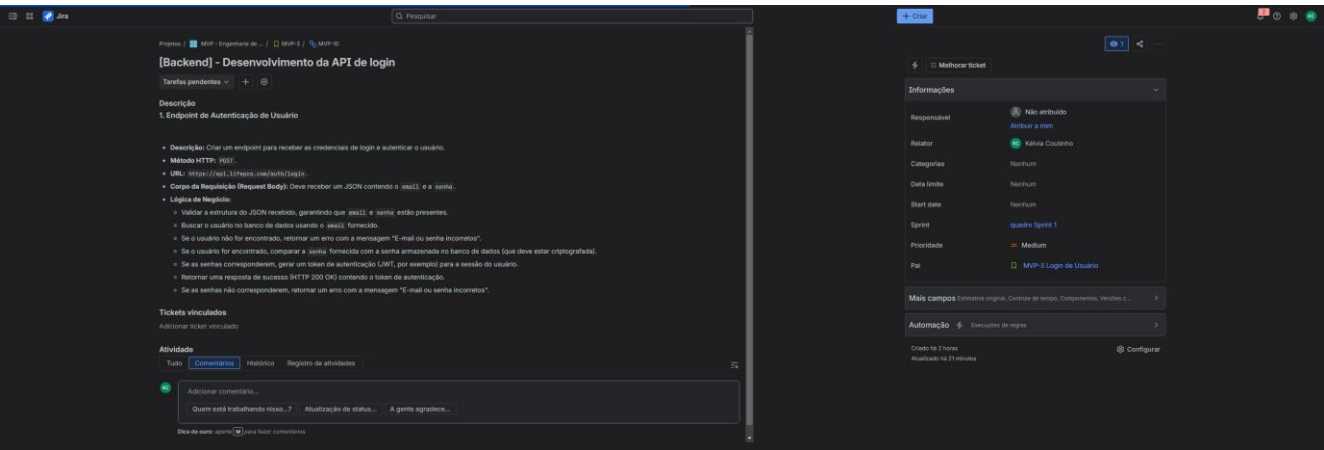
2. Gestão de compromissos

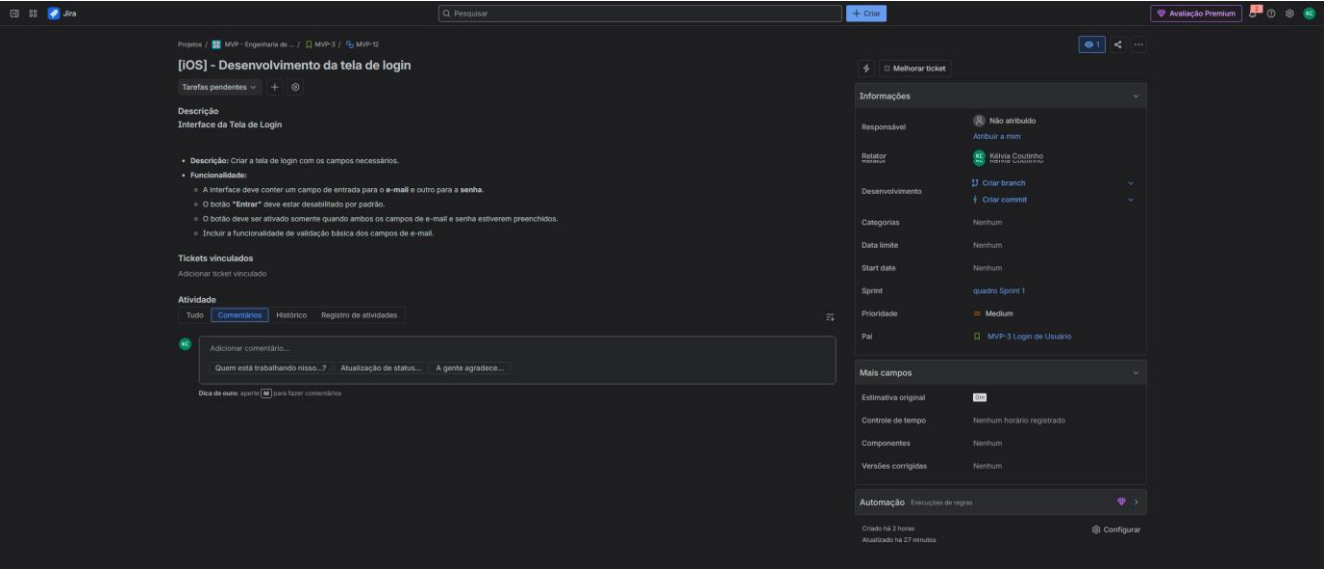
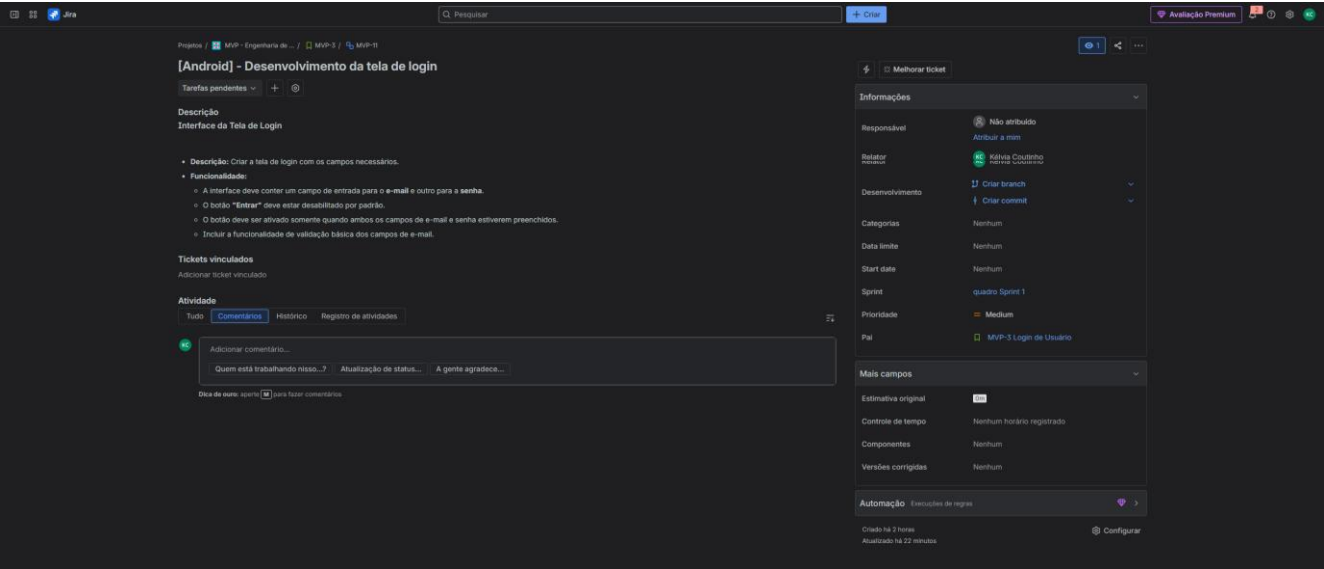


1. Login



1.1. Subtarefas





2. Cadastro de usuário

Projeto

MVP - Engenharia de ...

MVP-1

MVP-2

Cadastro de Usuário

Tarefas pendentes

Descrição

Descrição

Como um novo usuário, quero criar uma conta para ter acesso aos recursos do aplicativo e gerenciar meus compromissos.

Critérios de Aceitação:

- O usuário deve preencher nome completo, e-mail e senha.
- O campo de senha deve ter no mínimo 8 caracteres, incluindo letras, números e um caractere especial.
- A senha deve ser confirmada em um campo separado para evitar erros de digitação.
- O botão "Cadastrar" deve ser ativado após o preenchimento de todos os campos obrigatórios.
- Após o cadastro bem-sucedido, o usuário deve receber uma mensagem de sucesso e ser direcionado para a tela inicial.
- Em caso de e-mail já cadastrado, uma mensagem de erro clara deve ser exibida.

Requisitos Não Funcionais (DNF)

- Requisitos de Usabilidade**
  - A interface deve ser intuitiva e fácil de usar, mesmo para usuários com pouca experiência em tecnologia.
  - O tempo de carregamento das telas não deve exceder **3 segundos**.
  - Mensagens de erro devem ser claras e orientar o usuário.
  - O aplicativo deve ter navegação consistente.
- Requisitos de Performance**
  - Todas as operações de criação, leitura, atualização e exclusão (CRUD) devem responder em menos de **1 segundo**.
  - O aplicativo deve ser capaz de suportar **1.000 usuários simultâneos** sem degradação de performance.
- Requisitos de Segurança**
  - Todas as senhas devem ser armazenadas de forma segura, utilizando técnicas de **hashing e salting**.
  - A comunicação entre o cliente e o servidor deve ser criptografada (HTTPS).
  - O aplicativo deve estar em conformidade com as leis de proteção de dados (LGPD no Brasil).

Definition of Ready (DoR)

- Clareza e Entendimento:** A história de usuário foi escrita de forma clara, utilizando o formato "Como [pessoa], quero [funcionalidade] para [valor]". O objetivo e o valor de negócio são compreensíveis por todos a equipe.
- Critérios de Avaliação Detalhados:** Os critérios de aceitação foram definidos, estão claros, específicos e são testáveis.
- Design e UX Finalizados:** O design no Figma (ou outra ferramenta) está aprovado e disponível para o time. Todos os estados da interface (erros, sucesso, carregamento) foram mapeados.
- Estimativa de Esforço (Story Points):** O time de desenvolvimento discutiu a história e fez uma estimativa de esforço, atribuindo Story Points. A pontuação não excede o limite definido pelo time (ex: não é maior que 8).
- Estimativa de Esforço (Story Points):** O time de desenvolvimento discutiu a história e fez uma estimativa de esforço, atribuindo Story Points. A pontuação não excede o limite definido pelo time (ex: não é maior que 8).
- Sem Ambiguidade:** Qualquer dúvida sobre a história ou seus requisitos foi respondida e a equipe não tem questionamentos pendentes que impeçam o início do trabalho.

