**What made you choose a career of software engineer (Mobile Developer)….?**

1. **I am interested in sharing a passion for technology, creativity, and problem-solving. i enjoy working with the latest tools and technologies to create innovative and user-friendly applications.**
2. **In addition to these general interests, i may also have specific interests related to the types of apps i want to develop, such as games, social media, productivity tools, or enterprise solutions.** **Overall, the combination of passion for technology, growing demand, career opportunities, and the ability to make a positive impact drives me to choose mobile app development as a career.**

**What salary do you aspire to compared to your work experience and work in Flutter? Max duration is 3 minutes….?**

1. **Salaries in the tech industry can vary widely depending on several factors, including experience, location, company size and industry.**
2. **I have 2 years of experience in Mobile app development, developing skills in creating high-performing and user-friendly mobile applications. My experience includes specific projects and achievements that have made significant contributions and improvements.**
3. **Given my background, I am well-positioned within the competitive range for Flutter developers. Typically, Flutter developers with my level of experience earn between 14000 and 15000. My salary expectations are aligned with my professional capabilities and the level of responsibility I am prepared to take on.**

**Can you describe an instance where you optimized a Flutter application? What specific changes did you make, and what was the impact on the application's performance?Max duration is 3 minutes**

1. **In a recent project i optimized it by focusing on both performance and UI. I optimized rendering by replacing complex Column widgets with ListView.builder, which improved scroll performance.**
2. **Using FadeInImage to display placeholders while images are loading,**
3. **Minimize unnecessary rebuilds of widgets by using const constructors for immutable widgets and carefully managing state updates**
4. **Offload time-consuming tasks like network requests or complex calculations to background threads using async and await to prevent blocking the UI thread.**
5. **Regularly review your code for potential optimizations and refactor as needed.**

**What are the four pillars of OOP in detail?**

1. **Inheritance: represents an "is-a" relationship between classes. One class becomes more specific version of another class by acquiring its properties and behavior.**
2. **Encapsulation: group all the data members & member functions within one single unit**
3. **Polymorphism: Allows objects of different derived classes to be treated as objects of the base class, promoting flexibility and extensibility.**
4. **Abstraction: Hides the implementation details of a class, allowing clients to interact with it through a well-defined interface.**

* **6- How is Polymorphism implemented in Flutter**
* **Inheritance: establish relationships between classes, with super classes providing a common interface and derived classes overriding methods.**
* **Overriding: Derived classes can override methods defined in the base class to provide specific implementations tailored to their functionality.**
* **Dynamic Type Dispatch: determine which method implementation to call at runtime based on the object's dynamic type.**
* **Widget Trees: Flutter's UI is represented as a tree structure of widgets, and polymorphism allows for efficient rendering and management of complex widget trees.**
* **State Management: leverage polymorphism to update and rebuild widgets based on changes in their state.  
  7- How does Dart solve the problem of multiple inheritance**

**Dart handles multiple inheritance through a mechanism called mixins. Mixins allow classes to inherit functionality from multiple sources without the complexities associated with traditional multiple inheritance Avoids diamond problem.   
8- What is the difference between an interface and a mixin**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Interface** | **Mixin** |
| **Abstract methods** | **Must contain abstract methods** | **Can contain both abstract and concrete methods** |
| **Implementation** | **No implementation** | **Can contain implementation** |
| **Multiple implementations** | **Yes** | **No** |
| **Application** | **Implemented by classes** | **Applied to classes with “with” key word** |

**9- What is the difference between compile-time and run-time polymorphism, and how are they implemented in Dart**

|  |  |
| --- | --- |
| **Compile Time** | **Runtime** |
| **occurs before the program is actually run //** Before execution | **occurs after the program has been compiled and is running. //** During execution |
| **compiler analyzes the source code prepared for execution, checks for syntax errors, and translates it into machine code** | **the runtime environment executes the compiled code or byte , checks for code exceptions or memory leaks** |
| int result = 10 / 0 // syntax error | int result = 10 / 0; // divide by zero exception |

**10- Why do we need state management in Flutter  
11- Which state management solution do you prefer and why  
12- What are the main differences between Bloc and Riverpod  
13- What is the structure of Bloc  
14- Do state management solutions offer a way to cache data  
هان أنا جاوبته أنها بتوفر طريقة لتخزين البيانات مؤقتاً في الـ RAM لكن أكيد مش تخزين طويل المدى، فهو مباشرة انتقل لنقطة الـ Storage والتخزين طويل المدى.  
15- What do we need to cache data for a long time  
16- What packages do you use for handling data storage  
17- What is the difference between Shared Preferences, Sqflite, and Hive  
18- How do you check if cached data has expired and needs to be refreshed  
19- Do you use these packages directly or through service classes  
طبعاً أنا قلتله أني بعمل class ينظم استخدام الـ package وقلت أن هذا الاشي أفضل من ناحية الـ clean architecture ومن ناحية الـ dependency injection فهو مباشرة بدأ يسأل عنه 😀   
20- What is dependency injection, and why do we need it  
21- What packages do you use for dependency injection  
فقط سؤالين بسيطين وبعدها رجع لموضوع الـ storage  
22- How would you handle large file storage, such as a 10GB file  
23- How would you handle downloading and uploading files in Flutter  
ثم سؤال في الـ testing  
24- What do you know about testing in Flutter and what packages do you use  
ثم انتقلنا لـ Asynchronous Programming  
25- What is the difference between a Future and a Stream  
26- What are the different ways to handle Future errors in Flutter  
27- Can a Stream in Flutter have multiple subscribers  
28- How do you transform data in a Stream  
29- How do you handle errors in a Stream  
بعدها سألني عن الـ Authentication  
30- Have you implemented third-party sign-in options like Google Sign-In, LinkedIn, Apple, etc  
فأنا جاوبت Google Sign-In فسألني:  
31- What are the steps to perform Google Sign-In  
وانتقلنا لـ Google Maps  
32- Have you worked with Google Maps  
33- How would you track a location in Flutter  
34- Are you familiar with GeoJSON  
وبعدها موضوع الـ WebSockets والـ WebRTC  
35- Have you used WebSockets  
36- Have you used WebRTC  
وأخيراً  
37-Are you comfortable with learning new skills**