

BÀI 1

TỔNG QUAN VỀ LẬP TRÌNH TRÊN THIẾT BỊ DI ĐỘNG

GV: ThS. BÙI PHÚ KHUYÊN



*** Cập nhật: 30.08.2025**

- ☐ Tổng quan về thiết bị di động thông minh
- ☐ Khái quát về lập trình đa nền tảng
- ☐ Các công nghệ phát triển ứng dụng di động đa nền tảng phổ biến hiện nay



Thiết bị điện tử xử lý thông tin, kết nối mạng và chạy ứng dụng phần mềm.

Các loại thiết bị:

- Điện thoại thông minh
- Máy tính bảng
- Thiết bị đeo thông minh
- Thiết bị kết nối thông minh khác



Thiết bị di động cá nhân có thể gọi điện, nhắn tin và chạy ứng dụng

- Hệ điều hành: Android, iOS
- Kết nối: Wi-Fi, Bluetooth, 4G/5G, GPS
- Cảm biến: Gia tốc kế, con quay hồi chuyển, cảm biến ánh sáng, cảm biến vân tay
- Camera: Chất lượng cao
- Màn hình cảm ứng: Đa điểm



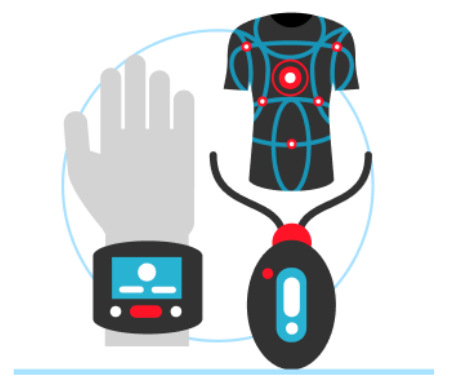
Thiết bị di động có màn hình cảm ứng lớn hơn điện thoại, thường từ 7 inch trở lên.

- Hệ điều hành: Android, iPadOS, Windows
- Màn hình: Từ 7 inch đến 12 inch hoặc hơn
- Pin: Dung lượng lớn hơn, thời gian sử dụng lâu hơn
- Phụ kiện: Bút cảm ứng, bàn phím rời



Các thiết bị đeo như đồng hồ thông minh, vòng tay sức khỏe

- Kết nối: Bluetooth, Wi-Fi, Cellular...
- Theo dõi sức khỏe: Nhịp tim, bước chân, giấc ngủ, calo...
- Thông báo: Tin nhắn, cuộc gọi, thông báo ứng dụng
- Khả năng chống nước
- Phục vụ giải trí



Android



Windows Phone



BlackBerry

symbian
OS

Hệ điều hành di động dựa trên nhân Linux, phát triển bởi Google

- Lịch sử phát triển: Thành lập 2003, ra mắt 2008, nhiều phiên bản cập nhật.
- Đặc điểm:
 - Mã nguồn mở
 - Google Play Store
 - Đa dạng thiết bị
 - Tùy biến cao



Baklava

Hệ điều hành di động của Apple cho iPhone, iPad

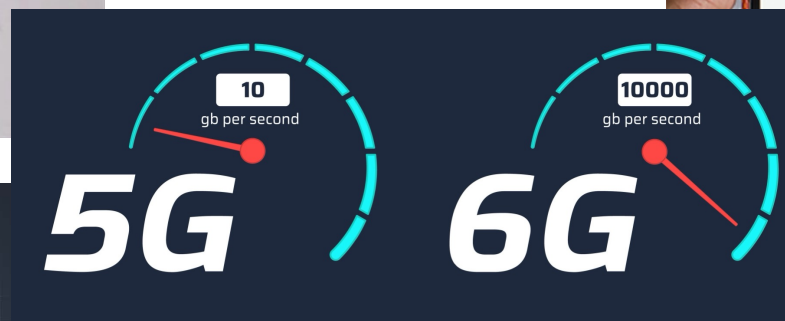
- Lịch sử phát triển: Ra mắt 2007, cập nhật liên tục.
- Đặc điểm:
 - Độc quyền
 - Appstore
 - Giao diện thân thiện
 - Bảo mật cao

