

Câu 1: Các nền tảng cho thiết bị di động thông minh hiện nay?
Hiện nay, có 2 nền tảng di động chính:

Android:

Đặc điểm: Hệ điều hành mã nguồn mở, phát triển bởi Google. Android chiếm thị phần lớn nhất trong thị trường di động.

Ưu điểm:

Mã nguồn mở, hỗ trợ nhiều loại phần cứng và nhà sản xuất.

Cộng đồng phát triển mạnh mẽ, thư viện phong phú.

Tính linh hoạt cao, có thể tùy biến nhiều tính năng.

Khuyết điểm:

Cập nhật phần mềm không đồng nhất giữa các thiết bị.

Vấn đề về bảo mật do mã nguồn mở.

Đa dạng phần cứng gây khó khăn trong tối ưu hóa ứng dụng.

iOS:

Đặc điểm: Hệ điều hành của Apple dành cho các thiết bị di động như iPhone, iPad, iPod.

Ưu điểm:

Quản lý phần mềm và phần cứng rất tốt, tối ưu hóa hiệu năng.

Bảo mật cao, ít bị tấn công từ bên ngoài.

Hệ sinh thái Apple đồng bộ rất tốt.

Khuyết điểm:

Hệ sinh thái đóng, giới hạn khả năng tùy biến.

Chi phí phát triển và phân phối ứng dụng cao (App Store).

Khó khăn trong việc tiếp cận thị trường vì phần cứng có hạn.

Câu 2: Các nền tảng phát triển ứng dụng di động phổ biến hiện nay và sự khác biệt chính giữa chúng?

Native Development (Phát triển ứng dụng gốc):

Android: Sử dụng Java hoặc Kotlin.

iOS: Sử dụng Swift hoặc Objective-C.

Ưu điểm: Tối ưu hiệu suất, quyền truy cập đầy đủ vào tính năng của hệ điều hành.

Nhược điểm: Phải viết mã riêng cho mỗi nền tảng, tốn nhiều thời gian và chi phí.

Cross-platform Development (Phát triển đa nền tảng):

React Native: Sử dụng JavaScript và React.

Flutter: Sử dụng Dart, phát triển bởi Google.

Xamarin: Sử dụng C# và .NET.

Ưu điểm: Phát triển một mã nguồn cho cả Android và iOS, tiết kiệm thời gian và chi phí.

Nhược điểm: Không tận dụng tối đa hiệu suất của từng nền tảng, đôi khi phải làm các điều chỉnh riêng cho từng hệ điều hành.

Progressive Web Apps (PWA):

Ưu điểm: Không cần phải cài đặt từ cửa hàng ứng dụng, dễ dàng tiếp cận qua trình duyệt.

Nhược điểm: Hạn chế về tính năng và hiệu suất so với ứng dụng gốc.

Câu 3: Điều gì làm cho Flutter trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng đa nền tảng? So sánh với các nền tảng khác như React Native và Xamarin.

Flutter:

Ưu điểm:

Hiệu suất cao: Flutter biên dịch trực tiếp sang mã máy, giúp ứng dụng chạy nhanh và mượt mà.

Cộng đồng và tài liệu mạnh mẽ: Được hỗ trợ mạnh mẽ bởi Google và cộng đồng mã nguồn mở.

Giao diện tùy chỉnh: Với Flutter, bạn có thể tạo giao diện đẹp và tùy chỉnh dễ dàng.

Khuyết điểm: Cần học ngôn ngữ Dart, hiện tại chưa phổ biến như JavaScript hay C#.

React Native:

Ưu điểm:

Cộng đồng rộng lớn: Được phát triển bởi Facebook, dễ dàng tìm tài liệu và hỗ trợ.

Dễ học: JavaScript là ngôn ngữ phổ biến, dễ dàng tiếp cận.

Khuyết điểm: Hiệu suất đôi khi không bằng ứng dụng gốc, cần phải viết mã native khi yêu cầu tính năng đặc biệt.

Xamarin:

Ưu điểm:

Cộng đồng Microsoft mạnh mẽ.

Chia sẻ mã giữa Android, iOS và Windows.

Khuyết điểm: Phụ thuộc vào .NET, cần kiến thức về C# và các công cụ của Microsoft, hiệu suất thấp hơn Flutter trong một số trường hợp.

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

Java:

Được sử dụng rộng rãi, dễ học, có cộng đồng lớn.

Tính tương thích ngược (backward compatibility) tốt.

Kotlin:

Được Google công nhận là ngôn ngữ chính cho phát triển Android.

An toàn hơn Java, ít lỗi hơn và dễ đọc.

Lý do chọn Kotlin và Java: Cả hai đều mạnh mẽ và phổ biến trong phát triển Android, Kotlin là lựa chọn ưu tiên nhờ tính năng hiện đại và tính tương thích với Java.

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

Swift:

Ngôn ngữ chính và hiện đại nhất do Apple phát triển, tối ưu cho phát triển iOS. An toàn, nhanh, dễ học và dễ duy trì.

Objective-C:

Ngôn ngữ cũ, vẫn được sử dụng trong nhiều dự án iOS hiện tại.

Khó học hơn Swift và có ít tính năng mới.

Lý do chọn Swift và Objective-C: Swift mang lại hiệu suất cao và tính an toàn, trong khi Objective-C vẫn còn được sử dụng rộng rãi trong các dự án cũ.

Câu 6: Thách thức mà Windows Phone phải đối mặt và nguyên nhân sụt giảm thị phần?

Thách thức:

Thiếu ứng dụng: Cửa hàng ứng dụng không đủ phong phú so với iOS và Android.

Hệ sinh thái đóng: Microsoft không cho phép các nhà phát triển tạo ứng dụng tùy chỉnh.

Thị phần quá nhỏ, dẫn đến ít người dùng và ít nhà phát triển.

Nguyên nhân sụt giảm:

Thiếu ứng dụng phổ biến: Các nhà phát triển không tập trung vào nền tảng này.

Chiến lược không rõ ràng: Microsoft thay đổi chiến lược quá nhanh chóng và không duy trì sự nhất quán.

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

HTML, CSS, JavaScript: Dùng để phát triển các ứng dụng web hoạt động trên trình duyệt di động.

Frameworks:

React, Angular, Vue.js: Các framework JavaScript giúp xây dựng các ứng dụng web động.

Ionic, Cordova: Các công cụ giúp phát triển ứng dụng di động web (PWA) với HTML, CSS và JavaScript.

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

Nhu cầu: Nhu cầu lập trình viên di động đang gia tăng mạnh mẽ, đặc biệt trong các công ty phát triển ứng dụng và các startup.

Kỹ năng yêu cầu:

Kỹ năng lập trình (Kotlin, Swift, React Native, Flutter).

Hiểu biết về UX/UI design.

Hiểu biết về các nền tảng phát triển ứng dụng (Android, iOS, Xamarin, Flutter).

Kiến thức về API và tích hợp với các dịch vụ bên ngoài.