Definition of Done (DoD)

- O código foi escrito e revisado (code review realizado).
- A história foi testada pelo time de desenvolvimento (testes unitários e de integração passaram).
- A funcionalidade foi testada e aprovada.
- O produto foi testado em ambiente de homologação e está funcionando como esperado.
- A documentação técnica foi atualizada.
- O design no Figma foi implementado conforme o layout aprovado.
- A história atende a todos os critérios de aceitação.
- Não há bugs críticos ou de alta prioridade.

Subtarefas

0% concluído

| Tarefa   | Prioridade | Est... | Responsável   | Status       |
|--|------------|--------|---------------|--------------|
| MVP-13 [Backend] - Desenvolvimento API de cadastro     | Medium     |        | Não atribuído | EM ANDAMENTO |
| MVP-14 [Android] - Desenvolvimento de tela de cadastro | Medium     |        | Não atribuído | EM ANDAMENTO |
| MVP-15 [iOS] - Desenvolvimento de tela de cadastro     | Medium     |        | Não atribuído | EM ANDAMENTO |

Adicionar subtarefa

Tickets vinculados

Adicionar ticket vinculado

Conteúdo do Confluence

Requisitos do produto

TESTAR TEMPLATO

Atividade

Tudo

Comentários

Histórico

Registro de atividades

Melhorar ticket

Informações

Responsável: Não atribuído

Relator: Rafaela Cristina

Desenvolvimento: IT

Categorias: Nenhum

Data limite: Nenhum

Start date: Nenhum

Story Points: 8

Sprint: quadro Sprint 1

Prioridade: Medium

Pal: IMPROVAR ACESSO AO APLICATIVO

Mais campos

Estimativa original: 8

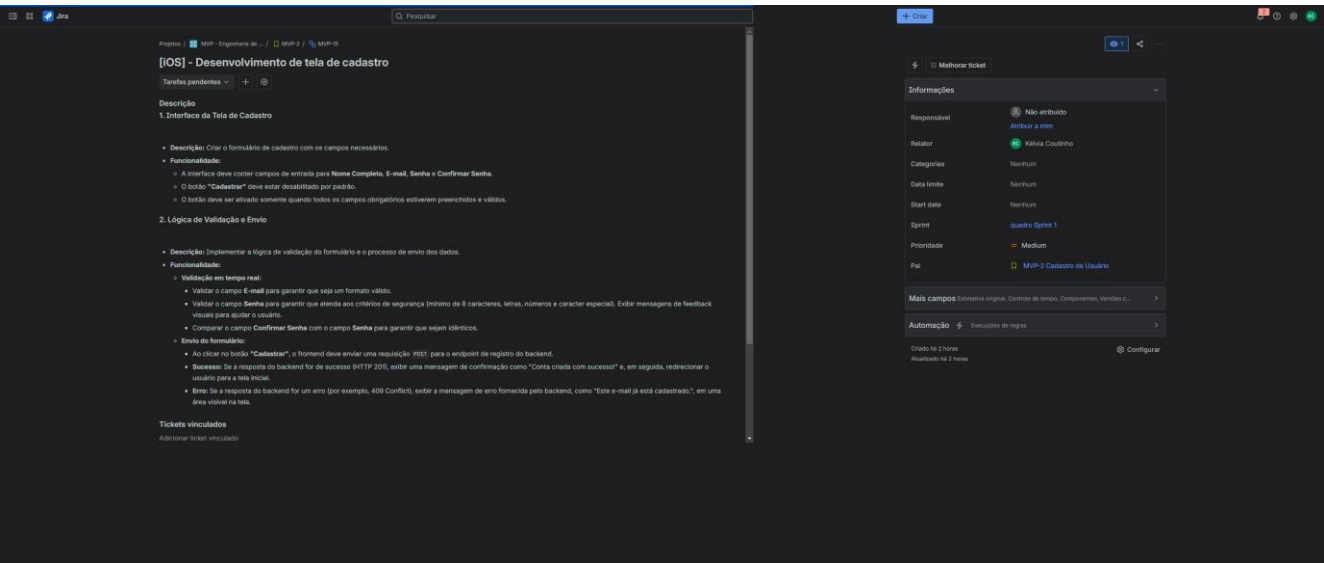
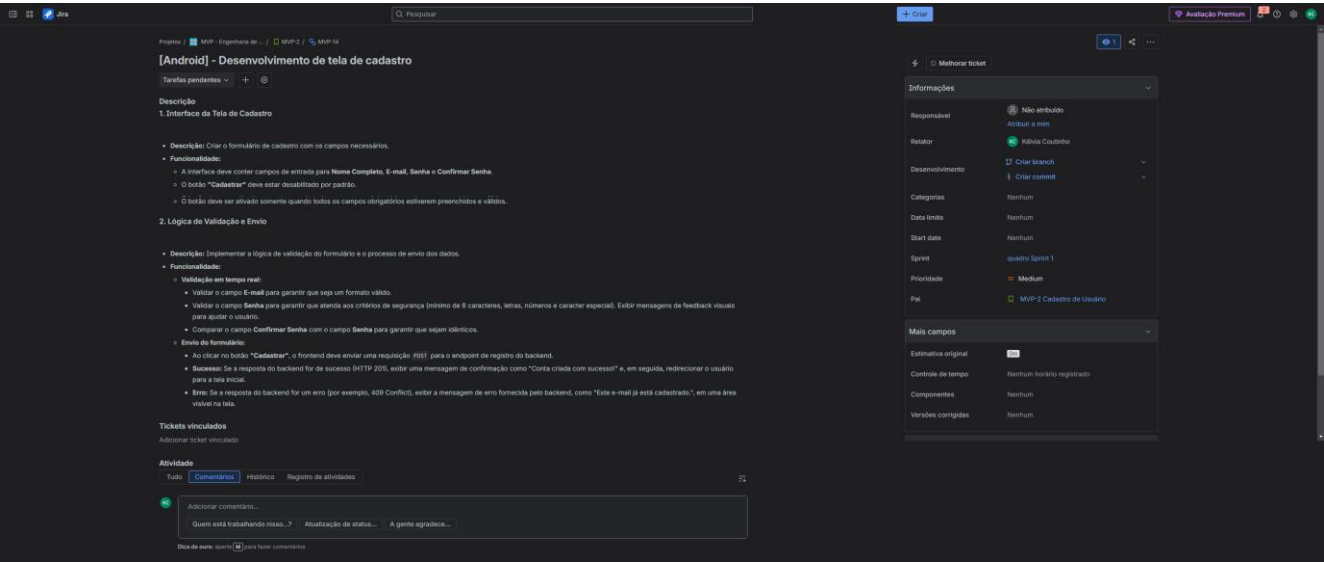
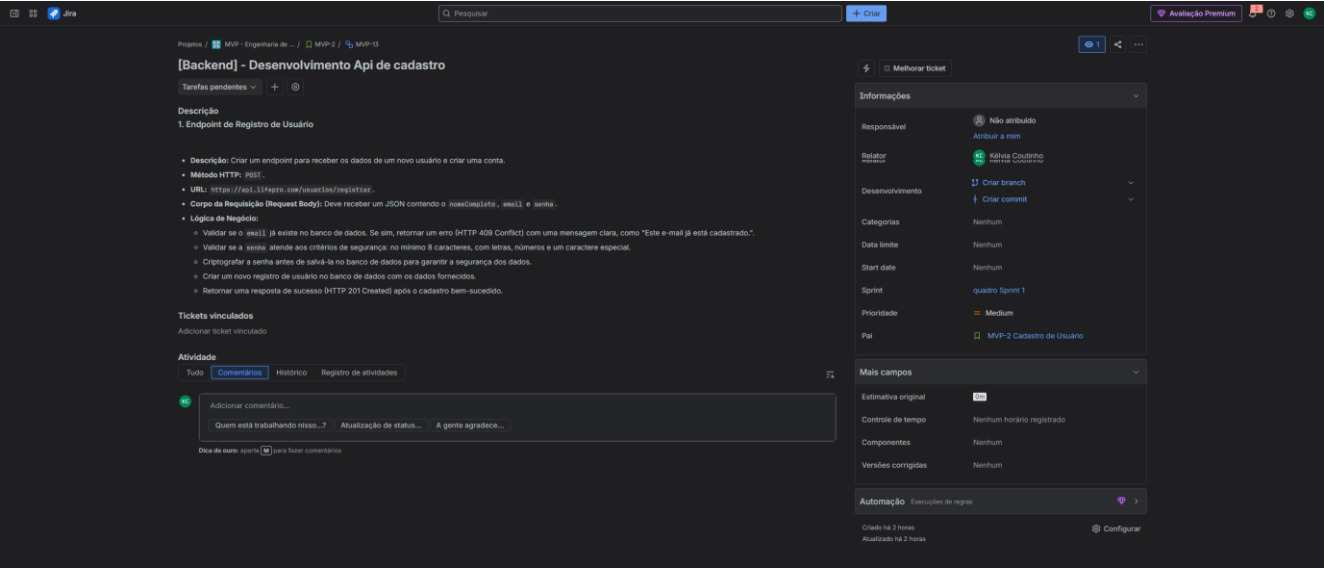
Controle de tempo: Nenhum horário registrado

