

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN 1
_____o0o_____



BÁO CÁO BÀI TẬP LỚN

**Chủ đề 5: Tìm hiểu chung về 2 họ hệ điều hành
MacOS và iOS**

**Môn học: Hệ điều hành
Số thứ tự nhóm: 4**

Nguyễn Đức Anh	MSV: B21DCAT031
Phạm Mai Chi	MSV: B21DCAT046
Nguyễn Mạnh Hùng	MSV: B21DCCN413
Vũ Quốc Huy	MSV: B21DCCN444
Lưu Trung Kiên	MSV: B21DCCN071
Phạm Công Minh	MSV: B21DCCN533
Phạm Văn Tiến	MSV: B21DCCN708

Giảng viên hướng dẫn: Đỗ Tiến Dũng

HÀ NỘI, 2023

Mục lục

1	Giới thiệu	2
1.1	Hệ điều hành iOS	2
1.2	Hệ điều hành macOS	4
2	Nội dung chính	6
2.1	Quản lý file	6
2.2	Quản lý bộ nhớ	9
2.3	Quản lý tiến trình	11
2.4	Các dịch vụ hệ điều hành cung cấp	16
2.5	Cấu trúc hệ thống	18
2.6	Các hàm Shell	21
2.7	Các đặc điểm hệ thống	23
3	Kết luận	25
4	Tài liệu tham khảo	26

1 Giới thiệu

1.1 Hệ điều hành iOS

Hệ điều hành iOS là gì?

Hệ điều hành iOS (trước đây là iPhoneOS) là hệ điều hành trên các thiết bị di động của Apple. Đây là hệ điều hành chạy trên các sản phẩm gồm iPhone, các mẫu iPad trước khi iPadOS được giới thiệu vào 2019, đồng thời cũng là hệ điều hành cho iPod Touch và là hệ điều hành phổ biến thứ 2 trên toàn cầu, sau Android của Google.

Khi sử dụng hệ điều hành này, người dùng được phép chạm, vuốt, thu, phóng đa cảm ứng trên màn hình điện thoại. Để nâng cao trải nghiệm của người dùng trong quá trình sử dụng, Apple luôn gửi đến người dùng các phiên bản mới nhất của hệ điều hành iOS.



Hình 1: Hệ điều hành iOS

Lịch sử hệ điều hành iOS

Lần đầu tiên iOS được ra mắt công chúng tại Hội nghị và Triển lãm Macworld diễn ra vào ngày 9 tháng 1 năm 2007. Trong tháng 9 năm đó cũng là thời điểm chiếc iPhone đầu tiên được phát hành. Lúc này, hệ điều hành này chưa có một cái tên riêng cụ thể mà nó chỉ là “iPhone OS”.



Hình 2: Steve Jobs giới thiệu chiếc Iphone đầu tiên tại triển lãm Macworld vào tháng 1/2007

Khoảng thời gian đầu, ứng dụng bên thứ ba không được hỗ trợ. Vào ngày 17 tháng 10 năm 2007, Apple công bố đang tiến hành xây dựng bộ công cụ phát triển phần mềm cho iPhone và sẽ đưa nó ra mắt để đến tay các nhà phát triển vào tháng 2 năm sau. Đến ngày 6 tháng 3 năm 2008, Apple đã cho ra mắt bản dùng thử đầu tiên, với một cái tên đầu tiên cho iOS lúc này là “iPhone OS”.

Tháng 6 năm 2010, Apple đổi tên iPhone OS thành iOS, tuy nhiên "iOS" đã được Cisco dùng để đặt tên cho hệ điều hành của mình trước đó. Để tránh các tranh chấp bản quyền trong tương lai, Apple đã xin giấy phép sử dụng thương hiệu iOS từ Cisco, điều này giống với cách Apple đã mua thương hiệu "iPhone" để sử dụng cho điện thoại thông minh của mình.



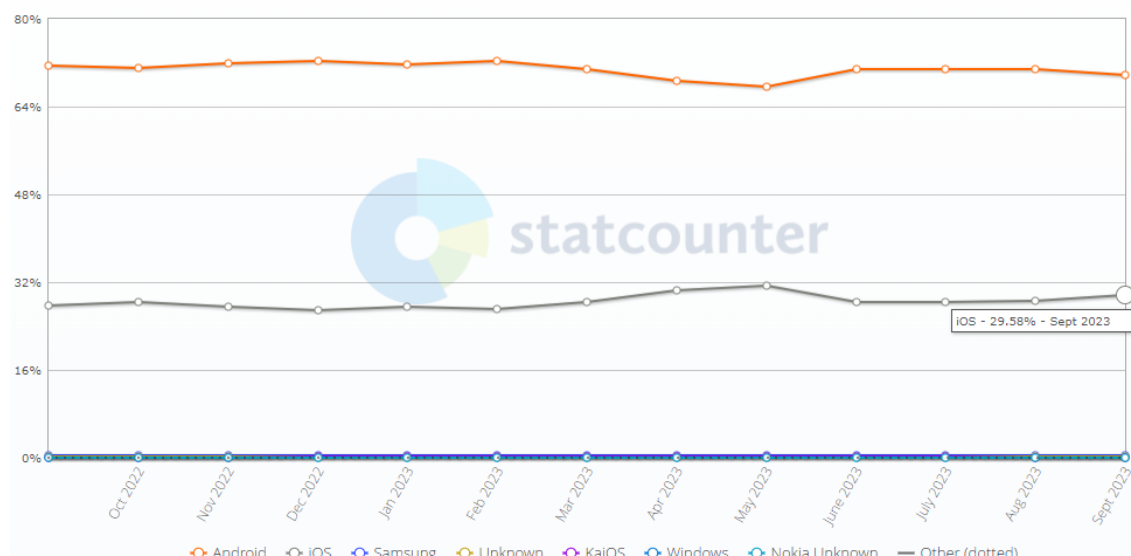
Hình 3: 17 phiên bản của hệ điều hành iOS từ năm 2007-2023

Tính đến năm 2023, phiên bản mới nhất của iOS là iOS 17 đã được Apple phát hành bản Beta đầu tiên vào ngày 7/6/2023 sau sự kiện WWDC.

Thống kê sử dụng

Theo dữ liệu từ trang web StatCounter, cho đến tháng 9 năm 2023 hệ điều hành iOS chiếm 29.58% thị phần hệ điều hành di động toàn cầu.

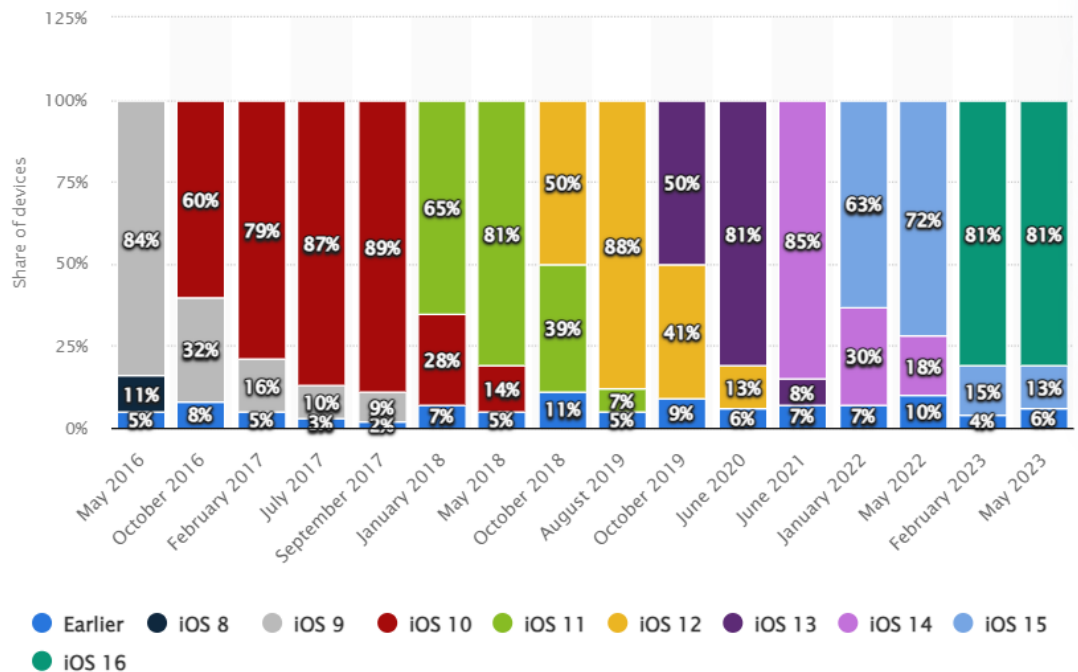
Mobile Operating System Market Share Worldwide Sept 2022 - Sept 2023



Hình 4: Thị phần hệ điều hành di động trên toàn thế giới từ tháng 9/2022 - 9/2023

Phiên bản iOS được sử dụng nhiều nhất: Vào tháng 5 năm 2023, hệ điều hành di

động mới iOS 16 đã được cài đặt trên 81% các thiết bị Apple truy cập vào App Store của Apple, trong khi phiên bản iOS 15 hay cũ hơn đang chạy trên 19% các thiết bị còn lại.



Hình 5: Tỷ lệ thiết bị Apple theo phiên bản iOS trên toàn thế giới từ năm 2016 - 2023

1.2 Hệ điều hành macOS

Hệ điều hành macOS là gì?

Hệ điều hành macOS (trước đây là OS X và ban đầu là Mac OS X) là hệ điều hành dựa trên Unix, được phát triển và phân phối bởi Apple Inc. kể từ 2001. Đây là hệ điều hành chính cho máy tính Mac. Trong thị trường PC và Laptop thì đây là hệ điều hành phổ biến thứ nhì, chỉ sau Windows.



Hình 6: Lễ ra mắt hệ điều hành macOS của Apple

Hệ điều hành macOS nổi tiếng bởi sự mượt mà, khả năng tương thích tuyệt vời đối với các sản phẩm nằm trong hệ sinh thái “Táo khuyết”. Đây cũng là nền tảng khép kín với giao diện người dùng tuyệt đẹp, được Apple liên tục cập nhật và tung ra nhiều bản nâng cấp mỗi năm.