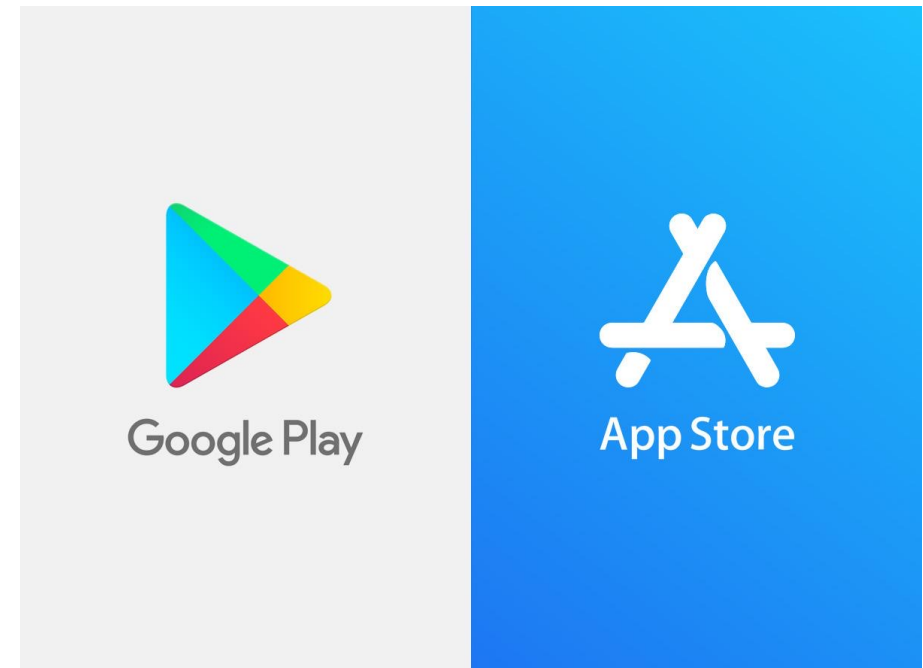


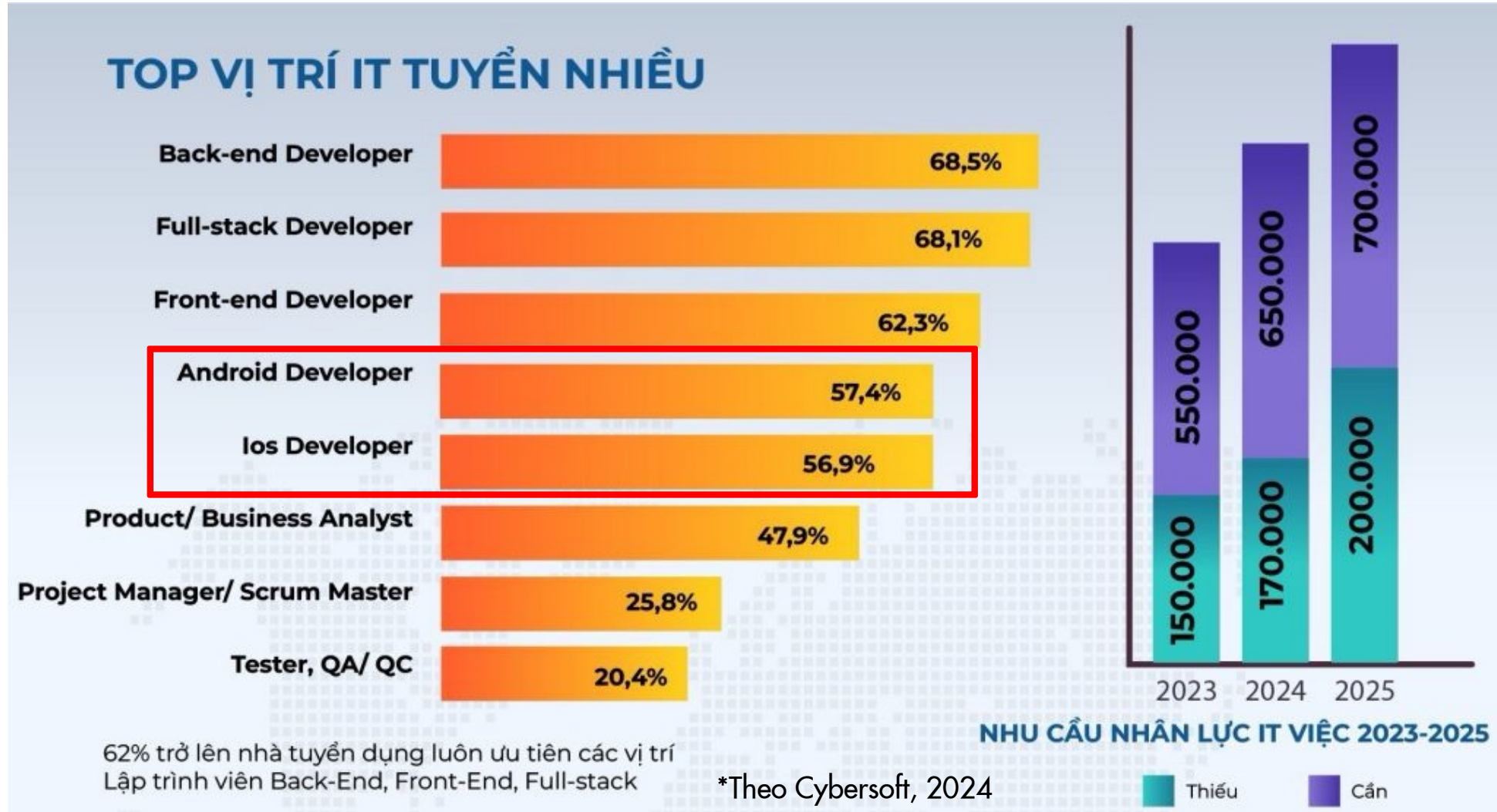
Hệ điều hành di động của Microsoft.

- Lịch sử phát triển: Ra mắt 2010, ngừng hỗ trợ 2017.
- Đặc điểm:
 - Giao diện Metro
 - Tích hợp Microsoft services
 - Đồng bộ hóa tốt







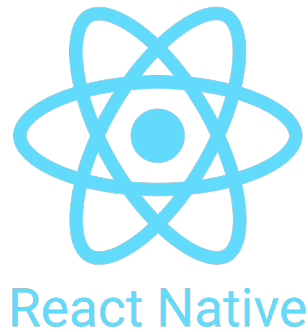




Lập trình Nguyên thủy (Native)



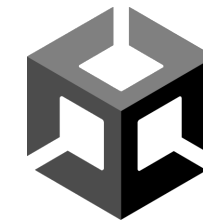
Lập trình Đa nền tảng (Cross Platform Mobile Apps)



Kotlin Multiplatform



APACHE
CORDOVA™



Unity

Lựa chọn công nghệ phù hợp?

Cảm ơn

Các bạn đã chú ý lắng nghe





Q&A