**Programa IT Academy – Processo Seletivo – Edição #20**

**Nome Completo:** Eduardo Amengual Garcia

**E-mail:**  eduardo.agarcia117@gmail.com

# **Exercício Técnico**

**RESUMO DA SOLUÇÃO**

Desenvolvi uma solução web full-stack com framework Nuxt para criar a parte visual e definir as rotas da API do back-end que se comunicam com o banco de dados PostgreSQL do Backend-As-A-Service Supabase.

Comecei lendo o enunciado e extraindo os requisitos não funcionais e funcionais para posteriormente passar para a modelagem do banco de dados relacional. Uma vez com a modelagem do banco pronta, ficou claro que o sistema teria apenas duas entidades: Aposta e Edição, onde uma Edição precisa ter no mínimo uma ou várias apostas para executar seu sorteio.

Após ter montado um modelo de dados sólido e consistente, ficou claro que o sistema se funcionaria como um dashboard de um administrador e, comecei a implementar as funcionalidades com base no fluxo de páginas (arquivos do diretório pages) a seguir:

OBS: nuxt é um framework que cria as rotas da aplicação com base na árvore de diretórios do folder “pages/”, logo o nome de alguns componentes e a forma com que eles são organizados pode ficar um pouco confuso para pessoas não familiarizadas. Segue abaixo o um exemplo de como interpretar a árvore de arquivos de pages/:

pages/edicao/[id].vue = cria a rota: url/edição/{idEdicao} (parâmetro de url)

- Página inicial (index.vue) = primeira página da aplicação, nela há uma listagem de todas as Edições disponíveis e a opção de criar uma Edição.

- Página da edição ([id].vue) = é a página contendo os dados e informações da Edição escolhida pelo usuário via Página inicial. Essa página possui dois estados, sendo o primeiro, quando o usuário não iniciou o sorteio, nesse estado é possível ver todas as apostas feitas até o momento para a edição, é possível criar as apostas e iniciar o sorteio e é possível iniciar o sorteio após confirmação. O segundo e último estado da página é quando o sorteio já foi realizado, nesse estado são mostradas as informações solicitadas pela funcionalidade 5 (números sorteados, quantidade de vencedores, número de rodadas, apostas vencedoras em ordem alfabética e frequência dos números escolhidos das apostas em formato de tabela).

- Página de criação de aposta (criar.vue) = é acessada uma vez dentro de uma página de edição, pois uma edição tem suas próprias apostas. Essa página permite que o usuário faça sua aposta informando nome, cpf e seus 5 números de escolha podendo usar a funcionalidade de gerá-los aleatoriamente, vale dizer que essa página possui todas as restrições e verificações necessárias para não deixar o usuário cadastrar uma aposta fora dos padrões.

- Página do sorteio (sorteio.vue) = é acessada após confirmar a mudança de estado da fase de apostas para fase de sorteio em uma edição, nessa página contém todas as apostas feitas e um botão para começar o sorteio. O sorteio é feito de maneira automática, isto é, uma vez sorteado o número, é feita uma verificação por vencedores, caso não haja vencedores, outro número é sorteado e a verificação é feita novamente, tudo de maneira automática com o limite de 25 vezes. Uma vez terminado o sorteio, seja por vencedores ou tentativas de sorteio esgotadas, a página se torna inutilizável, onde o usuário pode apenas conferir os números sorteados, não restando nada a não ser voltar conferir os status do fim do sorteio novamente na página da respectiva edição ou, criar uma edição pela página inicial e seguir com o fluxo novamente.

**TESTES (aqui você deverá colar capturas de tela de todas as funcionalidades desenvolvidas e realizar comentários, use o espaço que julgar necessário)**

**FUNCIONALIDADE 1 – Iniciar**

**Tela de computador com fundo preto

Descrição gerada automaticamente**

**Figura 1: Tela inicial**

É a primeira tela (Figura 1) que o usuário visualiza ao entrar na aplicação, ela possui uma tabela com todas as edições já criadas no banco de dados juntamente informações sobre elas (ID, data de criação, status (finalizado ou em andamento), fase (sorteio ou apostas), número de vencedores e total de números sorteados) e, a opção de criar uma edição (uma edição é criada automaticamente com o status = em andamento, fase = apostas, vencedores = 0 e números sorteados = 0). Vale dizer que basta o usuário clicar em qualquer linha de Edição da tabela para ser redirecionado automaticamente para respectiva página daquela edição.