Lịch sử hệ điều hành macOS

- OS X 10 beta: Kodiak - ngày 13 tháng 9 năm 2000.
- OS X 10.0: Cheetah - 24 March 2001.
- OS X 10.1: Puma - 25 tháng 9 năm 2001.
- OS X 10.2: Jaguar - 24 August 2002.
- OS X 10.3 Panther (Pinot) - 24 tháng 10 năm 2003.
- OS X 10.4: Tiger (Merlot) - 29 tháng 4 năm 2005.
- OS X 10.4.4: Tiger (Chardonnay).
- OS X 10.5: Leopard (Chablis) - 26 tháng 10 năm 2007.
- OS X 10.6 Snow Leopard - 28 tháng 8 năm 2009.
- OS X 10.7: Lion (Barolo) - 20 July 2011.
- OS X 10.8: Mountain Lion (Zinfandel) - 25 tháng 7 năm 2012.
- OS X 10.9: Mavericks (Cabernet) - ngày 22 tháng 10 năm 2013.
- **OS X 10.10: Yosemite (Syrah) - 16 tháng 10 năm 2014.**
- OS X 10.11: El Capitan (Gala) - 30 tháng 9 năm 2015.
- macOS 10.12: Sierra (Fuji) - ngày 20 tháng 9 năm 2016.
- **macOS 10.13: High Sierra (Lobo) - 25 tháng 9 năm 2017.**
- macOS 10.14: Mojave (Liberty) - 24 tháng 9 năm 2018.
- macOS 10.15: Catalina (Jazz) - ngày 7 tháng 10 năm 2019.
- macOS 11: Big Sur (GoldenGate) – 12 tháng 11 năm 2020.
- macOS 12: Monterey (Star) – 25 tháng 10 năm 2021.
- macOS 13: Ventura (Rome) – 24 tháng 10 năm 2022.

Hình 7: Dòng thời gian của hệ điều hành macOS

Hệ điều hành macOS là hệ điều hành kế nhiệm của Mac OS cổ điển, cũng là hệ điều hành Mac với 9 phiên bản từ 1984 cho đến 1999. Trong quãng thời gian đó, người đồng sáng lập Apple là Steve Jobs đã rời công ty và khởi nghiệp công ty của ông, NeXT, và phát triển hệ điều hành NeXTSTEP mà sau này Apple mua lại để hình thành nền tảng của macOS.

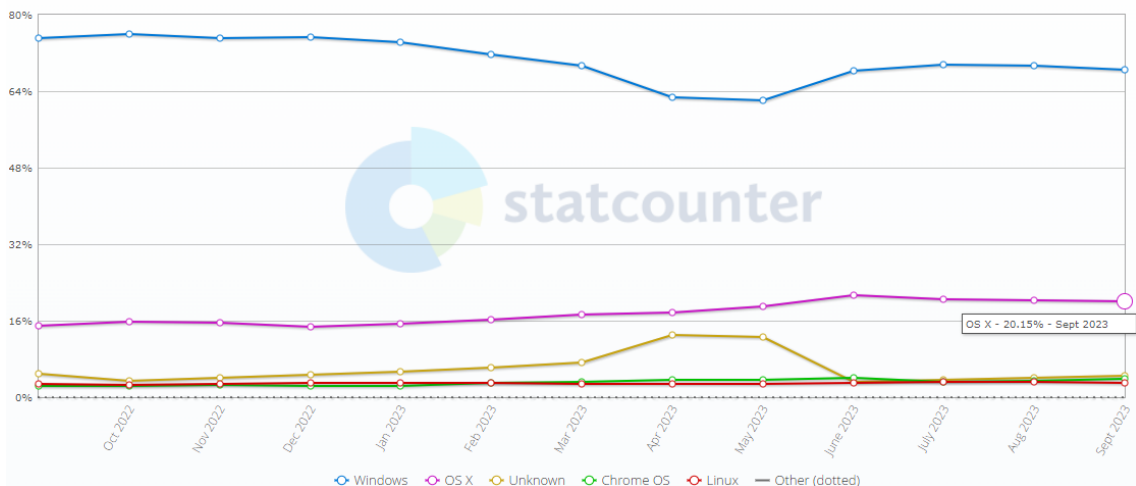
Sự nổi bật trong tên ban đầu của macOS chính là việc sử dụng số La Mã X ("ten") trong Mac OS X hay iPhone X, đi kèm với đặt tên mã các phiên bản theo các loài mèo lớn, hay đặt theo các thắng cảnh ở California. Apple rút ngắn tên hệ điều hành thành OS X vào 2011, rồi đặt lại tên thành macOS để đi liền với tên của các hệ điều hành khác của Apple, iOS, iPadOS, tvOS, và watchOS. Sau 16 bản macOS 10 khác nhau, macOS Big Sur được đề là bản 11 vào 2020, tiếp tục lên đến bản hiện tại, macOS 13.

Hệ điều hành macOS đã hỗ trợ tới ba loại kiến trúc khác nhau của các bộ vi xử lý, bắt đầu từ PowerPC năm 1999. Vào năm 2006, Apple chuyển sang bộ vi xử lý của Intel, từ đó ra mắt loạt máy Mac dựa trên Intel. Năm 2020, Apple chuyển tiếp sang Apple Silicon, sử dụng bộ vi xử lý M1 dựa trên ARM 64-bit được thiết kế bởi Apple trên các dòng máy Mac hiện tại.

Thống kê sử dụng

Theo dữ liệu từ trang web StatCounter, cho đến tháng 9 năm 2023 hệ điều hành macOS chiếm 20.15% thị phần hệ điều hành trên máy tính toàn cầu.

Desktop Operating System Market Share Worldwide Sept 2022 - Sept 2023



Hình 8: Thị phần hệ điều hành máy tính trên toàn thế giới từ tháng 9/2022 - 9/2023

2 Nội dung chính

2.1 Quản lý file

2.1.1 Hệ điều hành iOS

Hệ điều hành iOS của Apple sử dụng một hệ thống tệp tin dựa trên Unix, cụ thể là một phiên bản thay thế cho hệ thống tệp HFS+ được gọi là APFS (Apple File System). APFS được thiết kế để tối ưu hóa cho lưu trữ flash/SSD và cải thiện mã hóa dữ liệu.



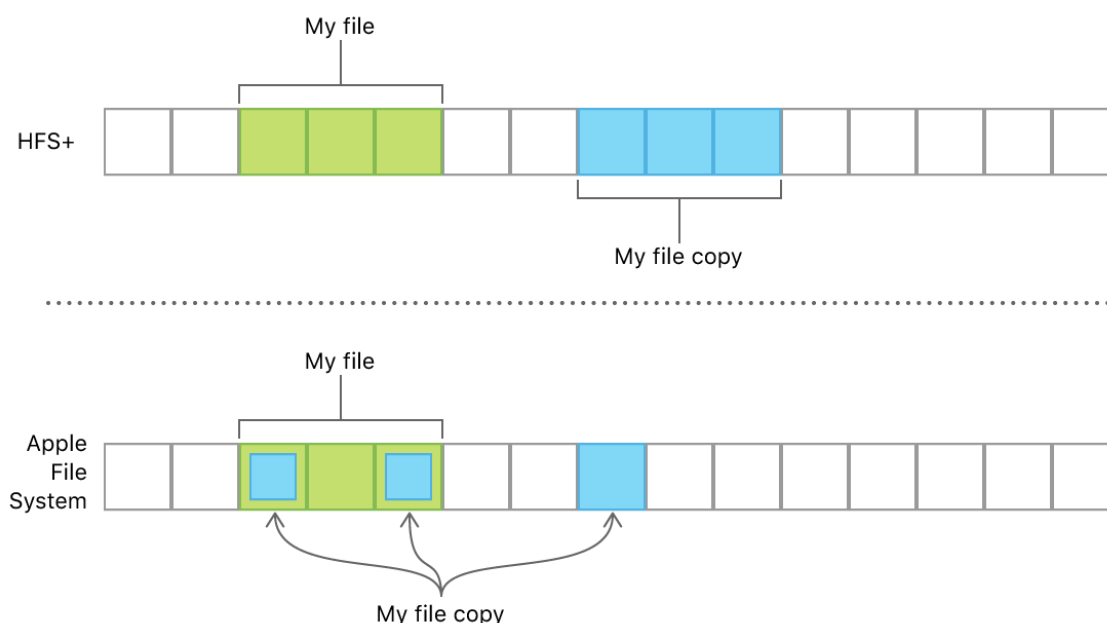
Hình 9: APFS (Apple File System)

APFS hỗ trợ inode 64bit. Inode là một cấu trúc dữ liệu trong hệ thống tệp kiểu Unix mô tả và xác định một đối tượng hệ thống tệp như tệp hoặc thư mục. Với 64bit inode, APFS có thể chứa khoảng 9 tỉ đối tượng hệ thống tệp. APFS còn đi kèm với các tính năng tiêu biểu như:

- *Tạo bản sao cho tệp và thư mục (Clones):* Cho phép sao chép tệp gần như ngay lập tức mà không cần sử dụng thêm dung lượng. Bản sao sẽ tham chiếu tệp gốc, chia sẻ các khối dữ liệu giống hệt nhau giữa hai tệp. Thực hiện các thay đổi đối với một tệp thì chỉ khối dữ liệu bị thay đổi mới được ghi vào bản sao mới, trong khi cả bản gốc và bản sao vẫn tiếp tục chia sẻ các khối dữ liệu

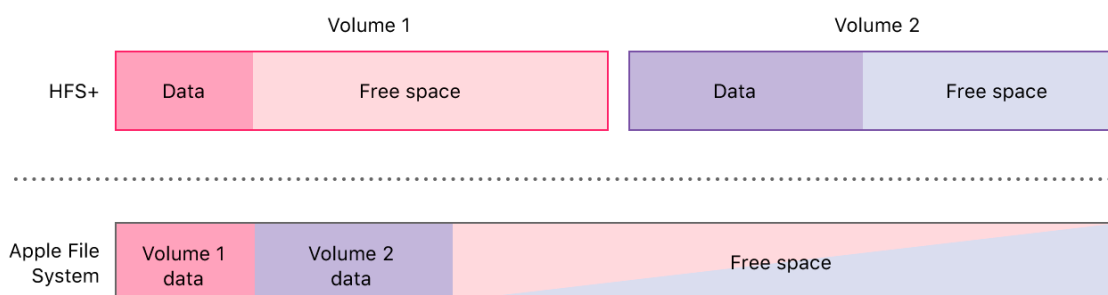
không thay đổi. Điều này không chỉ giúp việc sao chép và lưu tập tin cực kỳ nhanh chóng mà còn tiết kiệm dung lượng lưu trữ cần thiết.

