**Chương 1**

**GIỚI THIỆU HỆ ĐIỀU HÀNH ANDROID**

* 1. **Giới thiệu hệ điều hành Android.**

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho Zalo các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005. Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào tháng 10 năm 2008.

Android là mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache. Chính mã nguồn mở cùng với một giấy phép không có nhiều rằng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do. Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Vào tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới, vượt qua Symbian vào quý 4 năm 2010, và được các công ty công nghệ lựa chọn khi họ cần một hệ điều hành không nặng nề, có khả năng tinh chỉnh, và giá rẻ chạy trên các thiết bị công nghệ cao thay vì tạo dựng từ đầu. Kết quả là mặc dù được thiết kế để chạy trên điện thoại và máy tính bảng, Android đã xuất hiện trên TV, máy chơi game và các thiết bị điện tử khác. Bản chất mở của Android cũng khích lệ một đội ngũ đông đảo lập trình viên và những người đam mê sử dụng mã nguồn mở để tạo ra những dự án do cộng đồng quản lý. Những dự án này bổ các tính năng cao cấp cho những người dùng thích tìm tòi hoặc đưa sung Android vào các thiết bị ban đầu chạy hệ điều hành khác.  
 Android chiếm 75% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 3 năm 2012, với tổng cộng 500 triệu thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh.

* 1. **Lịch sử phát triển.**

Tổng công ty Android (Android, inc) được thành lập tại Palo Alto, California vào tháng 10 năm 2003 bởi Andy Rubin (đồng sáng lập công ty Danger),[20] Rich Miner (đồng sáng lập Tổng công ty Viễn thông Wildfire), Nick Sears (từng là Phó giám đốc T-Mobile), và Chris Whits (trưởng thiết kế và giao diện tại WebTV) để phát triển, theo lời của Rubin, "các thiết bị di động thông minh hơn có thể biết được vị trí và sở thích của người dùng". Dù những người thành lập và nhân viên đều là những người có tiếng tăm, tổng công ty. Android hoạt động một cách âm thẩm, chi tiết lộ rằng họ đang làm phần mềm dành cho điện thoại di động. Trong năm đó, Rubin hết kinh phí. Steve Perlman, một người bạn thân của Rubin, mang cho ông 10.000 USD tiền mặt nhưng từ chối tham gia vào công ty.   
 Google mua lại Tổng công ty Android vào ngày 17 tháng 8 năm 2005, biến nó thành một bộ phận trực thuộc Google. Những nhân viên của chủ chốt của Tổng công ty Androld, gồm Rubin, Miner và White, vẫn tiếp tục ở lại công ty làm việc sau thương vụ này. Vào thời điểm đó không có nhiều thông tin về Tổng công ty, những nhiều người đồn đoán rằng Google dự tính tham gia thị trường điện thoại di động sau bước đi này. Tại Cogle, nhóm do Rubin đứng đầu đã phát triển một nền tăng thiết bị di động phát triển trên nền nhân Linux, Google quảng bá nền tảng này cho các nhà sản xuất điện thoại và các nhà mạng với lời hứa sẽ cung cấp một hệ thống uyển chuyển và có khả năng nâng cấp. Google đã liên hệ với hàng loạt hãng phần cứng cũng như đối tác phần mềm, bắn tin cho các nhà mạng rằng họ sẵn sàng hợp tác với các cấp độ khác nhau.

Ngày cũng nhiều suy đoán rằng Google sẽ tham gia thị trường điện thoại đi động xuất hiện trong tháng 12 năm 2006. Tin tức của BBC và Nhật báo phố Wall chú thích rằng Google muốn đưa công nghệ tìm kiếm và các ứng dụng của họ vào điện thoại di động và họ đang nỗ lực làm việc để thực hiện điều này. Các phương tiện truyền thông, truyền thống lẫn online cùng viết về tin đồn rằng Google đang phát triển một thiết bị cầm tay mang thương hiệu Google. Một vài tờ báo còn nói rằng trong khi Google vẫn đang thực hiện những bản mô tả kỹ thuật chi tiết, họ đã trình diễn sản phẩm mẫu cho các nhà sản xuất điện thoại di động và nhà mạng.  
Tháng 9 năm 2007, InformationWeek đăng tải một nghiên cứu của Evalueserve cho biết Google đã nộp một số đơn xin cấp bằng sáng chế trong lĩnh vực điện thoại di động.

Ngày 5 tháng 11 năm 2007, Liên minh thiết bị cầm tay mở (Open Handset Alliance), một hiệp hội bao gồm nhiều công ty trong đó có Texas Instruments, Tập đoàn Broadcom, Google, HTC, Intel, LG, Tập đoàn Marvell Technology. Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel và T-Mobile được thành lập với mục đích phát triển các tiêu chuẩn mở cho thiết bị di động. Cùng ngày, Android cũng được ra mắt với vai trò là sản phẩm đầu tiên của Liên minh, một nền tảng thiết bị di động được xây dựng trên nhãn Linux phiên bản 2.6. Chiếc điện thoại chạy Android đầu tiên được bán ra là HTC Dream, phát hành ngày 22 tháng 10 năm 2008. Biểu trưng của hệ điều hành Android mới là một con rô bốt màu xanh lá cây do hãng thiết kế Irina Blok tại California vẽ.

Từ năm 2008, Android đã trải qua nhiều lần cập nhật để dần dần cải tiến hệ điều hành, bổ sung các tính năng mới và sửa các lỗi trong những lần phát hành trước. Mỗi bản nâng cấp được đặt tên lần lượt theo thứ tự bảng chữ cái, theo tâm của một món ăn tráng miệng; ví dụ như phiên bản 1.5 Cupcake (bánh bông lan nhỏ có kem) tiếp nối bằng phiên bản 1.6 Donut (bánh vòng). Phiên bản mới nhất hiện nay là 5.0 Lollipop. Vào năm 2010, Google ra mắt loạt thiết bị Nexus-một dòng sản phẩm bao gồm điện thoại thông minh và máy tính bảng chạy hệ điều hành Android, do các đối tác phần cứng sản xuất. HTC đã hợp tác với Google trong chiếc điện thoại thông minh Nexus đầu tiên, Nexus One. Kể từ đó nhiều thiết bị mới hơn đã gia nhập vào dòng sản phẩm này, như điện thoại Nexus 4 và máy tính bảng Nexus 10, lần lượt do LG và Samsung sản xuất, Google xem điện thoại và máy tính bảng Nexus là những thiết bị Android chủ lực của mình, với những tính năng phần cứng B: và phần mềm mới nhất của Android.

Năm 2014. Google công báo Android Wear, hệ điều hành dành cho các thiết bị đeo được.

