

**ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM
KHOA TIN**

-----∞ • ∞-----

Đề tài:

Tìm hiểu công nghệ Android - Xây dựng ứng dụng Lịch Vạn Sự Vạn Niên trên mobile.

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN KHOA HỌC

**GVHD : Lê Văn Mỹ
SVTH : Nguyễn Thị Diễm Hương
Đà Nẵng năm:2012**

LỜI CẢM ƠN

Em xin chân thành cảm thầy cô trong khoa Tin học cũng như các thầy cô giảng dạy trong trường Đại học Sư phạm – Đại học Đà Nẵng đã truyền đạt những kiến thức quý báu cho em trong những năm học vừa qua.

Đặc biệt, Em xin chân thành cảm ơn thầy giáo Lê Văn Mỹ đã tận tình giúp đỡ, hướng dẫn và cung cấp tài liệu liên quan đến đề án này.

Xin chân thành cảm ơn các anh chị và các bạn trong khoa Tin học đã ủng hộ, giúp đỡ, chia sẻ kiến thức, kinh nghiệm và tài liệu có được cho em trong quá trình nghiên cứu và thực hiện đề tài.

Một lần nữa em xin chân thành cảm ơn !

Đà Nẵng, ngày 07 tháng 05 năm 2012

Sinh viên thực hiện

Nguyễn Thị Diễm Hương

LỜI CAM ĐOAN

Em xin cam đoan :

- 1 Những nội dung trong luận văn này là do em thực hiện dưới sự hướng dẫn trực tiếp của Thầy giáo Lê Văn Mỹ.*
- 2 Mọi tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.*
- 3 Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo, hay gian trá, em xin chịu hoàn toàn trách nhiệm.*

Sinh viên

Nguyễn Thị Diễm Hương

MỤC LỤC

MỞ ĐẦU	1
1. LÝ DO CHỌN ĐỀ TÀI.....	1
2. MỤC TIÊU, NHIỆM VỤ	1
2.1 Mục tiêu.....	1
2.2 Nhiệm vụ	2
3. ĐỐI TƯỢNG, PHẠM VI NGHIÊN CỨU.....	2
3.1 Đối tượng.....	2
3.2 Phạm vi nghiên cứu	2
4. PHƯƠNG PHÁP NGHIÊN CỨU.....	2
4.1 Về mặt lý thuyết.....	2
4.2 Công cụ xây dựng đề tài.....	3
5. Ý NGHĨA KHOA HỌC, THỰC TIỄN CỦA ĐỀ TÀI	3
5.1 Ý nghĩa khoa học	3
5.2 Ý nghĩa thực tiễn	3
6. BỐ CỤC.....	3
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	4
1. TỔNG QUAN VỀ HỆ ĐIỀU HÀNH ANDROID.....	4
1.1 Lịch sử Android.....	4
1.2 Các phiên bản hệ điều hành Android (Android plaform)	5
1.3 Ứng dụng của Android và sự phát triển trong tương lai	6
2. KIẾN TRÚC HỆ ĐIỀU HÀNH ANDROID	7
2.1 Các thành phần cấu tạo nên Android (Android Architecture)	7
2.2 Activity.....	10
2.3 Broadcast receivers	14
2.4 DVM	15
3. LẬP TRÌNH ỨNG DỤNG ANDROID.....	15
3.1 Thiết lập môi trường phát triển ứng dụng.....	15
3.2 Các thành phần trong Android Project	16
4. LÝ THUYẾT CỦA BÀI TOÁN CHUYỂN ĐỔI NGÀY ÂM DƯƠNG 18	18
4.1 Cơ sở lý thuyết tính lịch âm Việt Nam	18
4.2 Các hàm cơ bản chuyển đổi ngày âm dương	20
CHƯƠNG 2: PHÂN TÍCH HỆ THỐNG	24
1. SƠ ĐỒ USE CASE	24
1.1 Sơ đồ use case tổng quát	24
1.2 Mô tả các ca sử dụng hay chức năng của nó.....	24
2. SƠ ĐỒ TUẦN TỰ	25
2.1 Quá trình xem lịch âm dương.....	25
2.2 Quá trình xem tham khảo trong ngày	26
2.3 Quá trình xem tuổi	27
2.4 Quá trình chọn ngày.....	29
2.5 Quá trình xem tử vi	30
3. TỔ CHỨC CÁC LỚP	31
3.1 Lớp ReadFileText.....	31
3.2 Lớp DateProcessing	31
3.3 Lớp CheckingForBuilding	32

3.4	Lớp MyDate	33
3.5	Các lớp giao diện	33
4.	TỔ CHỨC CƠ SỞ DỮ LIỆU	35
CHƯƠNG 3: TRIỂN KHAI VÀ KẾT QUẢ CHẠY DEMO.....		36
1.	MÀN HÌNH HIỂN THỊ LỊCH ÂM, LỊCH DƯƠNG.....	36
2.	MÀN HÌNH XEM THẢO TRỌNG NGÀY	37
3.	MÀN HÌNH XEM TUỔI	38
4.	MÀN HÌNH CHỌN NGÀY ÂM, NGÀY DƯƠNG	39
5.	MÀN HÌNH XEM TỬ VI	40
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN		42
1.	KẾT QUẢ ĐẠT ĐƯỢC	42
1.1.	Về mặt lý thuyết	42
1.2	Về thực nghiệm.....	42
2.	HẠN CHẾ.....	42
3.	HƯỚNG PHÁT TRIỂN ĐỀ TÀI.....	43
PHỤ LỤC.....		44
TÀI LIỆU THAM KHẢO.....		51
TÓM TẮT LUẬN VĂN TỐT NGHIỆP.....		52

DANH MỤC HÌNH ẢNH

Hình 1.1. Biểu tượng cho một số phiên bản	5
Hình 1.2. Samsung Galaxy S II.....	6
Hình 1.3. Kiến trúc hệ điều hành Android.....	7
Hình 1.4. Activity Lifecycle	13
Hình 1.5. Tạo mới 1 ứng dụng Android	16
Hình 2.1. Sơ đồ ca sử dụng tổng quát	24
Hình 2.2. Sơ đồ tuần tự quá trình xem lịch âm dương	26
Hình 2.3. Sơ đồ tuần tự quá trình xem nên kiêng trong ngày.....	26
Hình 2.4. Sơ đồ tuần tự quá trình xem hướng xuất hành	27
Hình 2.5. Sơ đồ tuần tự quá trình xem tuổi làm nhà	27
Hình 2.6. Sơ đồ tuần tự quá trình xem tuổi kết hôn	28
Hình 2.7. Sơ đồ tuần tự quá trình xem vận hạn của tuổi	28
Hình 2.8. Sơ đồ tuần tự quá trình chọn ngày dương	29
Hình 2.9. Sơ đồ tuần tự quá trình chọn ngày âm	29
Hình 2.10. Sơ đồ tuần tự quá trình xem tử vi đông phương.....	30
Hình 2.11. Sơ đồ tuần tự xem tử vi tây phương.....	30
Hình 2.12. Sơ đồ dữ liệu.....	35
Hình 3.1. Màn hình xem lịch dương	36
Hình 3.2. Màn hình xem lịch âm.....	37
Hình 3.3. Màn hình nên kiêng và hướng xuất hành trong ngày	37
Hình 3.4. Màn hình chọn tuổi, xem tuổi làm nhà, kết hôn và vận hạn	38
Hình 3.5. Màn hình thông tin về kết hôn, vận hạn, làm nhà	39
Hình 3.6. Màn hình chọn ngày dương và hiển thị.....	39
Hình 3.7. Màn hình chọn ngày âm và hiển thị.....	40
Hình 3.8. Màn hình chọn tuổi và xem tính cách	40
Hình 3.9. Màn hình chọn chòm sao và xem tính cách	41

