**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**---o0o---**



**TIỂU LUẬN CHUYÊN NGÀNH**

GVHD : TS. Đặng Thanh Dũng

SVTH : Nguyễn Thị Bích Tuyền

Huỳnh Tiến Huy

MSSV : 10110196

10110209

LỚP : 10110CLC

**Tp. Hồ Chí Minh, năm 2013**

Lời cm ơn

Trên thực tế không có sự thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp của người khác. Trong suốt thời gian từ khi bắt đầu học tập ở giảng đường đại học đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý thầy cô, gia đình và bạn bè.

Với lòng biết ơn sâu sắc nhất, em xin gửi đến quý thầy cô ở Khoa Đào tạo Chất Lượng Cao – Trường Đại Học Sư Phạm Kỹ Thuật đã cùng với tri thức và tâm huyết của mình để truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường. Và đặc biệt, trong học kỳ này, Khoa đã tổ chức cho chúng em được tiếp cận với môn học mà theo em là rất hữu ích đối với sinh viên ngành Công Nghệ Thông Tin. Em xin chân thành cảm ơn TS. Đặng Thanh Dũng đã tận tâm hướng dẫn chúng em qua từng buổi học trên lớp cũng như những buổi nói chuyện, thảo luận về lĩnh vực sáng tạo trong tiểu luận chuyên ngành. Nếu không có những lời hướng dẫn, dạy bảo của thầy thì em nghĩ bài báo cáo này của em rất khó có thể hoàn thiện được. Một lần nữa, em xin chân thành cảm ơn thầy.

Bước đầu đi vào thực tế, tìm hiểu về các lĩnh vực trong tiểu luận chuyên ngành, kiến thức của em còn hạn chế và còn nhiều bỡ ngỡ. Do vậy, không tránh khỏi những thiếu sót là điều chắc chắn, em rất mong nhận được những ý kiến đóng góp quý báu của quý Thầy Cô để kiến thức của em trong lĩnh vực này được hoàn thiện hơn.

Sau cùng, em xin kính chúc quý thầy cô trong Khoa Đào tạo Chất Lượng Cao và TS. Đặng Thanh Dũng thật dồi dào sức khỏe, niềm tin để tiếp tục thực hiện sứ mệnh cao đẹp của mình là truyền đạt kiến thức cho thế hệ mai sau.

Trân trọng.

TP. HCM, ngày tháng năm 20013

Sinh viên thực hiện

(ký và ghi họ tên)

# Chương 1 - Giới thiệu

Bản đồ tư duy (Mindmap) là phương pháp được đưa ra như là một phương tiện mạnh để tận dụng khả năng ghi nhận hình ảnh của [bộ não](http://vi.wikipedia.org/wiki/N%C3%A3o). Đây là cách để ghi nhớ chi tiết, để tổng hợp, hay để phân tích một vấn đề ra thành một dạng của lược đồ phân nhánh. Khác với [máy tính](http://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh), ngoài khả năng ghi nhớ kiểu tuyến tính (ghi nhớ theo 1 trình tự nhất định chẳng hạn như trình tự biến cố xuất hiện của 1 câu truyện) thì não bộ còn có khả năng liên lạc, liên hệ các dữ kiện với nhau. Phương pháp này khai thác cả hai khả năng này của bộ não.

Phương pháp này có lẽ đã được nhiều người Việt biết đến nhưng nó chưa bao giờ được hệ thống hóa và được nghiên cứu kĩ lưỡng và phổ biến chính thức trong nước mà chỉ được dùng tản mạn trong giới sinh viên học sinh trước các mùa thi.

Đây là một kĩ thuật để nâng cao cách ghi chép. Bằng cách dùng giản đồ ý, tổng thể của vấn đề được chỉ ra dưới dạng một hình trong đó các đối tượng thì liên hệ với nhau bằng các đường nối. Với cách thức đó, các dữ liệu được ghi nhớ và nhìn nhận dễ dàng và nhanh chóng hơn.

Thay vì dùng chữ viết để miêu tả một chiều biểu thị toàn bộ cấu trúc chi tiết của một đối tượng bằng hình ảnh hai chiều. Nó chỉ ra dạng thức của đối tượng, sự quan hệ hỗ tương giữa các khái niệm (hay ý) có liên quan và cách liên hệ giữa chúng với nhau bên trong của một vấn đề lớn.