Hình bên dưới là ví dụ hiển thị một tệp có tên “My file” và bản sao của nó “My file copy” có hai khối chung và một khối khác nhau giữa chúng. Trên các hệ thống tệp như HFS+ sẽ cần ba khối trên đĩa, nhưng trên ổ đĩa của hệ thống tệp APFS, hai khối chung sẽ được chia sẻ.



Hình 10: Clones

- *Snapshots*: APFS có thể tạo bản sao lưu của cả phân vùng ở chế độ read only. Snapshot tạo điều kiện sao lưu hiệu quả và cho phép bạn phục hồi lại (revision) trạng thái mọi thứ tại một thời điểm cụ thể trước đó.
- *Chia sẻ không gian (Space Sharing)*: Hỗ trợ nhiều ổ đĩa trong một phân vùng duy nhất, điều này cho phép tất cả các ổ đĩa đó chia sẻ không gian trống của chúng. Chia sẻ không gian cho phép nhiều ổ đĩa có thể tăng lên và thu nhỏ linh hoạt khi cần mà không cần phân vùng lại.



Hình 11: Space Sharing

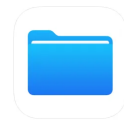
- *Sao chép khi ghi (Copy on Write)*: Cơ chế bảo vệ dữ liệu này giúp giảm thiểu nguy cơ lỗi trong quá trình sửa đổi tệp. Sau khi yêu cầu thay đổi (ghi), một bản sao duy nhất mới sẽ được tạo, đảm bảo bản gốc được giữ nguyên. Chỉ sau khi ghi xong, thông tin tệp mới được cập nhật để trở đến dữ liệu mới nhất.

- *Atomic Safe-Save*: Tương tự như sao chép khi ghi nhưng áp dụng cho mọi thao tác với tệp, chẳng hạn như đổi tên hoặc di chuyển tệp hoặc thư mục.

Hệ thống tệp được tổ chức theo một cấu trúc thư mục phân cấp. Mỗi ứng dụng có một thư mục riêng biệt trong hệ thống tệp, và các ứng dụng chỉ có quyền truy cập vào thư mục của chính nó. Điều này giúp tăng cường bảo mật và ngăn chặn việc truy cập không được phép vào dữ liệu của các ứng dụng khác.

Tuy nhiên, người dùng không thể truy cập trực tiếp vào hệ thống tệp của iOS như họ có thể làm với các hệ điều hành khác như Android hoặc Windows, thay vào đó, Apple cung cấp ứng dụng Files, cho phép người dùng xem và quản lý tệp tin được lưu trữ trên thiết bị của họ và trong iCloud Drive.

Ứng dụng Files cho phép người dùng tổ chức tệp tin của họ vào các thư mục, đánh dấu các tệp tin quan trọng và khả năng tìm kiếm tệp tin. Nó cũng tích hợp với các dịch vụ lưu trữ đám mây khác như Dropbox và Google Drive, cho phép người dùng quản lý tất cả các tệp tin của họ từ một nơi duy nhất.



Hình 12: Logo của Files

2.1.2 Hệ điều hành macOS

Tương tự với iOS, macOS cũng sử dụng hệ thống tệp APFS (từ trước bản 10.13 macOS sử dụng HFS+) và được tổ chức theo một cấu trúc thư mục phân cấp.

Người dùng có thể xem và quản lý các tệp tin bằng ứng dụng Finder. Ứng dụng Finder cho phép người dùng tổ chức tệp tin của họ vào các thư mục, đánh dấu các tệp tin quan trọng, và khả năng tìm kiếm tệp tin. Nó cũng tích hợp với các dịch vụ lưu trữ đám mây khác như Dropbox và Google Drive, cho phép người dùng quản lý tất cả các tệp tin của họ từ một nơi duy nhất.



Hình 13: Logo của Finder

2.1.3 So sánh

Giống nhau

- *Ứng dụng quản lý file*: Cả iOS và macOS đều có ứng dụng quản lý để quản lý tệp tin và thư mục của riêng mình, đó là Files (Tệp) trên iOS và Finder trên macOS.
- *iCloud Drive*: Cả hai hệ điều hành tích hợp với iCloud Drive, cho phép lưu trữ và chia sẻ tệp tin trực tuyến qua đám mây iCloud.
- *AirDrop*: Cả hai hệ điều hành đều hỗ trợ tính năng AirDrop để chia sẻ tệp tin giữa các thiết bị Apple.
- *Hệ thống tệp*: Ở những phiên bản mới nhất hiện nay của hai hệ điều hành iOS và macOS đều sử dụng hệ thống tệp tin mặc định là APFS (Apple File System).

Khác nhau

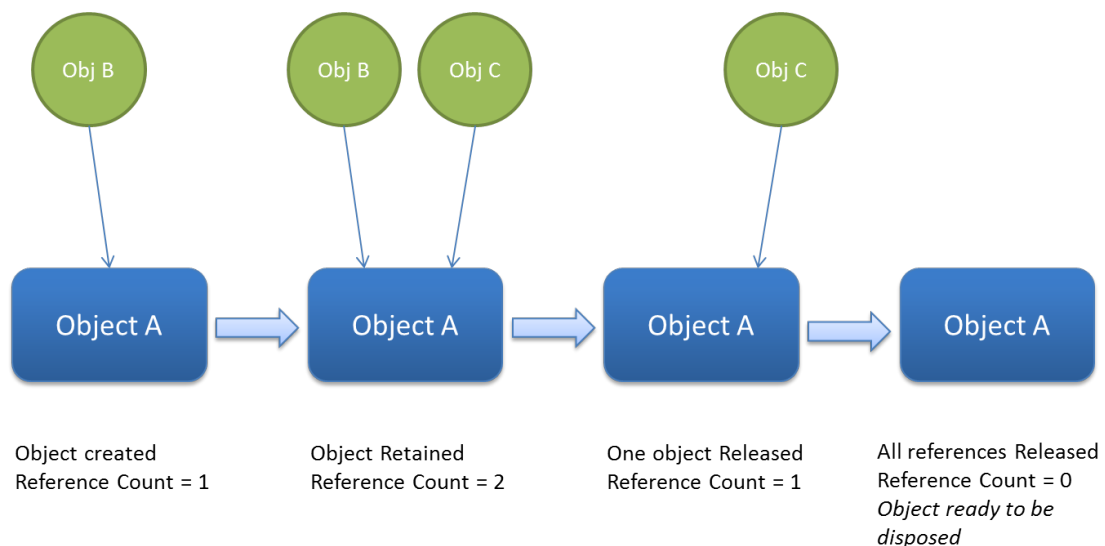
- *Giao diện*: Giao diện quản lý tệp trên iOS được thiết kế dành cho màn hình cảm ứng, trong khi Finder trên macOS dành cho máy tính cá nhân.
- *Quyền truy cập*: macOS cho phép người dùng có quyền cao hơn để truy cập và quản lý các tệp, thư mục trên hệ thống, trong khi iOS có thể hạn chế quyền truy cập hơn đối với ứng dụng và thư mục cụ thể.
- *Ứng dụng tiêu biểu*: iOS được sử dụng trên thiết bị di động như iPhone còn macOS dành cho các sản phẩm máy tính, vì vậy nên một số ứng dụng và tính năng quản lý tệp có thể khác nhau để phù hợp với nhu cầu sử dụng của từng nền tảng.

2.2 Quản lý bộ nhớ

2.2.1 Hệ điều hành iOS

Quản lý bộ nhớ trong iOS được dựa trên một hệ thống được gọi là ARC (Automatic Reference Counting). Đây là hệ thống được viết bằng Swift hoặc Objective - C, là kỹ thuật quản lý bộ nhớ tự động vì hệ thống có thể lần đầu và quản lý sự tham chiếu tới các đối tượng trong bộ nhớ.

ARC hoạt động bằng cách đếm số lượng tham chiếu tới một đối tượng. Khi một đối tượng được tạo ra, bộ đếm của ARC bằng 1. Mỗi khi có một đối tượng khác tham chiếu tới nó, bộ đếm tham chiếu tăng lên. Khi bộ đếm tham chiếu về 0, đối tượng được giải phóng khỏi bộ nhớ.



Hình 14: ARC

Khi bộ nhớ gần đầy, các phần của bộ nhớ mà chưa được sử dụng sẽ được viết ra đĩa để dành chỗ cho dữ liệu được cần ngay bây giờ. Phần đĩa chứa dữ liệu không được sử dụng được gọi là bộ nhớ phụ (backing store). Nếu lượng bộ nhớ trống xuống thấp ở một mức nhất định, hệ thống sẽ yêu cầu các ứng dụng đang chạy giải phóng bộ nhớ, nếu không thì các ứng dụng đó sẽ bị buộc phải dừng lại.

iOS có khả năng cung cấp thông tin cho người dùng biết hiện tại dung lượng còn

trông trong máy là bao nhiêu. Điều này giúp cho người dùng nắm bắt được tình trạng bộ nhớ của thiết bị di động.

Khi sử dụng ứng dụng này và chuyển sang ứng dụng khác, iOS cho phép các ứng dụng được thoát ra chạy ngầm trên RAM. iOS có thể cho phép người dùng quay lại ứng dụng đó để tiếp tục sử dụng, tạm dừng hoặc kết thúc các ứng dụng đó nếu cần giải phóng bộ nhớ cho ứng dụng hoạt động chính (giống Background App Refresh).

iOS có khả năng phát hiện khi thiếu bộ nhớ và gửi cảnh báo "Low Memory" cho các ứng dụng. Ứng dụng có thể phản ứng bằng cách giảm sử dụng bộ nhớ và tải lại dữ liệu cần thiết theo nhu cầu để giải phóng bộ nhớ.

iOS sử dụng cơ chế phân trang (paging) để quản lý bộ nhớ ảo. Khi bộ nhớ vật lý gần cạn kiệt, iOS có thể ghi dữ liệu không cần thiết lên bộ nhớ ảo để giải phóng bộ nhớ vật lý cho các ứng dụng quan trọng.