DANH MỤC BẢNG

Bảng 1.1: Bảng tiết khí	19
Bảng 2.1: Chức năng của từng ca sử dụng	25
Bảng 2.2: Các phương thức lớp ReadFileText	31
Bảng 2.3: Các phương thức lớp ProcessDate	32
Bảng 2.4: Các phương thức lớp CheckingForBuilding.....	33
Bảng 2.5: Các phương thức lớp MyDate.....	33
Bảng 2.6: Các lớp giao diện cơ bản.....	34

DANH MỤC TỪ VIẾT TẮT

- User: người sử dụng
- State: trạng thái
- Menu: trình đơn
- DVM: Dalvik virtual machine
- JVM: Java virtual machine
- JDK: Java Development Kit
- API: Application Programming Interface
- SDK: Software Development Kit
- GPRS: General Packet Radio Service
- WAP: Wireless Access Protocol
- SOAP: Simple Object Access Protocol
- UI: User Interface
- VM: Virtual Machine
- IDE: Integrated Development Environment
- J2ME: Java 2 Micro Edition
- ADT: Android Development Tools
- J2SE: Java 2 Standard Edition

MỞ ĐẦU

1. LÝ DO CHỌN ĐỀ TÀI

Hiện nay có rất nhiều công nghệ mới phát triển song song với việc phát triển công nghệ thông tin như Bluetooth, Wireless, WAP, SOAP,... nhằm giúp công nghệ thông tin ngày càng thân thiết với người dùng hơn. Một trong những công nghệ góp phần không nhỏ trong việc kết nối con người với thông tin cũng như con người với con người là công nghệ di động. Với tốc độ phát triển hiện nay và những lợi ích to lớn của công nghệ di động, có thể thấy nó có ảnh hưởng rất lớn đến cuộc sống của con người. Không giống như trước đây những chiếc điện thoại chỉ có chức năng rất đơn giản là đàm thoại, điện thoại hiện nay còn có thêm rất nhiều chức năng, ứng dụng khác như: email, truy cập Internet, video, nghe nhạc, chơi game, ... đồng thời với nó là sự phát triển vũ bão của các dịch vụ gia tăng trên điện thoại di động dựa trên công nghệ WAP và SOAP.

Vì những lý do trên, em chọn đề tài “Tìm hiểu công nghệ Android - Xây dựng ứng dụng Lịch Vạn Sự Vạn Niên trên mobile” để làm luận văn tốt nghiệp. Việc xây dựng một ứng dụng như thế nhằm tìm hiểu thêm một công nghệ mới trong lĩnh vực công nghệ di động và áp dụng những kiến thức mà em đã được học ở trường trong thời gian qua để áp dụng vào thực tiễn với mong muốn sẽ góp phần giải quyết một số nhu cầu cần thiết cho người sử dụng điện thoại di động.

2. MỤC TIÊU, NHIỆM VỤ

2.1 Mục tiêu

Xây dựng ứng dụng Lịch Vạn Sự Vạn Niên trên mobile cho điện thoại di động sử dụng hệ điều hành Android có một số chức năng như sau:

- Hiện thị lịch dương, xem ngày âm lịch, giờ hoàng đạo.
- Xem tuổi để làm nhà, kết hôn, vận hạn.
- Một số tham khảo trong ngày

- Chọn một ngày dương hay một ngày âm bất kỳ.
- Xem tử vi đông phương, tây phương.

2.2 Nhiệm vụ

- Tìm hiểu công nghệ Android
- Tìm hiểu các thuật toán tính ngày âm, ngày dương
- Lưu dữ liệu dưới dạng các file text

3. ĐỐI TƯỢNG, PHẠM VI NGHIÊN CỨU

3.1 Đối tượng

Lịch vạn sự vạn niên thế kỉ XXI và các điện thoại có hệ điều hành Android

3.2 Phạm vi nghiên cứu

Ứng dụng trên các thiết bị di động sử dụng trên hệ điều hành Android 1.6. Lịch vạn niên có các khả năng sau:

- Xem tử vi của nam và nữ
- Xem tham khảo trong ngày
- Xem bằng 2 loại màn hình: kích cỡ 380px và 420px

4. PHƯƠNG PHÁP NGHIÊN CỨU

4.1 Về mặt lý thuyết

- Tìm hiểu lý thuyết về các thiết bị di động, các ngôn ngữ cũng như các công nghệ để xây dựng các ứng dụng đó.
- Tìm hiểu các thuật toán chuyển đổi lịch âm dương, thu thập các dữ liệu về tử vi, các phong tục xem ngày tốt xấu.
- Đưa ra một số định hướng để phát triển đề tài.

4.2 Công cụ xây dựng đề tài

- Công cụ nghiên cứu cơ sở lý thuyết: các tài liệu tham khảo như các sách ngôn ngữ lập trình, các giáo trình, các Ebook, các trang web... lập trình cho thiết bị di động và lý thuyết mạng tính tham khảo trong cuộc sống..
- Công cụ thiết kế phần mềm: Eclipse, Android SDK, ADT plugin, J2SE.

5. Ý NGHĨA KHOA HỌC, THỰC TIỄN CỦA ĐỀ TÀI

5.1 Ý nghĩa khoa học

- củng cố các kiến thức đã học
- Nghiên cứu và nắm bắt được nền tảng lập trình trên hệ điều hành android. Sử dụng thành thạo một số công cụ hỗ trợ trong môi trường làm việc.
- Nâng cao khả năng lập trình, giải quyết vấn đề

5.2 Ý nghĩa thực tiễn

Tạo một ứng dụng giúp mọi người có thể giải trí, tham khảo trên điện thoại có hệ điều hành Android với giao diện trực quan.