* 1. **Giao diện Android**

Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt chạm, Kéo giãn và thu lại để xử lý các đối tượng trên màn hình. Sự phản ứng với tác động của người dùng diễn ra gần như ngay lập tức, nhằm tạo ra giao diện cảm ứng mượt mà, thường dùng tính năng rung của thiết bị để tạo phản hồi rung cho người dùng. Những thiết bị phần cứng bên trong như gia tốc kế, con quay hồi chuyển và cảm biến khoảng cách được một số ứng dụng sử dụng để phản hồi một số hành động khác của người dùng, ví dụ như điều chỉnh màn hình từ chế độ hiển thị dọc sang chế độ hiển thị ngang tùy theo vị trí của thiết bị, hoặc cho phép người dùng lái xe đua bằng xoay thiết bị, giống như đang điều khiển vô - lăng.

Các thiết bị Android sau khi khởi động sẽ hiển thị màn hình chính, điểm khởi đầu với các thông tin chính trên thiết bị, tương tự như khái niệm desktop (bàn làm việc) trên máy tính để bàn. Màn hình chính Android thường gồm nhiều biểu tượng (icon) và tiện ích (widget); biểu tượng ứng dụng sẽ mở ứng dụng tương ứng, còn tiện ích hiển thị những nội dung sống động, cập nhật tự động như dự báo thời tiết, hộp thư của người dùng, hoặc những mẫu tin thời sự ngay trên màn hình chính. Màn hình chính có thể gồm nhiều trang xem được bằng cách vuốt ra trước hoặc sau, mặc dù giao diện màn hình chính của Android có thể tùy chỉnh ở mức cao, cho phép người dùng tự do sắp đặt hình dáng cũng như hành vi của thiết bị theo sở thích. Những ứng dụng do các hãng thứ ba có trên Google Play và các kho ứng dụng khác còn cho phép người dùng thay đổi "chủ đề" của màn hình chính, thậm chí bắt chước hình dáng của hệ điều hành khác như Windows Phone chẳng hạn. Phần lớn những nhà sản xuất, và một số nhà mạng, thực hiện thay đổi hình dáng và hành vi của các thiết bị Android của họ để phân biệt với các hãng cạnh tranh.

Ở phía trên cùng màn hình là thanh trạng thái, hiển thị thông tin về thiết bị và tình trạng kết nối. Thanh trạng thái này có thể "kéo" xuống để xem màn hình thông báo gồm thông tin quan trọng hoặc cập nhật của các ứng dụng, như email hay tin nhắn SMS mới nhận, mà không làm gián đoạn hoặc khiến người dùng cảm thấy bất tiện. Trong các phiên bản đời đầu, người dùng có thể nhấn vào thông báo để mở ra ứng dụng tương ứng, về sau này các thông tin cập nhật được bổ sung theo tỉnh năng, như có khả năng lập tức gọi ngược lại khi có cuộc gọi nhỡ mà không cần phải mở ứng dụng gọi điện ra. Thông báo sẽ luôn nằm đó cho đến khi người dùng đã đọc hoặc xóa nó đi.

* 1. **Ứng dụng Android**

Android có lượng ứng dụng của bên thứ ba ngày cảng nhiều, được chọn lọc và đặt trên một cửa hàng ứng dụng như Google Play hay Amazon Appstore để người dùng lấy về, hoặc bằng cách tại xuống rồi cài đặt tập tin APK từ trang web khác. Các ứng dụng trên Cửa hàng Play cho phép người dùng duyệt, tài về và cập nhật các ứng dụng do Google và các nhà phát triển thứ ba phát hành. Cửa hàng Play được cài đặt sẵn trên các thiết bị thỏa mãn điều kiện tương thích của Google. Ứng dụng sẽ tự động lọc ra một danh sách các ứng dụng tương thích với thiết bị của người dùng, và nhà phát triển có thể giới hạn ứng dụng của họ chỉ dành cho những nhà mạng cố định hoặc những quốc gia cố định vì lý do kinh doanh. Nếu người dùng mua một ứng dụng mà họ cảm thấy không thích, họ được hoàn trả tiền sau 15 phút kể từ lúc tải về, và một vài nhà mạng còn có khả năng mua giúp các ứng dụng trên Google Play, sau đó tính tiền vào trong hóa đơn sử dụng hàng tháng của người dùng. Đến tháng 9 năm 2012, có hơn 675.000 ứng dụng dành cho Android, và số lượng ứng dụng tải về từ Cửa hàng Play ước tính đạt 25 tỷ.

Các ứng dụng cho Android được phát triển bằng ngôn ngữ Java sử dụng Bộ phát triển phần mềm Android (SDK), SDK bao gồm một bộ đầy đủ các công cụ dùng để phát triển, gồm có công cụ gỡ lỗi, thư viện phần mềm, bộ giả lập điện thoại dựa trên OEMU, tài liệu hướng dẫn, mã nguồn mẫu, và hướng dẫn từng bước. Môi trường phát triển tích hợp (IDE) được hỗ trợ chính thức là Eclipse sử dụng phần bổ sung Android Development Tools (ADT). Các công cụ phát triển khác cũng có sẵn, gồm có Bộ phát triển gốc dành cho các ứng dụng hoặc phần mở rộng viết bằng C hoặc C++, Google App Inventor, một môi trường đồ họa cho những nhà lập trình mới bắt đầu, và nhiều nền tảng ứng dụng web di động đa nền tảng phong phú.

* 1. **Quản lý bộ nhớ Android**

Vì các thiết bị Android được thiết kế để quản lý bộ nhớ (RAM) để giảm tối đa mức tiêu thụ điện năng, trái với hệ điều hành máy tính để bàn luôn cho rằng máy tính sẽ có nguồn điện không giới hạn. Khi một ứng dụng Android không còn được sử dụng, hệ thống sẽ tự động ngưng nó trong bộ nhớ - trong khi ứng dụng về mặt kỹ thuật vẫn “mở”, những ứng dụng này sẽ không tiêu thụ bất cứ tài nguyên nào (như năng lượng pin hay năng lượng xử lý) và năm đó cho đến khi nó được cần đến. Cách làm như vậy có lợi kép là vừa làm tăng khả năng phản hồi nói chung của thiết bị Android, vì ứng dụng không nhất phải đóng rồi mở lại từ đầu, vừa đảm bảo các ứng dụng nền không làm tiêu hao năng lượng một cách không cần thiết.

Android quản lý các ứng dụng trong bộ nhớ một cách tự động: khi bộ nhớ thấp, hệ thống sẽ bắt đầu diệt ứng dụng và tiến trình không hoạt động được một thời gian, sắp theo thời điểm cuối mà chúng được sử dụng (tức là cũ nhất sẽ bị tắt trước). Tiến trình này được thiết kế án đi với người dùng, để người dùng không cần phải quản lý bộ nhớ hoặc tự tay tắt các ứng dụng. Tuy nhiên, sự che giấu này của hệ thống quản lý bộ nhớ Android đã dẫn đến sự thịnh hành của các ứng dụng tắt chương trình của bên thứ ba trên cửa hàng Google Play, những ứng dụng kiểu như vậy được cho là có hại nhiều hơn có lợi.