Componentes: Incluir subtarefas

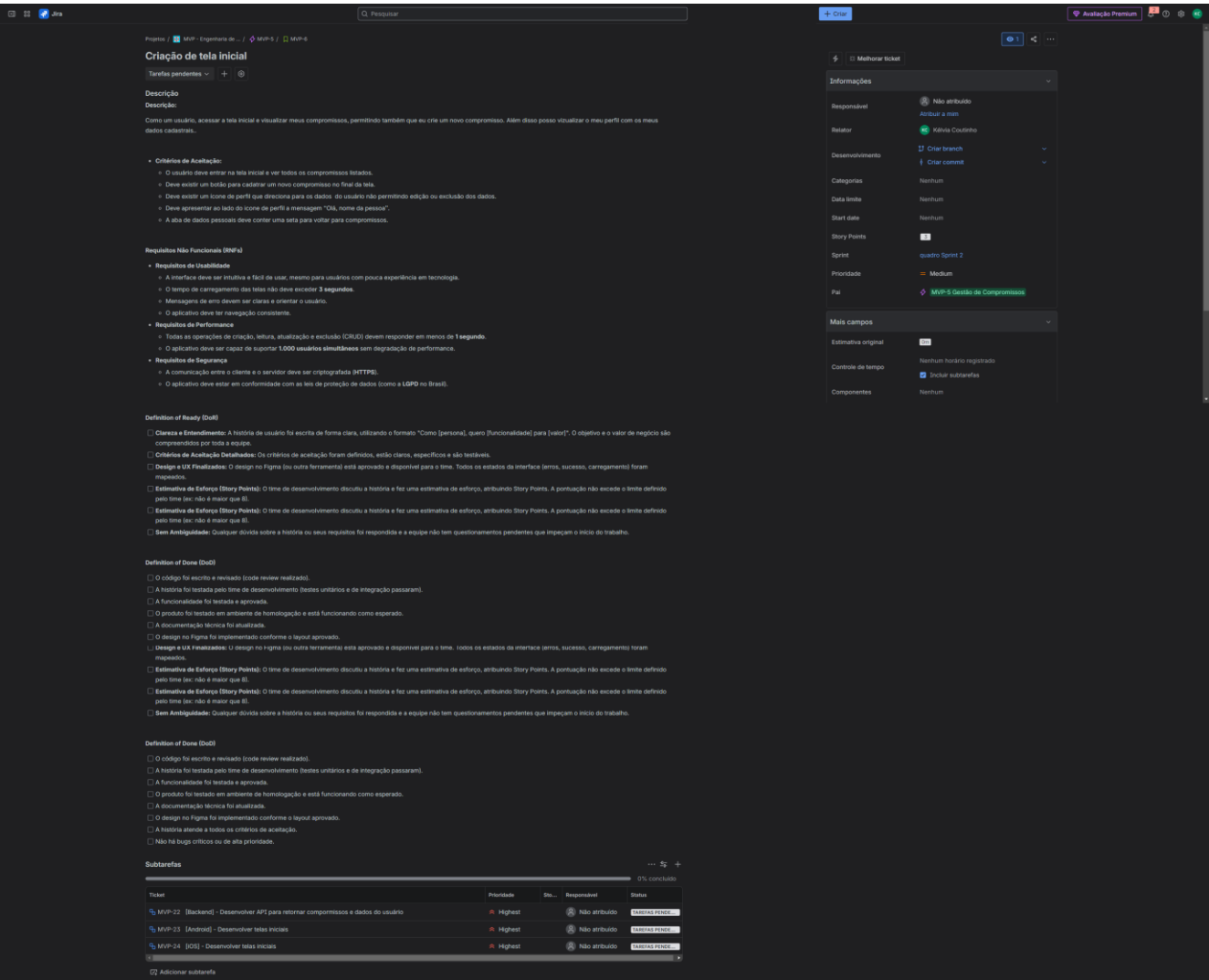
Componentes: Nenhum

Team: Nenhum

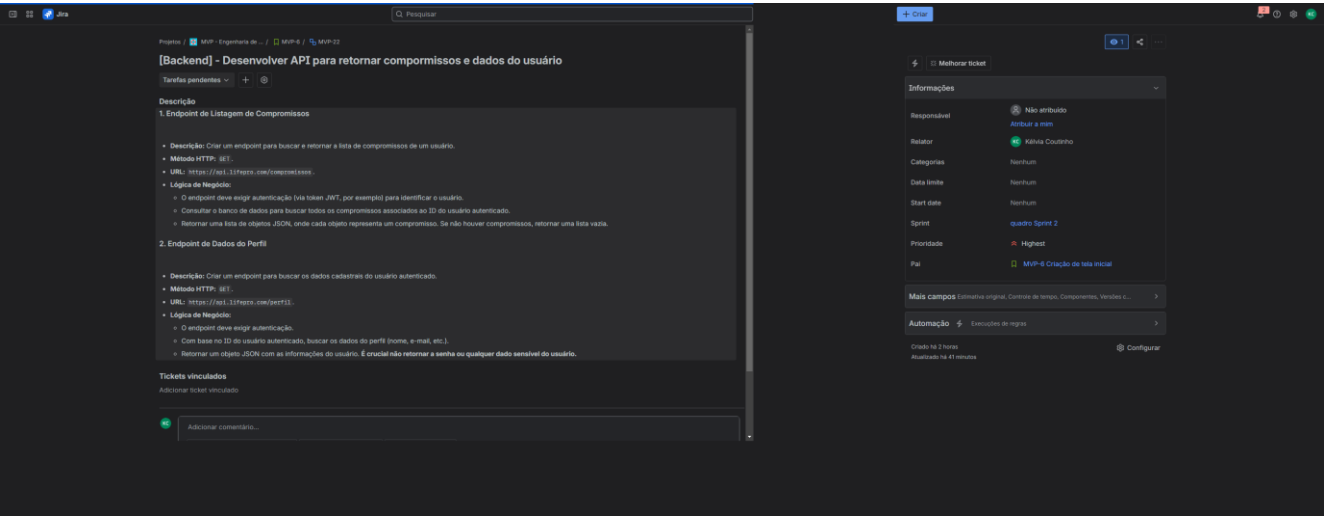
2.1. Subtarefas

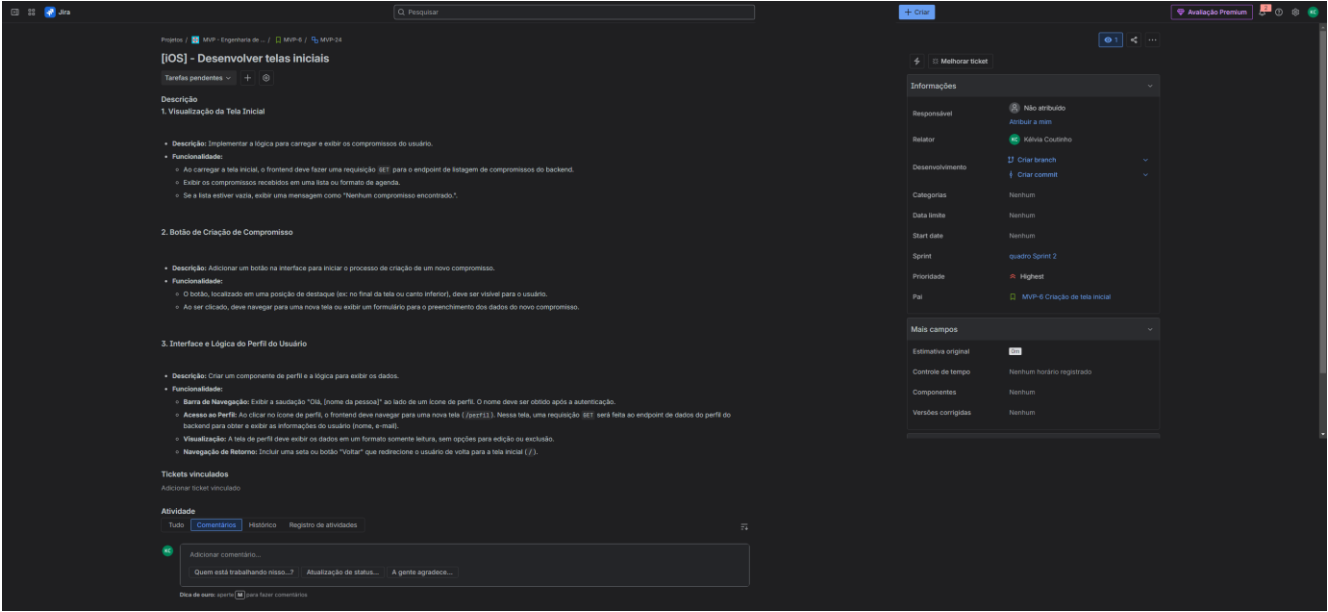
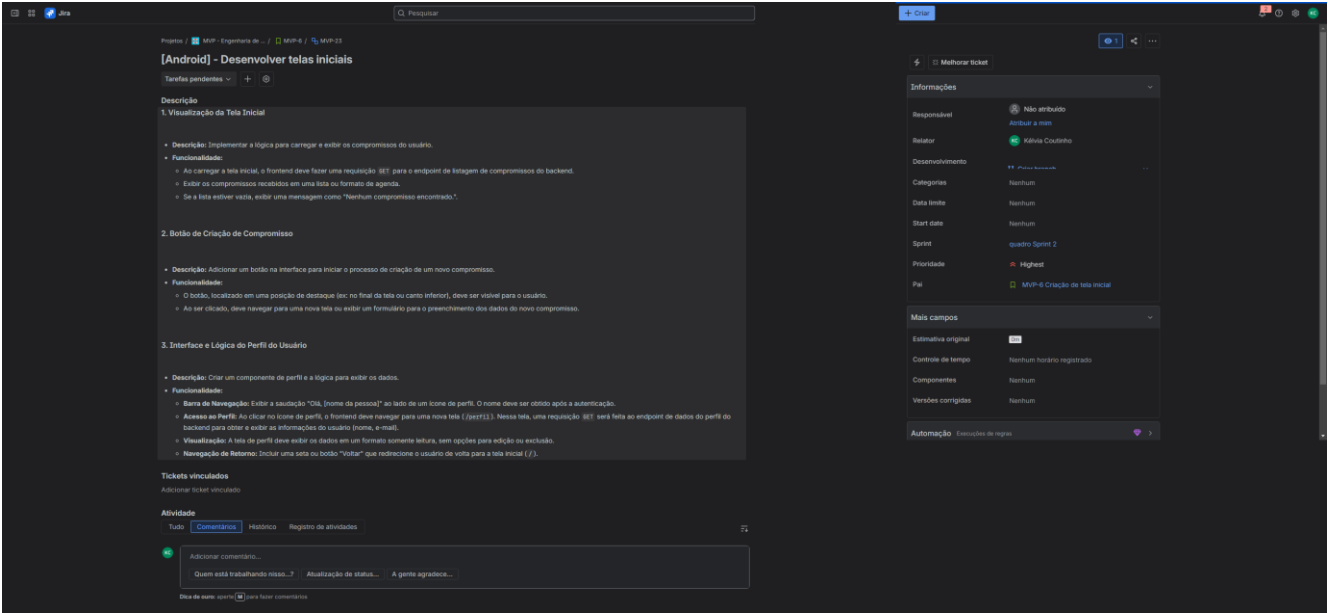