6. BỐ CỤC

Với kết quả dự kiến đạt được của ứng dụng, luận văn được tổ chức thành 3 chương. Cụ thể như sau:

- **Chương 1. Cơ sở lý thuyết:** giới thiệu tổng quan về công nghệ di động, ngôn ngữ lập trình, thuật toán chuyển đổi ngày âm dương.
- **Chương 2. Phân tích hệ thống:** phân tích các chức năng của hệ thống, từ đó thiết kế cho các chức năng và tổ chức dữ liệu.
- **Chương 3: Triển khai và kết quả demo:** kết quả chạy demo các chức năng của chương trình.
- **Kết luận. Kết luận và hướng phát triển:** nêu ra các nhận xét về kết quả đạt được và một số phương hướng phát triển tiếp theo của đề tài.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1. TỔNG QUAN VỀ HỆ ĐIỀU HÀNH ANDROID

1.1 Lịch sử Android

“Máy tính ngày càng trở nên “thân thiện” với con người hơn và ngày càng trở nên tiện dụng hơn, ở bất kỳ nơi đâu và bất kỳ lúc nào” – Y.HASHIMI & SATYA KOMATINENI (Pro Android Book – 2009).

Tháng 7 năm 2005, Google mua lại Android Inc, đó là tên một dự án do một số công ty nhỏ thành lập có trụ sở tại Palo Alto, California, USA. Những nhà đồng sáng lập nên Android được chuyển sang làm việc tại Google trong đó có Andy Rubin (đồng sáng lập công ty Danger), Rich Miner (đồng sáng lập công ty Wildlife Communication), Nick Sears (từng là phó chủ tịch T-Mobile) và Chris white (trưởng nhóm thiết kế và phát triển giao diện tại WebTV) . Tại thời điểm đó có rất ít thông tin về dự án Android, ngoại trừ việc họ đang phát triển phần mềm cho điện thoại di động.

Ở trụ sở của Google, nhóm do Rubin đứng đầu đã phát triển một nền tảng thiết bị di động dựa trên nhân Linux. Đến tháng 9 năm 2007, InformationWeek đăng tải một nghiên cứu của Evalueserve cho biết Google đã nộp 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 rộng OHA (Open Handset Alliance) bao gồm nhiều công ty trong đó có Google, HTC, Intel, LG, Motorola, Nvidia, Qualcomm, Samsung Elecstronic, T-Mobile, Sprint Nextel, Tập đoàn Marvell Technology, Texas Instruments, Tập đoàn Broadcom được thành lập với mục đích phát triển các tiêu chuẩn mở cho thiết bị di động, cùng với việc thành lập OHA, họ đồng thời giới thiệu Android một nền tảng cho thiết bị di động được xây dựng trên nhân Linux version 2.6.

1.2 Các phiên bản hệ điều hành Android (Android platform)



Hình 1.1 Biểu tượng cho một số phiên bản

1.3 Ứng dụng của Android và sự phát triển trong tương lai

Android chỉ mới hiện diện trên thị trường Mobile vào năm 2008 nhưng đã có những bước tiến vượt bậc trong lĩnh vực phát triển ứng dụng cho thiết bị Mobile và Smartphone.

Tuy là một nền tảng đi sau nhưng số lượng ứng dụng Android tăng nhanh chóng từ năm 2008 đến tháng 1 năm 2012 thì số ứng dụng của Android lên đến 430.000 ứng dụng.

Các hãng sản xuất thiết bị di động như Samsung, HTC, Motorola, LG, Acer, Sony ... đã có hướng chọn việc phát triển các thiết bị di động của mình chạy hệ điều hành Android làm chiến lược. Mặc dù hầu hết các ứng dụng hiện trên Google Play đa số là miễn phí (72 %) nhưng bù lại Android lại đang có đông đảo người sử dụng ưa thích cho đến hiện tại và số lượng sẽ còn tiếp tục tăng trong tương lai. Bên cạnh đó sự ra mắt của các dòng sản phẩm Smartphone chạy nền Android gần đây như sự bùng nổ mạnh mẽ của công nghệ này, chứng tỏ vai trò của mình trong thế giới thiết bị thông minh.

Lấy dẫn chứng cụ thể hãng điện thoại Samsung vừa cho ra mắt chiếc điện thoại Samsung Galaxy SII chạy hệ điều hành Android 2.3 vào ngày 22 tháng 3 năm 2011 và đã được nâng cấp lên hệ điều hành 4.0.3.

