

Câu 1: Các nền tảng cho thiết bị di động thông minh hiện nay và đặc điểm của chúng

1. iOS (Apple)

- **Đặc điểm:** Nền tảng độc quyền của Apple, được sử dụng trên iPhone, iPad và iPod Touch.
- **Ưu điểm:** Hiệu suất mượt mà, an toàn, nhiều ứng dụng chất lượng cao, cập nhật phần mềm thường xuyên.
- **Nhược điểm:** Tính tùy biến hạn chế, phụ thuộc vào hệ sinh thái của Apple, giá cả thiết bị cao.

2. Android (Google)

- **Đặc điểm:** Nền tảng mã nguồn mở, phổ biến nhất thế giới, có thể cài đặt trên nhiều thiết bị của các hãng khác nhau.
- **Ưu điểm:** Tính tùy biến cao, nhiều lựa chọn thiết bị với các mức giá khác nhau, kho ứng dụng phong phú.
- **Nhược điểm:** Phân mảnh hệ điều hành, độ bảo mật có thể thấp hơn iOS, cập nhật chậm hơn do phụ thuộc vào nhà sản xuất thiết bị.

3. HarmonyOS (Huawei)

- **Đặc điểm:** Hệ điều hành mới của Huawei, chủ yếu phát triển để thay thế Android trên các thiết bị Huawei.
- **Ưu điểm:** Tương thích với nhiều thiết bị trong hệ sinh thái của Huawei, tối ưu hóa cho phần cứng của hãng.
- **Nhược điểm:** Kho ứng dụng còn hạn chế, chưa được sử dụng rộng rãi ngoài các thiết bị của Huawei.

4. KaiOS

- **Đặc điểm:** Một hệ điều hành dành cho điện thoại phổ thông, giúp chúng có các tính năng thông minh như 4G, WiFi, và GPS.

- **Ưu điểm:** Nhẹ, phù hợp với thiết bị giá rẻ, hỗ trợ nhiều tính năng thông minh trên điện thoại cơ bản.
- **Nhược điểm:** Tính năng hạn chế so với Android và iOS, không có nhiều ứng dụng so với các nền tảng phổ biến khác.

Câu 2: Các nền tảng phát triển ứng dụng di động phổ biến hiện nay và so sánh

1. Flutter

- **Ngôn ngữ:** Dart
- **Ưu điểm:** Hiệu suất nhanh, có giao diện thống nhất trên nhiều nền tảng, dễ tùy biến UI.
- **Nhược điểm:** Cộng đồng hỗ trợ chưa lớn bằng các nền tảng khác, chưa hỗ trợ hoàn toàn cho web.

2. React Native

- **Ngôn ngữ:** JavaScript
- **Ưu điểm:** Có cộng đồng lớn, dễ học đối với lập trình viên JavaScript, dễ tích hợp với các dịch vụ web.
- **Nhược điểm:** Hiệu suất không mượt mà như Flutter, khó khăn trong việc tạo giao diện phức tạp.

3. Xamarin

- **Ngôn ngữ:** C#
- **Ưu điểm:** Tích hợp tốt với các ứng dụng và dịch vụ của Microsoft, có thể dùng mã nguồn chung cho iOS và Android.
- **Nhược điểm:** Ứng dụng có dung lượng lớn, cần kỹ năng về hệ sinh thái Microsoft, cộng đồng hỗ trợ không quá lớn.

Câu 3: Lý do Flutter trở thành một lựa chọn phổ biến cho phát triển ứng dụng đa nền tảng

- **Hiệu suất cao:** Flutter cho phép tạo ứng dụng gốc với hiệu suất mượt mà vì nó không dựa vào lớp trung gian giữa mã nguồn và mã gốc.
- **Công cụ UI mạnh mẽ:** Flutter cung cấp một bộ công cụ UI (widgets) rất phong phú, có thể tùy biến dễ dàng cho cả iOS và Android.
- **Cập nhật hot reload:** Giúp lập trình viên nhìn thấy các thay đổi ngay lập tức mà không cần biên dịch lại toàn bộ ứng dụng.

So với **React Native** và **Xamarin**:

- Flutter có hiệu suất cao hơn React Native vì không cần lớp trung gian.
- So với Xamarin, Flutter dễ tiếp cận hơn với lập trình viên không làm việc trong hệ sinh thái của Microsoft, và nó cũng nhẹ hơn khi triển khai ứng dụng.

