PARTICIPAÇÃO POPULAR

Especificação funcional e técnica consolidada

Documento de referência para uma plataforma SaaS multientidade de participação cidadã, com cadastro simplificado, votação por proposta, segurança em camadas, antifraude, privacidade, auditoria e acessibilidade.

ESCOPO	Plataforma SaaS multientidade
ACESSO	Web • celular • tablet
IDENTIFICAÇÃO	CPF + senha ou segurança do dispositivo
MODELO	Prevenção de duplicidade + gerenciamento de risco
DEPENDÊNCIAS EXTERNAS	Sem GOV.BR, Serpro ou validação cadastral paga

Documento consolidado • Setembro de 2026

Visão geral

A especificação define a jornada do cidadão, as garantias e limitações do modelo, a arquitetura de autenticação e antifraude, as regras de propostas e votos, a estrutura mínima de dados, os controles de segurança, o painel administrativo e os critérios finais de aceite.

01	Cadastro simples CPF, nome e senha; e-mail e telefone opcionais.
02	Participação ampla Várias propostas e voto em várias propostas diferentes.
03	Duplicidade controlada Uma conta por CPF e um voto por cidadão em cada proposta.
04	Antifraude em camadas Desafios, risco, rate limit, dispositivo, auditoria e revisão.
05	Privacidade CPF protegido, logs restritos e isolamento entre entidades.
06	Aceite verificável 60 testes obrigatórios e critérios objetivos de conclusão.

1. Escopo definitivo

A solução será uma plataforma SaaS multientidade, acessível pelo site da Prefeitura, navegador de computador, celular ou tablet.

O fluxo deverá ocorrer integralmente dentro da plataforma, sem:

- API do Serpro;
- autenticação obrigatória pelo GOV.BR;
- comparecimento presencial;
- aprovação manual prévia;
- e-mail obrigatório;
- telefone obrigatório;
- validação externa paga;
- redirecionamento para outro sistema.

O cidadão poderá:

- cadastrar-se imediatamente;
- acessar com CPF e senha ou com a segurança do celular;
- criar várias propostas;
- votar em várias propostas diferentes;
- votar somente uma vez em cada proposta;
- acompanhar propostas e votos;
- recuperar o acesso sem depender obrigatoriamente de e-mail.

---

2. Limite técnico e declaração de segurança

2.1. O que o sistema garantirá

O sistema deverá garantir tecnicamente que:

- CPFs com cálculo incorreto sejam recusados;
- sequências falsas evidentes sejam recusadas;
- um CPF não possua duas contas ativas;
- o mesmo CPF não vote duas vezes na mesma proposta;
- votos em propostas diferentes sejam permitidos;
- um cidadão possa cadastrar várias propostas;
- cliques duplicados não gerem votos repetidos;
- requisições simultâneas não gerem votos repetidos;
- robôs simples e ataques automatizados sejam bloqueados;
- comportamentos suspeitos sejam colocados em verificação;
- todas as operações relevantes sejam auditadas.

2.2. O que não será possível garantir

Sem consultar uma base cadastral confiável, o sistema não poderá comprovar que:

- o CPF foi efetivamente atribuído pela Receita Federal;
- o nome informado pertence ao CPF;

- a pessoa diante do aparelho é titular do CPF;
- uma pessoa não está utilizando CPFs válidos de terceiros;
- dois CPFs diferentes não estão sendo operados pela mesma pessoa.

Sites geradores conseguem produzir CPFs com dígitos verificadores corretos. Localmente, um CPF gerado e um CPF oficialmente atribuído possuem a mesma estrutura matemática.

Portanto, o sistema não deverá apresentar mensagens como:

> CPF confirmado na Receita Federal.
> Identidade oficialmente verificada.
> Titularidade do CPF confirmada.

Deverá utilizar:

> CPF com formato validado.

O modelo será de prevenção de duplicidade e gerenciamento de risco, não de certificação oficial da identidade civil.