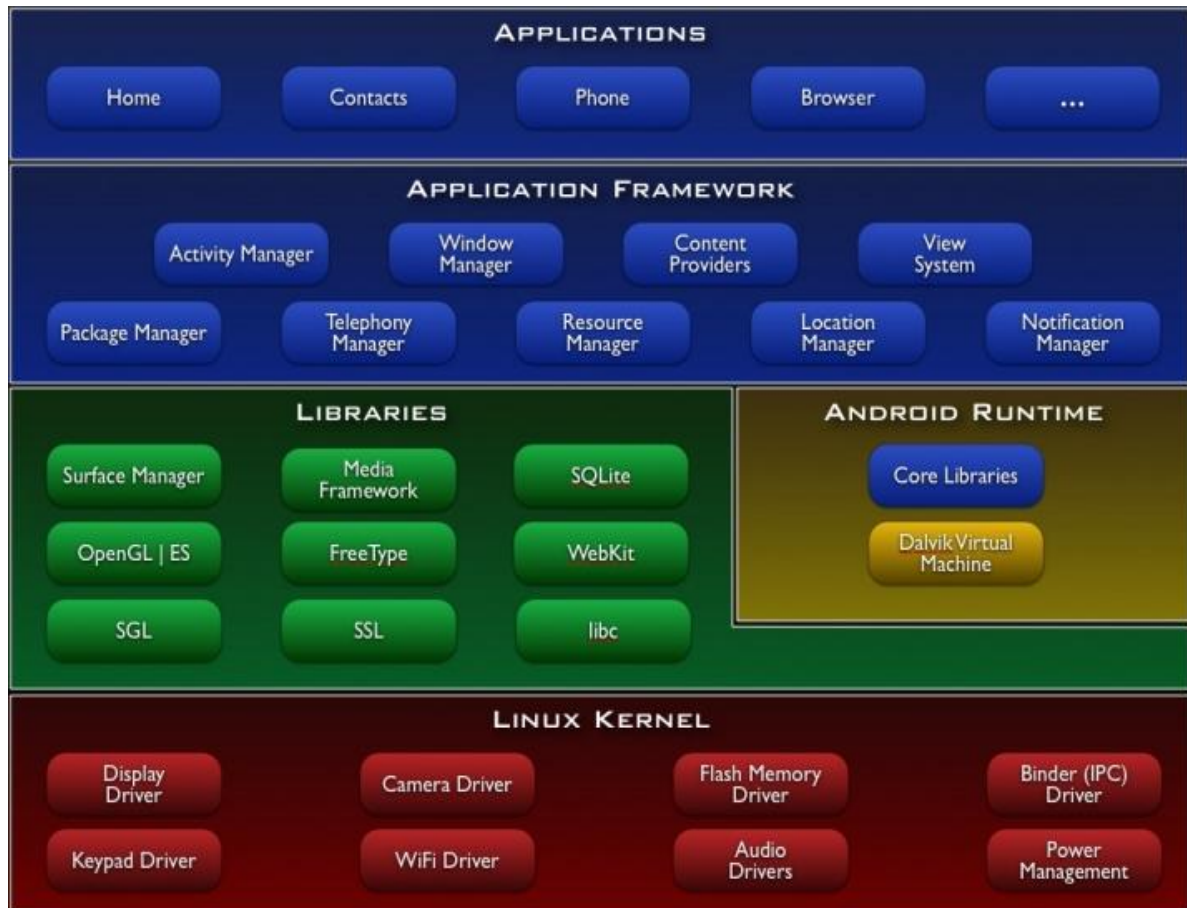


Hình 1.2 Samsung Galaxy S II

Đầu tháng 4 năm 2012 Google cho phát hành phiên bản SDK r19 hỗ trợ công cụ phát triển ứng dụng cho nền tảng Android 4.0, một nền tảng mới đầy hứa hẹn mở ra kỷ nguyên cho các thiết bị thông minh và tablet. Đặc biệt hơn phiên bản này đã được cài đặt trên một số dòng máy của Samsung và HTC.

2. KIẾN TRÚC HỆ ĐIỀU HÀNH ANDROID

2.1 Các thành phần cấu tạo nên Android (Android Architecture)



Hình 1.3 Kiến trúc hệ điều hành Android

2.1.1 Tầng Application

Android cung cấp bộ ứng dụng cốt lõi bao gồm một ứng dụng email, chương trình tin nhắn SMS, lịch, bản đồ, trình duyệt, địa chỉ liên hệ, ... Tất cả các ứng dụng được viết bằng ngôn ngữ lập trình Java.

2.1.2 Tầng Application Framework

Bằng cách cung cấp một nền tảng phát triển mở, Android cung cấp cho các nhà phát triển, lập trình viên khả năng xây dựng các ứng dụng vô cùng phong

phú và sáng tạo. Các nhà phát triển, lập trình viên có thể tự do tận dụng các thiết bị phần cứng, thông tin địa điểm truy cập, chạy các dịch vụ nền, thiết lập hệ thống cảnh báo, thêm các thông báo trạng thái và nhiều hơn nữa.

Các lập trình viên có toàn quyền truy cập đến các API trong cùng một Framework, thường được sử dụng bởi các ứng dụng cơ bản. Kiến trúc của tầng ứng dụng được thiết kế đơn giản cho việc tái sử dụng các thành phần; bất kỳ ứng dụng nào cũng có thể publish và các ứng dụng khác có thể sử dụng chúng. Cơ chế này cho phép các thành phần tương tự được thay thế bởi người dùng.

Tất cả các ứng dụng cơ bản là một tập hợp các dịch vụ và các hệ thống, bao gồm:

- Có nhiều thành phần phong phú và mở rộng cho việc thể hiện (Views) UI và xây dựng ứng dụng như List, Grid, Text box, Button và thậm chí các mã nhúng cho một trình duyệt web.
- Content Providers cho phép các ứng dụng có thể truy cập dữ liệu từ những ứng dụng khác (như là Contact) hoặc để chia sẻ dữ liệu với các ứng dụng khác.
- Một Resource Manager cung cấp truy cập đến các tài nguyên Phi Mã (Non-code) như Strings, Graphic, và các file Layout.
- Một Notification Manager cho phép tất cả các ứng dụng hiển thị những thông báo trên thanh trạng thái.
- Một Activity Manager thì quản lý vòng đời của một ứng dụng và cung cấp một Navigation Backstack phổ biến.

2.1.3 Tầng Libraries

Android cài đặt một số thư viện C/C++ được sử dụng bởi nhiều thành phần khác nhau của hệ thống Android. Một số thư viện cơ bản được liệt kê dưới đây:

- System library C – có nguồn gốc từ thư viện chuẩn của C (libc) được chỉnh để nhúng vào các thiết bị dựa trên Linux.
- Thư viện media dựa trên PacketVideo's OpenCore; thư viện này hỗ trợ playback và record nhiều định dạng audio và video phổ biến,

cũng như định dạng của một số ảnh bao gồm MPEG4, H.246, MP3, AAC, AMR, JPG và PNG.

- Sureface Manager: quản lý truy cập để hiển thị các subsystem và seamlessly
- Libwebcore một engine web mạnh mẽ chứa những tính năng của trình duyệt web Android và trình duyệt web nhúng.
- SGL có các thư viện hỗ trợ cho lập trình 2D
- 3D Libraries OPEN GL hỗ trợ cho lập trình 3D
- Freetype

SQLite một cơ sở dữ liệu nhỏ gọn nhưng mạnh mẽ cho tất cả các ứng dụng.

2.1.4 Tầng Android Runtime

Android bao gồm một tập hợp các thư viện cơ bản cung cấp hầu hết các chức năng có sẵn trong các thư viện lõi của ngôn ngữ lập trình Java.

Mỗi ứng dụng Android chạy trong quá trình riêng của nó với trường hợp riêng của mình, của máy ảo Dalvik. Dalvik được viết sao cho thiết bị có thể chạy nhiều máy ảo (VMs) một cách hiệu quả. Các máy ảo Dalvik thực thi các file trong một Dalvik Executable (.dex) được tối ưu hóa trong bộ nhớ một cách tối thiểu. VM dựa trên đăng ký và chạy các lớp được biên dịch bởi trình biên dịch ngôn ngữ Java mà được chuyển đổi thành định dạng file .dex bởi các công cụ “dx”.

Các máy ảo Dalvik dựa trên nhân Linux cho các chức năng cơ bản như luồng và quản lý bộ nhớ ở mức độ thấp.

2.1.5 Linux Kernel

Android được xây dựng dựa trên Linux phiên bản 2.6 cho hệ thống bao gồm các dịch vụ cốt lõi như bảo mật, quản lý bộ nhớ, quản lý tiến trình, mạng ngăn xếp (network stack) và mô hình điều khiển (driver model) . Hạt nhân (Kernel) cũng hoạt động như một lớp trừu tượng hóa giữa phần cứng và phần còn lại của phần mềm ngăn xếp.

2.2 Activity

➤ Activity là gì?

Activity là một thành phần cơ bản của ứng dụng Android. Mỗi activity thực hiện một chức năng của ứng dụng, trong một ứng dụng Android luôn có một MainActivity, khi ta chạy ứng dụng thì Activity này được gọi thực thi trước và thể hiện những giao diện hay các control... lên màn hình để người dùng có thể tương tác.

Một ứng dụng thường bao gồm nhiều activity có mối liên quan với nhau, và các activity này hoạt động theo cơ chế hàng đợi (queue mechanism) có nghĩa khi một activity đang hoạt động nhưng một activity khác được kích hoạt hoặc activity đó có độ ưu tiên lớn hơn (priority) thì activity trước đó sẽ chuyển sang trạng thái Stop hoặc Pause để activity này thực hiện.