* 1. **Nhân Linux**

Android có một hạt nhân dựa trên nhân Linux phiên bản 2.6, kể từ Android 4.0 Ice Cream Sandwich (bánh ngọt kẹp kem) trở về sau, là phiên bản 3.X, với middleware, thư viện và API viết bằng C. còn phần mềm ứng dụng chạy trên một nền tảng ứng dụng gồm các thư viện tương thích với Java dựa trên Apache Harmony. Android sử dụng máy ảo Dalvik với một trình biên dịch động để chạy 'mã dex' (Dalvik Executable) của Dalvik, thường được biên dịch sang Java bytecode. Nền tảng phần cứng chính của Android là kiến trúc ARM. Người ta cũng hỗ trợ x86 thông qua dự án Android x86, và Google V cũng sử dụng một phiên bản x86 đặc biệt của Android.

Một số tính năng cũng được Google đóng góp ngược vào nhân Linux, đáng chú ý là tính năng quản lý nguồn điện có tên wakelock, nhưng bị những người lập trình chính cho nhân từ chối vì họ cảm thấy Google không có định sẽ tiếp tục bảo trì đoạn mã do họ viết. Google thông báo vào tháng 4 năm 2010 rằng họ sẽ thuê hai nhân viên để làm việc với cộng đồng nhân Linux, nhưng Greg Kroah- Hartman, người bảo trì nhân Linux hiện tại của nhánh ổn định, đã nói vào tháng 12 năm 2010 rằng ông ta lo ngại rằng Google không còn muốn đưa những thay đổi của mình vào Linux dòng chính nữa. Một số lập trình viên Android của Google tỏ ý rằng "nhóm Android thấy chán với quy trình đó," vì nhóm họ không có nhiều người và có nhiều việc khẩn cấp cẩn làm với Android hơn.

Vào tháng 8 năm 2011, Linus Torvalds rằng "rốt cuộc thì Android và Linux cũng sẽ trở lại với một bộ nhân chung, nhưng điều đó có thể sẽ không xảy ra trong 4 hoặc 5 năm nữa". Vào tháng 12 năm 2011, Greg Kroah-Hartman thông báo kích hoạt Dự án Dòng chính Android, nhắm tới việc đưa một số driver, bản vá và tinh năng của Android ngược vào nhân Linux, bắt đầu từ Linux.

Trong phiên bản 3.3. Linux cũng đưa tính năng autosleep (tự nghĩ hoạt động) và wakelocks vào nhân 3.5, sau nhiều nỗ lực phối trộn trước đó. Tương tác thì vẫn vậy nhưng bản hiện thực trên Linux dòng chính cho phép hai chế độ nghỉ; bộ nhớ (dạng nghi truyền thống mà Android sử dụng), và đĩa (là ngủ đông trên máy tính để bàn). Việc trộn sẽ hoàn tất kể từ nhân 3.8, Google đã công khai kho mã nguồn trong đó có những đoạn thử nghiệm đưa Android về lại nhân 3.8.

Bộ lưu trữ flash trên các thiết bị Android được chia thành nhiều phân vùng như "system" dành cho hệ điều hành và "data" dành cho dữ liệu người dùng và cài đặt ứng dụng. Khác với các bản phân phối Linux cho máy tính để bàn, người sở hữu thiết bị Android không được trao quyền truy cập root vào hệ điều hành và các phân vùng nhạy cảm như "system" được thiết lập chỉ đọc. Tuy nhiên, quyền truy cập root có thể chiếm được bằng cách tận dụng những lỗ hổng bảo mật trong Android, điều mà cộng đồng mã nguồn mở thường xuyên sử dụng để nâng cao tính năng thiết bị của họ, kể cả bị những người ác ý sử dụng để cải virus và phần mềm ác ý.

Việc Android có được xem là một bản phân phối Linux hay không vẫn còn là một vấn đề gây tranh cãi, tuy được Linux Foundation và Chris DiBona, trưởng nhóm mã nguồn mở Google, ủng hộ. Một số khác, như linux-magazine.com thì không đồng b ý, do Android không không hỗ trợ nhiều công cụ GNU, trong đó có glibc.

* 1. **Lịch nâng cấp**

Google đưa ra các bản nâng cấp lớn cho Android theo chu kỳ từ 6 đến 9 tháng, mà phần lớn thiết bị đều có thể nhận được qua sóng khung dây. Bản nâng cấp lớn mới nhất là Android 8.0 Oreo.

So với các hệ điều hành cạnh tranh khác, như IOS, các bản nâng cấp Android thường mất thời gian lâu hơn để đến với các thiết bị. Với những thiết bị không thuộc dòng Nexus, các bản nâng cấp thường đến sau vài tháng kể từ khi phiên bản được chính thức phát hành. Nguyên nhân của việc này một phần là do sự phong phú về phần cứng của các thiết bị Android, nên người ta phải mất thời gian điều chỉnh bản nâng cấp cho phù hợp, vì mã nguồn chính thức của Google chỉ chạy được trên những thiết bị Nexus chủ lực của họ. Chuyển Android sang những phần cứng cụ thể là một quy trình tốn thời gian và công sức của các nhà sản xuất thiết bị, những người luôn ưu tiên các thiết bị mới nhất và thường bỏ rơi các thiết bị cũ hơn. Do đó, những chiếc điện thoại thông minh thế hệ cũ thường không được nâng cấp nếu nhà sản xuất quyết định rằng nó không đáng để bỏ thời gian, bất kể chiếc điện thoại đó có khả năng chạy bản nâng cấp hay không. Vấn đề này còn trầm trọng hơn khi những nhà sản xuất điều chỉnh Android để đưa giao diện và ứng dụng của họ vào, những thứ này cũng sẽ phải làm lại cho mỗi bản nâng cấp. Sự chậm trễ còn được đóng góp bởi nhà mạng, sau khi nhận được bản nâng cấp từ nhà sản xuất, họ còn điều chỉnh thêm cho phù hợp với nhu cầu rồi thử nghiệm kỹ lưỡng trên hệ thống mạng của họ trước khi chuyển nó đến người dùng.

Việc thiếu các hỗ trợ hậu mãi của nhà sản xuất và nhà mạng đã bị những nhóm người dùng và các trang tin công nghệ chỉ trích rất nhiều. Một số người viết còn nói rằng giỏi công nghiệp do cái lợi về tài chính đã cố tình không nâng cấp thiết bị, vì nếu thiết bị hiện tại không cập nhật sẽ thúc đẩy việc mua thiết bị mới, một thái độ được coi là "xúc phạm”. The Guardian đã than phiền rằng phương cách phân phối bản nâng cấp trở nên phức tạp chính vị những nhà sản xuất và nhà mạng đã cố tình làm nó như thế. Vào năm 2011, Google đã hợp tác cùng một số băng công nghiệp và ra mắt "Liên minh nâng cấp Android", với lời hứa sẽ nâng cấp thường xuyên cho các thiết bị trong vòng 18 tháng sau khi ra mắt. Tính đến năm 2012, người ta không còn nghe nhắc đến liên minh này nữa.