# Chương 2 - Giới thiệu về công nghệ được dùng trong tiểu luận

## Lập trình Android

### Tổng quan

#### Định nghĩa Android

Android là hệ thống tuần tự các phần mềm cho thiết bị di động bao gồm hệ điều hành, các trình điều khiển phần cứng và các ứng dụng cơ bản.

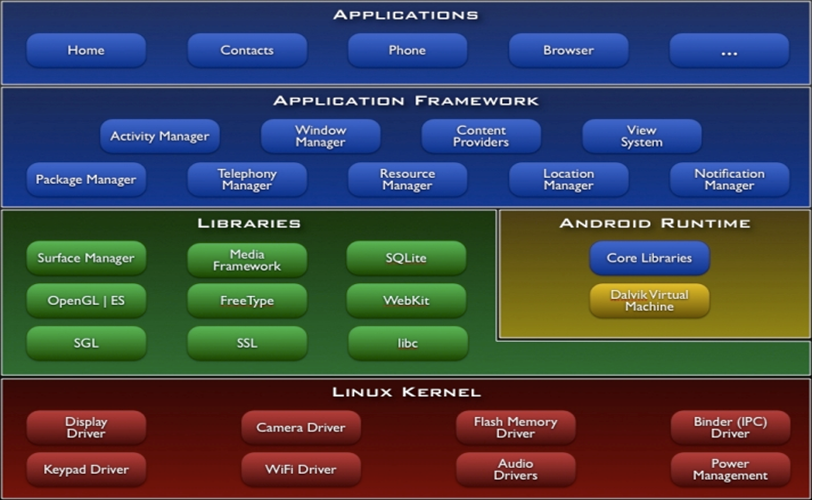
#### Tính năng và kiến trúc của android

##### Tính năng

Android gồm các tính năng như sau:

* Khung ứng dụng: cho phép tái sử dụng lại các ứng dụng đã có sẵn.
* Máy ảo Dalvik: là máy ảo được tinh chỉnh dành riêng cho thiết bị di động chạy trên nền Android. Các chương trình Java đa phần chạy trên máy ảo Java; trong khi đó, Dalvik là máy ảo dùng để chạy ứng dụng Java trên di động Android.
* Trình duyệt web được tích hợp sẵn: được dựa trên bộ mã nguồn mở Webkit.
* Hệ thống xử lý đồ họa được tối ưu hóa: gồm thư viện đồ họa 2D, thư viện đồ họa 3D dựa trên nền OpenGL (tùy theo từng loại phần cứng đồ họa của di động).
* SQLite: dùng cho lưu trữ cơ sở dữ liệu cục bộ.
* Media: hỗ trợ định dạng các tập tin âm thanh, video, và hình ảnh thông thường như: MPEG4, H.264, MP3, AAC, AMR, JPG, PNG và GIF.
* Điện thoại GSM với các chức năng nghe gọi, tin nhắn.
* Hỗ trợ hầu hết các phương thức kết nối Bluetooth, EDGE, 3G, và Wifi (tùy thuộc theo phần cứng điện thoại) .
* Hỗ trợ các phần cứng: Camera, GPS, la bàn số, vả cảm nhận gia tốc (tùy thuộc theo phần cứng điện thoại).
* Môi trường phát triển đa dạng: bao gồm thiết bị máy ảo, công cụ đểkiểm lỗi (debug), bộ nhớ và công cụ cung cấp thông tin giả lập và plugin cho Eclipse.

##### Kiến trúc



Hình 1 Kiến trúc Android theo từng tầng

##### Ứng dụng

Android đã bao gồm một bộ các ứng dụng chính như trình nhận email, trình nhận tin SMS, lịch, bản đồ, trình duyệt web, danh bạ và các chương trình phụ trợ khác. Tất cả các ứng dụng sử dụng ngôn ngữ lập trình Java.

##### Khung ứng dụng

Nhờ Android cung cấp nền tảng phát triển mở, các nhà phát triển có thể xây dựng các ứng dụng đa dạng và sáng tạo. Các nhà phát triển có thể dùng các phần cứng thiết bị, lấy thông tin vị trí, chạy ngầm các dịch vụ, thiết lập cảnh báo, thêm thông báo vào thanh trạng thái và nhiều cái khác nữa.