➤ Tạo một Activity

Để tạo một activity bạn cần tạo một lớp con kế thừa từ lớp Activity. Trong lớp con bạn cần thực hiện các phương thức callback (callback method) mà hệ thống sẽ thực hiện khi chuyển đổi giữa các activity trong vòng đời của nó, bao gồm các trạng thái như Khởi tạo, Dừng, Chạy lại hay Ngừng hẳn. Trong đó có 2 phương thức hay dùng nhất là:

Tên phương thức	Mô tả
onCreate()	Phương thức này dùng để khởi tạo một Activity, ở trong phương thức này bạn nên khởi tạo các thành phần thiết yếu cho activity đó. Quan trọng nhất là bạn phải gọi câu lệnh setContentView() để định nghĩa layout cho activity đó.
onPause()	Hệ thống sẽ gọi phương thức này khi người dùng chuyển sang một activity khác. Bạn có thể thêm những câu lệnh hay gọi activity khác khi điều này xảy ra.

➤ Thực hiện một giao diện người dùng

Giao diện người dùng cho một hoạt động được cung cấp bởi một hệ thống các quan điểm, đối tượng dẫn xuất từ lớp View. Mỗi đối tượng View được trình bày trên không gian hình chữ nhật (Màn hình Mobile) và ta có thể xử lý những sự kiện thông qua các câu lệnh trên các đối tượng View đó.

➤ Khai báo activity trong manifest

Bạn cần khai báo activity trong tập tin AndroidManifest để nó có thể truy cập vào hệ thống. Để khai báo cần mở tập tin AndroidManifest trong Project của bạn rồi thêm một thẻ <activity> ở bên trong cặp thẻ <application> như sau:

```
<manifest ..>
  <application >
    <activity android:name=".ExampleActivity"/>
    ..
  </application >
  ..
</manifest >
```

Bạn có thể thêm một số thuộc tính khác cho activity như label, icon, theme style.

➤ Sử dụng intent filters

Một Activity cũng có thể chỉ định những bộ lọc khác nhau tùy mục đích sử dụng, các yếu tố lọc được để trong thẻ <intent-filter> bao gồm các yếu tố để có thể kích hoạt activity đó.

```
<activity android:name=".ExampleActivity"
  android:icon="@drawable/app_icon">
  <intent-filter>
    <action android:name="android.intent.action.MAIN"/>
    <category android:name="android.intent.category.LAUNCHER"/>
  </intent-filter>
```

```
</activity>
```

Các phần tử trong thẻ tab <action> xác định thực hiện hành động của ứng dụng đối với activity đó. Còn các phần tử trong thẻ <category> xác định activity này nên được liệt kê khi khởi tạo ứng dụng hệ thống (cho phép người dùng gọi activity đó).

➤ Bắt đầu một Activity

Bạn có thể gọi một activity bằng phương thức startActivity(),_thông qua Intent mà bạn đã miêu tả về activity đó khi bạn muốn bắt đầu.

Ví dụ: Ở đây ta có một lớp SignInActivity

```
Intent intent= new Intent(this, SignInActivity.class);  
startActivity(intent);
```

Tuy nhiên, khi ứng dụng của bạn muốn thực hiện một vài action như là send mail, text message, hay cập nhật trạng thái, sử dụng dữ liệu từ activity của bạn. Trong trường hợp này, ứng dụng của bạn có thể không có những hoạt động riêng của mình để thực hiện action này, vì vậy bạn cần tận dụng những activities được cung cấp bởi thiết bị mà có thể thực hiện action đó cho bạn. Nếu có nhiều hoạt động có thể xử lý các intent, thì người dùng có thể chọn bất kỳ một trong các intent để sử dụng. Ví dụ nếu bạn muốn cho người dùng gửi email, bạn có thể tạo ra các intent sau đây:

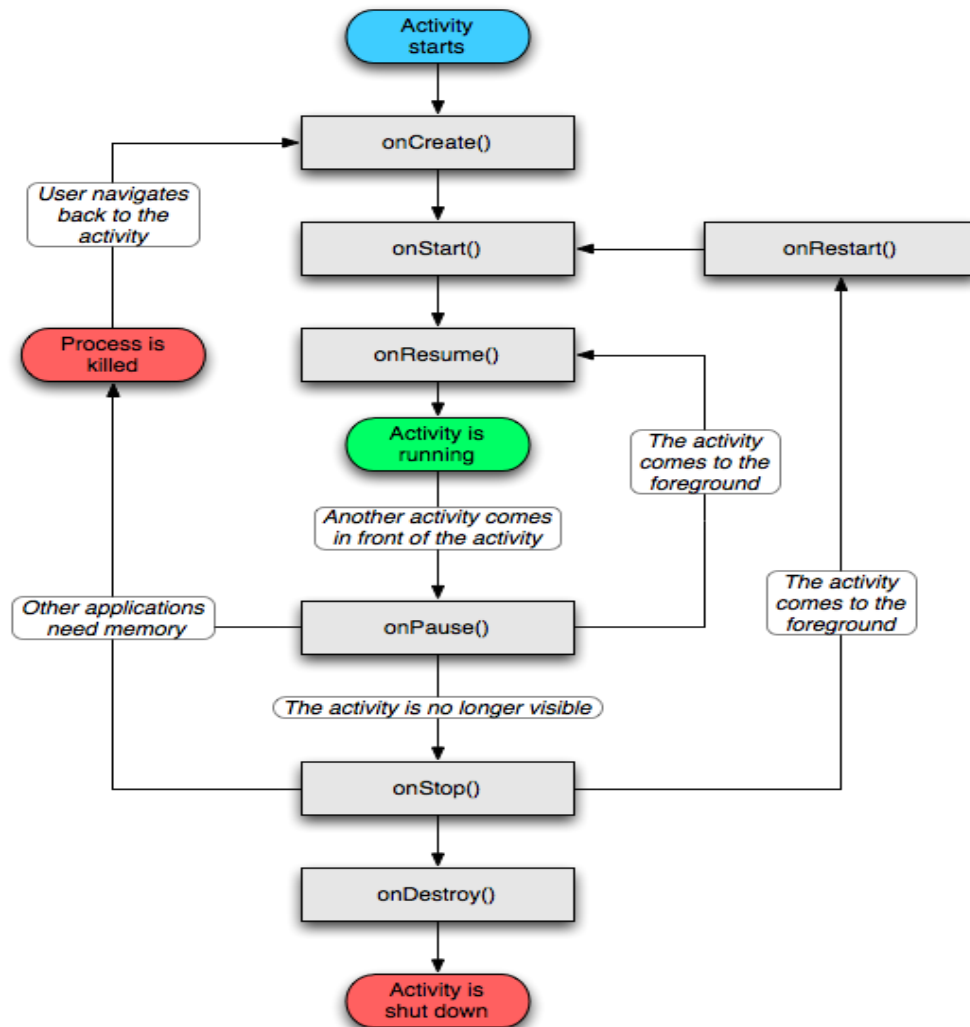
```
Intent intent = new Intent(Intent.ACTION_SEND);  
intent.putExtra(Intent.EXTRA_EMAIL, recipientArray);  
startActivity(intent);
```

Các EXTRA_MAIL được thêm vào intent là một mảng các chuỗi địa chỉ email mà sẽ được gửi. Khi một ứng dụng email hồi đáp intent này, nó sẽ đọc mảng chuỗi được cung cấp bên trong extra và thay thế chúng trong trường “To” của Form gửi mail. Trong trường hợp này activity của ứng dụng sẽ được gọi khi người dùng đã thực hiện và activity đó sẽ được resume lại.