2.2.2 Hệ điều hành macOS

Quản lý bộ nhớ trong macOS cũng được dựa trên hệ thống ARC tương tự như hệ điều hành iOS.

MacOS chứa hệ thống bộ nhớ ảo tích hợp đầy đủ (fully – integrated virtual memory) được khởi động sẵn. Để tiến trình được nạp vào bộ nhớ ảo này, macOS sử dụng đĩa cứng để chứa dữ liệu mà hiện tại chưa được sử dụng. Nếu bộ nhớ đầy, các phần của bộ nhớ mà không được sử dụng sẽ được ghi ra đĩa để dành chỗ cho dữ liệu cần thiết. Phần đĩa chứa dữ liệu chưa được sử dụng gọi là bộ nhớ phụ (backing store).

Một số phương thức được macOS sử dụng để quản lý bộ nhớ:

- *Phân vùng đổi (Swap Space)*: MacOS sử dụng phân vùng đổi (swap space) trên ổ đĩa để làm bộ nhớ đích (swap out) khi cần thiết để giải phóng bộ nhớ vật lý. Dữ liệu được trao đổi giữa bộ nhớ và phân vùng đổi khi bộ nhớ vật lý cạn kiệt.
- *Nén bộ nhớ (Memory Compression)*: MacOS sử dụng kỹ thuật nén bộ nhớ (Memory Compression) để giảm lượng bộ nhớ vật lý được hệ điều hành sử dụng, duy trì dữ liệu dưới dạng nén thay vì ghi trực tiếp lên đĩa, giúp tạo thêm vùng nhớ trống cho các chương trình ứng dụng của người dùng.
- *Áp lực bộ nhớ (Memory Pressure)*: MacOS theo dõi áp lực bộ nhớ và có thể thực hiện các biện pháp tùy thuộc vào mức độ áp lực. Các ứng dụng có thể bị tắt hoặc dời lên đĩa khi cần thiết để giảm áp lực bộ nhớ. Hệ thống sử dụng các “heuristics” để xác định kích thước bộ nhớ cần và xác định các đối tượng hoặc dữ liệu nào có thể được di chuyển ra khỏi bộ nhớ vật lý.
- *Tối ưu hoá sử dụng bộ nhớ (Optimizing Memory Usage)*: Các ứng dụng macOS cần tối ưu hóa việc sử dụng bộ nhớ bằng cách giảm tải dữ liệu không cần thiết, giải phóng tài nguyên không sử dụng và tránh làm bộ nhớ rò rỉ (memory leaks).

2.2.3 So sánh

Sự giống và khác nhau về phương thức quản lý bộ nhớ, bộ nhớ ảo giữa hệ điều hành iOS và macOS có một số điểm chính:

Giống nhau:

- Cả hai hệ điều hành đều quản lý bộ nhớ theo nguyên lý ARC.
- Cả hai hệ điều hành đều sử dụng bộ nhớ ảo để mở rộng bộ nhớ vật lý và sử dụng cơ chế phân trang.

Khác nhau:

- Máy tính Mac sử dụng hệ điều hành macOS thường có dung lượng bộ nhớ lớn, giúp macOS quản lý nhiều ứng dụng cùng lúc mà không gặp áp lực bộ nhớ cao, nhờ vậy mà macOS có khả năng quản lý bộ nhớ hệ thống rộng rãi hơn và cung cấp các công cụ và phương pháp cho ứng dụng để quản lý tài nguyên hệ thống, bao gồm quản lý bộ nhớ, CPU, và các tài nguyên phần cứng khác. Trong khi đó, các thiết bị di động như iPhone và iPad có hạn chế về dung lượng bộ nhớ vật lý, dẫn đến việc iOS phải thực hiện việc quản lý tài nguyên bộ nhớ nghiêm ngặt hơn để đảm bảo hiệu suất và tính ổn định trên các thiết bị di động có tài nguyên hạn chế.
- MacOS hỗ trợ bộ nhớ phụ, iOS lại không hỗ trợ cơ chế này. Nếu lượng bộ nhớ trống trong iOS xuống thấp ở một mức nhất định, hệ thống sẽ yêu cầu các ứng dụng đang chạy giải phóng bộ nhớ, nếu không thì các ứng dụng đó sẽ bị buộc dừng lại.
- MacOS cho phép thực hiện đa nhiệm nhiều chương trình ứng dụng cùng lúc, trong khi đó iOS cũng cho thực hiện đa nhiệm, nhưng một chương trình phải tạm ngưng lại để chương trình kia được chạy.

2.3 Quản lý tiến trình

2.3.1 Hệ điều hành iOS

Cách tiếp cận để quản lý các tiến trình trong iOS là thông qua việc sử dụng các lệnh gọi hệ thống.

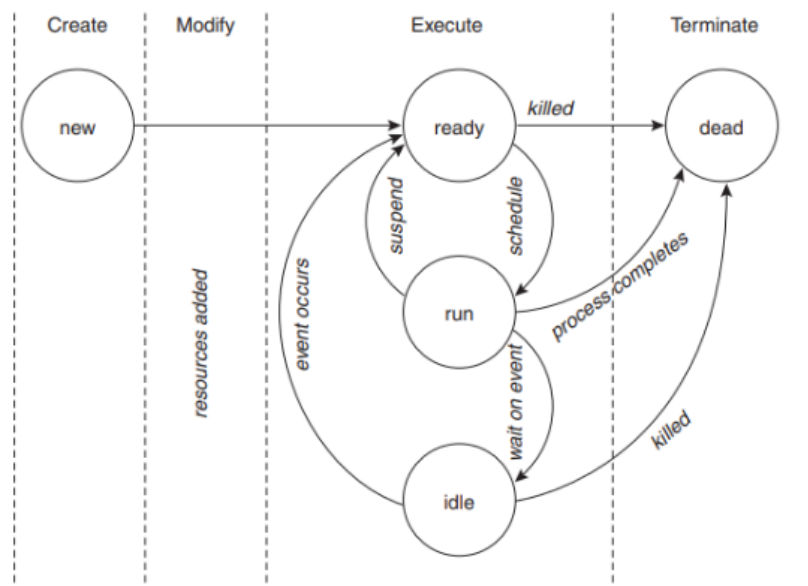
Các yếu tố liên quan đến quản lý tiến trình

- *Lập lịch trình*: Nhiệm vụ của bộ lập lịch/điều phối tiến trình là chọn một tác vụ từ các tác vụ được lưu trữ và tải nó vào bộ nhớ một cách tuần tự. Để phản hồi nhanh, iOS sử dụng bộ lập lịch tiến trình ngắn hạn. Bộ lập lịch ngắn hạn chịu trách nhiệm chính trong việc chọn tiến trình nào đã được tải vào bộ nhớ sẽ được thực thi tiếp theo tùy thuộc vào ngắt xung nhịp CPU, ngắt thiết bị I/O hoặc lệnh gọi hệ thống.
- *Điều phối*: Mục đích chính của bộ điều phối trong iOS là kiểm soát các tiến trình được CPU thực thi. Bộ điều phối trong iOS phải cực nhanh để tạo điều kiện chuyển đổi tiến trình nhanh.

Các thuật toán quản lý tiến trình

- *First In First Out*: Sử dụng phương pháp đến trước được phục vụ trước (FCFS) để phân bổ tài nguyên cho các luồng và tiến trình. Thủ tục xếp hàng dựa trên thời gian đến của tiến trình trong hàng đợi sẵn sàng.
- *Shortest remaining time*: Thuật toán lập lịch này sử dụng phương pháp Shortest Job First (SJF). Bộ lập lịch đặt các tiến trình yêu cầu ít thời gian CPU nhất tiếp theo vào hàng đợi sẵn sàng.
- *Round Robin algorithm*: Theo cách tiếp cận này, hệ điều hành chỉ định một lượng thời gian cố định cho mỗi tiến trình và quay vòng qua hàng đợi. Thuật toán quay vòng phát sinh nhiều chi phí trong trường hợp đơn vị thời gian cố định nhỏ. Nó thể hiện thông lượng cân bằng liên quan đến cả SJF và FCFS. Tình trạng bỏ đói được loại bỏ trong Round Robin vì có sự ưu tiên của các tiến trình dựa trên thứ hạng ưu tiên, thời gian xử lý hoặc thời gian đến.
- *Multi-level queue algorithm*: Thuật toán lập lịch tiến trình này được triển khai trong bối cảnh theo đó các tiến trình được phân tách thành các nhóm theo mức độ ưu tiên.

Các giai đoạn của một tiến trình



Hình 15: Các giai đoạn của một tiến trình iOS

- *Giai đoạn khởi tạo (Create)*: Khi một tiến trình mới được tạo, nó sẽ nhận được vùng ngăn xếp của chính nó và chuyển sang trạng thái mới. Các tiến trình sau đó có thể chuyển sang giai đoạn sửa đổi. Nếu không cần sửa đổi, chuyển sang giai đoạn thực hiện.
- *Giai đoạn sửa đổi (Modify)*: Không giống như hầu hết các hệ điều hành, iOS không tự động chuyển các tham số khởi động hoặc gán bảng điều khiển cho một tiến trình mới khi nó được tạo, bởi vì nó giả định hầu hết các tiến trình

không cần những tài nguyên này. Nếu một tiến trình cần nguồn tài nguyên này, luồng mà tạo nó có thể điều chỉnh để thêm vào.