3. Criação de tela inicial



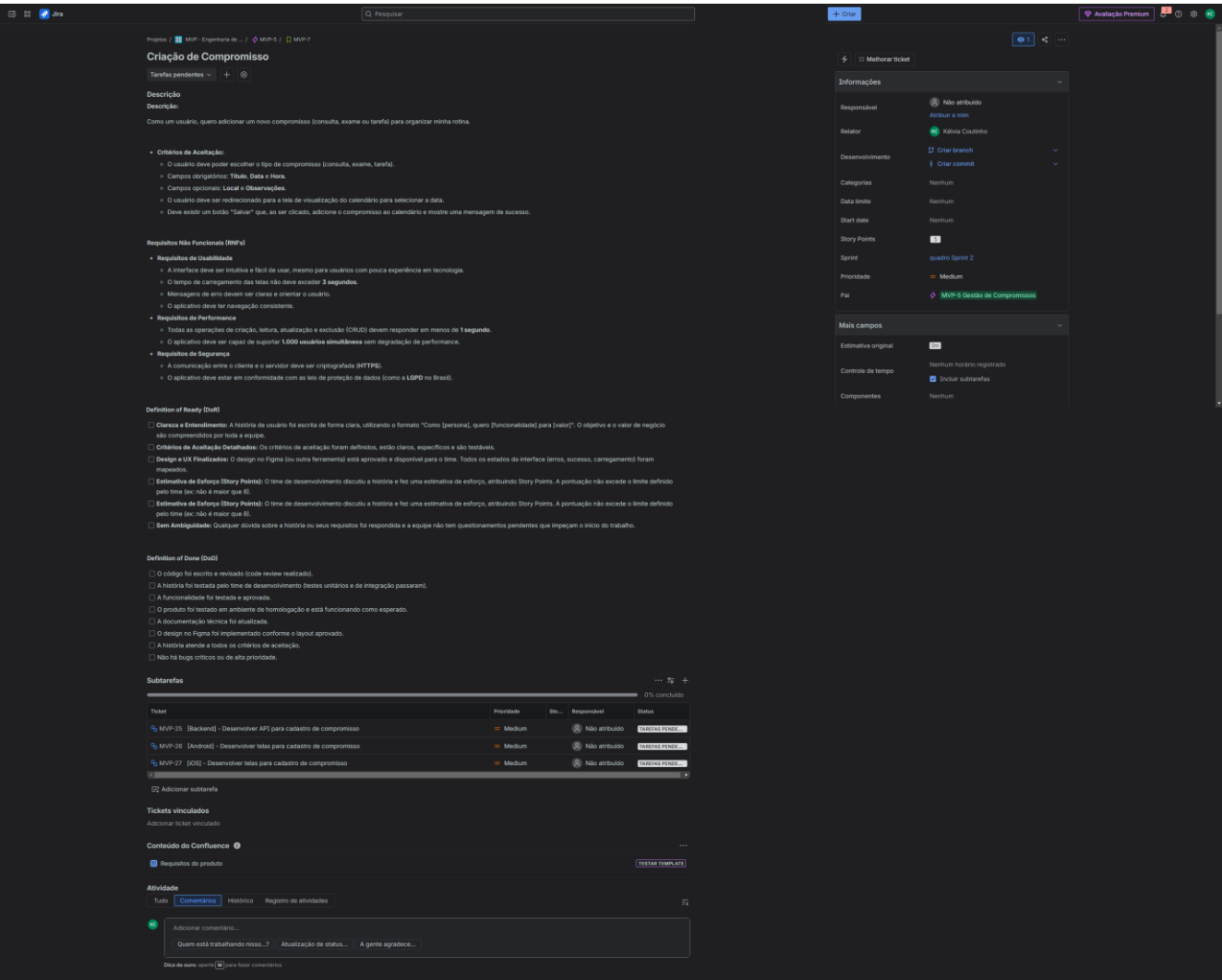
3.1. Subtarefa



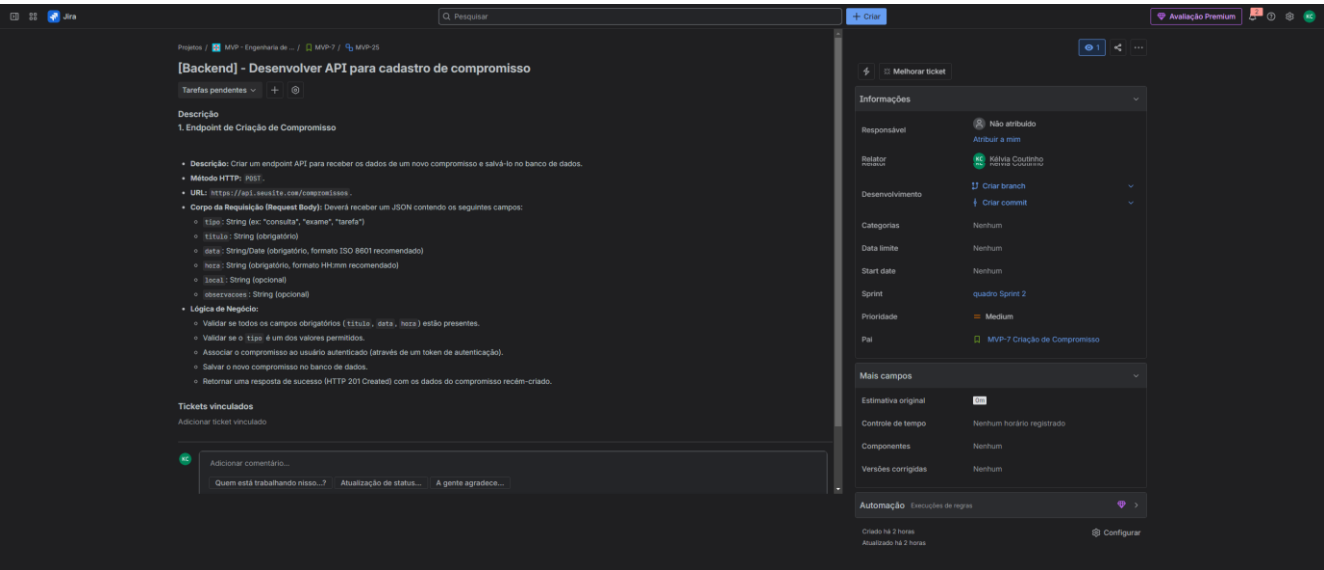


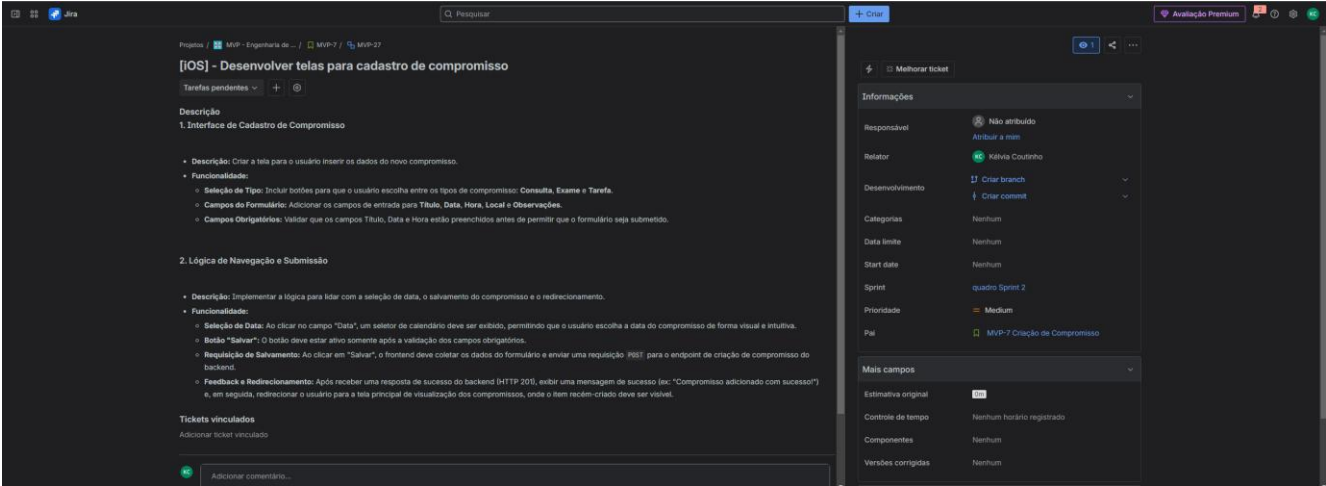
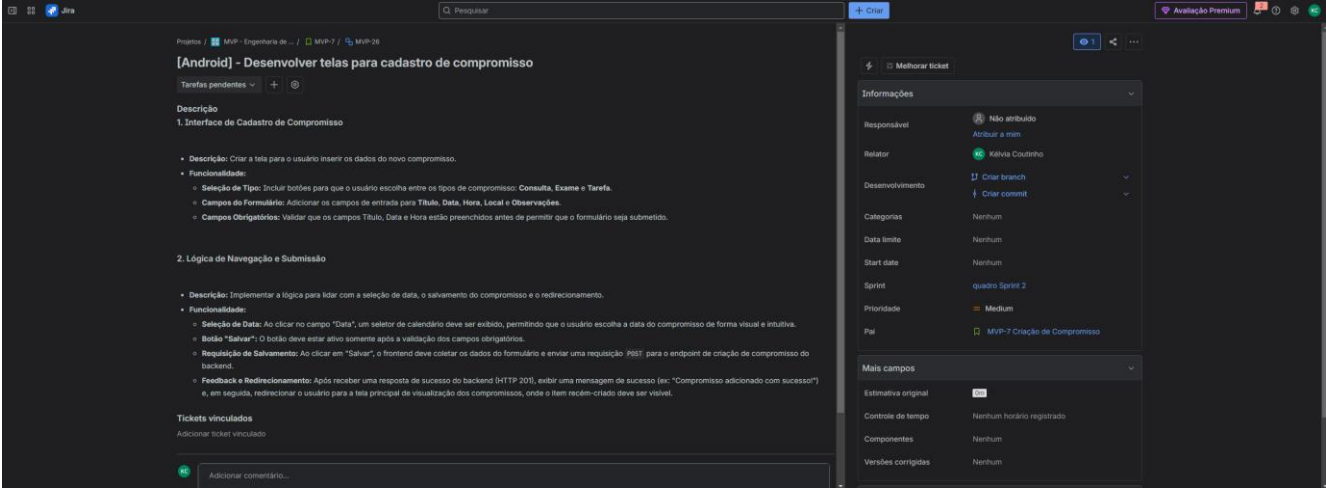