---

3. Princípios da solução

1. Cadastro simples e imediato.
2. Linguagem acessível.
3. Nenhuma exigência de e-mail.
4. Poucos campos obrigatórios.
5. Proteção anti-robô silenciosa.
6. Um cadastro por CPF.
7. Várias propostas por cidadão.
8. Um voto por cidadão em cada proposta.
9. Autenticação protegida por senha ou passkey.
10. Avaliação contínua de comportamento suspeito.
11. Revisão apenas de operações de alto risco.
12. Nenhuma exclusão automática definitiva.
13. Proteção integral dos dados pessoais.
14. Isolamento entre as entidades municipais.
15. Auditoria de todas as decisões antifraude.

---

4. Dados do cadastro cidadão

4.1. Campos obrigatórios

- CPF;
- nome completo;
- senha;
- confirmação da senha;
- aceite do aviso de privacidade;
- declaração de veracidade.

4.2. Campos opcionais

- telefone;
- e-mail;
- forma preferencial de notificação.

4.3. Data de nascimento

A data de nascimento não deverá ser solicitada, porque não haverá base externa para verificar sua correspondência com o CPF.

Ela somente poderá ser incluída se houver regra de idade formalmente configurada no processo. Mesmo nesse caso, será autodeclarada e não poderá ser apresentada como prova de identidade.

4.4. Declaração de veracidade

Texto recomendado:

> Declaro que as informações fornecidas são verdadeiras e que utilizarei somente meus próprios dados para participar. Estou ciente de que tentativas de manipulação, uso indevido de dados de terceiros ou votação automatizada poderão ser identificadas, auditadas e invalidadas.

Essa declaração não substitui os controles técnicos.

---

5. Jornada completa do cidadão

```
flowchart TD
    A["Enviar proposta ou votar"] --> B["Informar CPF"]
    B --> C["Validação local e anti-robô"]
    C --> D{"CPF já cadastrado?"}
    D -->|Não| E["Criar acesso"]
    D -->|Sim| F["Entrar"]
    E --> G["Retomar automaticamente"]
    F --> G
    G --> H["Criar várias propostas"]
    G --> I["Votar uma vez em cada proposta"]
```

5.1. Página inicial

Apresentar três ações principais:

- Enviar uma proposta
- Votar nas propostas
- Acompanhar minha participação

Texto:

> Para enviar propostas ou votar, faça um cadastro simples com CPF. Não é necessário possuir e-mail.

5.2. Identificação inicial

Ao escolher uma ação:

> Informe seu CPF para continuar
> O cadastro é gratuito, realizado neste sistema e não exige e-mail.

Campo:

- CPF.

Botão:

> Continuar

Recursos:

- teclado numérico no celular;
- máscara automática;
- possibilidade de colar o CPF;
- remoção automática de pontos e traços;
- orientação em linguagem simples;
- área de toque ampla;
- leitor de tela;
- foco visível.

5.3. Primeiro acesso

Se ainda não existir cadastro:

> Crie seu acesso
> Leva apenas alguns minutos. Você não precisa ter e-mail.

Campos:

- CPF já preenchido;
- nome completo;
- criar senha;
- repetir senha;
- aceite do aviso de privacidade;
- declaração de veracidade.

Botão:

> Criar meu acesso e continuar

A conta será criada imediatamente após as validações locais e antifraude.

5.4. Cidadão já cadastrado

Campos:

- CPF;
- senha.

Botão:

> Entrar e continuar

Links:

- Esqueci minha senha
- Entrar usando a segurança deste celular
- Preciso de ajuda

5.5. Retorno automático

Após cadastro ou login, o sistema deverá retornar exatamente para:

- a proposta que o cidadão pretendia votar;
- o formulário que estava preenchendo;
- a seção escolhida;
- os filtros utilizados;
- o último passo da jornada.

O usuário não deverá ser enviado genericamente para a página inicial.

---

6. Validação local do CPF

A validação deverá existir no frontend para orientação imediata e obrigatoriamente no backend para segurança.

6.1. Regras

1. Remover pontos, espaços e traços.
2. Exigir exatamente 11 dígitos.
3. Rejeitar caracteres não numéricos.
4. Rejeitar sequências repetidas:
   - `00000000000`;
   - `11111111111`;
   - demais repetições.
5. Calcular o primeiro dígito verificador.
6. Calcular o segundo dígito verificador.
7. Comparar os dígitos calculados.
8. Consultar se já existe conta para aquele CPF.
9. Verificar o desafio anti-robô.
10. Aplicar controle de tentativas.
11. Calcular o risco da operação.