* 1. **Cộng đồng mã nguồn mở**

Android có một cộng đồng các lập trình viên và những người đam mê rất năng động. Họ sử dụng mã nguồn Android để phát triển và phân phối những phiên bản chỉnh sửa của hệ điều hành. Các bản Android do cộng đồng phát triển thường đem những tính năng và cập nhật mới vào nhanh hơn các kênh chính thức của nhà sản xuất nhà mạng, tuy không được kiểm thử kỹ lưỡng cũng như không có đảm bảo chất lượng; cung cấp sự hỗ trợ liên tục cho các thiết bị cũ không còn nhận được bản cập nhật chính thức; hoặc mang Android vào những thiết bị ban đầu chạy một hệ điều hành khác, như HP Touchpad. Các bản Android của cộng đồng thường được root sẵn và có những điều chỉnh không phù hợp với những người dùng không rành rẽ, như khả năng ép xung hoặc tăng/giảm áp bộ xử lý của thiết bị. CyanogenMod là firmware của cộng đồng được sử dụng phổ biến nhất, và hoạt động như một tổ chức của số đông khác.

Trước đây, nhà sản xuất thiết bị và nhà mạng tỏ ra thiếu thân thiện chỉ với việc phát triển firmware của bên thứ ba. Những nhà sản xuất còn thể hiện lo ngại rằng các thiết bị chạy phần mềm không chính thức sẽ hoạt động không tốt và dẫn đến tổn tiền hỗ trợ. Hơn nữa, các firmware đã thay đổi như CyanogenMod đôi khi còn cung cấp những tính năng, như truyền tải mạng (tethering), mà người dùng bình thường phải trả tiền nhà mạng mới được sử dụng. Kết quả là nhiều thiết bị bắt đầu đặt ra hàng rào kỹ thuật như khóa bootloader hay hạn chế quyền truy cập root. Tuy nhiên, khi phần mềm do cộng đồng phát triển ngày càng trở nên phổ biến, và sau một thông cáo của Thư viện Quốc hội Hoa Kỳ cho phép “jailbreak" (vượt ngục) thiết bị di động, các nhà sản xuất và nhà mạng đã tỏ ra mềm mỏng hơn với các nhà phát triển thứ ba, thậm chí một số hãng như HTC. Motorola, Samsung và Sony, còn hỗ trợ và khuyến khích phát triển. Kết quả của việc này là dần dần nhu cầu tìm ra các hạn chế phần cứng để cài đặt được firmware không chính thức đã bớt đi do ngày càng nhiều thiết bị được phát hành với bootloader đã mở khóa sẵn hoặc có thể mở khóa, tương tự như điện thoại dòng Nexus, tuy rằng thông thường họ sẽ yêu cầu người dùng từ bỏ chế độ bảo hành nếu họ làm như vậy. Tuy nhiên, tuy được sự chấp thuận của nhà sản xuất, một số nhà mạng tại Mỹ vẫn bắt buộc điện thoại phải bị khóa.

Việc mở khóa và "hack" điện thoại thông minh và máy tính bảng vẫn còn là tác nhân gây căng thẳng giữa cộng đồng và công nghiệp. Cộng đồng luôn biện hộ rằng sự hỗ trợ không chính thức ngày càng trở nên quan trọng trước việc nền công nghiệp không cung cấp các bản cập nhật thường xuyên và/hoặc ngưng hỗ trợ cho chính các thiết bị của họ.

* 1. **Bảo mật và tính riêng tư của Android**

Các ứng dụng Android chạy trong một "hộp cát", là một khu vực riêng rẽ với hệ thống và không được tiếp cận đến phần còn lại của tài nguyên hệ thống, trừ khi nó được người dùng trao quyền truy cập một cách công khai khi cài đặt. Trước khi cài đặt ứng dụng, Cửa hàng Play sẽ hiển thị tất cả các quyền mà ứng dụng đòi hỏi: ví dụ như một trò chơi cần phải kích hoạt bộ rung hoặc lưu dữ liệu vào thẻ nhớ SD, nhưng nó không nên cần quyền đọc tin nhắn SMS hoặc tiếp cận danh bạ điện thoại. Sau khi xem xét các quyền này, người dùng có thể chọn đồng ý hoặc từ chối chúng, ứng dụng chỉ được cài đặt khi người dùng đồng ý.

Hệ thống hộp cát và hỏi quyền làm giảm bớt ảnh hưởng của lỗi bảo mật hoặc lỗi chương trình có trong ứng dụng, những sự bối rối của lập trình viên và tài liệu hướng dẫn còn hạn chế đã dẫn tới những ứng dụng hay đòi hỏi những quyền không cần thiết, do đó làm giảm đi hiệu quả của hệ thống này. Một số công ty bảo mật, như Lookout Mobile Security, AVG Technologies, và McAfee, đã phát hành những phần mềm diệt virus cho các thiết bị Android. Phần mềm này không có hiệu quả vì cơ chế hộp cát vẫn áp dụng vào các ứng dụng này, do vậy làm hạn chế khả năng quét sâu vào hệ thống để tìm nguy cơ.

Một nghiên cứu của công ty bảo mật Trend Micro đã liệt kê tình trạng lạm dụng dịch vụ trả tiền là hình thức phần mềm ác ý phổ biến nhất trên Android, trong đó tin nhắn SMS sẽ bị gửi đi từ điện thoại bị nhiễm đến một số điện thoại trả tiền mà người dùng không hề hay biết. Loại phần mềm ác ý khác hiển thị những quảng cáo không mong muốn và các khó chịu trên thiết bị hoặc ori thông tin cá nhân đến bên thứ ba khi chưa được phép. Đe dọa bảo mật trên Android được cho là tăng rất nhanh theo cấp số mũ; tuy nhiên, các kỹ sư Google phản bác rằng hiểm họa từ phần mềm ác ý và virus đã bị thổi phồng bởi các công ty bảo mật nhằm mục đích thương mại, và buộc tội ngành công nghiệp bảo mật đang lợi dụng sự sợ hãi để bán phần mềm diệt virus cho người dùng. Google vẫn giữ qu mềm diệt virus cho người dùng. Google vẫn giữ quan điểm rằng phần mềm ác ý thật sự nguy hiểm là cực kỳ hiếm, và một cuộc điều tra do F-Secure thực hiện cho thấy chỉ có 0,5% số phần mềm ác ý Android là len vào được cửa hàng Google Play.

Google hiện đang sử dụng bộ quét phần mềm ác ý Google Bouncer để theo dõi và quét các ứng dụng trên Cửa hàng Google Play. Nó sẽ đánh dấu các phần mềm bị nghi ngờ và cảnh bảo người dùng về những vấn đề có thể xảy ra trước khi họ tái nó về máy. Android phiên bản 4.2 Jelly Bean được phát hành vào năm 2012 cùng với các tính năng bảo mật được cải thiện, bao gồm một bộ quét phần mềm ác ý được cài sẵn trong hệ thống, hoạt động cùng với Google Play nhưng cũng có thể quét các ứng dụng được cài đặt từ nguồn thứ ba, và một hệ thống cảnh báo sẽ thông báo cho người dùng khi một ứng dụng cố gắng gửi một tin nhắn vào số tính tiền, chặn tin nhắn đó lại trừ khi người dùng công khai cho phép nó.