**FUNCIONALIDADE 2 – Registrar nova aposta**

É possível registrar uma nova aposta ao selecionar uma Edição na primeira tela (Figura 1), o que nos permitirá ir para a página da edição em específico (Figura 5), que por sua vez nos dará acesso a página de criação de apostas pelo botão “Criar aposta”.

Tela de celular

Descrição gerada automaticamente

**Figura 2: Formulário de criação de aposta**

A página de criação de apostas (Figura 2) trata-se de um simples formulário que permite que o usuário cadastra sua aposta apenas de todos os dados forem preenchidos. O campo de CPF possui uma máscara automática para deixar a string no formato ###.###.###-## e fica vermelho para valores inválidos, assim como o campo de número, onde o usuário pode selecionar o número manualmente escrevendo no input apenas valores entre 1 e 50 (Figura 3).

Tela de celular com fundo preto

Descrição gerada automaticamente

Figura 3: Tratamento de dados inválidos

Vale dizer que o usuário pode remover um número selecionado caso deseje, apenas clicando no respectivo quadrado do número como informado no input de número. Por fim, ao cadastrar corretamente uma Aposta, o sistema exibe uma mensagem de confirmação e limpa os campos do formulário para futuros cadastros (Figura 4).

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

**Figura 4: Mensagem de confirmação de cadastro de aposta**

**FUNCIONALIDADE 3 – Listar apostas**

Todas as apostas feitas em uma Edição se encontram disponíveis da página da respectiva edição quando o sorteio NÃO foi FINALIZADO.

Tela de computador com letras e números em fundo preto

Descrição gerada automaticamente

**Figura 5: Tela da respectiva edição selecionada**

Vale dizer que caso o usuário crie uma Edição e tente iniciar a fase de sorteio sem ter cadastrado nenhuma Aposta, uma mensagem de erro (Figura 6) ocorrerá indicando que para tal ação for realizada, é necessário ter pelo menos uma aposta cadastrada.

Tela de celular com fundo preto

Descrição gerada automaticamente

**Figura 6: Mensagem de erro ao tentar iniciar sorteio sem apostas registradas**

**FUNCIONALIDADE 4 – Finalizar apostas e executar o sorteio**

Uma vez com apostas já realizadas, o usuário pode começar o sorteio ao clicar no botão “Iniciar Sorteio”, o qual fará com que uma mensagem de confirmação apareça no canto inferior direito da tela exigindo a confirmação do usuário (Figura 7).

**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

**Figura 7: Mensagem de confirmação de início de sorteio**

Essa notificação desaparece com o tempo, mas só permite que o usuário vá para o começo do sorteio única e exclusivamente após ele clicar em confirmar. Uma vez confirmado, caso o usuário volte para a página de edição sem ter realizado o sorteio de fato, não será mais possível criar apostas pois o botão fica bloqueado (Figura 8).

Tela de celular

Descrição gerada automaticamente

**Figura 8: Botão de criar aposta bloqueado após início da fase de sorteio**

Uma vez na tela de sorteio (Figura 9), todas as apostas feitas ficam disponíveis e o sistema espera o clique do usuário no botão “Iniciar Sorteio”.

Tela de computador com fundo preto

Descrição gerada automaticamente

**Figura 9: Tela de sorteio da respectiva edição**

Após clicar no botão, o sorteio começa automaticamente (Figura 10), onde 5 números de 1 a 50 são sorteados e verificados com os números das apostas, se não houver nenhum ganhador, o sistema sorteia mais um número e faz a verificação novamente, e assim sucessivamente até houver um ganhador ou 25 números forem sorteados.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Figura 10: Tela de final de sorteio**

Vale dizer que a chance de houver um ganhador é bem baixa, entretanto, caso deseje-se forçar um ganhador do sorteio, ou vários ganhadores, acesse o arquivo **sorteio.vue** (Figura 11)e, tire dos comentários o trecho de código da linha 52 até a linha 57 e, comente a linha 60.

