**Projeto**

Por que escolhemos esse público alvo?

Porque escolhemos o campo de mercado de trabalho?

Porque escolhemos esse projeto?

Como vamos monetizar?

**Por que login com QR Code?**

Após uma visita à APAE, nos foi informado que muitos dos deficientes intelectuais possuem dificuldade em lembrar suas credenciais (emails e senhas), a partir deste ponto, decidimos desenvolver o login por QR Code afim de facilitar essa validação do usuário.

Por que utilizar GIT?

**Por que opção de gravar vídeo na descrição do currículo?**

Após uma visita à APAE, nos foi informado que deficientes intelectuais possuem muita dificuldade em lidar com longos textos, pois confundem e eles perdem um longo tempo tentando entender o que está sendo passado. Pensando nisso, habilitamos a opção de gravar vídeo para aqueles que se sentem mais à vontade falando do que escrevendo.

**Front-End**

Por que desenvolver o mobile/app primeiro?

**Por que utilizar sass?**

Sass é um framework para CSS que agiliza e facilita sua utilização, pois ele dispõe de vantagens como a criação de variáveis

Qual foi o critério para a escolha de cores?

Por que separar o cadastro em etapas?

Por que utilizar tantas imagens?

**Back-End**

**Por que usar php?**

O PHP é uma das linguagens mais utilizadas pelo mercado atualmente, assim, há bastante comunidades e fóruns na internet, o que auxilia quem está desenvolvendo.

Em questões de desenvolvimento, é uma linguagem madura e bem difundida, com várias funções prontas para uso, possui uma sintaxe fácil e, por ser fracamente tipada, proporciona uma liberdade maior para quem está programando. Em questão de conexão com banco de dados, ela dispõe de uma grande compatibilidade: Sybase, MySQL e MS-SQL aão alguns exemplos.

Além disso, dois integrantes do grupo possuem conhecimento prévio na linguagem, o que nos proporcionou maior confiança e agilidade no desenvolvimento do projeto.

**Por que usar um framework?**

Ele oferece uma base sólida a um projeto, implementando ferramentas e uma estrutura (a maioria MVC) que permite ao desenvolvedor ir direto à programação, ao invés de perder muito tempo criando essa mesma estrutura facilmente oferecida por um framework, além de uma grande estabilidade. Ele também se fez necessário devido ao curto período de tempo que nos foi dado para desenvolvimento.

Seus benefícios se estendem também à segurança, pois eles trazem configurações pré-definidas que protegem seu projeto de ataques comuns, como SQL Injection, por exemplo.

**Por que o Laravel?**

O Laravel, além de trazer todas as vantagens padrões que uma linguagem PHP e um framework disponibilizam, também traz algumas particulares:

**Blades**

Um compilador que separa o html do php, tornando o código mais organizado. Ele também possui heranças e sessões, o que facilita o reuso de código.

**Eloquent ORM e QueryBuilder**

Uma interação direita entre o PHP e o banco de dados, tornando possível manipular os dados do banco como se fossem uma classe do projeto.

Além disso, o queryBuilder é um “construtor” de querys, o que organiza seu código e te protege de ataques por meio de SQL Injection.

**Helpers**

Uma classe onde é possível criar uma função e acessá-la de qualquer parte do seu projeto, o que evita a duplicação de código.

**Migrations**

Uma pasta que é usada para guardar toda a estrutura do banco de dados, assim, não é preciso guardar um backup e depois rodar por um SGBD. Basta apenas executar um comandp e o banco de dados é criado.

**Sistema de rotas**

Realizam um mapeamento de URLs digitadas no navegador para alguma ação dentro da plataforma.

**Como o projeto está organizado?**

A organização tem como base o modelo MVC. No projeto, há uma estrutura organizada de pastas, essas são:

**Resources**

Contém: diretório JS que possui todos os arquivos JavaScript não compilados; lang que possui todas as línguas que podem ser traduzidas do inglês; Sass que possui todos os arquivos sass não compilados.

**Public**

Conta com as pastas: *CSS* e *JS* que salvam os arquivos JS e SASS compilados pelo npm; *image* que possui todas as imagens do sistema, como as fotos de perfil dos candidatos e os componentes do sistema;

**App**

Representa as funcionalidades Back-End. Nesse diretório, as classes modelos se encontram na raiz, elas representam cada tabela existente no banco de dados.

**Http**

Nesta pasta temos o Controllers, que contém as classes php responsáveis pela lógica da aplicação. Elas realizam validações, funções de CRUD, redirecionamento e o que mais for necessário para o tratamento daquele componente.

**Database**

Responsável por armazenar os arquivos php que gerenciam o banco de dados. Dentro da pasta migration você pode encontrar arquivos que geram as tabelas e, dentro da pasta seeds, o arquivo que popula as tabelas.

**Routes**

Armazena os arquivos que realizam os redirecionamentos. Dentro do arquivo web.php contém todas as rotas do projeto que utilizam o protocolo HTTP para ir a um Controller. Já o arquivo api.php faz as referencias às API’s utilizadas no projeto.

**Quais são as funcionalidades do projeto?**

Um candidato pode realizar um cadastro esse cadastro valida se o email e o nome de usuário ja existem, se a senha contem 8 dígitos e se ela foi confirmada.

Um candidato pode atualizar suas informações de se desejar.

Um usuario que nao tem cadastro não tem acesso a paginas que um usuário com login teria e um usuário que tenha login não tem acesso paginas como login e cadastro por exemplo.

Um candidato pode cadastrar um currículo que por sua vez pode ser editado.

E as vagas só aparecem se o candidato supre as necessidades da vaga

**Quais são medidas de segurança do sistema?**

Como estamos utilizado Laravel, toda a segurança do sistema é proporcionada automaticamente pelo framework, através de autenticações e validações de dados inseridos pelo usuário. Além disso, senhas de usuário são criptografadas pelo método hash password\_bcrypt, o que protege esse dado sensível caso um possível vazamento ocorra, além de não deixá-lo exposto a quem possui acesso ao banco de dados.

**Banco de Dados**

**Por que utilizar MySQL?**

**Como funciona o match?**

**Como o banco de dados armazena imagens e vídeos?**