- **Giai đoạn thực hiện (Execute):** Sau khi một tiến trình mới được tạo và sửa đổi thành công, nó sẽ chuyển sang trạng thái sẵn sàng và bước vào giai đoạn thực hiện. Trong giai đoạn này, một tiến trình có thể truy cập vào CPU và chạy. Trong giai đoạn này, một tiến trình có thể ở một trong ba trạng thái: sẵn sàng, đang chạy hoặc không hoạt động.
Một tiến trình ở trạng thái sẵn sàng đang đợi đến lượt truy cập vào CPU và bắt đầu thực thi các lệnh.
Một tiến trình ở trạng thái chạy sẽ được điều khiển bởi CPU và đang tích cực thực thi các lệnh.
Một tiến trình nhàn rỗi đang ở trạng thái ngủ, chờ đợi các sự kiện bên ngoài xảy ra trước khi nó có đủ điều kiện để chạy.
Một tiến trình chuyển từ trạng thái sẵn sàng sang trạng thái chạy khi nó được lên lịch chạy. Với đa nhiệm không được ưu tiên, một tiến trình đã được lên lịch sẽ tiếp tục chạy trên CPU cho đến khi nó tạm dừng hoặc chấm dứt.
Một tiến trình có thể tạm dừng theo một trong hai cách. Nó có thể tự tạm dừng một cách rõ ràng bằng cách báo với nhân Kernel của hệ điều hành rằng nó muốn từ bỏ CPU, chuyển sang trạng thái sẵn sàng và đợi đến lượt chạy tiếp theo. Tiến trình cũng có thể tạm dừng bằng cách chờ đợi một sự kiện bên ngoài xảy ra. Khi một tiến trình bắt đầu chờ đợi một sự kiện, Kernel sẽ ngừng tạm dừng nó bằng cách chuyển nó sang trạng thái không hoạt động, nơi nó sẽ tồn tại cho đến khi sự kiện xảy ra. Sau khi sự kiện xảy ra, Kernel sẽ chuyển tiến trình trở lại trạng thái sẵn sàng để chờ đến lượt chạy tiếp theo.
- **Giai đoạn kết thúc (Terminate):** Một tiến trình bước vào giai đoạn kết thúc khi nó hoàn thành chức năng của mình và tắt (được gọi là tự chấm dứt) hoặc khi một tiến trình khác chấm dứt nó. Khi một tiến trình bị hủy hoặc tự chấm dứt, tiến trình đó sẽ chuyển sang trạng thái chết. Một tiến trình đã kết thúc vẫn ở trạng thái chết, không hoạt động cho đến khi Kernel lấy lại được tất cả tài nguyên của nó. Kernel cũng có thể ghi lại số liệu thống kê về ngăn xếp của tiến trình khi nó kết thúc. Sau khi tài nguyên của nó được lấy lại, tiến trình đã kết thúc sẽ chuyển ra khỏi trạng thái chết và bị xóa hoàn toàn khỏi hệ thống.

Độ ưu tiên tiến trình

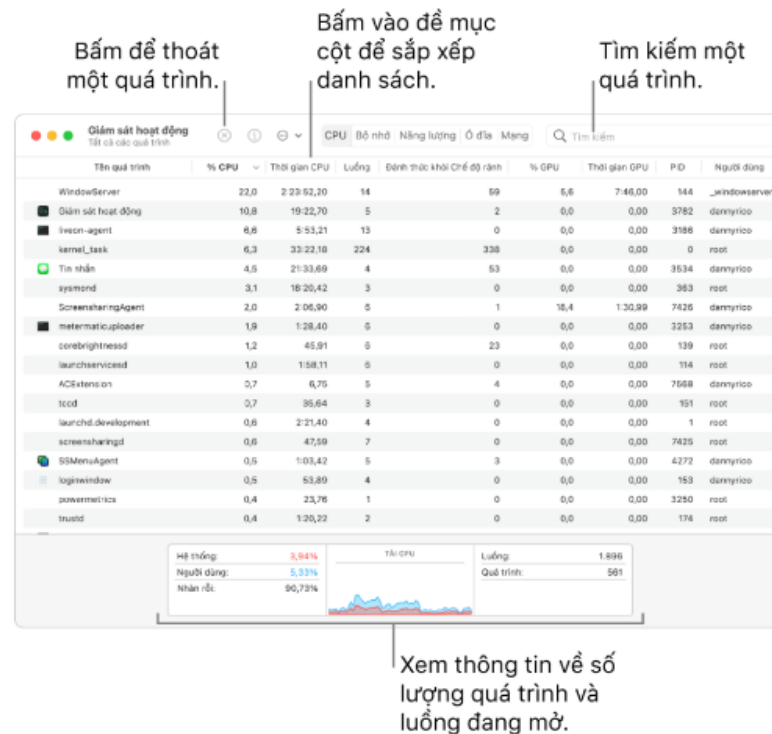
- **Quan trọng:** Dành riêng cho các tiến trình hệ thống thiết yếu nhằm giải quyết các vấn đề phân bổ tài nguyên.
- **Cao:** Được gán cho các tiến trình cung cấp thời gian phản hồi nhanh, chẳng hạn như tiến trình nhận các gói trực tiếp từ giao diện mạng.
- **Trung bình:** Mức độ ưu tiên mặc định được hầu hết các tiến trình sử dụng.
- **Thấp:** Được gán cho các tiến trình cung cấp các tác vụ nền định kỳ, chẳng hạn như ghi nhật ký tin nhắn.

Lưu ý: Tiến trình có độ ưu tiên cao hơn sẽ có nhiều cơ hội hơn để truy cập CPU và không thể làm gián đoạn tiến trình có mức ưu tiên thấp hơn.

2.3.2 Hệ điều hành macOS

Cả macOS và iOS đều là hệ điều hành do Apple Inc. tạo ra nên cách quản lý tiến trình của hai hệ điều hành tương tự nhau. Tuy nhiên, chúng được tạo ra cho các loại thiết bị khác nhau. MacOS có hệ thống đa nhiệm mạnh mẽ giúp người dùng dễ dàng chuyển đổi giữa nhiều cửa sổ và ứng dụng cùng một lúc. iOS cũng có hệ thống đa nhiệm, nhưng vì được tạo ra cho thiết bị di động nên nó bị hạn chế so với macOS.

Trong macOS, người dùng có thể theo dõi hoạt động của các tiến trình thông qua ứng dụng giám sát hoạt động (Activity Monitor).



Hình 16: Theo dõi tiến trình qua Activity Monitor

Thông tin về các quá trình trong ứng dụng Giám sát hoạt động:

- *Tất cả quá trình:* Hiển thị tất cả các quá trình đang chạy trên máy Mac.
- *Tất cả các quá trình, phân cấp:* Hiển thị các quá trình thuộc về các quá trình khác, do đó bạn có thể xem mối quan hệ cấp trên/cấp dưới giữa chúng.
- *Quá trình của tôi:* Hiển thị các quá trình do tài khoản người dùng của bạn sở hữu.
- *Quá trình Hệ thống:* Hiển thị các quá trình do macOS sở hữu.
- *Quá trình khác của Người dùng:* Hiển thị các quá trình không do người dùng gốc hoặc người dùng hiện tại sở hữu.
- *Quá trình Đang hoạt động:* Hiển thị các quá trình đang chạy mà không ở chế độ ngủ.
- *Quá trình Không hoạt động:* Hiển thị các quá trình đang chạy mà ở chế độ

ngủ.

- *Quá trình GPU*: Hiển thị các quá trình đang chạy được sở hữu bởi GPU của máy tính.
- *Quá trình Có cửa sổ*: Hiển thị các quá trình có thể tạo cửa sổ. Những quá trình này thường là các ứng dụng.
- *Quá trình Đã chọn*: Chỉ hiển thị các quá trình bạn đã chọn trong cửa sổ Giám sát hoạt động.
- *Ứng dụng trong 12 giờ qua*: Chỉ hiển thị các ứng dụng đang chạy các quá trình trong 12 giờ qua.
- *Các quá trình, bởi GPU*: Hiển thị các quá trình GPU đang chạy được nhóm theo GPU.

2.3.3 So sánh

	macOS	iOS
Quản lý tiến trình	Tiến trình được quản lý trong môi trường đa nhiệm đầy đủ, với khả năng chạy nhiều ứng dụng đồng thời. Activity Monitor là công cụ được sử dụng để theo dõi và quản lý các tiến trình trên macOS.	Các ứng dụng chạy trong môi trường "sandbox", có hạn chế quyền truy cập vào hệ thống và tương tác với các tiến trình khác. Người dùng có thể xem danh sách các ứng dụng đang chạy và đóng chúng thông qua App Switcher.
Quản lý bộ nhớ khi thực hiện tiến trình	Người dùng có kiểm soát lớn hơn về quản lý bộ nhớ. Có thể sử dụng Activity Monitor để xem và kiểm soát sử dụng bộ nhớ của các tiến trình.	Hệ thống quản lý bộ nhớ tự động, và ứng dụng có thể bị kết thúc tự động khi hệ thống cần giải phóng bộ nhớ.
Điều khiển tiến trình	Người dùng có thể đóng ứng dụng thông qua cửa sổ hoặc bằng cách sử dụng tổ hợp phím hoặc menu. Có khả năng chấm dứt tiến trình không mong muốn thông qua Terminal hoặc Activity Monitor.	Người dùng có khả năng đóng ứng dụng thông qua App Switcher, nhưng không có cách chính thức để "kết thúc" một ứng dụng hoàn toàn ngoại trừ việc khởi động lại thiết bị.

Theo dõi hiệu suất	Activity Monitor cung cấp thông tin chi tiết về tài nguyên hệ thống và hiệu suất của các tiến trình.	Có hạn chế về khả năng theo dõi chi tiết về hiệu suất của ứng dụng trên thiết bị. Có thể sử dụng Instruments để phân tích hiệu suất trong môi trường phát triển.
--------------------	--	--

Tóm lại, quản lý tiến trình trên iOS và macOS phản ánh sự khác biệt giữa môi trường di động và máy tính cá nhân. Trong khi iOS tập trung vào việc giữ cho hệ thống ổn định và hiệu quả trên các thiết bị có tài nguyên hạn chế, macOS mang lại sự linh hoạt lớn hơn trong việc quản lý và kiểm soát tiến trình.

2.4 Các dịch vụ hệ điều hành cung cấp

iOS và macOS là hai hệ điều hành thuộc cùng hệ sinh thái Apple, được phát triển dựa trên cùng một nền tảng. Do đó, hai hệ điều hành này có một số dịch vụ chung.

App Store

- *App Store* là một nền tảng phân phối ứng dụng của Apple: Cho phép người dùng tải xuống và cài đặt ứng dụng trên các thiết bị của họ như iPhone, iPad và Mac.
- *Đa dạng ứng dụng*: App Store chứa hàng triệu ứng dụng đa dạng từ trò chơi đến các ứng dụng làm việc và học tập.
- *Mục đích của Apple khi tạo ra App Store*: Apple hướng đến mục tiêu cung cấp một nền tảng an toàn, dễ truy cập để người dùng có thể khám phá và tải về ứng dụng một cách thuận tiện.



