



DESAFIO FLUTTER

PROCESSO SELETIVO CHECKMOB

Tema: Criar um projeto em **Flutter** chamado "Fast Trivia".

Instruções:

- 1- Após finalizado, **o projeto deverá ser hospedado publicamente no GitHub**, e o link para o repositório enviado através do link: <https://forms.gle/Tbr3ZsvB8vcnuant7>
- 2- Você pode utilizar bibliotecas externas quando achar necessário;
- 3- Projetos incompletos podem ser enviados e serão avaliados de acordo com o que foi entregue;
- 4- A partir do recebimento do e-mail com o desafio, você terá até **Terça-feira, 22 de maio até às 11:30 da manhã** para completar a tarefa e enviar.

Projeto:

Aplicação em que o usuário receba um questionário, com uma ou mais perguntas, tendo alternativas, e possa responder de acordo com seus conhecimentos, ao final ser retornado a sua pontuação total. No mínimo 4 telas:

- Tela onde podemos ver os questionários respondidos e iniciar um;
- Tela para responder o questionário;
- Tela após finalizar o questionário mostra total de questões acertadas;
- Tela para visualizar o questionário respondido e as respostas escolhidas, podendo ver a correta;

Requisitos:

- Ao selecionar a opção de responder um novo questionário, a aplicação deverá receber o modelo de questionário a ser respondido via uma requisição mock HTTP;
- Permitir que o usuário responda questionários com questões do tipo seleção única;
- Exibir a lista de questionários respondidos pelo usuário;
- Permitir que o usuário selecione um questionário respondido anteriormente e exibir quais foram as respostas informadas.
- Inclua instruções sobre o projeto no **readme.md** do repositório, tais como versão do Flutter e outras ferramentas utilizadas.



Descrição:

A aplicação deverá possuir uma lista com o histórico de todos os questionários respondidos pelo usuário e uma forma de iniciar um novo questionário, esse histórico deve estar em um banco de dados e persistir além da sessão.

Um questionário pode possuir várias questões e suas alternativas, tendo seu gabarito para verificação e correção.

Para iniciar uma nova resposta de questionário, o aplicativo deve receber as informações necessárias utilizando uma requisição mock assíncrona, ou seja, uma simulação de requisição.

- O JSON abaixo pode ser utilizado como exemplo de resposta da requisição mock:

```
{
  "questionario": {
    "id": 1,
    "titulo": "Natureza brasileira",
    "questoes": [
      {
        "id": 1,
        "titulo": "Florestas brasileiras",
        "pergunta": "Qual o nome da maior floresta brasileira?",
        "gabarito": 4,
        "alternativas": [
          {
            "id": 1,
            "titulo": "Caatinga"
          },
          {
            "id": 2,
            "titulo": "Cerrado"
          },
          {
            "id": 3,
            "titulo": "Mata Atlântica"
          },
          {
            "id": 4,
            "titulo": "Amazônica"
          }
        ]
      }
    ]
  },
  {
    "id": 2,
    "titulo": "Praias brasileiras",
```

```

"pergunta": "Qual estado tem a maior faixa litorânea?",
"gabarito": 3,
"alternativas": [
  {
    "id": 1,
    "titulo": "São Paulo"
  },
  {
    "id": 2,
    "titulo": "Maranhão"
  },
  {
    "id": 3,
    "titulo": "Bahia"
  },
  {
    "id": 4,
    "titulo": "Rio de Janeiro"
  }
]
}
]
}
}

```

Observação: Considerar que o JSON que será utilizado para testes pode conter mais questões, e todas deverão ser exibidas corretamente assim como suas alternativas.

Utilizando o JSON recebido, a aplicação deve permitir que o usuário responda às questões presentes no modelo, as respostas devem ser mantidas em memória, persistindo durante a sessão atual, ou seja, até o usuário fechar o aplicativo.

Depois que todas as questões forem respondidas, e o questionário finalizado, o aplicativo deve exibir uma tela mostrando o resultado, e logo após o questionário respondido completo, com todas as respostas do usuário e sua devida correção. Essa tela também deverá possuir uma forma de retornar para a tela de listagem de histórico, para que o usuário possa responder novos questionários.

Ao selecionar um questionário respondido no histórico, a aplicação deve permitir que o usuário abra-o, exibindo a tela com questionário respondido completo.