Các nhà phát triển có thể tác động sâu vào khung API mà các ứng dụng chính đang dùng. Kiến trúc ứng dụng được thiết kế đơn giản để tái sử dụng lại các thành phần trong ứng dụng đó. Như ứng dụng mua sắm xem giá cả có thể sử dụng lại ứng dụng chụp mã vạch BarcodeZXing.

Tất cả các ứng dụng nền là một bộ các dịch vụ và hệ thống, bao gồm:

* Bộ các View đa dạng và có thể tùy biến bao gồm: danh sách (listView), lưới (gridView), textbox, nút (Button), và các View khác.
* Content Provider: cho phép ứng dụng lấy dữ liệu từ ứng dụng khác (ví dụ danh bạ) hoặc chúng chia sẽ dữ liệu với nhau..v .v . . .
* Resource Manager: cho phép truy xuất các nguồn tài nguyên cục bộ như chuỗi tĩnh, hình ảnh và tập tin khai báo giao diện, .v. v…
* Notification Manager: cho phép ứng dụng trình bày cảnh báo tùy biến trên thanh trạng thái.
* Activity Manager: quản lý vòng đời ứng dụng.

##### Thư viện

Android bao gồm bộ thư viện C/C++ được dùng bởi ứng dụng chính của hệ thống Android. Thông qua khung ứng dụng (Application framework), các nhà phát triển có thể tiếp cận sâu vào lõi thư viện bao gồm:

* Thư viện hệ thống C: được rút ra từ thư viện hệ thống C chuẩn (libc).
* Thư viện media: dựa tên OpenCORE của PacketVideo, một bộ thư viện hỗ trợ đọc, ghi lại nhiều định dạng âm thanh, video, hình ảnh như: MPEG4, H.264, MP3, AAC, AMR, JPG, và PNG.
* Surface Manager: quản lí giao diện bao gồm lớp giao diện 2D và 3D.
* LibWebCore: là bộ thư viện trình duyệt web mới hỗ trợ trình duyệt Android và WebView.
* SGL: hệ thống đồ họa 2D cơ sở.
* Thư viện 3D: dựa trên OpenGL ES 1.0 API; thư viện gồm phần cứng hỗ trợ 3D và phần mềm hỗ trợ tối ưu 3D.
* FreeType: trình bày kiểu chữ và hình ảnh.
* SQLite: hệ thống cơ sở dữ liệu có kích thước nhỏ nhưng mạnh mẽ cho tất cả ứng dụng.

##### Android runtime

Android bao gồm bộ thư viện chính, cung cấp hầu hết các tính năng cho bộ thư viện kể trên và dùng ngôn ngữ Java.

Mỗi ứng dụng Android chạy trên tiến trình riêng của nó với một thể hiện (instance) riêng của nó trên máy ảo Dalvik. Dalvik giúp thiết bị chạy nhiều máy ảo cùng lúc. Máy ảo Dalvik chạy nhiều tập tin định dạng .dex được tối ưu hóa cho máy có bộ nhớ thấp. Máy ảo được khai báo và chạy các lớp được biên dịch bởi trình biên dịch ngôn ngữ Java mà mới vừa được chuyển sang định dạng .dex bởi công cụ có sẵn “dx”.

Máy ảo Dalvik lệ thuộc vào lõi Linux cho các tính năng cơ bản như là luồng và quản lý bộ nhớ mức độ thấp.

##### Lõi Linux

Android dựa trên nền Linux phiên bản 2.6 cho các chương trình hệ thống chính như an ninh, quản lý bộ nhớ, quản lý tiến trình, mạng và mẫu trình điều khiển thiết bị. Nhân Linux cũng đóng vai trò như là l ớp ảo giữa phần cứng và phần còn lại của kiến trúc Android.

