

Documento de Integração Frontend <-> Backend

Plataforma Nevú

1. Visão Geral

A plataforma Nevú é um sistema de recrutamento e seleção que atende dois tipos principais de usuários: Candidatos (pessoas em busca de vagas) e Empresas (organizações que publicam vagas e buscam talentos). O frontend já está implementado e atualmente utiliza dados mockados/localStorage, mas será migrado para consumir uma API real fornecida pelo backend.

2. Principais Funcionalidades do Frontend

Cadastro e login de usuários (candidato e empresa)

Gestão e edição de perfil

Planos de assinatura (upgrade, downgrade, cancelamento)

Publicação, edição e gestão de vagas (empresa)

Candidatura a vagas (candidato)

Troca de mensagens entre usuários

Notificações de eventos importantes

Integração com sistemas ATS (empresa)

Geração e uso de chave de API empresarial (empresa)

3. Fluxos de Usuário

Cadastro e Login

Usuários podem se cadastrar como candidato ou empresa.

O login diferencia o tipo de usuário e retorna os dados do perfil e assinatura.

Gestão de Perfil

Usuários podem visualizar e editar seus dados pessoais ou empresariais.

Empresas possuem campos específicos como CNPJ, setor, etc.

Planos de Assinatura

Usuários podem visualizar planos disponíveis, realizar upgrade/downgrade e cancelar assinatura.

O sistema controla limites de uso (ex: número de vagas, mensagens) conforme o plano.

Publicação e Gestão de Vagas

Empresas podem criar, editar, publicar e remover vagas.

Empresas visualizam o uso de vagas conforme o limite do plano.

Candidatura a Vagas

Candidatos podem se candidatar a vagas, visualizar status e cancelar candidaturas.

Mensagens

Usuários podem trocar mensagens (chat) entre candidatos e empresas.

Notificações

O sistema exibe notificações de eventos relevantes (ex: candidatura aprovada, nova mensagem, etc).

Integração ATS e API Empresarial

Empresas com planos avançados podem integrar sistemas ATS e gerar uma chave de API para integrações externas.

4. Requisitos Gerais para a API Backend

Autenticação:

O frontend espera autenticação baseada em token (preferencialmente JWT), com controle de sessão e permissões por tipo de usuário.

Endpoints RESTful:

A API deve seguir boas práticas REST, com endpoints claros para cada recurso (usuário, vaga, candidatura, mensagem, assinatura, etc).

Controle de Permissões:

O backend deve garantir que apenas usuários autorizados possam acessar/alterar recursos (ex: só empresa pode publicar vaga).

Limites e Regras de Plano:

O backend deve validar limites de uso conforme o plano do usuário (ex: número de vagas, mensagens, candidaturas).

Validação de Dados:

O backend deve validar todos os dados recebidos, retornando mensagens de erro amigáveis em caso de inconsistências.

Paginação e Filtros:

Para listas grandes (vagas, mensagens, notificações), a API deve suportar paginação e filtros.

Retorno de Dados:

As respostas da API devem ser rápidas, claras e conter todos os dados necessários para o frontend renderizar as telas.

Mensagens de Erro:

Mensagens de erro devem ser padronizadas e informativas para facilitar o tratamento no frontend.

5. Observações Importantes

O frontend já está preparado para consumir endpoints REST e trabalhar com autenticação por token.

O controle de limites de uso (vagas, mensagens, etc) deve ser feito no backend, não no frontend.

O backend deve garantir a integridade dos dados e a segurança das operações.

O sistema de planos e permissões é central para a experiência do usuário e deve ser rigorosamente implementado.

O backend deve estar preparado para evoluir, permitindo a inclusão de novas funcionalidades e integrações futuras.