Texto

Descrição gerada automaticamente

**Figura 11: Tutorial de como forçar um vencedor**

Esse código, ao invés de adicionar o próximo número aleatório sorteado na fila para verificação com as apostas, força a combinação escolhida pelo usuário a ser o “próximo número sorteado” na posição que o usuário escolher. Por exemplo, se a variável position for 10 e, a combinação do vetor [1,2,3,4,5], sabe-se que a décima combinação do sorteio será os números [1,2,3,4,5].

**FUNCIONALIDADE 5 – Fim da apuração**

A apuração do sorteio pode ser feita após a ação do mesmo, voltando para a página da respectiva edição do sorteio, onde o conteúdo da página irá mudar e, passaremos a ver a lista de vencedores caso haja algum, todos os números sorteados, a quantidade de rodadas do sorteio, a quantidade de vencedores em formato de tabela organizados em ordem alfabética (Figura 12).

**Tela de computador com jogo

Descrição gerada automaticamente**

**Figura 12: Parte 1 da apuração do sorteio**

E para fechar essa funcionalidade, logo abaixo da tabela de ganhadores, há uma lista com todos os números escolhidos pelos usuários, bem como a quantidade de vezes que foram escolhidos no total da edição (Figura 13).

Tela de computador com fundo preto

Descrição gerada automaticamente

**Figura 13: Frequência dos números escolhidos**

**FUNCIONALIDADE 6 – Premiação**

[cole aqui a(s) captura(s) de tela mostrando a funcionalidade em ação]

[você pode incluir uma explicação ou detalhamento em texto]

**AUTOAVALIAÇÃO**

Você concluiu a implementação de 100% das funcionalidades solicitadas?

( X ) Sim ( ) Não

Para as 6 principais funcionalidades solicitadas, como você avalia a sua solução?

Marque um ‘X’.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Inexistente/ Insuficiente | Pouco satisfeito(a) | Satisfeito(a) | Muito satisfeito(a) |
| Funcionalidade 1 |  |  |  | X |
| Funcionalidade 2 |  |  |  | X |
| Funcionalidade 3 |  |  |  | X |
| Funcionalidade 4 |  |  | X |  |
| Funcionalidade 5 |  |  |  | X |
| Funcionalidade 6 |  |  |  |  |

**Principais dificuldades**

A minha principal dificuldade foi a utilização das duas principais tecnologias do projeto: Nuxt e o Supabase. São tecnologias das quais eu nunca tive um contato prévio, apenas de uma maneira bem indireta (Nuxt é feito em cima do framework Javascript Vue.js, o qual eu possuo experiência e, o Supabase possui um banco de dados relacional e funcionalidades muito semelhantes ao Google Firebase, os quais eu já tenho uma certa familiaridade).

Eu acabei escolhendo essas tecnologias mesmo sem ter o domínio sobre elas pois vi no exercício técnico a oportunidade de aprender coisas novas juntamente com tecnologias novas, onde eu poderia realizar meu primeiro projeto full-stack com o front-end e o back-end totalmente sozinho. No fim, essa dificuldade foi algo que me motivou ainda mais a resolver o exercício técnico, tornando a jornada da resolução do exercício prazerosa, onde a documentação dessas tecnologias foram minhas principais aliadas.

**Desempenho Geral**

Começando pela minha organização, eu foquei em ler bastante o enunciado e levantar os requisitos necessários para a modelagem do banco para então ter uma visão sólida da estrutura e do relacionamento dos dados. Tais ações somadas à eficiência do meu estudo sobre as tecnologias Nuxt e Supabase, tecnologias essas que são muito poderosas e que facilitaram e muito o desenvolvimento full-stack, resultaram no trabalho aqui presente, uma aplicação web full-stack 100% funcional, que abrange todos os requisitos solicitados e, que me proporcionou novos conhecimentos e habilidades durante meu tempo de desenvolvimento.

Obrigado por participar deste processo seletivo.  
Salve o documento em PDF com o seu nome.