4. Criação de compromissos

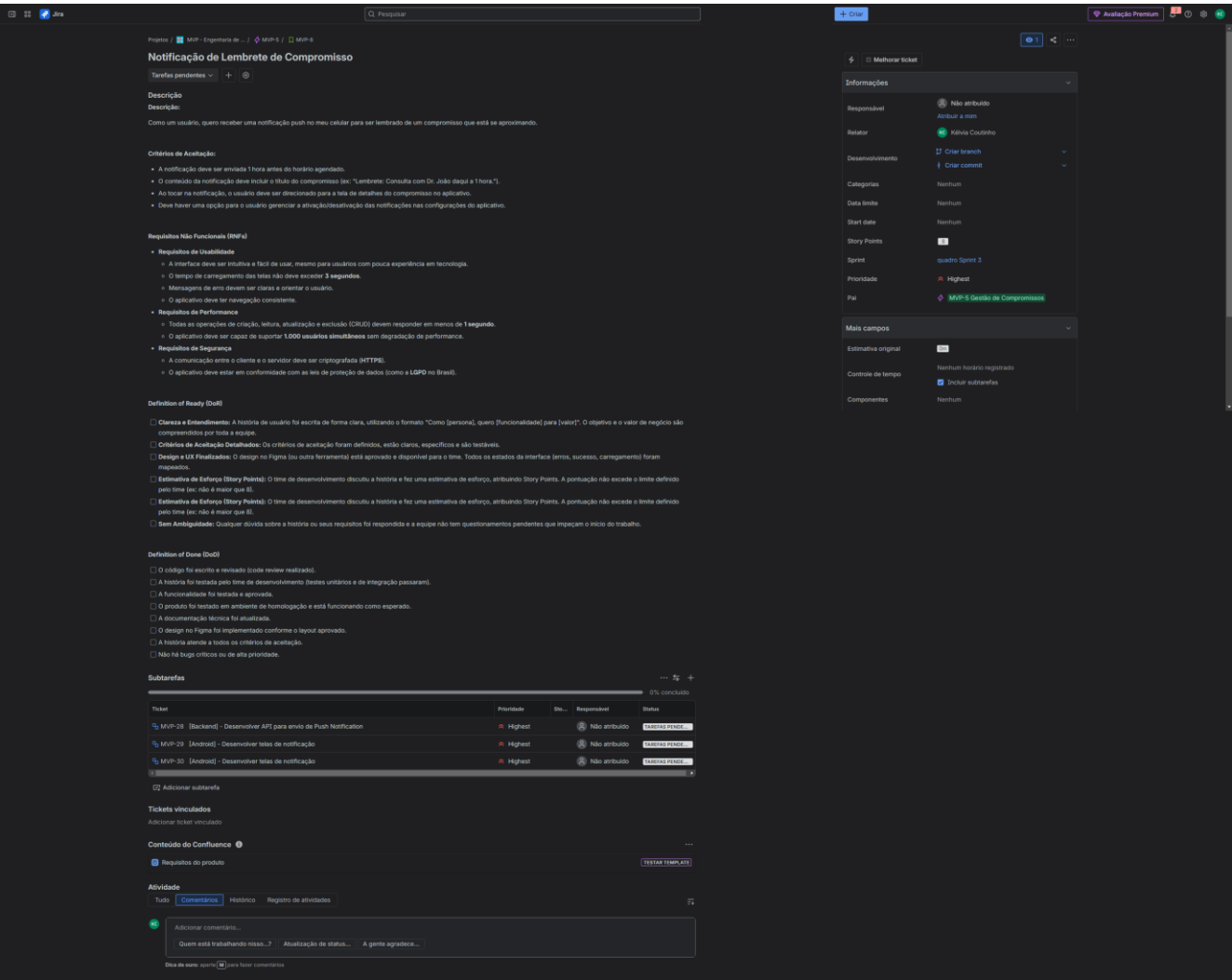


4.1 Subtarefas

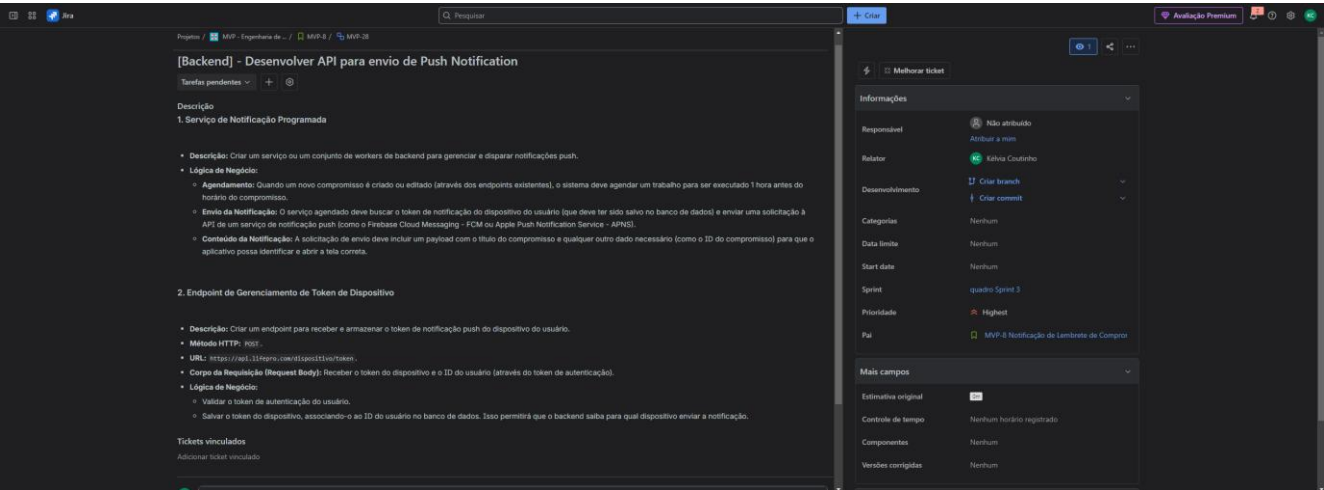


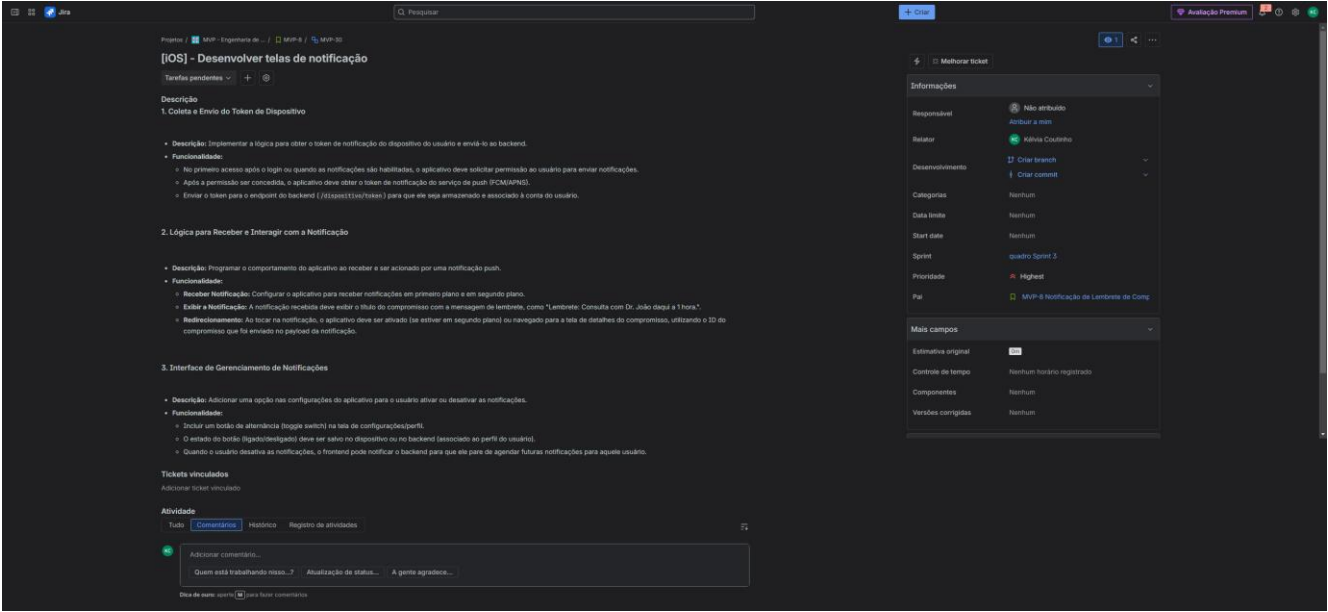
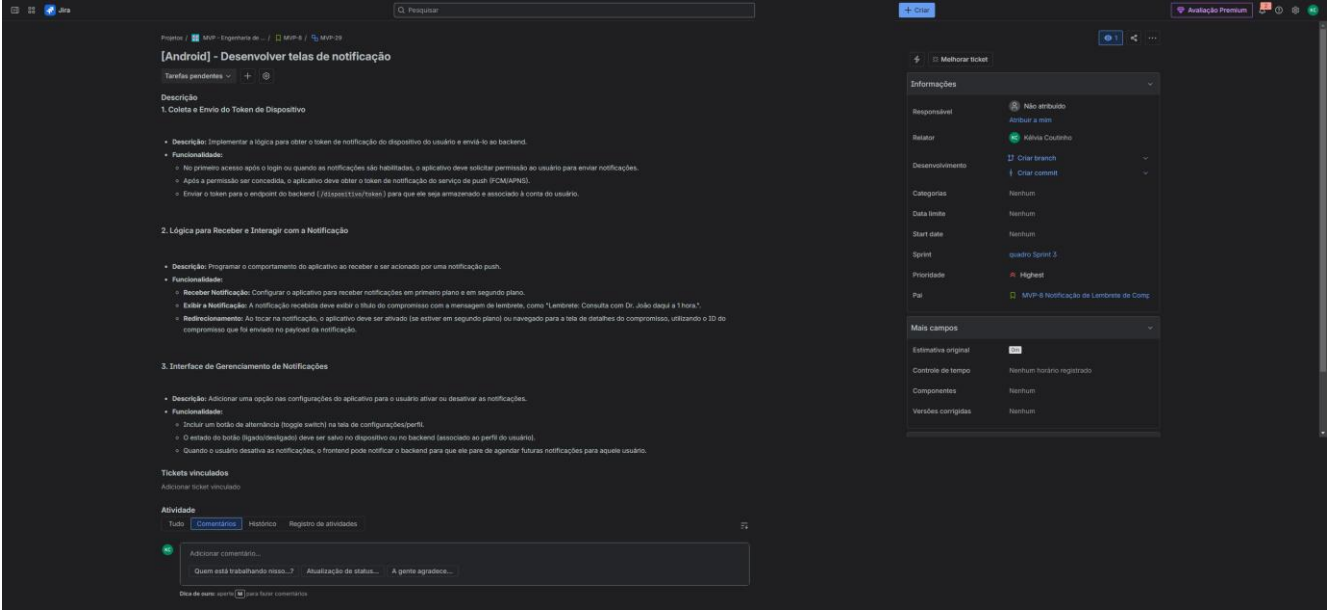