6.2. Mensagens

CPF incorreto:

> Confira o CPF informado
> Digite os 11 números do CPF.

CPF já cadastrado:

> Este CPF já possui acesso
> Digite sua senha ou use outra forma de entrar.

Falha temporária:

> Não foi possível concluir agora
> Seus dados foram preservados. Tente novamente.

---

7. Proteção gratuita contra robôs

7.1. Prova de trabalho auto-hospedada

Utilizar uma solução gratuita e auto-hospedada, como o núcleo aberto do ALTCHA, ou equivalente auditável.

Ela deverá proteger:

- criação de conta;
- recuperação de acesso;
- login após repetidos erros;
- envio anormal de propostas;
- votação em comportamento suspeito;
- chamadas diretas à API.

Não utilizar desafios com imagens, letras distorcidas ou perguntas difíceis como regra geral.

7.2. Validação obrigatória no backend

O backend deverá validar:

- assinatura do desafio;
- prazo de validade;
- uso único;
- sessão;
- operação autorizada;
- origem esperada;
- proteção contra repetição.

Um desafio utilizado no cadastro não poderá ser utilizado para votar.

7.3. Dificuldade dinâmica

O desafio deverá ser simples para usuários normais e mais rigoroso para operações suspeitas.

Risco	Comportamento
Baixo	Desafio silencioso
Médio	Nova verificação automática
Alto	Desafio adicional e espera progressiva
Crítico	Bloqueio temporário da operação

A dificuldade deverá ser testada em celulares simples e conexões lentas.

7.4. Campo-armadilha

Adicionar campos invisíveis destinados à detecção de robôs.

O preenchimento desses campos deverá:

- elevar o risco;
- bloquear ou retardar a requisição;

- registrar o evento;
- não revelar qual proteção foi acionada.

7.5. Formulário assinado

Cada formulário de cadastro, proposta e voto deverá possuir:

- identificador aleatório;
- assinatura do servidor;
- prazo de validade;
- vínculo com a sessão;
- uso único;
- proteção CSRF.

---

8. Identificador anônimo do dispositivo

A plataforma deverá gerar um identificador aleatório, assinado e não invasivo para o navegador.

Ele ajudará a identificar:

- muitos CPFs no mesmo aparelho;
- criação acelerada de contas;
- troca constante de usuários;
- repetição do mesmo padrão de votos;
- uso automatizado;
- reutilização de sessão.

Limitações

O dispositivo não poderá ser utilizado como prova de identidade, pois:

- famílias podem compartilhar aparelho;
- pontos públicos podem atender várias pessoas;
- cookies podem ser apagados;
- o usuário pode trocar de navegador;
- aparelhos podem ser substituídos.

O dispositivo será um sinal de risco, não uma regra definitiva de bloqueio.

Proibições

Não coletar desnecessariamente:

- lista de aplicativos;
- contatos;
- arquivos;
- fontes instaladas;
- sensores;
- identificadores permanentes;
- biometria;
- dados de outras aplicações.

---

9. Passkey e autenticação

9.1. Passkey recomendada

Após o cadastro, oferecer:

> Entrar usando a segurança deste celular

Explicação:

> Na próxima vez, você poderá entrar da mesma forma que desbloqueia seu celular: impressão digital, rosto, PIN ou padrão.

A biometria permanece no aparelho. O SaaS armazena apenas a chave pública e os dados técnicos da credencial.

A passkey:

- protege a conta;
- reduz roubo de senha;
- dificulta acesso automatizado;
- simplifica o login;
- evita SMS;
- não comprova que o CPF pertence ao usuário.

9.2. Senha alternativa

Manter CPF e senha para:

- aparelhos incompatíveis;
- usuários que não desejem passkey;
- acesso em computadores;
- contingência.

Requisitos:

- mínimo de 8 caracteres;
- permitir frases;
- botão mostrar/ocultar;
- armazenamento com Argon2id;
- proteção contra tentativas;
- não exigir combinações excessivamente complexas.