Điện thoại thông minh Android có khả năng báo cáo vị trí của điểm truy cập Wi-Fi, phát hiện ra việc di chuyển của người dùng điện thoại, để xây dựng những cơ sở dữ liệu có chứa vị trí của hàng trăm triệu điểm truy cập. Những cơ sở dữ liệu này tạo nên một bản đồ điện tử để tìm vị trí điện thoại thông minh, cho phép chúng chạy các ứng dụng như Foursquare, Google Lavitude, Facebook Places, và gửi những đoạn quảng cáo dựa trên vị trí. Phần mềm theo dõi của bên thứ ba như TaintDroid, một dự án nghiên cứu trong trường đại học, đôi khi có thể biết được khi nào thông tin cá nhân bị gửi đi từ ứng dụng đến các máy chủ đặt ở xa.

Bản chất mã nguồn mở của Android cho phép những nhà thầu bảo mật lấy những thiết bị sẵn có rồi điều chỉnh để sử dụng ở mức độ bảo mật cao hơn. Ví dụ như Samsung đã cộng tác với General Dynamics sau khi họ thâu tóm Open Kernel Labs để xây dựng lại Jellybean trên nền bộ vi kiểm soát dành cho dự án "Knox".

* 1. **Giấy phép phát hành**

Mã nguồn của Android được cấp phép theo các giấy phép phần mềm mã nguồn mở tự do. Google đưa phần lớn mã nguồn (bao gồm cả các lớp mạng và điện thoại) theo Giấy phép Apache phiên bản 2.0, và phần còn lại, các thay đổi đối với nhân Linux, theo Giấy phép Công cộng GNU phiên bản 2. Liên minh Thiết bị cầm tay mở đã thực hiện các thay đổi trên nhân Linux, với mã nguồn lúc nào cũng công khai. Phần còn lại của Android được Google phát triển một mình, và mã nguồn chi được công bố khi phát hành một phiên bản mới. Thông thường Google cộng tác với một nhà sản xuất phần cứng để cung cấp một thiết bị chủ lực' (thuộc dòng Google tay mở đã thực hiện các thay đổi trên nhân Linux, với mã nguồn lúc nào cũng công khai. Phần còn lại của Android được Google phát triển một mình, và mã nguồn chi được công bố khi phát hành một phiên bản mới. Thông thường Google cộng tác với một nhà sản xuất phần cứng để cung cấp một thiết bị chủ lực (thuộc dòng Google Nexus) với phiên bản mới nhất của Android, sau đó phát hành mã nguồn sau khi Dán thiết bị này được bán ra.

Vào đầu năm 2011, Google quyết định tạm ngưng phát hành mã nguồn Android phiên bản 3.0 Honeycomb dành riêng cho máy tính bảng. Lý do, theo Andy Rubin trong một bài blog Android chính thức, là vì Honeycomb đã được làm gấp gáp để phục vụ cho Motorola Xoom, và họ không muốn các bên thứ ba tạo ra một "trải nghiệm người dùng cực kỳ tồi tệ" bằng cách cố gắng đưa vào điện thoại thông minh một phiên bản dành riêng cho máy tính bảng. Mã nguồn một lần nữa được xuất bản công khai vào tháng 11 năm 2011 với sự ra mắt của Android 4.0.

Mặc dù phần mềm là mã nguồn mở, các nhà sản xuất thiết bị không thể sử dụng thương hiệu Android của Google trừ khi Google chứng nhận rằng thiết bị của họ phù hợp với Tài liệu Định nghĩa Tương thích (Compatibility Definition Document - CDD). Các thiết bị cũng phải thỏa mãn định nghĩa này thì mới được cấp phép để cài các ứng dụng mã nguồn đóng của Google, gồm cả Google Play. Vì Android không hoàn toàn được phát hành theo giấy phép tương thích GPL, ví dụ như mã nguồn của Google là theo giấy phép Apache license, và cũng vì Google Play cho phép các phần mềm có bản quyền, Richard Stallman và Quỹ phần mềm tự do luôn chỉ trích Android và khuyên người dùng sử dụng hệ điều hành khác như Replicant.

* 1. **Giai đoạn đầu khi ra mắt của Android**

Android được đón nhận bằng một thái độ thờ ơ khi ra mắt vào năm 2007. Mặc dù những nhà phân tích rất ấn tượng với việc các công ty công nghệ có tiếng tăm hợp tác cùng Google để tạo ra Liên minh thiết bị di động mở, người ta vẫn không rõ liệu các nhà sản xuất có sẵn sàng thay thế hệ điều hành mà họ đang dùng bằng Android hay không. Ý tưởng về một nền tảng phát triển mã nguồn mở dựa trên Linux đã thu hút sự quan tâm, nhưng cũng dấy lên những lo ngại rằng Android sẽ phải đối mặt với sự cạnh tranh mạnh mẽ từ những tay chơi có hạng trong thị trưởng điện thoại thông minh, như Nokia và Microsoft, và các hệ điều hành di động đối thủ cũng sử dụng Linux đang trong quá trình phát triển. Những công ty hàng đầu không giấu sự hoài nghi: Nokia được trích nói rằng"chúng tôi không xem đó là một sự đe dọa," và một thành viên của nhóm Windows Mobile của Microsoft nói rằng "tôi không hiểu rồi họ sẽ có tác động ra sao".

Kể từ đó Android đã phát triển để trở thành hệ điều hành dành cho điện thoại thông minh phổ biến nhất trên thế giới và là "một trong những trải nghiệm di động nhanh nhất hiện nay." Các nhà bình luận thì nhấn mạnh vào bản chất mã nguồn mở của hệ điều hành chính là một trong những yếu tố quyết định sức mạnh, cho phép các công ty như (Kindle Fire), Barnes & Noble (Nook), Ouya, Baidu, và những hãng khác đối hướng phần mềm và phát hành những phần cứng chạy trên phiên bản Android đã thay đổi của riêng họ. Kết quả, nó được trang web công nghệ Ars Technica mô tả là "đương nhiên là hệ điều hành mặc định khi phát hành phần cứng mới" cho những công ty không có nền tảng di động riêng của họ. Chính sự mở và uyển chuyển này cũng hiện diện ở cấp độ người dùng cuối: Android cho phép người dùng điện thoại điều chỉnh thoải mái thiết bị của họ và ứng dụng thì có sẵn trên các cửa hàng ứng dụng và trang web không phải của Google. Những đặc điểm này được xem là đóng góp vào những thế mạnh chính của điện thoại Android so với các điện thoại khác.