➤ **Dừng một Activity**

Để tắt một activity thì bạn cần gọi phương thức finish(). Bạn cũng có thể tắt một activity mà bạn đã gọi trước đó bằng phương thức finishActivity().

➤ **Vòng đời của một Activity**



Hình 1.4 Activity Lifecycle

Trong đó:

- `onCreate` : Khởi tạo activity
- `onStart` : Thực hiện activity
- `onResume` : Thực hiện tiếp tục một activity đã bị dừng trước đó
- `onPause` : Tạm dừng một activity

- onStop : Dừng hẳn một activity
- onRestart : Thực hiện lại activity
- onDestroy: Phá hủy một activity

Android có cơ chế quản lý các tiến trình theo độ ưu tiên. Các tiến trình có độ ưu tiên thấp sẽ bị Android giải phóng mà không hề cảnh báo nhằm đảm bảo tài nguyên.

➤ **Một Activity bao gồm 4 trạng thái:**

- active (running): Activity đang hiển thị lên màn hình (foreground)
- paused: Activity vẫn hiển thị (visible) nhưng không thể tương tác (lost focus)
- stop: Một activity được thay thế hoàn toàn bởi một activity khác sẽ chuyển sang trạng thái stop.
- killed: khi hệ thống thiếu bộ nhớ nó sẽ giải phóng các tiến trình theo nguyên tắc độ ưu tiên. Các activity ở trạng thái stop hoặc pause cũng có thể bị giải phóng và khi nó được hiển thị lại thì các Activity này phải khởi động lại hoàn toàn và phục hồi lại trạng thái trước đó.

➤ **Vậy ta có thể hiểu vòng đời của Activity như sau:**

- Entire lifetime : Bắt đầu từ phương thức onCreate() cho tới phương thức onDestroy()
- Visible lifetime : Từ phương thức onStart() cho tới phương thức onStop().
- Foreground lifetime : Từ phương thức onResume() đến phương thức onPause().

2.3 Broadcast receivers

Là một thành phần của ứng dụng có chức năng thu nhận và phản ứng lại những thông báo được broadcast. Nhiều broadcast gốc trong mã hệ thống và các ứng dụng cũng có thể khởi tạo broadcast.

Broadcast receivers không hiển thị giao diện người dùng. Tuy nhiên chúng có thể khởi tạo một Activity phản hồi lại những thông tin được nhận. Chúng cũng có thể dùng Notification Manager để thông báo đến người dùng.

2.4 DVM

DVM là máy ảo giúp các ứng dụng Java chạy được trên các thiết bị di động Android. Nó chạy các ứng dụng đã được chuyển đổi thành một file thực thi Dalvik(dex). Định dạng phù hợp cho các hệ thống mà thường bị hạn chế về bộ nhớ và tốc độ xử lý. DVM đã được thiết kế và viết bởi Dan Bornstein, ông đã đặt tên cho máy ảo này sau khi đến thăm một ngôi làng có tên Dalvík ở đảo Eyjafjörður, nơi mà một số tổ tiên của ông sinh sống.

Từ góc nhìn của một nhà phát triển, DVM giống như một máy ảo Java (Java virtual machine) nhưng thực tế thì hoàn toàn khác. Khi nhà phát triển viết một ứng dụng dành cho Android, anh ta thực hiện các đoạn mã trong môi trường Java. Sau đó nó sẽ được biên dịch sang bytecode của Java, tuy nhiên để thực thi được ứng dụng này trên Android thì nhà phát triển phải thực thi một công cụ có tên là dx. Đây là công cụ dùng để chuyển đổi bytecode sang một dạng khác gọi là dex bytecode. “Dex” là từ viết tắt của Dalvik Executable đóng vai trò như cơ chế ảo thực thi các ứng dụng Android.

3. LẬP TRÌNH ỨNG DỤNG ANDROID

3.1 Thiết lập môi trường phát triển ứng dụng

Môi trường được thiết lập trên các hệ điều hành Windows và cụ thể là Windows 7. Để thực hiện việc này bạn cần có các gói cài đặt sau:

- JDK Java for windows

Link: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

- SDK Android for windows

Link: <http://developer.android.com/sdk/index.html>

- IDE Eclipse for Java Developer

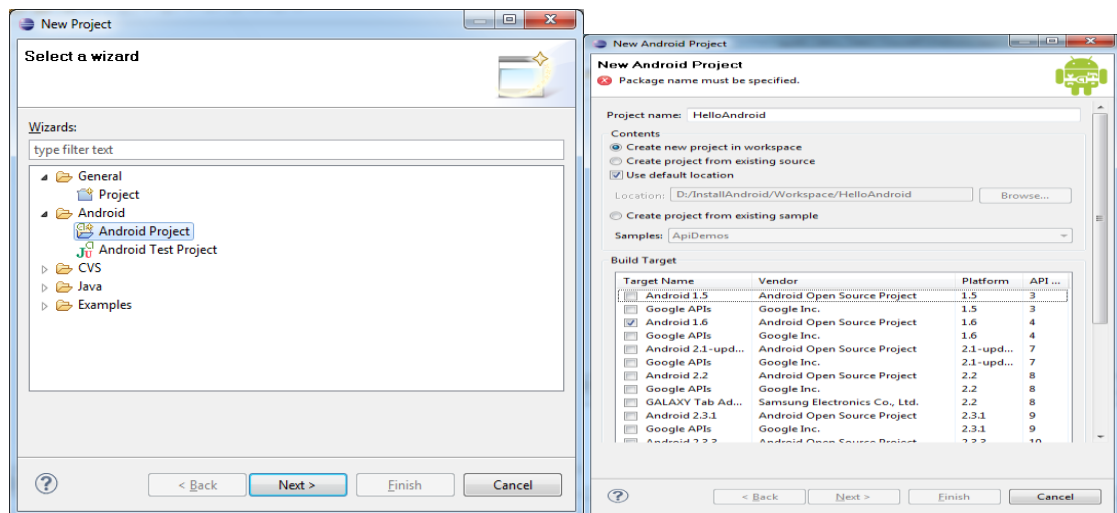
Link: <http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/heliossr2>

3.2 Các thành phần trong Android Project

➤ Tạo mới Android Project:

Click menu **File-> New -> Project** chọn **Android-> Android Project**.

Hoặc có thể tạo nhanh bằng cách click vào biểu tượng  trên thanh công cụ.



