Disciplina: Programação de dispositivos moveis

Aluno: Francisco Klayrton

Atividade: Avaliação-08

**Questionário: Programação de Dispositivos Móveis com Flutter**

1. O que é o Flutter, de acordo com as fontes, frequentemente utilizado para?  
   * a) Exclusivamente o desenvolvimento de aplicativos iOS
   * b) Apenas o design de bancos de dados
   * c) A construção de aplicações multiplataforma (mobile, web e desktop) a partir de uma única base de código
   * d) A criação de sistemas operacionais de baixo nível
2. De acordo com as fontes, o que o "Networking" no Flutter essencialmente permite?  
   * a) Apenas a manipulação de dados armazenados localmente
   * b) Interagir com serviços externos, recuperar dados e atualizar informações pela internet
   * c) Somente a criação de interfaces de usuário estáticas
   * d) O gerenciamento de animações em tempo real
3. Qual pacote é descrito nas fontes como a maneira mais simples e popular para emitir requisições HTTP em Flutter?  
   * a) dio
   * b) chopper
   * c) http
   * d) retrofit
4. Por que as fontes recomendam evitar o uso direto de dart:io ou dart:html para fazer requisições HTTP em um aplicativo Flutter?  
   * a) Eles são mais lentos para processar respostas
   * b) Eles não suportam requisições autenticadas
   * c) Eles são dependentes da plataforma e vinculados a uma única implementação
   * d) Eles não permitem o tratamento de erros
5. Para adicionar o pacote http como uma dependência em seu projeto Flutter, qual comando é o mais comum, conforme indicado nas fontes?  
   * a) flutter run http
   * b) flutter build http
   * c) flutter pub add http
   * d) flutter init http
6. Ao realizar uma requisição de rede assíncrona, como http.get(), o que é retornado, representando um valor ou erro potencial disponível no futuro?  
   * a) Um Stream
   * b) Um Future
   * c) Uma Response síncrona
   * d) Um Map diretamente
7. Para converter a resposta de uma requisição HTTP (geralmente JSON) em um objeto Dart personalizado de forma organizada, o que é frequentemente utilizado em uma classe, conforme as fontes?  
   * a) Um construtor padrão
   * b) Um método estático parseJson
   * c) Um construtor de fábrica (factory constructor), como fromJson
   * d) Um método de extensão
8. Qual widget do Flutter é projetado para facilitar o trabalho com fontes de dados assíncronas e exibir a interface do usuário com base no estado de uma Future (carregamento, sucesso ou erro)?  
   * a) StatelessWidget
   * b) StatefulWidget
   * c) FutureBuilder
   * d) ListView.builder
9. As fontes desaconselham colocar chamadas de API diretamente no método build() de um widget. Qual é o motivo principal para essa recomendação?  
   * a) O método build() não consegue lidar com operações assíncronas
   * b) O método build() é chamado com pouca frequência, o que atrasaria a exibição dos dados
   * c) Chamadas repetidas no build() a cada reconstrução podem deixar o aplicativo lento
   * d) Chamadas de API devem ser feitas apenas em main()
10. Qual das seguintes opções NÃO é listada como uma causa comum de NetworkException em aplicativos Flutter, de acordo com as fontes?  
    * a) Indisponibilidade da rede
    * b) URLs inválidas
    * c) Erros no lado do servidor (por exemplo, status 500)
    * d) Uso excessivo de widgets Material Design
11. Qual é uma das práticas essenciais para lidar com NetworkException em aplicativos Flutter, conforme as fontes?  
    * a) Desativar todas as funcionalidades de rede para evitar erros
    * b) Ignorar as exceções de rede e deixar o aplicativo travar
    * c) Implementar tratamento robusto de erros, envolvendo operações de rede em blocos try-catch
    * d) Usar apenas requisições HTTP síncronas
12. Na classificação dos códigos de status HTTP, qual categoria indica que a requisição foi recebida, compreendida e processada com sucesso?  
    * a) 1xx (Informacionais)
    * b) 2xx (Sucesso)
    * c) 3xx (Redirecionamento)
    * d) 4xx (Erros do Cliente)
13. Se um servidor retornar um código de status HTTP 400 Bad Request, a qual categoria esse erro pertence, indicando uma falha do lado do cliente?  
    * a) 1xx (Informacionais)
    * b) 2xx (Sucesso)
    * c) 4xx (Erros do Cliente)
    * d) 5xx (Erros do Servidor)
14. Uma das principais vantagens do pacote dio sobre o http é a sua capacidade de:  
    * a) Ser mais leve e ter menos abstrações
    * b) Exigir decodificação manual de JSON
    * c) Suportar decodificação automática de JSON
    * d) Não oferecer suporte a requisições autenticadas
15. Qual recurso do dio é particularmente útil para adicionar tokens de autenticação automaticamente, registrar requisições e respostas para depuração, ou lidar com códigos de status específicos globalmente?  
    * a) HttpClientAdapter
    * b) BaseOptions
    * c) CancelToken
    * d) Interceptores
16. Em cenários onde um usuário pode sair de uma tela antes que uma requisição seja concluída, qual funcionalidade do dio é útil para cancelar uma requisição em andamento e economizar recursos?  
    * a) Reconfiguração da URL base
    * b) Utilização de um CancelToken
    * c) Desativação de logs de requisição
    * d) Apenas esperar pelo timeout da requisição
17. Quando seu aplicativo Flutter precisa fazer upload ou download de arquivos com relatórios de progresso, qual pacote é preferível devido ao seu suporte embutido e API simplificada (FormData)?  
    * a) http
    * b) dio
    * c) dart:convert
    * d) path\_provider
18. Qual é o principal benefício de usar interceptores HTTP (como os oferecidos por http\_interceptor ou dio) em um aplicativo Flutter para fins de depuração?  
    * a) Eles automaticamente otimizam o tamanho do aplicativo
    * b) Eles permitem interceptar e modificar requisições e respostas, facilitando a depuração e o registro
    * c) Eles fornecem uma interface gráfica para o usuário final
    * d) Eles substituem a necessidade de qualquer tratamento de erro
19. Em Flutter, para executar código Dart em segundo plano, mesmo quando o aplicativo não é o ativo, qual modelo para multithreading é utilizado, não compartilhando memória com o programa principal?  
    * a) Thread
    * b) Process
    * c) Isolate
    * d) Coroutine
20. Qual ferramenta é descrita nas fontes como um "HTTP Interceptor poderoso" que pode interceptar, gravar, depurar e modificar requisições HTTP, além de ser útil para testes entre dispositivos?  
    * a) Zipy
    * b) Firebase Crashlytics
    * c) Requestly
    * d) Google AdMob