

Qual é o número?

Visão geral

O problema consiste em receber um número através de uma requisição e implementar um jogo para acertar este número através de palpites. Ao errar um palpite, irá ser informado se o número obtido é maior ou menor do que o palpite feito. O palpite realizado ou *status code* de erro de requisição devem ser exibidos na tela no formato de LED de 7 segmentos. O palpite será obtido como entrada em um campo de texto, que deverá ser processado apenas quando o botão ENVIAR for clicado.

Para solucionar o problema proposto, você deverá implementar a solução utilizando, HTML, CSS e Javascript.

Layout

Baixe o [arquivo de layout](#).

O arquivo foi construído no [Figma](#). Você pode criar uma conta e fazer upload do arquivo.

The image displays three vertical panels representing different states of a web application titled "QUAL É O NÚMERO?".

- Panel 1 (Left):** Shows the title "QUAL É O NÚMERO?" at the top. Below it is a large 7-segment LED display showing the number "0". At the bottom, there is a text input field containing the number "81" and an orange button labeled "ENVIAR".
- Panel 2 (Middle):** Shows the title "QUAL É O NÚMERO?". Below it, the text "É menor" is displayed in orange. Underneath is a 7-segment LED display showing the number "08". At the bottom, there is a text input field with the placeholder text "Digite o palpite" and an orange button labeled "ENVIAR".
- Panel 3 (Right):** Shows the title "QUAL É O NÚMERO?". Below it, the text "É maior" is displayed in orange. Underneath is a 7-segment LED display showing the number "98". At the bottom, there is a text input field with the placeholder text "Digite o palpite" and an orange button labeled "ENVIAR".



Especificação

Você deverá enviar uma requisição para receber um valor aleatório utilizando o endpoint abaixo, com parâmetros especificando o limite inferior e superior. Para o modo padrão de jogo, utilize o limite inferior como 1 e o limite superior como 300, conforme a URL abaixo:

`https://us-central1-ss-devops.cloudfunctions.net/rand?min=1&max=300`

A resposta estará no formato JSON com o valor aleatório especificado no campo `value`:

```
{
  "value": 15
}
```

Esteja preparado para eventuais falhas: ausência de parâmetros ou parâmetros incorretos irão causar falha na requisição. Além disso, há uma pequena chance desta requisição retornar com erro mesmo com parâmetros corretos. O *status code* em casos de falha deverá ser mostrado no LED e pode ser obtido em métodos específicos da tecnologia utilizada ou obtido do JSON respondido em casos de falha, no campo `StatusCode`:

```
{
  "Error": "Bad Gateway",
  "StatusCode": 502
}
```

Você deverá informar em um texto acima do LED os resultados possíveis:

- "Erro": quando houver erro na requisição
- "É menor": quando o palpite enviado é **maior** que o número obtido
- "É maior": quando o palpite enviado é **menor** que o número obtido
- "Você acertou!!!": quando o palpite enviado é igual ao número obtido

Além disso, deverá existir na tela um botão NOVA PARTIDA, cujo clique deverá criar uma nova partida obtendo um novo número aleatório (o que implica em fazer uma nova requisição).

É importante mencionar que o número a ser adivinhado pelo usuário DEVE impreterivelmente ser obtido por meio da requisição descrita anteriormente. O objetivo do problema envolve testar a sua habilidade de lidar com requisições. Portanto gerar um número aleatório dentro da própria aplicação foge do propósito desta prova.

Implementação

Segmentos:

- O display pode conter números não-negativos de 1 a 3 algarismos.
- O valor numérico exibido nos segmentos deve representar o palpite realizado ou o status code obtido quando a requisição falhar.
- Só deve ser apresentada a quantidade de algarismos necessária (O número 29 precisar ser exibido sem zero à esquerda).
- Você deve implementar sua própria solução para exibir os segmentos. Não serão aceitas bibliotecas de terceiros para esta etapa. Em especial, **NÃO** utilize uma fonte pronta para exibir os segmentos.

Botão NOVA PARTIDA:

- O botão ficará visível apenas quando houve erro ao receber o número ou quando o jogador acertou o palpite

Campo de entrada:

- Mostrar o texto "Digite o palpite" como placeholder
- O valor deve aparecer nos segmentos assim que o botão *ENVIAR* for clicado e o input deve ser resetado para o estado inicial.
- O botão de enviar deverá ficar desabilitado quando houve erro ao receber o número ou quando o jogador acertou o palpite. O usuário deve clicar em "NOVA PARTIDA" neste caso.

Comentários e documentação

Você DEVE documentar qual a lógica utilizada no funcionamento do programa. Se alguma estrutura de dados especial tiver sido utilizada e for relevante para o funcionamento do algoritmo, citá-la também.

Exemplo:

```
/**
 * A lógica do programa se baseia em (...)
 * Para algarismo do display, existe uma estrutura
 * correspondente (...)
 */
```

Critérios de avaliação

- Compilar e executar sem crashes
- Descrição do funcionamento do programa (deve condizer com o que foi implementado)
- Implementação da requisição e tratamento correto dos possíveis status
- Tratamento correto do input do usuário
- Funcionamento correto e robustez da solução
- Controle de estado do botão NOVA PARTIDA
- Layout condizente com os exemplos, incluindo responsividade
- Organização e clareza do código

Pontos extras

As funcionalidades a seguir **NÃO SÃO OBRIGATÓRIAS** para que seu programa seja avaliado, mas contarão como pontos extras para avaliarmos seu nível técnico de programação. Fique à vontade para escolher um ou mais itens abaixo para usar na sua implementação.

★ Testes automatizados:

- Você pode garantir o funcionamento correto da sua aplicação com escrita de testes (unitários ou de ponta a ponta).

- Os testes escritos devem executar sem falhas

★ **Configuração de bundler:**

- Você pode configurar um bundler para a entrega final do seu projeto, simulando um ambiente de produção ou não. A escolha do bundler é opcional.

Vale ressaltar: Você não deve utilizar qualquer outra biblioteca de terceiros: ao invés disso, construa sua própria solução. Em especial, implemente você mesmo a lógica relacionada com o LED de sete segmentos, sem utilizar nenhuma biblioteca que porventura o faça.

Boa prova =)