Hình 1.5 Tạo mới 1 ứng dụng Android

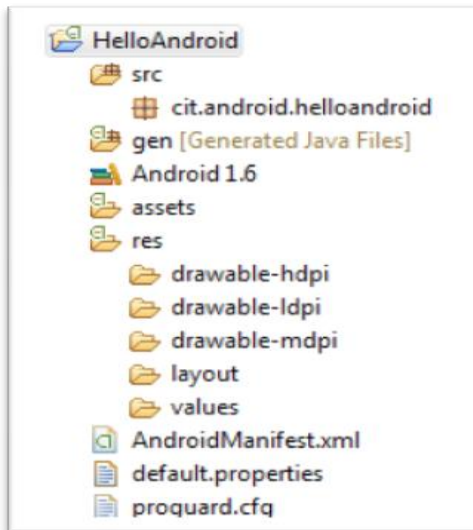
Project name: Tên cho Project

Application name: Tên ứng dụng

Package name: Tên package, bạn đặt sao cũng được nhưng sao cho dễ nhớ và không có các ký tự đặc biệt (chỉ gồm chữ thường và các số từ 0 ~ 9).

Create Activity: Đặt tên cho Activity chính sẽ khởi tạo đầu tiên khi chạy ứng dụng, hoặc không muốn tạo bằng cách bỏ dấu Tick.

Min SDK Version: Tương ứng với API Level của Target. Sau đó click Finish ta sẽ có 1 project như hình dưới:



/src folder: chứa các mã nguồn (source code) là các file Java.

/gen: Chứa file R.java dùng để quản lý các tài nguyên được tạo ra từ các tập tin nguồn, file này tự tạo nên không được chỉnh sửa hoặc thay đổi.

/Android 1.6: bao gồm các API 1.6 tương ứng với HĐH Android 1.6, các thư viện này được Import khi ta khởi tạo Project

bằng cách chọn Target ở trên.

/assets: chứa các file tài nguyên như database sẽ đi kèm với Project khi ta đóng gói ứng dụng.

/res/drawable-hdpi: nơi lưu hình ảnh sử dụng cho ứng dụng hdpi (High dpi: độ phân giải cao) tương thích với điện thoại có màn hình WVGA800 (480x800), WVGA854 (480x854).

/res/drawable-ldpi: cũng là thư mục để chứa các file ảnh có độ phân giải thấp hơn Low dpi: QVGA (240x320), WQVGA400 (240x400), WQVGA432 (240x432).

/res/drawable-mdpi: chứa các file ảnh có độ phân giải trung bình Medium dpi: HVGA (320x480), WVGA800 (480x800), WVGA854 (480x854).

/res/layout/main.xml: file thiết kế giao diện cho ứng dụng, có thể tạo thêm nhiều file khác cho những màn hình khác nhau.

/res/values/strings.xml: định nghĩa giá trị cho các tài nguyên trong project, giúp tối ưu hóa trong việc quản lý giá trị mặc định các đối tượng String trên mỗi layout.

/AndroidManifest.xml: file cấu hình cho ứng dụng

Ngoài ra còn có một số folder và file do người lập trình tự định nghĩa.

4. LÝ THUYẾT CỦA BÀI TOÁN CHUYỂN ĐỔI NGÀY ÂM DƯƠNG

4.1 Cơ sở lý thuyết tính lịch âm Việt Nam

Âm lịch Việt Nam là một loại lịch thiên văn. Nó được tính toán dựa trên sự chuyển động của mặt trời, trái đất và mặt trăng. Ngày tháng âm lịch được tính dựa theo các nguyên tắc sau:

- Ngày đầu tiên của tháng âm lịch là ngày chứa điểm Sóc.
- Một năm bình thường có 12 tháng âm lịch, một năm nhuận có 13 tháng âm lịch.
- Đông chí luôn rơi vào tháng 11 âm lịch.
- Trong một năm nhuận, nếu có 1 tháng không có Trung khí thì tháng đó là tháng nhuận. Nếu nhiều tháng trong năm nhuận đều không có Trung khí thì chỉ tháng đầu tiên sau Đông chí là tháng nhuận.
- Việc tính toán dựa trên kinh tuyến 105° đông.

4.1.1 Điểm sóc

Sóc là thời điểm hội diện, đó là khi trái đất, mặt trăng và mặt trời nằm trên một đường thẳng và mặt trăng nằm giữa trái đất và mặt trời. (Như thế góc giữa mặt trăng và mặt trời bằng 0°). Gọi là “hội diện” vì mặt trăng và mặt trời ở cùng một hướng đối với trái đất. Chu kỳ của điểm Sóc là khoảng 29,530588 ngày tức là 29 ngày 12 giờ 44 phút 2,8032 giây. Ngày chứa điểm Sóc được gọi là ngày Sóc và đó là ngày bắt đầu tháng âm lịch.

4.1.2 Tiết khí

Trung khí là các điểm chia đường hoàng đạo thành 12 phần bằng nhau. Trong đó, bốn Trung khí giữa bốn mùa là đặc biệt nhất: Xuân phân (khoảng 20/3), Hạ chí (khoảng 22/6), Thu phân (khoảng 23/9) và Đông chí (khoảng 22/12). Thời giữa 2 Trung khí là khoảng 30,43685 ngày tức là 30 ngày 10 giờ 29 phút 3,84 giây. Đông chí luôn rơi vào tháng 11 âm lịch. Sau đây là bảng tiết khí:

Số thứ tự	Độ	Tiết khí			Dương lịch Gregorius
		Trung khí	Tiết khí	Việt Nam	
1	270°	Đông chí		Giữa đông	21(22) -12
2	285°		Tiểu hàn	Chớm rét	4(5)-1
3	300°	Đại hàn		Giá rét	20(21)-1

4	315°		Lập xuân	Đầu xuân	4(5)-2
5	330°	Vũ thủy		Ấm ướt	18(19)-2
6	345°		Kinh trập	Sâu nở	5(6)-3
7	0°	Xuân phân		Giữa xuân	20(21)-3
8	15°		Thanh minh	Trong sáng	5(6)-4
9	30°	Cốc vũ		Mưa rào	20(21)-4
10	45°		Lập hạ	Đầu hè	5(6)-5
11	60°	Tiểu mãn		Kết hạt	21(22)-5
12	75°		Mang chủng	Tua rua	6(7)-6
13	90°	Hạ chí		Giữa hè	21(22)-6
14	105°		Tiểu thử	Nắng oi	7(8)-7
15	120°	Đại thử		Nóng nực	23(24)-7
16	135°		Lập thu	Đầu thu	7(8)-8
17	150°	Xử thử		Mang chủng	23(24)-8
18	165°		Bạch lộ	Nắng nhạt	7(8)-9
19	180°	Thu phân		Giữa thu	23(24)-9
20	195°		Hàn lộ	Mát mẻ	8(9)-10
21	210°	Sương giáng		Sương sa	23(24)-10
22	225°		Lập đông	Đầu đông	7(8)-11
23	240°	Tiểu tuyết		Hạnh heo	22(23)-11
24	255°		Đại tuyết	Khô úa	7(8)-12

Bảng 1.1: Bảng tiết khí

4.1.3 Tháng nhuận

Do độ dài giữa 2 Trung khí (30,43685) dài hơn độ dài của tuần trăng hay còn gọi là chu kỳ trăng (29,530588) nên có những tuần trăng không chứa Trung khí và ta đặt đó là tháng nhuận.