9.3. Sessões

- cookies `HttpOnly`;
- cookies `Secure`;
- `SameSite` apropriado;
- rotação de sessão;
- expiração por inatividade;
- encerramento remoto;
- limite de sessões simultâneas;

- reautenticação em operação sensível;
- revogação após troca de senha.

A opção “Manter conectado neste celular” deverá ser apresentada apenas em dispositivo pessoal.

---

10. Recuperação sem e-mail obrigatório

10.1. Códigos de recuperação

Gerar um conjunto de códigos de uso único.

Exemplo:

```
PK7M-42TX
R9BV-73KL
FG2N-18QP
```

Permitir:

- copiar;
- baixar;
- imprimir;
- fotografar.

Mensagem:

> Guarde seus códigos de recuperação
> Eles serão necessários caso você esqueça a senha ou troque de celular.

10.2. Formas de recuperação

- passkey já cadastrada;
- código de recuperação;
- telefone opcional;
- e-mail opcional;
- sessão ainda ativa em outro aparelho.

10.3. Limitação inevitável

Se o cidadão perder simultaneamente:

- senha;
- passkey;
- códigos de recuperação;
- telefone opcional;
- e-mail opcional;
- todas as sessões ativas;

não haverá meio seguro de recuperar a conta apenas com CPF, porque a plataforma não consulta uma base externa e não realiza atendimento presencial.

Essa informação deverá ser apresentada claramente no cadastro.

O sistema deverá incentivar:

- criação de passkey;
- armazenamento dos códigos;
- inclusão opcional de telefone ou e-mail;
- cadastro de uma segunda passkey em outro dispositivo.

---

11. Múltiplas propostas

11.1. Regra

Um cidadão poderá criar várias propostas dentro do mesmo processo.

Não criar restrição:

```
UNIQUE (processo_id, cidadao_id)
```

Cada proposta deverá possuir:

- UUID;
- entidade;
- processo;
- cidadão autor;
- protocolo;
- título;
- descrição;
- tema;
- território;
- anexos;
- situação;
- risco;
- data e hora;
- histórico.

11.2. Jornada

Etapas:

1. Título da ideia.
2. Descrição.
3. Tema.
4. Localidade.
5. Anexos opcionais.
6. Revisão.
7. Envio.
8. Protocolo.

Após o envio:

> Proposta enviada com sucesso
> Você pode acompanhar o protocolo ou enviar outra proposta.

Botões:

- Enviar outra proposta
- Ver minhas propostas

11.3. Rascunho

- salvamento automático;
- preservação durante login;
- recuperação após falha;
- indicação “Salvo”;
- limpeza de dados em dispositivo compartilhado;
- armazenamento no servidor após autenticação.

11.4. Limite de velocidade

Permitir múltiplas propostas sem permitir automação ilimitada.

Parâmetros iniciais sugeridos:

- até 5 propostas por hora por conta;
- até 20 propostas por dia por conta;
- sem limite total permanente;
- limites configuráveis pela entidade;
- excesso colocado em verificação.

O limite será de velocidade, não de participação total.

---

12. Um voto por CPF em cada proposta

12.1. Regra

O cidadão poderá votar em quantas propostas diferentes desejar, mas somente uma vez em cada uma.

A proteção deverá existir no PostgreSQL:

```
UNIQUE (
    entidade_id,
    processo_id,
    proposta_id,
    cidadao_id
)
```

12.2. Inserção transacional

```
INSERT INTO votos (
    id,
    entidade_id,
    processo_id,
    proposta_id,
    cidadao_id,
    situacao,
    criado_em
)
VALUES (
    $1,
    $2,
    $3,
    $4,
    $5,
```

```
    'CONFIRMADO',
    NOW()
)
ON CONFLICT (
    entidade_id,
    processo_id,
    proposta_id,
    cidadao_id
)
DO NOTHING
RETURNING id;
```

Se não houver retorno, o voto já existe.