5. Notificação de lembretes de compromissos

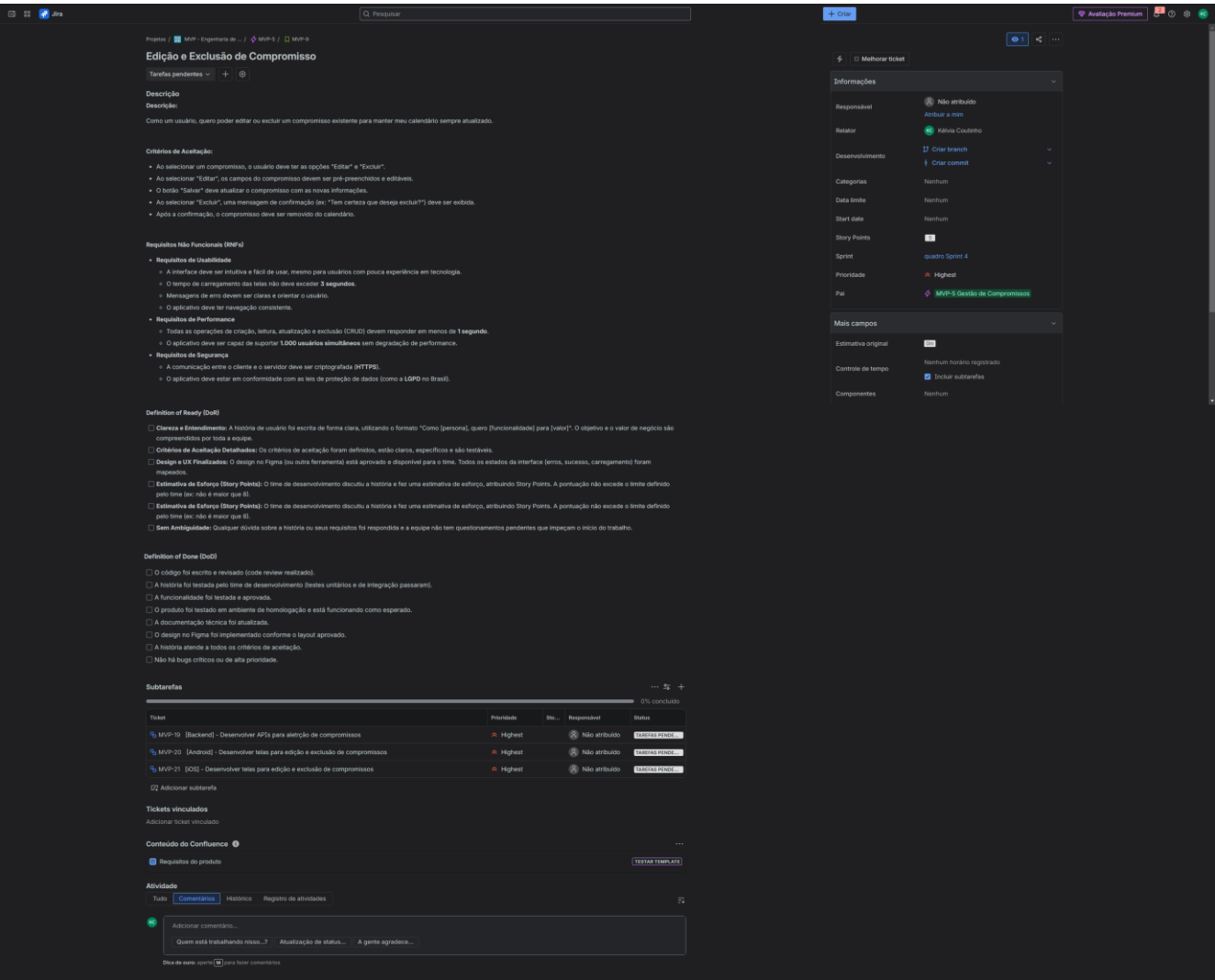


5.1 Subtarefas

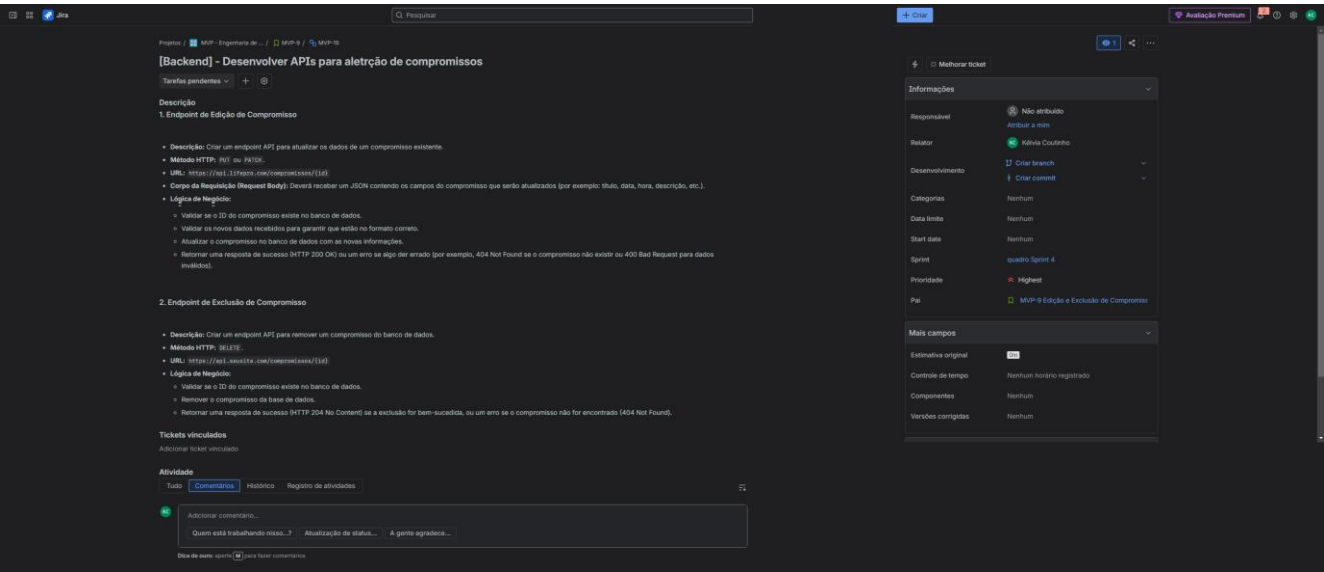


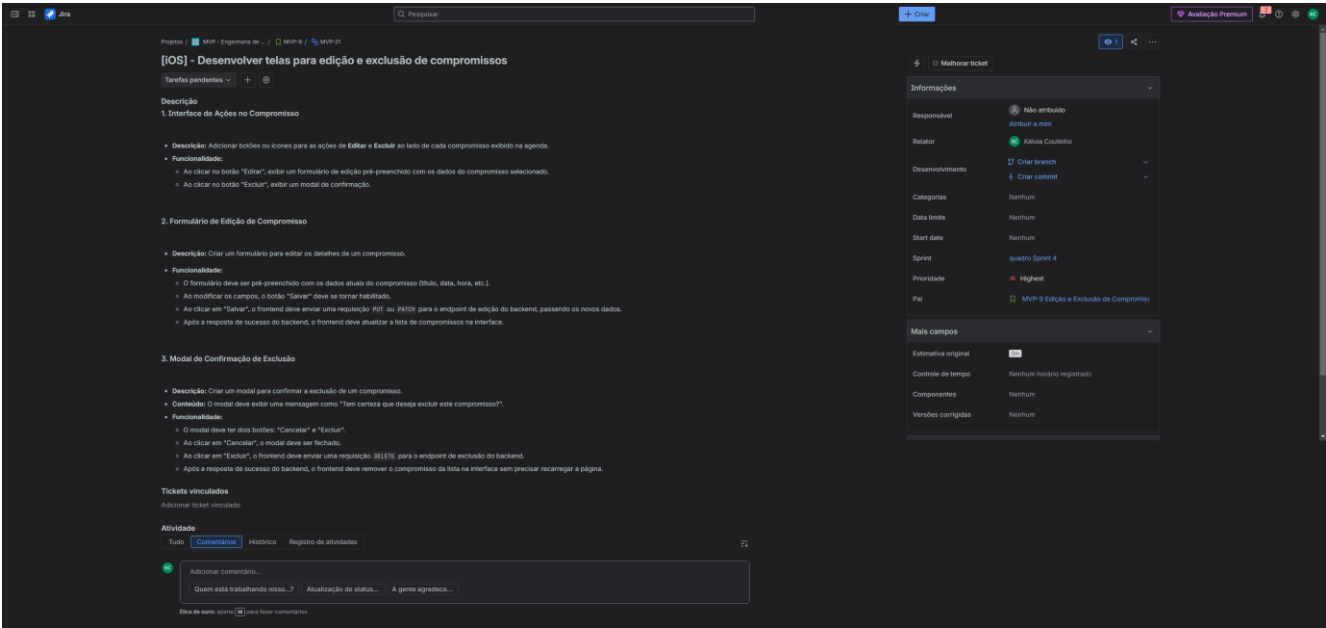
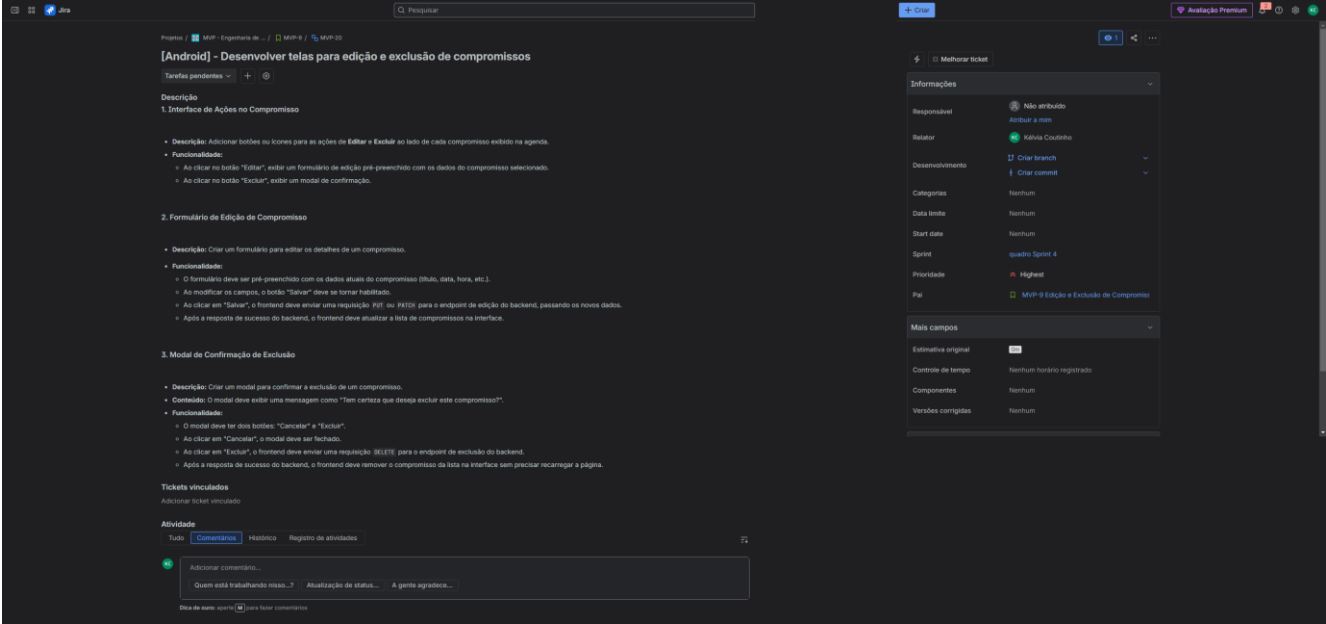


6. Edição e exclusão de compromisso



6.1 Subtarefas





7. Recuperação de senha

Projetos / MPP - Engenharia de ... / MPP-1 / MPP-4

Recuperação de Senha

Tarefas pendentes

Descrição

Descrição

Como um usuário que esqueceu a senha, quero poder recuperá-la de forma segura para ter acesso novamente à minha conta.

Critérios de Aceitação:

- Na tela de login, deve haver um link "Esqueceu a senha?".
- O usuário deve inserir o e-mail de cadastro para receber o link de redefinição.
- Um e-mail com um link único e temporário deve ser enviado para o endereço fornecido.
- O link deve direcionar o usuário para uma tela de criação de uma nova senha.
- A nova senha deve seguir as mesmas regras de segurança do cadastro.
- Após a redefinição, o usuário deve ser notificado por e-mail e direcionado para a tela de login.
- No topo da tela deve conter um botão de voltar para a tela anterior.

Requisitos Não Funcionais (RNFs)

- Requisitos de Usabilidade**
  - A interface deve ser intuitiva e fácil de usar, mesmo para usuários com pouca experiência em tecnologia.
  - O tempo de carregamento das telas não deve exceder 3 segundos.
  - Mensagens de erro devem ser claras e orientar o usuário.
  - O aplicativo deve ter navegação consistente.
- Requisitos de Performance**
  - Todas as operações de criação, leitura, atualização e exclusão (CRUDs) devem responder em menos de 1 segundo.
  - O aplicativo deve ser capaz de suportar 1.000 usuários simultâneos sem degradação de performance.
- Requisitos de Segurança**
  - Todas as senhas devem ser armazenadas de forma segura, utilizando técnicas de hashing e salting.
  - A comunicação entre o cliente e o servidor deve ser criptografada (HTTPS).