Android cũng bị phê phán vì thiếu sự hỗ trợ hậu mãi từ nhà sản xuất và nhà mạng. néu so sánh với iOS của Apple. Với những thiết bị không mang nhãn hiệu Nexus, nhà mạng luôn kiểm tra các tiêu chuẩn của họ rồi thực hiện thay đổi cho riêng từng thiết bị (bắt nguồn từ sự điều chỉnh của nhà sản xuất và sự đa dạng của thiết bị Android) được xem là tác nhân chính trì hoãn việc cập nhật. Những nhà bình luận cũng nói rằng ngành công nghiệp thiết bị di động vì lý do lợi nhuận đã cổ tinh không cập nhật thiết bị của họ, vì thiếu cập nhật trên thiết bị hiện tại sẽ thúc đẩy việc mua thiết bị mới.

**CHƯƠNG 2.**

**LẬP TRÌNH ỨNG DỤNG QUẢN LÝ CHI TIÊU CÁ NHÂN**

1. **Tổng quan về đề tài**
2. **Giới thiệu ứng dụng:**
   1. **Tên của ứng dụng:** ExpenseManager.

Tên gọi của ứng dụng "Expense Manager" có thể được định nghĩa như sau:

* "Expense" có nghĩa là "chi phí", "chi tiêu", "khoản chi".
* "Manager" có nghĩa là "người quản lý" hoặc "công cụ quản lý".

Vì vậy, tên gọi "Expense Manager" có nghĩa là "Công cụ quản lý chi tiêu", "Công cụ quản lý khoản chi" hoặc "Người quản lý chi tiêu". Tên gọi này cho thấy mục đích chính của ứng dụng.

* 1. **Logo của ứng dụng:**

****

(Hình ảnh minh họa)

Logo của ứng dụng Expense Manager là một biểu tượng đồ họa kết hợp giữa các chữ cái và biểu tượng với màu sắc đặc trưng để đại diện cho thương hiệu của nhà phát triển, mô tả một cách đơn giản công dụng của ứng dụng và giúp người dùng dễ dàng nhận biết và nhớ tên nó.

* Về chữ cái:
* Về biểu tượng:
* Về màu sắc:  
  1. **Công dụng và tính năng của ứng dụng:**

Ứng dụng Expense Manager được nhà phát triển tạo ra như là một công cụ tiện ích cho phép người dùng quản lý, kiểm soát và theo dõi các khoản chi tiêu của mình một cách hiệu quả hơn. Dưới đây là một số công dụng, tính năng của ứng dụng này:

1. Quản lý và theo dõi chi tiêu hàng ngày, hàng tuần, hàng tháng hoặc theo một khoảng thời gian tùy chọn.

2.Tạo danh sách các khoản chi tiêu và phân loại chúng để dễ dàng nhận biết.

3. Ngoài ra, ứng dụng còn cho phép người dùng thiết lập ngân sách và mục tiêu chi tiêu, giúp họ quản lý tài chính cá nhân hiệu quả hơn.

4. Thêm các ghi chú vào các khoản chi tiêu để giúp người dùng nhớ lại các khoản chi tiêu đó.

5.Tính toán tổng số tiền chi tiêu và so sánh với ngân sách đã đặt ra để giúp người dùng hiểu rõ hơn về tình trạng tài chính của mình.

6. Cung cấp các báo cáo và biểu đồ để thể hiện trực quan về tình hình tài chính của người dùng.

7.Cung cấp các tính năng nhắc nhở hoặc thông báo khi người dùng đến gần hoặc vượt quá ngân sách đã đặt ra.

* 1. **Nhà phát triển ứng dụng**

Trên cơ sở học tập môn “Lập trình trên thiết bị di động”, nhằm phát triển những kiến thức đã học thành kỹ năng thực hành, nhóm học viên lớp B15D52 đã lên ý tưởng, nghiên cứu, và phát triển ứng dụng Expense Manager này.

Nhóm phát triển ứng dụng gồm các thành viên:  
1. Châu Tấn Pháp  
2. Trần Đình Tuấn An  
3. Nguyễn Đình Hùng  
4. Thân Văn Thanh  
5. Dương Hoàng Nam

* 1. **Môi trường phát triển ứng dụng:**

Ứng dụng được thiết kế để chạy trên thiết bị di động môi trường hệ điều hành Android, và chạy tối ưu nhất trên điện thoại di động.

1. **Mục đích và đối tượng sử dụng của ứng dụng quản lý chi tiêu**
   1. **Mục đích**

1. Ứng dụng quản lý chi tiêu cá nhân có mục đích giúp người dùng quản lý và theo dõi các khoản thu chi của mình một cách dễ dàng và hiệu quả hơn. Với ứng dụng này, người dùng có thể ghi lại các khoản thu nhập, chi tiêu, lập kế hoạch chi tiêu, đặt mục tiêu tiết kiệm, tạo các báo cáo và phân tích chi tiêu của mình theo thời gian, loại hình hoặc nguồn gốc.

2.Quản lý chi tiêu cá nhân còn giúp người dùng có thể tối ưu hóa ngân sách, tăng cường kiểm soát chi tiêu và định hướng quyết định tài chính của mình. Ngoài ra, ứng dụng còn cung cấp các tính năng như hiển thị thông báo nhắc nhở chi tiêu cho người dung, cung cấp các biểu đồ cho người dung biết được các khoản thu chi của mình so với tháng trước, tuần trước giúp người dùng quản lý tài chính của mình một cách toàn diện và thuận tiện hơn.

3. Với sự phát triển của công nghệ và cuộc sống ngày càng phức tạp, việc quản lý tài chính cá nhân trở nên cần thiết hơn bao giờ hết. Và ứng dụng quản lý chi tiêu cá nhân được xem là một công cụ hữu ích để hỗ trợ người dùng trong việc quản lý tài chính của mình.

* 1. **Đối tượng sử dụng.**

Sau khi nghiên cứu tham khảo thị trường khách hàng, người dùng của các ứng dụng quản lý chi tiêu tương tự, chúng em xác định mục tiêu người dùng của ứng dụng Expense Manager hướng đến các đối tượng sau:

*1. Người đang lao động chính*:

Đây là những người có thu nhập đều đặn và cần quản lý chi tiêu của mình để có thể đầu tư và tiết kiệm cho tương lai. Họ có thể là các người lao động trẻ trong độ tuổi 22-28 đang củng cố tài chính để phục vụ, nâng cao chất lượng cuộc sống cá nhân, hay tiết kiệm chuẩn bị các cơ sở cho việc hôn nhân, lập gia đình. Hoặc cũng có thể là người lao động từ 28 - 40 tuổi trở đi phải quản lý tài chính không chỉ để đảm bảo cho cuộc sống cá nhân, mà phải chăm lo cho cuộc sống của cả gia đình, đặc biệt là chi phí sinh hoạt, học tập của con cái, có những người còn tiết kiệm một khoản tài chính lo cho tương lai sau này (mua nhà, mua xe, kết hôn,...) cho con cái của họ. Những người lao động trong độ tuổi 40 trở đi thường có xu hướng muốn tiết kiệm một khoản tài chính để đảm bảo nuôi dưỡng bản thân khi ở hết độ tuổi lao động trong xã hội.