12.3. Garantias

A restrição impedirá duplicidade em caso de:

- clique duplo;
- atualização da página;
- internet instável;
- repetição automática;
- outro navegador;
- outro celular;
- requisições simultâneas;
- chamada direta à API;
- despublicação e republicação;
- mudança de título;
- alteração de categoria.

12.4. Interface

Antes do voto:

> Votar nesta proposta

Confirmação:

> Confirmar seu voto?
> Você está votando em: [título].

Depois:

> Voto registrado com sucesso
> Você pode continuar votando em outras propostas.

Se já houver voto:

> Você já votou nesta proposta
> Seu voto continua registrado. Escolha outra proposta para continuar participando.

---

13. Alteração ou retirada de voto

A entidade deverá configurar uma das políticas.

Política recomendada

O cidadão poderá retirar ou alterar seu voto enquanto a votação estiver aberta.

Entretanto, o sistema não criará um segundo registro. Atualizará o voto existente:

```
CONFIRMADO
RETIRADO
REATIVADO
```

A restrição única permanecerá.

Registrar:

- data do voto;
- data da retirada;
- data da reativação;
- histórico;
- sessão;
- motivo, quando aplicável.

Após o encerramento, nenhum voto poderá ser alterado.

---

14. Avaliação antifraude

14.1. Situações

Conta

- `ATIVA`;
- `DESAFIO_NECESSARIO`;
- `BLOQUEADA_TEMPORARIAMENTE`;
- `SUSPENSA`;
- `ENCERRADA`.

Proposta

- `RASCUNHO`;
- `ENVIADA`;
- `EM_VERIFICACAO`;
- `EM_ANALISE`;
- `PUBLICADA`;
- `NAO_APTA`;
- `ARQUIVADA`.

Voto

- `CONFIRMADO`;
- `RECEBIDO_EM_VERIFICACAO`;
- `INVALIDADO`;
- `RESTABELECIDO`;
- `RETIRADO`.

14.2. Sinais de risco

- muitos CPFs pelo mesmo dispositivo;
- CPFs em sequência;
- muitos erros de CPF;
- várias contas em poucos minutos;
- tempo de cadastro incompatível;
- votação sem navegação normal;
- votos em intervalos idênticos;
- várias contas votando no mesmo conjunto;
- uso direto da API;
- alteração constante de IP;
- acesso por rede de datacenter;
- proxy ou automação conhecida;
- desafio inválido;
- reutilização de desafio;
- formulário sem assinatura;
- excesso de propostas;
- anexos repetidos em várias contas;
- texto idêntico enviado por muitas contas.

14.3. Pontuação inicial sugerida

Evento	Pontos
Passkey cadastrada	-20
Conta antiga com histórico regular	-10
Telefone opcional confirmado	-15
Desafio inválido	+40
Mais de 3 contas no dispositivo	+30
Muitos CPFs sequenciais	+40
Votação excessivamente rápida	+25
Chamada direta irregular	+50
Reutilização de desafio	+50
Rede de automação conhecida	+25
Padrão coordenado de votos	+30

Classificação inicial:

Pontuação	Ação
0 a 29	Confirmar
30 a 49	Novo desafio
50 a 69	Espera progressiva

Pontuação	Ação
70 a 89	Receber em verificação
90 ou mais	Bloqueio temporário

Os pesos deverão ser configuráveis, testados e recalibrados. Não poderão ser tratados como prova absoluta de fraude.

---

15. Limites de velocidade

Cadastro

- 5 CPFs inválidos por sessão em 15 minutos;
- 3 contas por dispositivo em 24 horas;
- múltiplas contas por IP geram alerta;
- bloqueio progressivo, não definitivo.

Login

- 5 falhas em 15 minutos por conta;
- espera progressiva;
- novo desafio;
- proteção contra enumeração.

Propostas

- 5 por hora;
- 20 por dia;
- excesso em verificação;
- nenhum limite total permanente por cidadão.

Votação

- bloquear operações simultâneas;
- permitir votação normal em propostas diferentes;
- exigir novo desafio diante de velocidade anormal;
- colocar padrões automatizados em verificação.

Os pontos oficiais de internet pública poderão receber configuração específica, mas nunca ficarão dispensados da auditoria.