### Giới thiệu Java SE JDK, Android SDK , ADT, Eclipse

#### Giới thiệu chung

Để viết ứng dụng Android, chúng ta cần Java SE Development Kit (JDK), Android SDK, môi trường phát triển phần mềm (IDE) và các công cụ hỗ trợ phát triển phần mềm Android trên IDE đó.  
Một cách “ngông cuồng” mà nói, ta có thể lập trình trên bất cứ trình soạn thảo văn bản nào nhưng làm vậy sẽ mất rất nhiều thời gian trong khi lập trình ứng dụng Android với IDE và các công cụ đi kèm có sự hỗ trợ cao về các công cụ hỗ trợlập trình giao diện, kiểm lỗi, hỗ trợ biên dịch chương trình,… Kể từ các mục sau trở đi ta đi sâu vào tính năng từng thành phần và cách thức tải về đối từng thành phần.

#### Giới thiệu Java SE Development Kit (JDK)

Java SE là hệ thống nền tảng Java và thường được cài đặt lên các máy trạm, máy PC hay laptop để tiện lợi cho các nhà lập trình tiến hành phát triển các ứng dụng có nền tảng Java. Cho nên dù muốn dù không thì khi lập trình với ứng dụng Java trên máy trạm hay laptop hay PC thì trên máy phải có Java SE Development Kit (JDK).

Android SDK là bộ phát triển ứng dụng Android trên nền Java và nó đòi hỏi Java SE JDK phiên bản 5 hay cao hơn. Ta có thể tải về Java SE JDK tại địa chỉ sau: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. Hiệp hội các thiết bị cầm tay mở và Google chọn Java là nền tảng để xây dựng Android không chỉ có lí do là nó là mã nguồn mở mà còn lí do quan trọng không kém là nó được hỗ trợ trên nhiều hệ điều hành khác nhau từ Windows, Macintosh (x86), Solaris cho đến Linux.

#### Giới thiệu Eclipse IDE

Về nguyên tắc mà nói, bạn có thể dùng bất kì môi trường phát triển phần mềm Java nào mà bạn thích để phát triển ứng dụng. Thậm chí bạn có thể dùng NotePad của Windows để viết các mã lệnh nhưng làm thế sẽ tốn rất nhiều thời gian và khá phức tạp. Đối với Android, thì theo như Google khuyến cáo dùng Eclipse để có sự hỗ trợ tối đa từ các công cụ lập trình Android mà Google cung cấp. Theo

chúng tôi biết, Netbean cũng có thể lập trình được ứng dụng Android nhưng về mặt ổn định và hỗ trợ thì chúng tôi chưa phân tí ch kĩ và dùng thử. Tổ chức OHA và Google khuyến cáo dùng Eclipse vì Eclipse có thể thực thi trên nhiều hệ điều hành khác nhau mà không cần cài đặt trên chúng. Toàn bộ gói ứng dụng Eclipse có thể tải về tại địa chi sau: <http://www.eclipse.org/downloads/>.

Như vậy có Elipse và JDK ta có thể lập trình ứng dụng Java ngay trên các máy trạm được. Do chúng ta lập trình ứng dụng Android nên cần JDK để có nền tảng để lập trình Java, có IDE Eclipse để lập trình ứng dụng trên đó, bây giờ ta cần thêm một Android SDK để hỗ trợ viết ứng dụng đặc thù Android với những công cụ hỗ trợ của nó cùng với Eclipse có thể viết ứng dụng Android hoàn chỉnh. Để Eclipse và Android SDK hiểu nhau thì ta cần có công cụ phát triển phần mềm Android có tác “mồi” để Eclipse và Android SDK hiểu nhau. Mặc dù theo lí thuyết mà nói, ADT là những công cụ nằm trong Android SDK nhưng ADT phải được cài đặt hay đồng bộ hóa trước với Eclipse để “tạo đà” để lấy tất cả các bảng cập nhật nền tảng Android về cho Android SDK.

#### Giới thiệu plugin ADT

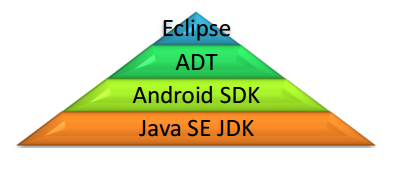
ADT là plugin mở rộng được tích hợp vào môi trường phát triển phần mềm Eclipse. Nó cho phép ta tạo và kiểm lỗi ứng dụng Android dễ dàng hơn và nhanh chóng hơn. Nếu bạn dùng Eclipse. Plugin ADT sẽ giúp bạn được các công việc sau:

