

## **Câu 1: Các nền tảng cho thiết bị di động thông minh hiện nay**

### **1. Android**

- **Đặc điểm:**
  - Nền tảng mã nguồn mở, phát triển bởi Google.
  - Hỗ trợ nhiều thiết bị từ nhiều nhà sản xuất khác nhau.
- **Ưu điểm:**
  - Tính linh hoạt cao, dễ dàng tùy chỉnh.
  - Thị phần lớn nhất toàn cầu.
- **Khuyết điểm:**
  - Phân mảnh cao, khó khăn trong việc cập nhật phiên bản mới.
  - Nhiều ứng dụng kém chất lượng trên cửa hàng Google Play.

### **2. iOS**

- **Đặc điểm:**
  - Nền tảng đóng, phát triển bởi Apple.
  - Chỉ chạy trên các thiết bị của Apple.
- **Ưu điểm:**
  - Tính bảo mật cao và trải nghiệm người dùng mượt mà.
  - Cập nhật hệ điều hành đồng bộ cho tất cả thiết bị.
- **Khuyết điểm:**
  - Chi phí phát triển cao hơn.
  - Hạn chế về tùy chỉnh và sự đa dạng của ứng dụng.

### **3. Windows Phone**

- **Đặc điểm:**
  - Nền tảng phát triển bởi Microsoft.
  - Tích hợp tốt với dịch vụ Microsoft.
- **Ưu điểm:**
  - Giao diện người dùng thân thiện.
  - Tích hợp tốt với các dịch vụ của Microsoft.
- **Khuyết điểm:**

- Thị phần thấp, ít ứng dụng.
- Không còn được hỗ trợ mạnh mẽ từ Microsoft.

## Câu 2: Nền tảng phát triển ứng dụng di động phổ biến

### 1. Native Development

- **Đặc điểm:** Sử dụng ngôn ngữ lập trình gốc (Java/Kotlin cho Android, Swift cho iOS).
- **Ưu điểm:** Hiệu suất cao, trải nghiệm người dùng tốt nhất.
- **Khuyết điểm:** Phát triển tốn thời gian và công sức hơn cho mỗi nền tảng.

### 2. Cross-Platform Development

- **Flutter:** Sử dụng Dart. Tạo ra giao diện đẹp, hiệu suất cao.
- **React Native:** Sử dụng JavaScript. Khả năng phát triển nhanh chóng, dễ dàng tái sử dụng mã nguồn.
- **Xamarin:** Sử dụng C#. Tích hợp tốt với các dịch vụ .NET.

### So sánh

- **Hiệu suất:** Native > Flutter > React Native > Xamarin.
- **Tốc độ phát triển:** React Native > Flutter > Xamarin > Native.
- **Cộng đồng hỗ trợ:** React Native lớn nhất, Flutter đang phát triển nhanh.

## Câu 3: Flutter trở thành lựa chọn phổ biến

### Đặc điểm nổi bật

- **Giao diện đẹp:** Tạo ra UI mượt mà với widget tùy chỉnh.
- **Hiệu suất cao:** Biên dịch trực tiếp thành mã máy, giúp ứng dụng chạy nhanh.
- **Cộng đồng hỗ trợ:** Ngày càng lớn mạnh.

### So sánh với React Native và Xamarin

- **Flutter:**
  - Ngôn ngữ: Dart.
  - Tính năng: Hot reload, UI phong phú.
- **React Native:**
  - Ngôn ngữ: JavaScript.
  - Tính năng: Cộng đồng lớn, dễ học.
- **Xamarin:**

- Ngôn ngữ: C#.
- Tính năng: Tích hợp tốt với .NET, nhưng phức tạp hơn.

#### **Câu 4: Ngôn ngữ lập trình chính trên Android**

##### **1. Java**

- **Giải thích:** Ngôn ngữ chính thức từ khi Android ra đời, có cộng đồng mạnh mẽ.

##### **2. Kotlin**

- **Giải thích:** Ngôn ngữ hiện đại, hỗ trợ tính năng lập trình hàm, được Google công nhận chính thức.

##### **3. C++**

- **Giải thích:** Sử dụng cho các ứng dụng yêu cầu hiệu suất cao, như game.

#### **Câu 5: Ngôn ngữ lập trình chính trên iOS**

##### **1. Swift**

- **Giải thích:** Ngôn ngữ hiện đại, an toàn và nhanh chóng, thay thế Objective-C.

##### **2. Objective-C**

- **Giải thích:** Ngôn ngữ cũ hơn, vẫn được sử dụng trong các dự án lớn.

#### **Câu 6: Thách thức của Windows Phone**

- **Thị phần thấp:** Không thu hút được nhiều nhà phát triển ứng dụng.
- **Thiếu ứng dụng:** Nhiều ứng dụng phổ biến không có mặt trên Windows Phone.
- **Chậm cập nhật:** Không có chiến lược rõ ràng để cải thiện thị phần.

#### **Câu 7: Ngôn ngữ và công cụ phát triển ứng dụng web trên di động**

##### **Ngôn ngữ**

- **HTML/CSS/JavaScript:** Cơ bản cho mọi ứng dụng web.
- **TypeScript:** Phiên bản mở rộng của JavaScript, cung cấp tính năng kiểu tĩnh.

##### **Công cụ**

- **Frameworks:** React, Angular, Vue.js.
- **Responsive Design:** Bootstrap, Tailwind CSS.

#### **Câu 8: Nhu cầu nguồn nhân lực lập trình viên di động**

##### **Nhu cầu**

- **Tăng trưởng ổn định:** Ngành công nghiệp di động đang phát triển mạnh mẽ.

- **Kỹ năng phổ biến:**

- Kinh nghiệm với các framework Cross-Platform (Flutter, React Native).
- Kiến thức về UI/UX.
- Kỹ năng phát triển backend (Node.js, Python).

**Kỹ năng yêu cầu**

- **Lập trình:** Java, Kotlin, Swift.
- **Công cụ phát triển:** Git, Docker.
- **Kiến thức về API:** RESTful services, GraphQL.