

## Câu hỏi bài tập

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

#### 1. iOS

- Đặc điểm: Được phát triển bởi Apple, chỉ hoạt động trên các thiết bị của Apple như iPhone, iPad.
- Ưu điểm: Hệ sinh thái bảo mật cao, trải nghiệm người dùng mượt mà, chất lượng ứng dụng cao, hỗ trợ cập nhật thường xuyên.
- Khuyết điểm: Chi phí phát triển cao hơn do yêu cầu phần cứng đặc thù, quá trình xét duyệt ứng dụng nghiêm ngặt.

#### 2. Android

- Đặc điểm: Được phát triển bởi Google, mã nguồn mở, hoạt động trên nhiều loại thiết bị với các mức giá khác nhau.
- Ưu điểm: Phổ biến nhất thế giới, linh hoạt, dễ tùy chỉnh, hỗ trợ nhiều thiết bị và mức giá khác nhau.
- Khuyết điểm: Vấn đề phân mảnh (nhiều phiên bản Android chạy song song), bảo mật kém hơn iOS.

### 3. Windows Phone (đã ngừng phát triển)

- Đặc điểm: Được Microsoft phát triển cho thiết bị Lumia nhưng đã dừng hỗ trợ.
- Ưu điểm: Giao diện khác biệt, tích hợp tốt với các dịch vụ Microsoft.
- Khuyết điểm: Kho ứng dụng hạn chế, ngừng phát triển.

## **Câu 2: Các nền tảng phát triển ứng dụng di động phổ biến hiện nay**

### 1. Native Development (iOS sử dụng Swift/Objective-C, Android sử dụng Kotlin/Java)

- Ưu điểm: Hiệu suất cao, truy cập trực tiếp vào các tính năng phần cứng, hỗ trợ đầy đủ từ Apple và Google.
- Khuyết điểm: Cần phát triển riêng cho từng hệ điều hành, tốn kém thời gian và chi phí.

### 2. React Native

- Ưu điểm: Hỗ trợ đa nền tảng, phát triển nhanh, sử dụng JavaScript.
- Khuyết điểm: Hiệu suất thấp hơn so với native, hạn chế khi truy cập các tính năng phần cứng phức tạp.

### 3. Flutter

- Ưu điểm: Hiệu suất cao nhờ sử dụng ngôn ngữ Dart và kiến trúc Widget, đồng bộ giao diện trên các nền tảng.
- Khuyết điểm: Thư viện có thể thiếu so với native, ngôn ngữ Dart ít phổ biến.

### 4. Xamarin

- Ưu điểm: Dùng C#, phát triển cho cả Android và iOS với một codebase.
- Khuyết điểm: Hiệu suất thấp hơn native, gói thư viện không phong phú.

### **Câu 3: Điều gì làm cho Flutter trở thành lựa chọn phổ biến?**

- Tốc độ phát triển nhanh: Flutter cung cấp thư viện phong phú và khả năng xây dựng UI nhất quán cho các nền tảng.
- Hiệu suất cao: Flutter dùng Dart, kết hợp với engine mạnh giúp giảm độ trễ.
- So sánh với React Native và Xamarin:
  - React Native: Gần gũi với web developer, dùng JavaScript, nhưng có hạn chế về hiệu suất khi xử lý giao diện phức tạp.
  - Xamarin: Dùng C#, phù hợp với hệ sinh thái Microsoft, nhưng có các hạn chế về hiệu suất và tính nhất quán giao diện.

#### **Câu 4: Các ngôn ngữ lập trình chính để phát triển ứng dụng trên Android**

1. Java: Ngôn ngữ truyền thống, được dùng nhiều năm, hỗ trợ thư viện phong phú và cộng đồng lớn.
2. Kotlin: Được Google khuyến nghị, hiện đại, dễ học hơn Java, khả năng viết code gọn và dễ bảo trì

#### **Câu 5: Các ngôn ngữ lập trình chính để phát triển ứng dụng trên iOS**

##### **1. Objective-C**

- Đặc điểm: Là ngôn ngữ truyền thống, được Apple phát triển từ đầu để xây dựng ứng dụng trên iOS.
- Ưu điểm: Tích hợp tốt với các thư viện cũ và có sự hỗ trợ ổn định từ Apple.
- Khuyết điểm: Cú pháp phức tạp, khó học hơn cho người mới.

##### **2. Swift**

- Đặc điểm: Ngôn ngữ hiện đại của Apple, ra mắt vào năm 2014, được thiết kế để thay thế Objective-C.
- Ưu điểm: Cú pháp đơn giản, tốc độ nhanh, an toàn hơn với các lỗi phổ biến, tối ưu hiệu suất.

- Khuyết điểm: Các phiên bản đầu có sự thay đổi nhanh chóng, cần cập nhật thường xuyên, nhưng đã dần ổn định hơn.

## **Câu 6: Những thách thức của Windows Phone và nguyên nhân sụt giảm thị phần**

### **1. Thiếu ứng dụng và hỗ trợ từ nhà phát triển**

- Chi tiết: Hệ sinh thái ứng dụng của Windows Phone không phong phú như Android và iOS, do ít nhà phát triển hỗ trợ.

### **2. Không thu hút được người dùng**

- Chi tiết: Giao diện và trải nghiệm người dùng tuy khác biệt nhưng không phù hợp với phần đông người dùng. Các ứng dụng phổ biến thường không được cập nhật hoặc có trải nghiệm thấp hơn so với các nền tảng khác.

### **3. Thiếu sự hợp tác với các nhà sản xuất thiết bị**

- Chi tiết: Các thiết bị hỗ trợ Windows Phone chủ yếu là từ Nokia, khiến sự đa dạng và tính cạnh tranh của Windows Phone kém hơn so với Android.

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

### **1. Ngôn ngữ lập trình**

- HTML: Cung cấp cấu trúc cho các ứng dụng web, giúp tạo ra giao diện người dùng mượt mà.
- CSS: Định kiểu giao diện và hỗ trợ tối ưu hóa cho màn hình nhỏ.
- JavaScript: Giúp phát triển các chức năng và tính năng tương tác.

### **2. Framework và thư viện phổ biến**

- React.js và Vue.js: Các framework JavaScript hiện đại, giúp xây dựng ứng dụng web linh hoạt và hiệu suất cao.

### **3. Công cụ phát triển**

- Visual Studio Code: Trình biên tập mã phổ biến với nhiều tiện ích và plugin hỗ trợ.

## **Câu 8: Nhu cầu nguồn nhân lực lập trình viên trên thiết bị di động và kỹ năng yêu cầu**

### **1. Nhu cầu tuyển dụng**

- Chi tiết: Nhu cầu lập trình viên di động tiếp tục tăng do các doanh nghiệp đầu tư mạnh vào phát triển ứng dụng di động để cải thiện trải nghiệm người dùng.
- Các ngành nghề: Thương mại điện tử, tài chính, giáo dục, y tế đều cần lập trình viên di động để xây dựng ứng dụng phục vụ người dùng.

### **2. Kỹ năng cần thiết**

- Ngôn ngữ lập trình: Thành thạo Swift cho iOS, Kotlin cho Android là lợi thế lớn.
- Kiến thức UI/UX: Khả năng thiết kế giao diện người dùng, đảm bảo trải nghiệm mượt mà trên màn hình nhỏ.
- Kỹ năng phát triển đa nền tảng: Kiến thức về Flutter, React Native rất được ưa chuộng, giúp tiết kiệm chi phí phát triển.
- Kiến thức bảo mật: Quan trọng trong việc bảo vệ dữ liệu người dùng và ngăn ngừa tấn công