  - O sistema deve implementar a lógica de redefinição de senha com links temporários e únicos.
  - O aplicativo deve estar em conformidade com as leis de proteção de dados (como a LGPD no Brasil).

Definition of Ready (DoR)

- Clareza e Entendimento: A história de usuário foi escrita de forma clara, utilizando o formato "Como [persona], quero [funcionalidade] para [valor]". O objetivo e o valor de negócio são compreendidos por toda a equipe.
- Critérios de Aceitação Detalhados: Os critérios de aceitação foram definidos, estão claros, específicos e são testáveis.
- Design e UX Finalizados: O design no Figma (ou outra ferramenta) está aprovado e disponível para o time. Todos os estados da interface (erros, sucesso, carregamento) foram respondidos.
- Estimativa de Esforço (Story Points): O time de desenvolvimento discutiu a história e fez uma estimativa de esforço, atribuindo Story Points. A pontuação não excede o limite definido pelo time (se não é maior que 8).
- Estimativa de Esforço (Story Points): O time de desenvolvimento discutiu a história e fez uma estimativa de esforço, atribuindo Story Points. A pontuação não excede o limite definido pelo time (se não é maior que 8).
- Sem Ambiguidade: Qualquer dúvida sobre a história ou seus requisitos foi respondida e a equipe não tem questionamentos pendentes que impeçam o início do trabalho.

Definition of Done (DoD)

- O código foi escrito e revisado (code review realizado).
- A história foi testada pelo time de desenvolvimento (testes unitários e de integração passaram).
- A funcionalidade foi testada e aprovada.
- O produto foi testado em ambiente de homologação e está funcionando como esperado.
- A documentação técnica foi atualizada.
- O design no Figma foi implementado conforme o layout aprovado.
- A história atende a todos os critérios de aceitação.
- Não há bugs críticos ou de alta prioridade.