Hình 17: Logo của AppStore

Apple Books

- *Apple Books* là dịch vụ sách điện tử của Apple, cho phép người dùng mua, tải xuống và đọc sách trên các thiết bị của họ như iPhone, iPad, và Mac. Nó cung cấp một thư viện rộng lớn các sách điện tử từ nhiều thể loại khác nhau, bao gồm cả tiểu thuyết, sách kỹ năng, sách giáo dục, và nhiều thể loại khác.
- *Chức năng chính của Apple Books*: cung cấp một trải nghiệm đọc sách số thuận tiện, với các tính năng như làm dấu trang, tìm kiếm, và chế độ đọc ban đêm. Nó cũng tích hợp với các dịch vụ khác của Apple, như iCloud, để đồng bộ hóa nội dung giữa các thiết bị.
- *Mục đích của Apple khi tạo ra Apple Books*: mở rộng trải nghiệm người dùng bằng cách cung cấp một nền tảng tiện lợi để truy cập và thưởng thức sách điện tử, tạo điều kiện cho người dùng tiếp cận với nhiều loại thông tin và giải trí khác nhau.



Hình 18: Logo của Apple Books

FaceTimes

- *FaceTime* là một dịch vụ video call của Apple, cho phép người dùng thực hiện cuộc gọi video giữa các thiết bị của họ như iPhone, iPad, Mac và Apple Watch.
- *Chức năng chính của FaceTime*: kết nối người dùng qua video, tạo ra trải nghiệm giao tiếp trực tuyến chất lượng cao.
- *Mục đích của Apple khi tạo ra FaceTimes*: tạo ra một phương tiện giao tiếp cá nhân và chất lượng cao trên các thiết bị của họ, đồng thời cung cấp trải nghiệm gọi video để sử dụng và tích hợp chặt chẽ với hệ sinh thái Apple.



Hình 19: Logo của FaceTimes

Find My

- *Find My* là một dịch vụ của Apple, cung cấp khả năng theo dõi và định vị vị trí của các thiết bị thuộc hệ sinh thái Apple như iPhone, iPad, Mac, Apple Watch và các thiết bị khác.
- *Chức năng chính của Find My*: giúp người dùng tìm lại và bảo vệ thiết bị của họ khi chúng bị mất hoặc bị đánh cắp.
- *Mục đích của Apple khi tạo ra Find My*: cung cấp một công cụ an toàn và hiệu quả để bảo vệ và quản lý thiết bị của người dùng. Bằng cách tích hợp chặt chẽ với hệ sinh thái của Apple, Find My đóng vai trò quan trọng trong việc tăng cường trải nghiệm người dùng và bảo vệ thông tin cá nhân.



Hình 20: Logo của Find My

Apple Podcasts

- *Apple Podcasts* là một ứng dụng của Apple, được thiết kế để nghe và quản lý các chương trình podcast trên các thiết bị của họ như iPhone, iPad, Mac và Apple Watch.
- *Chức năng chính của Apple Podcasts*: được thiết kế để nghe và quản lý các chương trình podcast trên các thiết bị của họ như iPhone, iPad, Mac, và Apple Watch.
- *Mục đích của Apple khi tạo ra Apple Podcasts*: mở rộng trải nghiệm giải trí của người dùng, đồng thời cung cấp một nền tảng thuận tiện để khám phá và tiếp cận nội dung podcast đa dạng từ trên toàn thế giới. Đây là một phần của mục tiêu tổng thể của Apple là tạo ra các dịch vụ và ứng dụng tích hợp chặt chẽ trong hệ sinh thái của mình.



Hình 21: Logo của Apple Podcasts

Safari

- *Safari* là trình duyệt web do Apple phát triển và tích hợp sẵn trên hầu hết các thiết bị của họ, bao gồm iPhone, iPad, Mac và Apple Watch. Trình duyệt này tích hợp nhiều tính năng bảo mật, bao gồm cả công nghệ Intelligent Tracking Prevention để ngăn chặn theo dõi người dùng qua các trang web.
- *Chức năng chính của Safari*: cung cấp một trải nghiệm duyệt web nhanh chóng, an toàn và hiệu quả trên các thiết bị của hệ sinh thái Apple.
- *Mục đích của Apple khi tạo ra Safari*: mang đến một trình duyệt web tích hợp chặt chẽ với hệ sinh thái của mình, tối ưu hóa hiệu suất và bảo mật trên các thiết bị của Apple. Safari cũng được thiết kế để tối ưu hóa thời lượng pin trên các thiết bị di động và tích hợp nhiều tính năng tiện ích như đồng bộ lịch sử trình duyệt qua iCloud giữa các thiết bị.



Hình 22: Logo của Safari

iCloud

- *iCloud* là một dịch vụ đám mây của Apple, được tích hợp chặt chẽ với hệ sinh thái của họ, bao gồm iPhone, iPad, Mac, Apple Watch và nhiều thiết bị khác.
- *Chức năng chính của iCloud*: lưu trữ dữ liệu trên đám mây giúp giảm bớt áp lực lưu trữ trực tiếp trên thiết bị và tạo ra một phương tiện để truy cập dữ liệu từ bất kỳ nơi nào có kết nối internet; đồng bộ hoá thông tin giữa các thiết bị giúp người dùng duy trì một trạng thái đồng nhất trên nhiều thiết bị và giảm thiểu việc nhập lại thông tin, sao lưu thiết bị giúp bảo vệ dữ liệu quan trọng của người dùng, bao gồm cả hình ảnh, ứng dụng và cài đặt.
- *Mục đích của Apple khi tạo ra iCloud*: tạo ra một môi trường tích hợp, linh hoạt và an toàn để người dùng lưu trữ, quản lý và đồng bộ hóa dữ liệu của họ qua các thiết bị. iCloud đóng vai trò quan trọng trong việc kết nối và tối ưu hóa trải nghiệm người dùng trong hệ sinh thái Apple.



Hình 23: Logo của iCloud

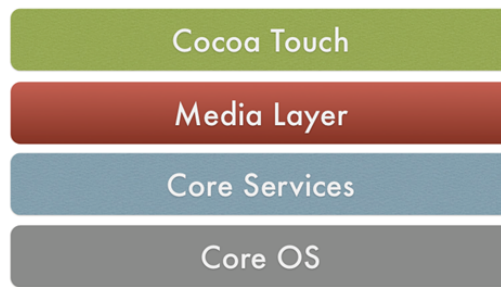
Ngoài ra, vì hai hệ điều hành này định hướng cho hai kiểu người dùng khác nhau đó là macOS cho người dùng máy tính và iOS cho người dùng điện thoại di động, nên chúng có một số dịch vụ riêng biệt hiện tại chỉ tồn tại trên một hệ điều hành. Ví dụ, macOS đã trang bị Time Machine, một công cụ sao lưu mạnh mẽ, giúp bảo vệ dữ liệu hay Universal Control, một tính năng cho phép người dùng điều khiển đồng thời nhiều iPad và máy Mac bằng một bộ chuột và bàn phím,...

2.5 Cấu trúc hệ thống

2.5.1 Cấu trúc hệ thống của Hệ điều hành iOS

Cấu trúc hệ thống của iOS được chia thành bốn cấp độ:

- **Lớp Hệ điều hành lõi (Core OS)** Lớp Core OS là lớp cơ sở của hệ thống, cung cấp các dịch vụ thấp nhất như quản lý bộ nhớ, quản lý tiến trình, quản lý năng lượng, quản lý thiết bị và quản lý bảo mật.
- **Lớp Core Services** Lớp Core Services là lớp cung cấp các dịch vụ cơ bản cho

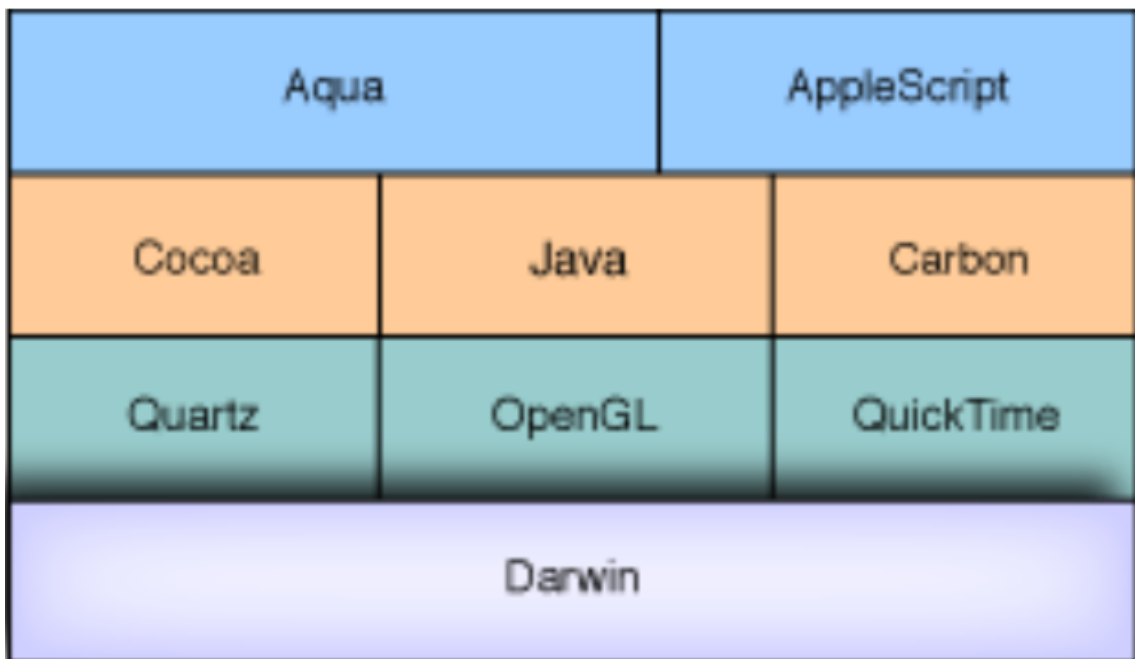


Hình 24: Cấu trúc hệ thống iOS

các ứng dụng, như quản lý tập tin, quản lý mạng, quản lý luồng, quản lý sự kiện và quản lý dữ liệu.