*2. Sinh viên*:

Sinh viên đang theo học ở trường đại học thường là những đối tượng vừa xa gia đình và tập độc lập trong cuộc sống. Họ thường phải quản lý các khoản chi tiêu liên quan đến học phí, chi phí ăn uống, phương tiện, giải trí, v.v. Sinh viên có thể đi làm công việc part-time ngoài giờ học để có thêm thu nhập nhưng nguồn này thường hạn chế. Do vậy, sinh viên cần phải quản lý chi tiêu của mình để có thể sống tự lập, tiết kiệm, chi tiêu hợp lý và tránh nợ nần.

*3. Người đang tiết kiệm hoặc lên kế hoạch tài chính*:

Đây là nhóm đối tượng đặc biệt, có thể nằm trong phạm vi nhóm đối tượng người đang trong độ tuổi lao động chính hoặc sinh viên, nhưng ở họ phát sinh nhu cầu cần phải tiết kiệm hay quản lý tài chính trong một khoảng thời gian nhất định. Đối với những người này, việc quản lý chi tiêu là rất quan trọng để đạt được mục tiêu tiết kiệm và đầu tư, có thể họ đang cần chuẩn bị một khoản tiền, chẳng hạn cho cho một chuyến du lịch, công tác, mua nhà, mua đất, mua xe.

*4. Người muốn kiểm soát chi tiêu*:

Những người này đang vấn đề trong việc chi tiêu, họ tiêu xài quá mức, tiêu phung phí, có thể lớn hơn cả thu nhập cá nhân; những người này thường không nhớ được mình đã tiêu gì hoặc hoặc để ý những khoản tiền nhỏ mà mình tiêu hằng ngày cho đến khi cuối tháng những khoản này tổng kết lại quá lớn. Đối với những người này, việc sử dụng Expense Manager sẽ giúp họ đánh giá mức độ tiêu dùng của mình và giúp họ kiểm soát chi tiêu hợp lý hơn.

1. **Phương pháp và phạm vi nghiên cứu:**
   1. **Phương pháp nghiên cứu:**

1.Nghiên cứu về các quy trình phát triển ứng dụng phổ biến hiện nay như SDLC (Software Development Life Cycle), Agile, DevOps (Development and Operations), RAD (Rapid Application Development).

2. Đối với ứng dụng quản lý chi tiêu cá nhân của chúng ta, nhóm quyết định lựa chọn phương pháp RAD để phát triển ứng dụng nhanh chóng và hiệu quả. Các giai đoạn phát triển được tập trung vào việc xây dựng các tính năng chính và đưa ra thị trường một cách nhanh chóng.

3. Để đảm bảo tính chính xác và hiệu quả của ứng dụng, chúng tôi sẽ sử dụng phương pháp thăm dò ý kiến khách hàng và thu thập thông tin phản hồi từ người dùng. Từ đó, chúng tôi có thể đánh giá và cải thiện ứng dụng sao cho phù hợp với nhu cầu và mong đợi của người dùng.

* 1. **Phạm vi nghiên cứu:**

Chúng em tập trung vào phát triển một ứng dụng quản lý chi tiêu cá nhân trên nền tảng Android, hướng đến việc cung cấp các tính năng quản lý chi tiêu chuyên nghiệp và thuận tiện cho người dùng. Tuy nhiên, chúng em không bao gồm tính năng quản lý tài khoản ngân hàng trong phạm vi của nghiên cứu này. Nghiên cứu sẽ tập trung vào các trường hợp sử dụng ứng dụng của người dùng thông thường để đưa ra các cải tiến và tối ưu hóa trải nghiệm người dùng. Chúng em sẽ sử dụng phương pháp thăm dò ý kiến khách hàng và thu thập thông tin phản hồi từ người dùng để đảm bảo tính chính xác và hiệu quả của ứng dụng.

1. **Thiết kế và triển khai ứng dụng:**
   1. **Mô tả kiến trúc của ứng dụng:**

Kiến trúc của ứng dụng Expense Manager được xây dựng theo mô hình ba lớp(three-tier architecture). Kiến trúc này được chia thành ba phần chính:

1. Presentation Layer: Đây là lớp giao diện người dùng của ứng dụng, nơi người dùng tương tác trực tiếp với hệ thống. Lớp này sẽ đảm nhận việc hiển thị dữ liệu cho người dùng, cũng như thu thập thông tin từ người dùng để gửi đến lớp xử lý dữ liệu.

2. Business Logic Layer: Lớp xử lý dữ liệu là nơi thực hiện các tính toán, xử lý logic và quản lý dữ liệu của ứng dụng. Lớp này cũng đảm nhận việc truy cập và lưu trữ dữ liệu vào cơ sở dữ liệu.

3. Data Access Layer: Đây là lớp tương tác với cơ sở dữ liệu, nơi cung cấp các phương thức để thực hiện truy vấn và cập nhật dữ liệu trong cơ sở dữ liệu.  
  
Ngoài ra, ứng dụng cũng sử dụng một số công nghệ và tiêu chuẩn phổ biến như:

1. Ngôn ngữ lập trình: Chúng em sử dụng ngôn ngữ Java để xây dựng ứng dụng. Java là một ngôn ngữ lập trình phổ biến và được sử dụng rộng rãi trong việc phát triển ứng dụng.

2. Cơ sở dữ liệu: Ứng dụng sử dụng cơ sở dữ liệu quan hệ để lưu trữ dữ liệu của ứng dụng. Chúng tôi sử dụng hệ quản trị cơ sở dữ liệu SQLite vì nó là một hệ thống quản lý cơ sở dữ liệu miễn phí và phổ biến.

3. RESTful API: Chúng tôi sử dụng kiến trúc RESTful để thiết kế API cho ứng dụng. Kiến trúc này cho phép chúng tôi thiết kế các API đơn giản, dễ sử dụng và dễ dàng mở rộng trong tương lai.

4. Thư viện: Chúng tôi sử dụng các thư viện phổ biến để giúp xây dựng ứng dụng nhanh chóng và hiệu quả hơn, bao gồm: Spring Framework, Hibernate, và Gson.

