

Histórias do usuário:

- Eu como cidadão, quero criar uma conta no aplicativo informando meus dados básicos para poder registrar ocorrências.
 - Eu como cidadão cadastrado, quero acessar minha conta para utilizar o aplicativo
 - Eu como cidadão cadastrado, quero descrever minha ocorrência para o COPOM
 - Eu como cidadão cadastrado, quero receber uma confirmação de que minha ocorrência foi encaminhada para saber que o envio foi realizado com sucesso.
 - Eu como usuário cadastrado, quero poder realizar a avaliação do aplicativo para demonstrar minha opinião
-

Analisar cada história segundo os critérios INVEST. (SM)

- Eu como cidadão, quero criar uma conta no aplicativo informando meus dados básicos para poder registrar ocorrências.
 - I – Independente: A funcionalidade é majoritariamente independente, embora seja pré-requisito para uso de outras funcionalidades.
 - N – Negociável: Os dados solicitados podem ser ajustados conforme requisitos do sistema.
 - V – Valiosa: Permite identificação do usuário e garante rastreabilidade das ocorrências.
 - E – Estimável: O esforço é previsível, pois trata-se de funcionalidade comum.
 - S – Pequena: Pode ser desenvolvida dentro de uma Sprint.
 - T – Testável: É possível validar cadastro válido, inválido e armazenamento correto dos dados.
- Eu como cidadão cadastrado, quero acessar minha conta para utilizar o aplicativo
 - I – Independente: Pode ser implementada separadamente das demais funcionalidades.
 - N – Negociável: A forma de autenticação pode ser discutida e ajustada.
 - V – Valiosa: Garante segurança e controle de acesso ao sistema.
 - E – Estimável: Complexidade conhecida e previsível.
 - S – Pequena: Adequada para desenvolvimento em uma Sprint.
 - T – Testável: Pode-se validar login correto, incorreto e bloqueio de acesso.
- Eu como cidadão cadastrado, quero descrever minha ocorrência para o COPOM

- I – Independente: Não depende de outras funcionalidades complexas.
N – Negociável: Pode-se definir limite de caracteres e obrigatoriedade do campo.
V – Valiosa: Essencial para fornecer detalhes à central.
E – Estimável: Funcionalidade simples de implementar.
S – Pequena: Adequada para uma Sprint.
T – Testável: Pode-se validar envio com e sem preenchimento.
- Eu como cidadão cadastrado, quero receber uma confirmação de que minha ocorrência foi encaminhada para saber que o envio foi realizado com sucesso.
 - I – Independente: É parcialmente dependente da funcionalidade de envio da ocorrência, mas pode ser tratada como história separada.
N – Negociável: O formato da confirmação pode ser definido pelo time.
V – Valiosa: Gera segurança e confiança ao usuário.
E – Estimável: Complexidade moderada e previsível.
S – Pequena: Pode ser entregue em uma Sprint.
T – Testável: Pode-se validar se a mensagem de confirmação é exibida corretamente após o envio.
- Eu como usuário cadastrado, quero poder realizar a avaliação do aplicativo para demonstrar minha opinião
 - I – Independente: Não depende das funcionalidades principais do sistema.
N – Negociável: Pode-se definir formato da avaliação (nota, comentário, ambos).
V – Valiosa: Contribui para melhoria contínua do aplicativo.
E – Estimável: Implementação simples e previsível.
S – Pequena: Pode ser desenvolvida em uma Sprint.
T – Testável: Pode-se validar registro correto da avaliação no sistema.

PRIORIDADE M – MUST HAVE (Deve ter)

1. Criar conta
Justificativa: O sistema exige identificação do usuário para rastreabilidade e segurança. Sem cadastro, não há controle nem responsabilização.
2. Acessar conta
Justificativa: A autenticação é essencial para segurança e uso do sistema. Sem login, não há controle de acesso.

3. Descrever ocorrência

Justificativa: É o núcleo da funcionalidade. Sem descrição, não há informação útil para a central.

PRIORIDADE S – SHOULD HAVE (Deveria ter)

6. Receber confirmação de envio

Justificativa: Muito importante para confiança e transparência ao usuário.

Porém, o sistema ainda pode funcionar sem essa confirmação inicial, embora com menor qualidade de experiência.

PRIORIDADE C – COULD HAVE (Poderia ter)

7. Avaliar aplicativo

Justificativa: Agrega valor para melhoria contínua e feedback dos usuários, mas não impacta o funcionamento principal do sistema. Pode ser implementada em versão futura.

HISTÓRIA 1

Eu como usuário, quero criar uma conta no aplicativo informando meus dados básicos para poder registrar ocorrências.

Critérios Funcionais

Cenário: Cadastro realizado com sucesso

Dado que o usuário está na tela de cadastro

Quando ele preencher todos os campos obrigatórios corretamente

Então o sistema deve criar a conta e exibir mensagem de sucesso

Cenário: Cadastro com campos obrigatórios vazios

Dado que o usuário está na tela de cadastro

Quando ele tentar enviar o formulário com campos obrigatórios vazios

Então o sistema deve impedir o cadastro e exibir mensagem de erro

Critério Não Funcional

Cenário: Tempo de resposta do cadastro

Dado que o usuário enviou o formulário corretamente

Quando o sistema processar o cadastro

Então a resposta deve ocorrer em até 3 segundos

HISTÓRIA 2

Eu como usuário cadastrado, quero acessar minha conta para utilizar o aplicativo.