---

16. Resultado e contagem dos votos

16.1. Durante a votação

Recomenda-se não divulgar resultados completos em tempo real.

Isso reduz:

- manipulação coordenada;
- acompanhamento da eficácia de robôs;
- voto estratégico;
- influência indevida;
- tentativa de ultrapassar outra proposta artificialmente.

Poderão ser divulgados:

- número geral de participantes;
- quantidade geral de votos;
- período;
- temas disponíveis;
- metodologia.

16.2. Encerramento

Após o prazo:

1. bloquear novos votos;
2. concluir verificações pendentes;
3. analisar padrões anormais;
4. documentar invalidações;
5. permitir revisão;
6. consolidar o resultado;
7. publicar dados agregados;
8. preservar a auditoria.

16.3. Transparência

Publicar:

- total de participantes;
- votos confirmados;
- votos invalidados em quantidade agregada;
- critérios gerais antifraude;
- resultados por proposta;
- metodologia;
- data de consolidação.

Não publicar:

- CPF;
- nome do votante;
- IP;
- dispositivo;
- voto individual associado ao cidadão;
- pontuação individual de risco.

---

17. Estrutura mínima do PostgreSQL

17.1. Contas

```
CREATE TABLE cidadaos (
    id UUID PRIMARY KEY,
    entidade_id UUID NOT NULL,
    cpf_criptografado BYTEA NOT NULL,
    cpf_fingerprint VARCHAR(64) NOT NULL,
    nome VARCHAR(200) NOT NULL,
    senha_hash TEXT,
    situacao VARCHAR(40) NOT NULL DEFAULT 'ATIVA',
    risco_atual INTEGER NOT NULL DEFAULT 0,
    criado_em TIMESTAMPTZ NOT NULL DEFAULT NOW(),
    atualizado_em TIMESTAMPTZ NOT NULL DEFAULT NOW(),

    CONSTRAINT uq_cidadao_cpf_entidade
        UNIQUE (entidade_id, cpf_fingerprint)
);
```

17.2. Passkeys

```
CREATE TABLE cidadao_passkeys (
    id UUID PRIMARY KEY,
    cidadao_id UUID NOT NULL,
    credential_id TEXT NOT NULL UNIQUE,
    public_key BYTEA NOT NULL,
    sign_count BIGINT NOT NULL DEFAULT 0,
    criado_em TIMESTAMPTZ NOT NULL DEFAULT NOW(),
    ultimo_uso_em TIMESTAMPTZ,

    FOREIGN KEY (cidadao_id) REFERENCES cidadaos(id)
);
```

17.3. Propostas

```
CREATE TABLE propostas (
    id UUID PRIMARY KEY,
    entidade_id UUID NOT NULL,
    processo_id UUID NOT NULL,
    cidadao_id UUID NOT NULL,
    protocolo VARCHAR(50) NOT NULL UNIQUE,
    titulo VARCHAR(250) NOT NULL,
    descricao TEXT NOT NULL,
    situacao VARCHAR(40) NOT NULL,
    risco INTEGER NOT NULL DEFAULT 0,
    criado_em TIMESTAMPTZ NOT NULL DEFAULT NOW(),

    FOREIGN KEY (cidadao_id) REFERENCES cidadaos(id)
);
```

17.4. Votos

```
CREATE TABLE votos (
    id UUID PRIMARY KEY,
    entidade_id UUID NOT NULL,
    processo_id UUID NOT NULL,
    proposta_id UUID NOT NULL,
    cidadao_id UUID NOT NULL,
    situacao VARCHAR(40) NOT NULL,
    risco INTEGER NOT NULL DEFAULT 0,
    criado_em TIMESTAMPTZ NOT NULL DEFAULT NOW(),
    atualizado_em TIMESTAMPTZ NOT NULL DEFAULT NOW(),

    CONSTRAINT uq_voto_cidadao_proposta
        UNIQUE (
            entidade_id,
            processo_id,
            proposta_id,
            cidadao_id
        )
);
```

17.5. Eventos antifraude

```
CREATE TABLE eventos_antifraude (
    id UUID PRIMARY KEY,
    entidade_id UUID NOT NULL,
    cidadao_id UUID,
    proposta_id UUID,
    voto_id UUID,
    tipo VARCHAR(80) NOT NULL,
    pontuacao INTEGER NOT NULL,
    detalhes_protegidos JSONB,
    criado_em TIMESTAMPTZ NOT NULL DEFAULT NOW()
);
```

---

18. Proteção do CPF e demais dados

CPF

Utilizar:

- criptografia autenticada para armazenamento recuperável;
- HMAC-SHA-256 para unicidade e pesquisa;
- chave em cofre de segredos;
- rotação planejada de chaves;
- mascaramento na interface.