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

1. Java

- **Lý do chọn:** Java là ngôn ngữ chính thức đầu tiên của Android, có hỗ trợ mạnh từ Google, và có rất nhiều tài liệu, thư viện hỗ trợ.

2. Kotlin

- **Lý do chọn:** Kotlin hiện là ngôn ngữ chính thức của Android, thân thiện hơn với lập trình viên, cú pháp ngắn gọn và dễ đọc hơn so với Java.

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

1. Swift

- **Lý do chọn:** Swift là ngôn ngữ chính thức của Apple dành cho iOS và macOS, được phát triển để dễ tiếp cận, an toàn, và hiệu suất cao. Nó cung cấp cú pháp đơn giản, giúp giảm lỗi và tăng tốc độ phát triển.

2. Objective-C

- Lý do chọn: Đây là ngôn ngữ gốc của iOS, được sử dụng trước khi Swift ra đời. Objective-C vẫn được sử dụng cho các dự án cũ và có tính tương thích tốt với hệ sinh thái của Apple.

Câu 6: Thách thức của Windows Phone và nguyên nhân dẫn đến sự sụt giảm thị phần

1. **Thiếu ứng dụng:** Kho ứng dụng Windows Phone thiếu nhiều ứng dụng phổ biến và cập nhật, khiến người dùng cảm thấy hạn chế. Nhiều nhà phát triển không ưu tiên phát triển ứng dụng trên Windows Phone do lượng người dùng ít.
2. **Cạnh tranh khốc liệt từ iOS và Android:** Windows Phone khó cạnh tranh với hai nền tảng lớn này, cả về hệ sinh thái và tính năng. Android và iOS đã phát triển hệ sinh thái ứng dụng phong phú và liên kết chặt chẽ với các dịch vụ khác.
3. **Chiến lược không rõ ràng từ Microsoft:** Microsoft đã thay đổi chiến lược phát triển và bỏ cuộc sớm đối với Windows Phone, khiến hệ điều hành này thiếu đầu tư và mất niềm tin từ người dùng lẫn nhà phát triển.

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

1. **HTML, CSS và JavaScript:**
 - Đây là ba ngôn ngữ cốt lõi của lập trình web. HTML tạo cấu trúc trang, CSS giúp định dạng giao diện, và JavaScript xử lý logic và tương tác người dùng.
2. **Frameworks và Libraries:**
 - **React.js và React Native:** Được dùng để xây dựng giao diện web và ứng dụng đa nền tảng.
 - **Vue.js và Angular:** Các framework JavaScript phổ biến giúp quản lý dữ liệu, xây dựng giao diện động, và tương thích tốt với các dự án lớn.
3. **Progressive Web Apps (PWAs):**
 - PWAs là các trang web có thể cài đặt giống như ứng dụng, cung cấp trải nghiệm gần giống với ứng dụng gốc.
4. **Công cụ phát triển:**
 - **Ionic và Apache Cordova:** Các công cụ giúp lập trình viên tạo ứng dụng web và chuyển đổi thành ứng dụng gốc cho Android và iOS

Câu 8: Nhu cầu nhân lực lập trình viên di động hiện nay và kỹ năng cần thiết

1. Nhu cầu tuyển dụng:

- Công ty công nghệ, startup, fintech, và thương mại điện tử đều cần lập trình viên di động để xây dựng ứng dụng thân thiện, đáp ứng nhanh chóng nhu cầu của người dùng trên điện thoại.

2. Kỹ năng cần có:

- **Thành thạo ngôn ngữ chính:** Biết Swift (cho iOS) hoặc Kotlin (cho Android) là yêu cầu cơ bản.
- **Phát triển đa nền tảng:** Flutter và React Native là điểm cộng lớn vì giúp tạo ứng dụng chạy được trên cả iOS và Android, tiết kiệm thời gian và chi phí.
- **Hiểu UI/UX di động:** Kỹ năng này giúp thiết kế app dễ dùng, tối ưu trải nghiệm cho người dùng.
- **Tối ưu hiệu suất:** Khả năng viết mã hiệu quả, tiết kiệm pin và bộ nhớ trên thiết bị di động là điểm mạnh.
- **Cập nhật công nghệ mới:** Luôn học hỏi để nắm xu hướng công nghệ, không bị tụt hậu.