Critérios Funcionais

Cenário: Login com dados válidos

Dado que o usuário possui cadastro ativo

Quando ele informar login e senha corretos

Então o sistema deve permitir acesso à conta

Cenário: Login com dados inválidos

Dado que o usuário está na tela de login

Quando ele informar credenciais incorretas

Então o sistema deve negar o acesso e exibir mensagem de erro

Critério Não Funcional

Cenário: Segurança da autenticação

Dado que o usuário realiza login

Quando os dados forem transmitidos ao servidor

Então a comunicação deve ocorrer via protocolo HTTPS com criptografia TLS

HISTÓRIA 3

Eu como cidadão cadastrado, quero descrever minha ocorrência para o COPOM.

Critério Funcional

Cenário: Descrição registrada com sucesso

Dado que o cidadão está autenticado e na tela de registro de ocorrência

Quando ele preencher o campo de descrição e confirmar o registro

Então o sistema deve salvar a descrição vinculada à ocorrência

Cenário: Bloqueio de registro sem descrição

Dado que o cidadão está autenticado e na tela de registro de ocorrência

Quando ele tentar confirmar o registro sem preencher o campo de descrição

Então o sistema deve impedir o envio e exibir uma mensagem solicitando o preenchimento

Critério Não Funcional

Cenário: Persistência da descrição da ocorrência

Dado que o cidadão confirmou o registro da ocorrência com descrição preenchida

Quando o sistema armazenar os dados

Então a descrição deve permanecer íntegra e disponível para consulta posterior

HISTÓRIA 4

Eu como cidadão cadastrado, quero receber uma confirmação de que minha ocorrência foi encaminhada para saber que o envio foi realizado com sucesso.

Critério Funcional

Cenário: Confirmação exibida após encaminhamento bem-sucedido

Dado que o cidadão registrou uma ocorrência e solicitou o envio

Quando o sistema concluir o encaminhamento para o COPOM com sucesso

Então o sistema deve exibir uma confirmação ao cidadão informando que a ocorrência foi encaminhada

Cenário: Mensagem de falha e possibilidade de nova tentativa

Dado que o cidadão registrou uma ocorrência e solicitou o envio

Quando o sistema não conseguir encaminhar a ocorrência ao COPOM

Então o sistema deve informar a falha e oferecer a opção de tentar novamente

Critério Não Funcional

Cenário: Tempo de retorno da confirmação

Dado que o encaminhamento da ocorrência foi concluído com sucesso

Quando o sistema processar a resposta do encaminhamento

Então a confirmação ao cidadão deve ser exibida em até 5 segundos

HISTÓRIA 5

Eu como usuário cadastrado, quero poder realizar a avaliação do aplicativo para demonstrar minha opinião.

Critério Funcional

Cenário: Avaliação registrada com sucesso

Dado que o usuário está autenticado e acessou a área de avaliação

Quando ele selecionar uma nota e enviar sua avaliação

Então o sistema deve registrar a avaliação vinculada ao usuário

Cenário: Bloqueio de envio sem nota

Dado que o usuário está autenticado e acessou a área de avaliação

Quando ele tentar enviar uma avaliação sem selecionar uma nota

Então o sistema deve impedir o envio e exibir uma mensagem solicitando a seleção da nota

Critério Não Funcional

Cenário: Tempo de resposta ao enviar avaliação

Dado que o usuário enviou uma avaliação válida

Quando o sistema processar o envio

Então o sistema deve registrar a avaliação e retornar a resposta em até 3 segundos

RNF-01 — Desempenho no Cadastro

A resposta ao envio do cadastro deve ocorrer em até 3 segundos.

História Impactada: H1

RNF-02 — Segurança da Autenticação

- Comunicação via HTTPS com TLS 1.2 ou superior.
- Senhas armazenadas com hash criptográfico seguro.

Histórias Impactadas: H1, H2

RNF-03 — Integridade e Persistência dos Dados

- 100% das ocorrências confirmadas devem permanecer armazenadas.
- Dados não podem ser alterados sem registro interno.

História Impactada: H3

RNF-04 — Tempo de Retorno da Confirmação

A confirmação deve ser exibida em até 5 segundos após envio bem-sucedido.

História Impactada: H4

RNF-05 — Desempenho no Envio de Avaliação

O registro da avaliação deve ocorrer em até 3 segundos.

História Impactada: H5

RNF-06 — Disponibilidade

Disponibilidade mínima de 99% ao mês.

Histórias Impactadas: H1, H2, H3, H4, H5

RNF-07 — Usabilidade

- O usuário deve ser capaz de concluir o registro de uma ocorrência em até **2 minutos**, considerando conexão estável e preenchimento adequado dos dados obrigatórios.

- Máximo de 3 telas no fluxo de envio.

Histórias Impactadas: H1, H3, H5

RNF-08 — Compatibilidade

- Compatível com diversos dispositivos.
- Interface responsiva para telas iguais ou superiores a 5 polegadas.

Histórias Impactadas: Todas

RNF-09 — Conformidade com LGPD

- Consentimento explícito no cadastro.
- Opção de exclusão de conta.
- Dados pessoais acessíveis apenas ao usuário autenticado.

Histórias Impactadas: H1, H2, H3, H5

RNF-10 — Tecnologia de Desenvolvimento

- O sistema deve ser desenvolvido utilizando JavaScript.

Tipo: Restrição Tecnológica

RNF-11 — Banco de Dados Relacional

- O sistema deve utilizar banco de dados relacional.
- Estrutura baseada em tabelas com chaves primárias e estrangeiras.
- Garantia de integridade referencial.

Tipo: Restrição Arquitetural

RNF-12 — Banco de Dados

- O sistema deve utilizar Supabase.
- Autenticação gerenciada pelo Supabase.

Tipo: Restrição de Infraestrutura

RNF-13 — Arquitetura do Sistema

- Arquitetura em camadas (Frontend, Backend e Banco de Dados).
- Comunicação via API REST.
- Separação clara entre lógica de apresentação e lógica de negócio.

Tipo: Restrição Arquitetural