Não utilizar SHA-256 simples.

Senha

- Argon2id;
- salt exclusivo;
- parâmetros atualizáveis;
- bloqueio contra força bruta;
- nunca registrar em log;
- nunca permitir visualização administrativa.

Logs

Não registrar:

- CPF completo;
- senha;
- código de recuperação;
- dados de passkey;
- conteúdo sensível desnecessário.

O IP completo poderá ser mantido criptografado por prazo estritamente necessário à segurança e depois anonimizado, conforme política de retenção.

---

19. Segurança da aplicação

Implementar:

- HTTPS obrigatório;
- HSTS;
- CSP;
- proteção CSRF;
- validação de origem;
- consultas parametrizadas;
- prevenção de SQL injection;
- codificação contra XSS;
- proteção contra upload malicioso;
- antivírus auto-hospedado para anexos;
- limitação de tamanho e tipo de arquivo;
- sessões revogáveis;
- rotação de segredos;
- backup criptografado;
- testes de restauração;
- monitoramento;
- alertas;
- trilha imutável de auditoria;
- idempotência;
- proteção contra repetição;
- separação de permissões;
- atualização de dependências;
- análise de vulnerabilidades no CI/CD.

---

20. Administração antifraude

Criar painel com:

- contas criadas;
- cadastros bloqueados;
- CPFs estruturalmente inválidos;
- dispositivos com múltiplas contas;
- IPs com comportamento anormal;
- propostas em verificação;
- votos em verificação;
- padrões coordenados;
- histórico;
- justificativas;
- revisão;
- restabelecimento.

Regras administrativas

- CPF sempre mascarado;
- nenhuma exclusão física;

- invalidação com justificativa;
- dupla confirmação em ações em massa;
- registro do servidor responsável;
- impossibilidade de apagar auditoria;
- exportação protegida;
- acesso por perfil;
- MFA obrigatório para administradores.

---

21. Notificações

Como e-mail e telefone são opcionais, manter central de notificações dentro do portal.

Notificar:

- conta criada;
- passkey adicionada;
- novo login;
- proposta enviada;
- proposta em análise;
- complementação;
- proposta publicada;
- voto confirmado;
- voto em verificação;
- voto restabelecido ou invalidado;
- alteração de senha;
- novo código de recuperação.

Canais opcionais:

- portal;
- push no navegador;
- SMS;
- WhatsApp;
- e-mail.

Nenhum canal externo será obrigatório.

---

22. Usabilidade e acessibilidade

- linguagem simples;
- uma decisão por tela;
- campos grandes;
- teclado numérico para CPF;
- botão mostrar senha;
- progresso visível;
- salvamento automático;
- mensagens sem códigos técnicos;
- contraste adequado;

- foco visível;
- navegação por teclado;
- leitor de tela;
- áreas de toque adequadas;
- funcionamento em orientação vertical e horizontal;
- conexão lenta;
- prevenção de perda de dados;
- opção de imprimir;
- botão “Voltar” sem apagar;
- nenhuma dependência exclusiva de cor;
- nenhuma exigência de conhecimento sobre autenticação.

Evitar palavras como:

- WebAuthn;
- passkey;
- token;
- hash;
- CAPTCHA;
- idempotência.

Esses termos ficam restritos à documentação técnica.

---

23. Testes obrigatórios

Cadastro

1. CPF inválido é rejeitado.
2. CPF repetitivo é rejeitado.
3. CPF estruturalmente válido é aceito.
4. Mesmo CPF não cria duas contas.
5. Cadastro funciona sem e-mail.
6. Cadastro funciona sem telefone.
7. Cadastro funciona pelo celular.
8. Desafio reutilizado é bloqueado.
9. Robô não ignora o frontend.
10. Chamada direta à API exige validação.