- Lớp Media Lớp Media là lớp cung cấp các dịch vụ liên quan đến đa phương tiện, như âm thanh, video, hình ảnh và đồ họa.
- Lớp Cocoa Touch Lớp Cocoa Touch là lớp cao nhất của hệ thống, cung cấp các giao diện người dùng và các khung năng lực cho các ứng dụng. Lớp này bao gồm các thành phần như UIKit, Foundation, MapKit, GameKit và SiriKit.

2.5.2 Cấu trúc hệ thống của Hệ điều hành macOS



Hình 25: Cấu trúc hệ thống macOS

Cấu trúc hệ thống MacOS là một hệ điều hành chạy trên mọi máy tính của nhà Apple, nó chứa một số lượng lớn các chương trình được thiết kế tốt. Cấu trúc MacOS được cấu tạo bởi nhiều lớp khác nhau:

- Lớp cơ bản của hệ thống đó là Darwin – core Unix.
Mach là phần chính của core Darwin và thực hiện các hoạt động như: Sử dụng bộ nhớ, đường truyền dữ liệu,... Core OS cũng là một nguồn mở, điều này

đồng nghĩa với việc bất cứ ai cũng có thể lấy được mã nguồn của nó và chỉnh sửa nó. Các phiên bản khác nhau của Darwin có thể được sử dụng để nâng cao MacOS. Một số tính năng nổi bật của Darwin đó chính là bảo vệ bộ nhớ, quản lý bộ nhớ tự động, bộ nhớ ảo nâng cao, ưu tiên đa nhiệm, ... Ngoài ra nó cũng cung cấp các dịch vụ input output cho MacOS và có hỗ trợ plug-and-play, hot-swapping và quản lý năng lượng.

- Lớp tiếp theo là hệ thống đồ họa bao gồm: Quartz, Open-GL và QuickTime. Trong MacOS thì hệ thống con đồ họa chứa ba phần chính đó là: Quartz, OpenGL và QuickTime. Mỗi bộ phận này đều sẽ đảm nhận các chức năng và nhiệm vụ khác nhau. Trong đó, Quartz sẽ đảm nhận nhiệm vụ là quản lý đồ họa 2D trong hệ thống con đồ họa, ngoài ra nó còn cung cấp phông chữ, kiểu chữ, hiển thị hình ảnh, đồ họa giao diện,... OpenGL cung cấp, hỗ trợ đồ họa 3D trong hệ thống như ánh xạ kết cấu, độ trong suốt, khử răng cưa, hiệu ứng khí quyển, hiệu ứng đặc biệt, ... Ngoài ra, nó cũng có thể được sử dụng trong các hệ thống Unix và Windows. QuickTime được sử dụng thường xuyên trong các phương tiện kỹ thuật số khác nhau như video kỹ thuật số, truyền phát video, âm thanh,... Nó cũng cho phép sử dụng các ứng dụng sáng tạo trong hệ thống như là iMovie, iTunes,...
- Lớp ứng dụng được tạo thành bởi bốn thành phần chính đó là: Classic, Carbon, Cocoa và Java. Hệ thống con ứng dụng trong MacOS cung cấp môi trường cổ điển (Classic Environment) để có thể chạy các ứng dụng cổ điển. Ba môi trường phát triển ứng dụng có sẵn có thể kể đến như là: Carbon, Cocoa và Java. Môi trường cổ điển sẽ đảm bảo rằng các ứng dụng được viết cho các phiên bản trước của hệ điều hành có thể chạy một cách mượt mà và trơn tru. Môi trường Carbon được sử dụng để chuyển các ứng dụng hiện đang có sang giao diện mới là giao diện chương trình ứng dụng Carbon. Quá trình này được gọi là Carbon hóa ứng dụng. Còn về môi trường Cocoa, nó cung cấp môi trường phát triển ứng dụng hướng đối tượng. Các ứng dụng Cocoa sẽ sử dụng các lợi ích của Cấu trúc MacOS là nhiều nhất. Các ứng dụng Java và Java Applet có thể chạy bằng môi trường Java.
- Lớp giao diện người dùng nằm ở trên cùng đó là lớp Aqua. Giao diện người dùng của MacOS còn được gọi là Aqua. Nó cung cấp, hỗ trợ các tính năng trực quan tốt cũng như các công cụ có chức năng tùy chỉnh giao diện người dùng phù hợp theo yêu cầu, sở thích của người dùng. Aqua sử dụng rất nhiều màu sắc và kết cấu khác nhau, các biểu tượng của nó cực kỳ chi tiết, chân thực, bắt mắt, thú vị và hiệu quả để sử dụng.

2.5.3 So sánh

- Hệ điều hành macOS và iOS đều được phát triển bởi Apple Inc. Tuy nhiên, chúng có cấu trúc khác nhau. macOS là một hệ điều hành dựa trên Unix, được thiết kế đặc biệt để tương thích với các thiết bị trong hệ sinh thái của Apple như MacBook, iMac, Mac Pro và Mac Mini. Trong khi đó, iOS là một hệ điều hành chạy trên thiết bị di động của Apple là iPhone.

- Cấu trúc của macOS và iOS khác nhau vì chúng được thiết kế để phù hợp với các thiết bị khác nhau, cơ bản nhất là phần giao diện. Với macOS, ta có màn hình desktop và thanh taskbar. Với iOS lại có bố cục khác do màn hình có kích thước nhỏ hơn và được điều khiển bằng cảm ứng. iOS có giao diện dựa trên ứng dụng với màn hình chính hiển thị các biểu tượng ứng dụng. Người dùng có thể tương tác với iOS bằng các thao tác như vuốt, chạm, thu, phóng đa cảm ứng trên màn hình. Tuy nhiên, cả hai đều có các thành phần chính như tiến trình, nhân, bộ đệm gói, trình điều khiển thiết bị và phần mềm chuyển mạch nhanh.

2.6 Các hàm Shell

2.6.1 Hệ điều hành iOS

Hệ điều hành iOS, được phát triển bởi Apple, không cho phép người dùng truy cập trực tiếp vào môi trường dòng lệnh (command-line shell) như các hệ điều hành Unix/Linux hoặc hệ điều hành Android. iOS được thiết kế để cung cấp một môi trường ứng dụng đóng và an toàn hơn, với sự kiểm soát chặt chẽ hơn việc truy cập hệ thống.

Tuy nhiên, có một số ứng dụng từ bên thứ ba có thể cung cấp môi trường shell trên các thiết bị đã jailbreak. Jailbreaking là một quá trình không được hỗ trợ bởi Apple, và nó mở khóa hệ thống để cho phép truy cập vào các chức năng mà Apple ban đầu không cho phép. Jailbreaking có thể gây ra các vấn đề bảo mật và vận hành không ổn định, và nó có thể vi phạm các điều khoản của Apple. Sau khi thiết bị iOS của bạn đã được jailbreak, bạn có thể sử dụng một số ứng dụng (MobileTerminal, NewTerm. . .) để gọi tới hệ thống và sử dụng môi trường shell.

2.6.2 Hệ điều hành macOS

Về mặt lịch sử, shell mặc định cho macOS là Bash. Nhưng với sự ra đời của macOS Catalina, Apple đã thay đổi mặc định từ Bash Shell thành Zsh (Zshell). Cả hai đều thuộc họ vỏ Bourne.

Các lệnh shell phổ biến và hữu ích:

- *cat*: đọc và hiển thị nội dung của một hoặc nhiều file.
- *cd*: thay đổi thư mục bạn đang làm việc.
- *chmod*: sửa đổi quyền của tập tin hoặc thư mục.
- *chown*: sửa đổi quyền sở hữu tập tin hoặc thư mục.
- *cp*: sao chép tập tin.
- *echo*: trả kết quả của lệnh về đầu ra tiêu chuẩn.
- *grep*: tìm kiếm một tệp cho một mẫu nhất định.
- *killall*: dừng một tiến trình hoặc ứng dụng đang chạy.
- *ls*: liệt kê nội dung của một thư mục.

- *mkdir*: tạo một thư mục hoặc thư mục mới.
- *pwd*: xem thư mục bạn đang làm việc.
- *rm*: xóa tập tin.
- *rmdir*: xóa thư mục.
- *mv*: di chuyển một hoặc nhiều file hoặc thư mục.

2.6.3 So sánh

	macOS	iOS
Shell được sử dụng	Shell có thể được sử dụng dễ dàng trên Terminal với mặc định sử dụng Zshell.	Nếu bạn cài đặt ứng dụng Terminal từ App Store hoặc sử dụng các ứng dụng thứ ba, bạn có thể sử dụng Bash hoặc các shell khác.
Cú pháp lệnh	Cú pháp lệnh trên macOS giống với các lệnh shell trên các hệ thống Unix-based khác.	Cú pháp lệnh tương tự macOS, nhưng có thể có một số hạn chế tùy thuộc vào ứng dụng Terminal bạn đang sử dụng.
Quyền truy cập hệ thống	Người dùng có quyền truy cập đầy đủ vào hệ thống và có thể thực hiện nhiều lệnh với quyền root thông qua sudo.	Với quyền jailbreak, người dùng có thể có quyền root và thực hiện lệnh với quyền cao hơn. Tuy nhiên, trong trạng thái không jailbreak, quyền truy cập có thể bị giới hạn.
Ứng dụng Terminal	Terminal là một ứng dụng có sẵn, mạnh mẽ và linh hoạt, hỗ trợ nhiều tính năng như kịch bản shell, đa cửa sổ, và chế độ xem toàn màn hình.	Terminal không phải là ứng dụng mặc định nhưng có thể được cài đặt từ App Store. Các ứng dụng Terminal trên iOS có thể có các tính năng khác nhau tùy thuộc vào ứng dụng cụ thể.
Kịch bản và Automation	Người dùng có thể viết và chạy các kịch bản shell để tự động hóa nhiều công việc.	Tùy thuộc vào ứng dụng Terminal bạn đang sử dụng, việc viết và chạy kịch bản có thể được hỗ trợ, nhưng không linh hoạt như trên macOS.