* Truy xuất trực tiếp đến các công cụ phát triển Android khác bên trong Eclipse. Ví dụ, ADT cho phép truy xuất đến chức năng của công cụ DDMS như: chụp màn hình, quản lý liên kết với thiết bị, đặt điểm dừng để kiểm lỗi, và xem thông tin các luồng và tiến trình trự tiếp từ Eclipse.
* Nó cung cấp New Project Wizard, mà giúp ta tạo và thiết lập tất cả các tập tin cơ bản cần thiết cho một ứng dụng Android mới.
* Nó tự động hóa và làm đơn giản hóa tiến trình xây dựng ứng dụng Android.
* Nó cung cấp trình biên tập mã lệnh Android mà giúp bạn viết lệnh XML cho tập tin AndroidManifest và các tập tin tài nguyên khác.
* Xuất ra tập tin .apk “đã đăng ký” (signed) hoặc “chưa đăng ký” (unsigned) để mà phân phối ứng dụng cho người dùng cuối.

Nói chung, dùng Eclipse với plugin ADT, chúng ta có thể lợi dụng tính năng cập nhật từ xa của Eclipse. Bằng cách thiết lập một trang cập nhật từ xa, bạn có thể dễ dàng tải về, cài đặt, và kiểm tra những cập nhật ADT. Thật chất ADT không cài đặt riêng lẻ hay chạy trên tập tin thực thi mà nó được “đính vào” Eclipse dưới dạng plugin.Ta tải về ADT dưới dạng plugin dành cho Eclipse thông qua 2 hình thức là:

* Cập nhật về 2 công cụ thông dụng là DDMS và ADT lên Eclipse qua đường liên kết https://dl-ssl.google.com/android/eclipse/ thì sẽ có mục chọn như đã nói là DDMS và ADT. Hai phần này sẽ được đính kèm vào Android và đây là hai plugin mồi để tải về nền tảng Android dựa trên thư mục Android SDK mà ta thiết lập sau đó.
* Hay tải về tập tin ADT tại http://developer.Android.com/SDK/eclipse-adt.html và sau đó đính kèm nó vào Eclipse cũng qua đường Help/Install New Software có mục Add trên khung thoại Install thì ta sẽ Add achive cho nó. Ngay lập tức Eclipse sẽ hỗ trợ các công cụ và trình đơn cần thiết để thiết lập Android SDK với nền tảng hệ điều hành các các công cụ khác của Android SDK.

ADT là công cụ hỗ trợ lập trình đắc lực được gắn thêm trên Eclipse để các nhà phát triển Android làm việc tốt hơn. Ta cũng có thể nhận định ADT như là chiếc cầu nối giữa Eclipse và Android SDK. Ta tóm tắt lại theo mô hình sau:



Hình 2 Biểu diễn trình tự và quan hệ giữa Java SE, Android SDK, ADT và Eclipse.

Một khi thiết lập ADT xong, ta sẽ tải về tập tin Android SDK để nhờ ADT trên Eclipse để thiết lập giữa Eclipse IDE và Android SDK.

#### Giới thiệu Android SDK

Android SDK là nền tảng xây dựng ứng dụng Android trên nền Java và được cung cấp trực tiếp trên trang web: <http://developer.Android.com/SDK/index.html> với 3 phiên bản dành cho ba hệ điều hành khác nhau gồm Windows, Mac và Linux.

Android SDK là một tập tin dạng nén .zip và khi giải nén ra thì ta sẽ có một thư mục /android-sdk-windows cùng với các thư mục /tools với các công cụ đi kèm phía bên trong nhưng thư mục /platform thì vẫn còn trống chưa có gì. Nhiệm vụ duy nhất của ta hiện giờ là kết nối Eclipse và Android SDK hiểu nhau sau đó cập nhật lại nền tảng và các công cụ cho Android SDK.

Một khi thiết lập tham chiếu đến thư mục Android SDK cho Eclipse thì trên Eclipse có mục Android SDK and AVD Manager để tải về các thành phần nền tảng mà chúng ta cần như là nền tảng Android, các SDK add-on, các công cụ, và tài liệu hướng dẫn.

## Google API

# Chương 3 - Thiết kế giao diện

# Chương 4 - Thiết kế chương trình và CSDL

# Chương 5 - Cài đặt & Demo

# Chương 6 - Kết luận và hướng phát triển