Propostas

11. Cidadão cria várias propostas.
12. Protocolos são diferentes.
13. Rascunho é preservado.
14. Login não apaga o formulário.
15. Limite de velocidade funciona.
16. Excesso não exclui conteúdo.
17. Anexo malicioso é bloqueado.

18. Propostas suspeitas entram em verificação.

Votos

19. Primeiro voto é registrado.
20. Segundo voto na mesma proposta é bloqueado.
21. Voto em outra proposta é permitido.
22. Clique duplo não duplica.
23. Requisições simultâneas não duplicam.
24. Outro navegador não duplica.
25. Outro celular não duplica.
26. Chamada direta não duplica.
27. Republicação não permite novo voto.
28. Mudança de título não permite novo voto.
29. Retirada atualiza o mesmo registro.
30. Encerramento bloqueia novos votos.

Antifraude

31. Muitos CPFs no mesmo dispositivo geram risco.
32. CPF sequencial gera risco.
33. Alta velocidade gera risco.
34. IP compartilhado não causa bloqueio definitivo.
35. Desafio inválido é rejeitado.
36. Formulário expirado é rejeitado.
37. Assinatura inválida é rejeitada.
38. Voto suspeito não entra imediatamente no resultado.
39. Restabelecimento mantém histórico.
40. Invalidação exige justificativa.

Segurança e privacidade

41. CPF não aparece em URL.
42. CPF não aparece em logs.
43. CPF não aparece publicamente.
44. Senha utiliza Argon2id.
45. Passkey valida origem e assinatura.
46. Sessão expirada não vota.
47. CSRF inválido é rejeitado.
48. Tenant não acessa outro tenant.
49. Backup é restaurável.
50. Auditoria não pode ser apagada.

Usabilidade

51. Fluxo completo em celular simples.
52. Fluxo em conexão lenta.
53. Navegação por teclado.

54. Leitor de tela.
55. Aumento de fonte.
56. Mensagens compreensíveis.
57. Retorno automático após login.
58. Código de recuperação pode ser impresso.
59. Botões são adequados para toque.
60. Usuário consegue enviar outra proposta sem novo cadastro.

---

24. Critérios finais de aceite

A funcionalidade somente será considerada concluída quando:

- todo o cadastro ocorrer dentro do SaaS;
- nenhuma conta depender de e-mail;
- nenhuma conta depender de GOV.BR;
- nenhuma conta depender de comparecimento;
- CPF inválido for bloqueado;
- robôs simples forem bloqueados;
- comportamentos anormais forem classificados;
- conta for criada imediatamente;
- ação original for retomada;
- múltiplas propostas forem permitidas;
- um voto por proposta for garantido no PostgreSQL;
- votos em propostas diferentes forem permitidos;
- votos suspeitos forem separados;
- resultados definitivos passarem por consolidação;
- dados pessoais permanecerem protegidos;
- a solução funcionar em celular, tablet e desktop;
- os 60 testes forem aprovados;
- a limitação do modelo sem validação cadastral externa estiver formalmente documentada.

Resultado do modelo

Objetivo	Resultado
Impedir CPF matematicamente inválido	Garantido
Impedir duas contas com o mesmo CPF	Garantido
Impedir dois votos do mesmo CPF na mesma proposta	Garantido
Permitir votos em propostas diferentes	Garantido
Permitir várias propostas	Garantido
Reduzir robôs	Forte mitigação
Detectar criação massiva de contas	Forte mitigação
Confirmar que o CPF existe oficialmente	Não garantido

Objetivo	Resultado
Confirmar que o usuário é titular do CPF	Não garantido
Impedir absolutamente uma pessoa de usar CPFs diferentes	Não garantido

Esta orientação substitui integralmente as versões anteriores e estabelece o modelo definitivo sem consulta cadastral externa: controle determinístico de duplicidade, cadastro imediato e uma arquitetura antifraude baseada em múltiplas camadas, avaliação de risco, auditoria e revisão de operações suspeitas.