Tóm lại, kiến trúc của ứng dụng quản lý chi tiêu cá nhân được xây dựng dựa trên phương pháp RAD để phát triển nhanh chóng và hiệu quả. Đối với việc đảm bảo tính chính xác và hiệu quả của ứng dụng, chúng tôi sử dụng phương pháp thăm dò ý kiến khách hàng và thu thập phản hồi từ người dùng để đánh giá và cải thiện ứng dụng sao cho phù hợp với nhu cầu và mong đợi của người dùng. Tuy nhiên, không đề cập rõ ràng về kiến trúc của ứng dụng. Thường thì trong phương pháp RAD, kiến trúc của ứng dụng được thiết kế đơn giản và tập trung vào các tính năng chính của sản phẩm, vì vậy đối với ứng dụng quản lý chi tiêu cá nhân, kiến trúc sẽ tập trung vào việc xây dựng các tính năng quản lý chi tiêu, bao gồm lưu trữ, phân loại, thống kê chi tiêu, và hiển thị thông tin

* 1. **Mô tả các chức năng và tính năng của ứng dụng:**

1. Quản lý và theo dõi chi tiêu:  
Người dùng có thể dễ dàng thêm, sửa và xóa các khoản chi tiêu một cách nhanh chóng và tiện lợi. Các khoản chi tiêu này sẽ được lưu trữ và hiển thị trong ứng dụng để người dùng có thể dễ dàng theo dõi các khoản chi tiêu của mình.

2. Tạo danh sách các khoản chi tiêu và phân loại chúng để dễ dàng nhận biết:  
Ứng dụng cho phép người dùng tạo danh sách các khoản chi tiêu và phân loại chúng theo các nhóm khác nhau như thức ăn, giải trí, đi lại...vv. Giúp người dùng dễ dàng nhận biết và quản lý các khoản chi tiêu của mình.

3. Ngoài ra, ứng dụng còn cho phép người dùng thiết lập ngân sách và mục tiêu chi tiêu, giúp họ quản lý tài chính cá nhân hiệu quả hơn:  
Ứng dụng cung cấp tính năng thiết lập ngân sách và mục tiêu chi tiêu, giúp người dùng quản lý tài chính cá nhân hiệu quả hơn. Người dùng có thể đặt một ngân sách cụ thể cho mỗi loại khoản chi tiêu hoặc cho toàn bộ chi tiêu trong một tháng.

4. Thêm các ghi chú vào các khoản chi tiêu để giúp người dùng nhớ lại các khoản chi tiêu đó:  
Người dùng có thể thêm các ghi chú vào các khoản chi tiêu để giúp nhớ lại các khoản chi tiêu đó. Điều này sẽ giúp người dùng có thể đánh giá lại các khoản chi tiêu của mình và điều chỉnh lại chi tiêu của mình nếu cần thiết.

5. Tính toán tổng số tiền chi tiêu và so sánh với ngân sách đã đặt ra giúp người dùng hiểu rõ hơn về tính trạng tài chính của mình:  
Ứng dụng tính toán tổng số tiền chi tiêu và so sánh với ngân sách đã đặt ra giúp người dùng hiểu rõ hơn về tính trạng tài chính của mình. Người dùng có thể xem tổng số tiền chi tiêu của mình trong một tháng hoặc trong một khoảng thời gian nhất định.

6. Cung cấp các báo cáo và biểu đồ để thể hiện trực quan về tình hình tài chính của người dùng:  
Cung cấp các báo cáo và biểu đồ: Ứng dụng cung cấp các báo cáo và biểu đồ để thể hiện trực quan về tình hình tài chính của người dùng. Bằng cách sử dụng các biểu đồ và báo cáo này, người dùng có thể dễ dàng theo dõi các khoản chi tiêu của mình và đưa ra những quyết định tài chính phù hợp. Các báo cáo và biểu đồ có thể bao gồm:

1. Biểu đồ cột: thể hiện tổng số tiền chi tiêu của người dùng theo các danh mục chi tiêu khác nhau, giúp người dùng dễ dàng nhận biết khoản chi tiêu nào chiếm phần lớn nhất trong tài khoản của mình.

2. Biểu đồ tăng trưởng: thể hiện sự thay đổi của số tiền chi tiêu theo thời gian, giúp người dùng theo dõi được xu hướng chi tiêu của mình.

3. Biểu đồ so sánh: so sánh giữa số tiền chi tiêu của người dùng trong một khoảng thời gian cụ thể với một khoảng thời gian khác, giúp người dùng đánh giá hiệu quả của việc quản lý tài chính của mình trong thời gian dài hạn.

**7.** Cung cấp các tính năng nhắc nhở hoặc thông báo khi người dùng đến gần hoặc vượt quá ngân sách đã đặt ra.

Tính năng nhắc nhở người dung khi chi tiêu gần đến ngân sách giúp cho người dùng theo dõi tiến độ của ngân sách và tránh chi tiêu vượt quá mức đã đặt ra. Khi người dùng tiêu thụ gần đến ngưỡng ngân sách hoặc đã vượt quá ngân sách, ứng dụng sẽ gửi thông báo để cảnh báo. Điều này giúp người dùng có thể kiểm soát tình hình tài chính của mình một cách chủ động và tránh chi tiêu quá mức.

* 1. **Mô tả cách thức triển khai và phát triển ứng dụng.**

1. **Kết quả và đánh giá.**
2. **Kiểm thử và đánh giá hiệu quả của ứng dụng.**
3. **So sánh với các ứng dụng quản lý chi tiêu cá nhân khác.**
4. **Những vấn đề cần cải thiện và phát triển trong tương lai.**

Để ứng dụng quản lý chi tiêu cá nhân ngày càng hoàn thiện và đáp ứng tốt nhu cầu của người dùng, cần có sự cải tiến liên tục và phát triển đầy đủ tính năng mới, đồng thời bảo đảm được tính bảo mật và tương thích trên nhiều phiên bản Android khác nhau.

* 1. **Giao diện người dùng.**

Giao diện người dùng cần được cải tiến để trở nên thân thiện hơn với người dùng, dễ sử dụng hơn và thích hợp với nhiều loại màn hình khác nhau.

* 1. **Tính năng.**

Ứng dụng cần được bổ sung thêm nhiều tính năng và chức năng hơn để đáp ứng nhu cầu của người dùng, ví dụ như tính năng kết nối với tài khoản ngân hàng để tự động cập nhật các giao dịch mới nhất.

* 1. **Bảo mật.**

Bảo mật thông tin cá nhân của người dùng là một vấn đề quan trọng, do đó ứng dụng cần được bảo vệ bằng các biện pháp bảo mật như mã hóa dữ liệu và xác thực người dùng.

* 1. **Tương thích.**

Ứng dụng cần được tối ưu để hoạt động tốt trên các phiên bản Android khác nhau, từ phiên bản cũ đến phiên bản mới nhất.

* 1. **Phát triển:**

Để cải tiến và phát triển ứng dụng, cần đầu tư thêm vào nghiên cứu và phát triển các công nghệ mới để đáp ứng nhu cầu ngày càng tăng của người dùng.

1. **Kết luận:**
2. **Tài liệu tham khảo:**

1. XÂY DỰNG KẾ HOẠCH PHÁT TRIỂN MOBILE APP | CO-WELL Asia .

2.Quy trình thiết kế phần mềm toàn tập từ A-Z dành cho bạn (got-it.ai).

3.Tìm hiểu về Quy trình phát triển phần mềm (SDLC) | VietnamWorks InTECH.