Tóm lại, lời gọi hàm shell trên macOS và iOS có nhiều điểm tương đồng, nhưng có sự khác biệt tùy thuộc vào cấu hình cụ thể của hệ thống và ứng dụng sử dụng.

2.7 Các đặc điểm hệ thống

2.7.1 Hệ điều hành iOS

Để phù hợp cho sử dụng trên điện thoại thông minh, hệ điều hành iOS của Apple có một số đặc điểm quan trọng như sau:

- *Giao diện người dùng (UI)*: iOS nổi tiếng với giao diện người dùng đơn giản, trực quan và thẩm mỹ; sử dụng cơ chế màn hình cảm ứng đa điểm để tạo ra trải nghiệm tương tác dễ dàng cho người dùng.



Hình 26: Hình ảnh giao diện Iphone trên hệ điều hành iOS

- *Hệ thống khóa và bảo mật*: iOS được thiết kế với nhiều lớp bảo mật, bao gồm Face ID và Touch ID để mở khóa thiết bị. Apple luôn cập nhật iOS để bảo vệ người dùng khỏi các lỗ hổng bảo mật và phần mềm độc hại.
- *Ứng dụng*: Người dùng có thể tải ứng dụng từ App Store, nơi có hàng triệu ứng dụng có sẵn cho nhiều mục đích khác nhau, từ giải trí đến công việc và giáo dục.



Hình 27: App Store - Kho tải ứng dụng cho hệ sinh thái Apple

- *Tích hợp với hệ sinh thái Apple*: iOS hoàn toàn tích hợp với các sản phẩm và dịch vụ khác của Apple như iCloud, iMessage, Apple Music, Siri và Apple Watch, tạo ra một hệ sinh thái hoàn chỉnh.
- *Cập nhật phần mềm*: Apple thường xuyên phát hành cập nhật phần mềm để sửa lỗi, cải thiện hiệu suất và bảo mật. Người dùng iOS có thể dễ dàng cập nhật hệ điều hành của họ.

- *Hỗ trợ đa nhiệm:* iOS hỗ trợ đa nhiệm, cho phép bạn chạy nhiều ứng dụng cùng một lúc và chuyển đổi giữa chúng dễ dàng.
- *Hỗ trợ đám mây:* iOS tích hợp với dịch vụ lưu trữ đám mây của Apple là iCloud để đồng bộ hóa dữ liệu và tài khoản trên nhiều thiết bị.
- *Khả năng tùy chỉnh:* Người dùng iOS có khả năng tùy chỉnh khá giới hạn so với hệ điều hành Android, nhưng họ có thể thay đổi nền, bố cục màn hình chính và thay đổi ứng dụng nền.
- *Khả năng quản lý thiết bị:* Apple cung cấp các tính năng quản lý thiết bị cho doanh nghiệp và người dùng cá nhân để bảo vệ dữ liệu và thông tin riêng tư.

2.7.2 Hệ điều hành macOS

Hệ điều hành macOS của Apple có nhiều đặc điểm độc đáo và quan trọng, giúp nó phù hợp cho máy tính cá nhân và máy tính xách tay của Apple. Dưới đây là một số đặc điểm của macOS:

- *Giao diện người dùng đồ họa:* macOS có giao diện người dùng thân thiện, với menu trên thanh đầu tiên (menu bar) và giao diện cửa sổ đa nhiệm cho phép người dùng mở nhiều ứng dụng và cửa sổ cùng một lúc.



Hình 28: Hình ảnh giao diện hệ điều hành macOS

- *Khả năng tìm kiếm:* Tính năng Spotlight cho phép người dùng tìm kiếm nhanh chóng tệp tin, ứng dụng, liên hệ, thông tin máy tính và nhiều thông tin khác.
- *Khả năng sao lưu và khôi phục:* Time Machine là một công cụ sao lưu tích hợp cho phép người dùng sao lưu dữ liệu của họ một cách định kỳ và khôi phục phiên bản trước của các tệp tin hoặc hệ thống hoàn chỉnh.
- *Cài đặt đơn giản:* Việc cài đặt và cập nhật phần mềm trên macOS thường rất dễ dàng. Apple thường cung cấp cập nhật phần mềm định kỳ để cải thiện hiệu suất và bảo mật.
- *Tích hợp với hệ sinh thái Apple:* macOS hoàn toàn tích hợp với các sản phẩm và dịch vụ khác của Apple như iCloud, iMessage, Handoff (cho phép bạn làm việc trên nhiều thiết bị), Apple Music và AirDrop.
- *Bảo mật:* macOS được thiết kế với nhiều tính năng bảo mật như Gatekeeper, XProtect, và FileVault để bảo vệ người dùng khỏi sự xâm nhập của các phần

mềm độc hại.

- *Khả năng thiết lập kiểm soát của phụ huynh*: Tính năng này của macOS cho phép phụ huynh có thể kiểm soát thời gian con cái họ dành trên máy Mac và các trang web chúng truy cập.
- *Tích hợp công cụ phát triển*: Để hỗ trợ việc phát triển ứng dụng cho macOS, Apple cung cấp các công cụ như Xcode và Interface Builder cho các nhà phát triển.
- *Tích hợp hệ thống tệp hỗn hợp*: macOS sử dụng hệ thống tệp hệ điều hành APFS (Apple File System), cung cấp khả năng quản lý tệp tin hiệu quả và đáng tin cậy.
- *Hỗ trợ đám mây và đồng bộ hóa*: macOS tích hợp với iCloud Drive, cho phép người dùng lưu trữ và đồng bộ hóa tệp tin trực tuyến trên các thiết bị của họ.

3 Kết luận

Dựa trên các phân tích và so sánh trên, chúng ta có thể thấy rằng cả hệ điều hành iOS và macOS đều có những ưu điểm và nhược điểm riêng. Trong quản lý file, quản lý bộ nhớ, cấu trúc hệ thống, và các dịch vụ hệ điều hành cung cấp, cả hai hệ điều hành đều cho thấy sự linh hoạt và hiệu quả. Tuy nhiên, có những khác biệt đáng kể giữa hai hệ điều hành này trong các hàm Shell và các đặc điểm hệ thống.

iOS và macOS đều là những hệ điều hành hiện đại, ổn định, bảo mật và dễ sử dụng. Chúng đều được phát triển bởi Apple và có nhiều tính năng tương thích với nhau.

iOS là hệ điều hành dành cho các thiết bị di động như iPhone, iPad và iPod touch. Nó có giao diện đơn giản, trực quan và thân thiện với người dùng. Nó cũng có nhiều ứng dụng và dịch vụ đa dạng cho nhu cầu giải trí, học tập và làm việc của người dùng.

macOS là hệ điều hành dành cho các máy tính cá nhân như MacBook, iMac và Mac mini. Nó có giao diện đẹp mắt, chuyên nghiệp và linh hoạt. Nó cũng có nhiều công cụ và chức năng cao cấp cho nhu cầu sáng tạo, nghiên cứu và phát triển của người dùng.

iOS và macOS có nhiều điểm giống nhau trong cách quản lý file, bộ nhớ, tiến trình và các dịch vụ hệ điều hành. Chúng đều sử dụng hệ thống file APFS (Apple File System), bộ nhớ ảo, thuật toán lập lịch tiến trình Round Robin và các dịch vụ như iCloud, Siri, FaceTime, AirDrop và Apple Pay.

iOS và macOS cũng có nhiều điểm khác nhau trong cấu trúc hệ thống, các hàm Shell và các đặc điểm hệ thống. iOS là một hệ điều hành mô-đun (modular), có kiến trúc lõi (core) gọn nhẹ và ít mở rộng. Nó không có giao diện Shell cho người dùng và chỉ cho phép chạy các ứng dụng được phê duyệt bởi Apple. Nó cũng có nhiều giới hạn về quyền truy cập file, bộ nhớ và tiến trình của người dùng. macOS là một hệ điều hành lớp (layered), có kiến trúc lõi (core) phức tạp và nhiều mở rộng. Nó có giao diện Shell cho người dùng và cho phép chạy các ứng dụng từ

nhiều nguồn khác nhau. Nó cũng có nhiều tự do về quyền truy cập file, bộ nhớ và tiến trình của người dùng.

4 Tài liệu tham khảo

- [1] Tom Nelson (2021), *What Is APFS (Apple's File System for macOS)?*, *Lifewire*. [Online]. Available: <https://www.lifewire.com/apple-apfs-file-system-4117093> (visited on 10/17/2023)
- [2] *Memory usage performance guidelines (2013) About the Virtual Memory System*. [Online]. Available: <https://developer.apple.com/library/archive/documentation/Performance/Conceptual/ManagingMemory/Articles/AboutMemory.html> (visited on 10/17/2023)
- [3] Available: <https://www.slideshare.net/caothanhvang/nghe-n-cu-xy-dng-ng-dng-cho-h-iu-hnh-ios> (visited on 10/17/2023)
- [4] Kandji Team (2022), *Guide for Apple IT: Introduction to Mac Shell Scripts*. Available: [https://blog.kandji.io/guide-for-apple-it-introduction-to-mac-scripting#:~:text=The%20shell%20is%20the%20command,shell%20to%20zsh%20\(zshell\)](https://blog.kandji.io/guide-for-apple-it-introduction-to-mac-scripting#:~:text=The%20shell%20is%20the%20command,shell%20to%20zsh%20(zshell)) (visited on 10/20/2023)
- [5] MobileWorld (2017), *Memory management – MRC vs Arc, nileshIOS*. Available: <https://swift007blog.wordpress.com/2017/01/14/what-is-arc-in-ios/> (visited on 10/21/2023)
- [6] Available: <https://cdn.ttgtmedia.com/searchNetworking/downloads/InsideIOS.pdf> (visited on 10/23/2023)
- [7] *Tìm hiểu về hệ điều hành macOS*. Available: <https://www.studocu.com/en-us/document/hoc-vien-ngan-hang/kien-truc-may-tinh-va-he-dieu-hanh/tim-hieu-ve-he-dieu-hanh-mac-os/54178340> (visited on 10/23/2023)
- [8] *About Apple File System*. [Online]. Available: https://developer.apple.com/documentation/foundation/file_system/about_apple_file_system (visited on 11/01/2023)