Subtarefas

3/5 concluído

| Tarefa   | Prioridade | Est... | Responsável   | Status          |
|--|------------|--------|---------------|-----------------|
| MPP-16 [Backend] - Desenvolver API para recuperação de senha | Medium     |        | Não atribuído | TAREFAS PENDING |
| MPP-17 [Android] - Desenvolver telas de recuperação de senha | Medium     |        | Não atribuído | TAREFAS PENDING |
| MPP-18 [iOS] - Desenvolver telas de recuperação de senha     | Medium     |        | Não atribuído | TAREFAS PENDING |

Adicionar subtarefa

Tickets vinculados

Adicionar ticket vinculado

Conteúdo do Confluence

Requisitos do produto

Atividade

Tudo

Comentários

História

Registro de atividades

Adicionar comentário...

Quem está trabalhando nisso...? Atualização do status... A gente agradece...

Diálogo de ação: para fazer comentários

Malheur Ticket

Informações

Responsável: Não atribuído

Atribuir a mim

Relator: Kátia Coutinho

Desenvolvimento: Citar Branch

Citar commit

Categorias: Nenhum

Data limite: Nenhum

Start date: Nenhum

Story Points: 1

Speed: quadro Sprint 4

Prioridade: Medium

Pa: MPP-16 Backend do Aplicativo

Mais campos

Estimativa original: 1

Controle de tempo: Nenhum horário registrado

Incluir subtarefas

Comentários: Nenhum

7.1 Subtarefas

Projetos

MVP - Engenharia de ...

MVP-4 / MVP-6

[Backend] - Desenvolver API para recuperação de senha

Tarefas pendentes

Descrição

1. Endpoint de Solicitação de Redefinição de Senha

- Descrição: Criar um endpoint API que recebe o e-mail de um usuário e inicia o processo de recuperação de senha.
- Método HTTP: POST.
- URL: `https://api.11factory.com/auth/requests-senha`.
- Corpo da Requisição (Request Body): Deve receber um JSON contendo o campo `email`.
- Lógica de Negócio
  - Verificar se o e-mail fornecido existe no banco de dados.
  - Gerar um token de redefinição único e seguro: Este token deve ser associado ao usuário e ter um tempo de expiração limitado (ex: 30 minutos) para segurança.
  - Enviar o e-mail: Compor um e-mail com uma mensagem clara e um link de redefinição de senha, incluindo o token gerado como parte da URL (ex: `https://11factory.com/reset/senha?token=...`).
  - Retornar uma resposta de sucesso genérica para o frontend, independentemente de o e-mail ter sido encontrado ou não. Isso previne que usuários mal-intencionados descubram quais e-mails estão cadastrados no sistema.

2. Endpoint de Redefinição de Senha

- Descrição: Criar um endpoint que recebe o token e a nova senha para concluir a redefinição.
- Método HTTP: POST.
- URL: `https://api.11factory.com/auth/reset/senha`.
- Corpo da Requisição (Request Body): Deverá receber um JSON com o token e a nova senha.
- Lógica de Negócio
  - Validar a nova senha, garantindo que ela atenda aos critérios de segurança (mínimo de 8 caracteres, com letras, números e um caractere especial).
  - Buscar o usuário no banco de dados usando o token de redefinição.
  - Verificar se o token é válido e se não está expirado. Se o token for inválido, retornar um erro.
  - Criptografar a nova senha e atualizar o registro do usuário no banco de dados.
  - Inativar o token de redefinição para que ele não possa ser usado novamente.
  - Retornar uma resposta de sucesso.

Tickets vinculados

Adicionar ticket vinculado

Atividade

Tudo

Comentários

Histórico

Registro de atividades

Adicionar comentário...

Quem está trabalhando nisso...?

Atualização de status...

A gente agradece...

Olá de novo, aqui para fazer comentários

+ Criar

1

Avançado Premium

Melhorar ticket

Informações

Responsável

Não atribuído

Atribuir a mim

Relator

Kahira Coutinho

Desenvolvimento

Criar branch

Criar commit

Categorias

Frontend

Data limite

Frontend

Start date

Frontend

Sprint

quadro Sprint 4

Prioridade

Medium

Pal

MVP-4 Recuperação de Senha

Mais campos

Estimativa original

1

Controle de tempo

Frontend horário registrado

Componentes

Frontend

Verdes corrigidas

Frontend

Projetos

MVP - Engenharia de ...

MVP-4 / MVP-7

[Android] - Desenvolver telas de recuperação de senha

Tarefas pendentes

Descrição

1. Fluxo de Solicitação de Redefinição

- Descrição: Criar a interface e a lógica para o usuário solicitar a redefinição de senha.
- Funcionalidade:
  - Link na tela de login: Adicionar um link "Esqueceu a senha?" na tela de login.
  - Tela de solicitação: Ao clicar no link, navegar para uma nova tela com um campo para o e-mail e um botão "Enviar".
  - Tela de validação: Incluir um botão no topo da tela no topo da tela para retornar à tela de login.
  - Requisição: Ao clicar em "Enviar", fazer uma requisição POST para o endpoint `/auth/requests-senha` com o e-mail fornecido.
  - Feedback: Exibir uma mensagem de sucesso, como "Se o e-mail estiver cadastrado, você receberá um link para redefinir a senha."

2. Fluxo de Redefinição de Senha

- Descrição: Criar a interface e a lógica para o usuário inserir a nova senha.
- Funcionalidade:
  - Recuperação do link: O aplicativo deve ser capaz de processar o link de redefinição de senha recebido por e-mail, capturando o token da URL.
  - Tela de redefinição: Navegar para uma tela com campos para "Nova Senha" e "Confirmar Nova Senha".
  - Validação: Implementar a validação da nova senha em tempo real, garantindo que ela atenda aos critérios de segurança e que os dois campos de senha correspondam.
  - Tela de sucesso: Adicionar um botão de voltar para retornar à tela de login.
  - Requisição: Fazer uma requisição POST para o endpoint `/auth/reset/senha`, enviando a nova senha e o token capturado da URL.
  - Feedback e redirecionamento: Em caso de sucesso, exibir uma mensagem como "Sua senha foi redefinida com sucesso!" e redirecionar o usuário para a tela de login. Em caso de erro, exibir uma mensagem clara (ex: "Token inválido ou expirado").

Tickets vinculados

Adicionar ticket vinculado

Atividade

Tudo

Comentários

Histórico

Registro de atividades

Adicionar comentário...

Quem está trabalhando nisso...?

Atualização de status...

A gente agradece...

Olá de novo, aqui para fazer comentários

+ Criar

1

Avançado Premium

Melhorar ticket

Informações

Responsável

Não atribuído

Atribuir a mim

Relator

Kahira Coutinho

Desenvolvimento

Criar branch

Criar commit

Categorias

Frontend

Data limite

Frontend

Start date

Frontend

Sprint

quadro Sprint 4

Prioridade

Medium

Pal

MVP-4 Recuperação de Senha

Mais campos

Estimativa original

1

Controle de tempo

Frontend horário registrado

Componentes

Frontend

Verdes corrigidas

Frontend

Projetos

MVP - Engenharia de ...

MVP-4 / MVP-8

[iOS] - Desenvolver telas de recuperação de senha

Tarefas pendentes

Descrição

1. Fluxo de Solicitação de Redefinição

- Descrição: Criar a interface e a lógica para o usuário solicitar a redefinição de senha.
- Funcionalidade:
  - Link na tela de login: Adicionar um link "Esqueceu a senha?" na tela de login.
  - Tela de solicitação: Ao clicar no link, navegar para uma nova tela com um campo para o e-mail e um botão "Enviar".
  - Tela de validação: Incluir um botão no topo da tela no topo da tela para retornar à tela de login.
  - Requisição: Ao clicar em "Enviar", fazer uma requisição POST para o endpoint `/auth/requests-senha` com o e-mail fornecido.
  - Feedback: Exibir uma mensagem de sucesso, como "Se o e-mail estiver cadastrado, você receberá um link para redefinir a senha."

2. Fluxo de Redefinição de Senha

- Descrição: Criar a interface e a lógica para o usuário inserir a nova senha.
- Funcionalidade:
  - Recuperação do link: O aplicativo deve ser capaz de processar o link de redefinição de senha recebido por e-mail, capturando o token da URL.
  - Tela de redefinição: Navegar para uma tela com campos para "Nova Senha" e "Confirmar Nova Senha".
  - Validação: Implementar a validação da nova senha em tempo real, garantindo que ela atenda aos critérios de segurança e que os dois campos de senha correspondam.
  - Tela de sucesso: Adicionar um botão de voltar para retornar à tela de login.
  - Requisição: Fazer uma requisição POST para o endpoint `/auth/reset/senha`, enviando a nova senha e o token capturado da URL.
  - Feedback e redirecionamento: Em caso de sucesso, exibir uma mensagem como "Sua senha foi redefinida com sucesso!" e redirecionar o usuário para a tela de login. Em caso de erro, exibir uma mensagem clara (ex: "Token inválido ou expirado").

Tickets vinculados

Adicionar ticket vinculado

Atividade

Tudo

Comentários

Histórico

Registro de atividades

Adicionar comentário...

Quem está trabalhando nisso...?

Atualização de status...

A gente agradece...

Olá de novo, aqui para fazer comentários

+ Criar

1

Avançado Premium

Melhorar ticket

Informações

Responsável

Não atribuído

Atribuir a mim

Relator

Kahira Coutinho

Desenvolvimento

Criar branch

Criar commit

Categorias

Frontend

Data limite

Frontend

Start date

Frontend

Sprint

quadro Sprint 4

Prioridade

Medium

Pal

MVP-4 Recuperação de Senha

Mais campos

Estimativa original

1

Controle de tempo

Frontend horário registrado

Componentes

Frontend

Verdes corrigidas

Frontend