Đông chí luôn rơi vào tháng 11 của năm âm lịch. Bởi vậy chúng ta cần tính 2 điểm Sóc: Sóc A ngay trước ngày Đông chí thứ nhất và Sóc B ngay trước ngày Đông chí thứ hai. Nếu khoảng cách giữa A và B là dưới 365 ngày thì năm âm lịch có 12 tháng, và những tháng đó có tên là: tháng 11, tháng 12, tháng 1, tháng 2, ..., tháng 10. Ngược lại, nếu khoảng cách giữa hai Sóc A và B là trên 365 ngày thì năm âm lịch này là năm nhuận, và chúng ta cần tìm xem đâu là tháng nhuận. Để làm việc này ta xem xét tất cả các tháng giữa A và B, tháng đầu tiên không chứa Trung khí sau ngày Đông chí thứ nhất là tháng nhuận. Tháng đó sẽ được mang tên của tháng trước nó kèm chữ "nhuận".

Chú ý : Khi tính ngày Sóc và ngày chứa Trung khí cần lưu ý xem xét chính xác múi giờ. Múi giờ của Việt Nam (GMT+7, kinh tuyến 105° đông).

4.2 Các hàm cơ bản chuyển đổi ngày âm dương

4.2.1 Chuyển ngày dương ra số ngày Julius

Trong tính toán thiên văn người ta lấy ngày 1/1/4713 trước công nguyên của lịch Julius (tức ngày 24/11/4714 trước CN theo lịch Gregorius) làm điểm gốc. Số ngày tính từ điểm gốc này gọi là số ngày Julius (Julian day number) của một thời điểm. Ví dụ: số ngày Julius của 1/1/2000 là 24515455.

Lịch Julius do hoàng đế Julius Caesar đưa ra vào năm 45 trước Công nguyên. Lịch Julius chia thành 12 tháng với 365 ngày, cứ 4 năm thì thêm một ngày vào cuối tháng 2 tạo thành năm nhuận. Vì vậy theo lịch Julius thì một năm có 365,25 ngày. Nhưng độ dài của năm mặt trời là 365,242216 ngày cho nên lịch Julius dài hơn khoảng 0,0078 ngày trong một năm, tức là khoảng 11 phút 14 giây.

Để bù vào sự khác biệt này thì cứ 400 năm ta sẽ bỏ bớt đi 3 ngày năm nhuận. Cho đến năm 1582, thì sự sai biệt đã lên đến 10 ngày. Giáo Hoàng Gregory XIII quyết định bỏ 10 ngày trong tháng 10 năm đó để cho lịch và mùa màng tương ứng trở lại. Sau ngày 4 tháng 10 năm 1582 là ngày 15 tháng 10 (số ngày Julius là 2299161). Và để tránh sai biệt, lịch lấy năm nhuận là năm có số thứ tự chia chắn cho 4 và các năm tận cùng bằng 00 phải chia chắn cho 400 mới là năm nhuận. Lịch đã sửa mang tên lịch Gregory và được áp dụng cho đến bây giờ.

Để đổi ngày dương dd/mm/yyyy ra ngày Julius jd ta dùng thuật toán sau:

- $a = (14 - mm) \bmod 12$
- $y = yyyy + 4800 - a$
- $m = mm + 12 * a - 3$
- $jd = dd + (153 * m + 2) \bmod 5 + 365 * y + y \bmod 4 - y \bmod 100 + y \bmod 400 - 32045$
- Nếu $jd < 2299161$ thì tính lại jd :
- $jd = dd + (153 * m + 2) \bmod 5 + 365 * y + y \bmod 4 - 32083$

Ghi chú : Xem code ở phụ lục 1.

4.2.2 Chuyển đổi số ngày Julius ra ngày dương

Nếu số ngày Julius nhập vào mà lớn hơn 2299161 sẽ được tính theo lịch Gregorius.

- Nếu $jd > 2299160$ thì:
 - $a = jd + 32044;$
 - $b = (4 * a + 3) \bmod 146097;$
 - $c = a - (b * 146097) \bmod 4;$
- Ngược lại:
 - $b = 0;$
 - $c = jd + 32082;$
- $d = (4 * c + 3) \bmod 1461;$
- $e = c - (1461 * d) \bmod 4$
- $m = (5 * e + 2) \bmod 153;$
- $day = e - (153 * m + 2) \bmod 5 + 1$
- $month = m + 3 - 12 * (m \bmod 10)$
- $year = b * 100 + d - 4800 + m \bmod 10$

Ghi chú: Xem code ở phụ lục 2

4.2.3 Tính ngày Sóc

Như trên đã nói, việc quan trọng đầu tiên khi tính lịch âm là tính xem các điểm Sóc (tức Hội diện) rơi vào ngày nào. Phương thức sau cho phép tính thời điểm (tính bằng số ngày Julius) của Sóc thứ k tính từ điểm Sóc lúc 13:51 GMT ngày 1/1/1900 (ngày Julius 2415021.076998695).

Ghi chú: Xem code ở phụ lục 3 và 4.

4.2.4 Tính tọa độ mặt trời

Để biết Trung khí nào nằm trong tháng âm lịch nào, ta chỉ cần tính xem mặt trời nằm ở khoảng nào trên đường hoàng đạo vào thời điểm bắt đầu một tháng âm lịch. Ta chia đường hoàng đạo làm 12 phần và đánh số các cung này từ 0 đến 11: từ Xuân phân đến Cốc vũ là 0; từ Cốc vũ đến Tiểu mãn là 1; từ Tiểu mãn đến Hạ chí là 2; v.v.. Cho jd_n là số ngày Julius của bất kỳ một ngày, phương pháp sau này sẽ trả lại số cung nói trên.

Ghi chú: Xem code ở phụ lục 5.

Với hàm này ta biết được một tháng âm lịch chứa Trung khí nào. Giả sử một tháng âm lịch bắt đầu vào ngày N1 và tháng sau đó bắt đầu vào ngày N2 và hàm `getSunLongitude` cho kết quả là 8 với N1 và 9 với N2. Như vậy tháng âm lịch bắt đầu ngày N1 là tháng chứa Đông chí: trong khoảng từ N1 đến N2 có một ngày mặt trời di chuyển từ cung 8 (sau Tiểu tuyết) sang cung 9 (sau Đông chí). Nếu hàm `getSunLongitude` trả lại cùng một kết quả cho cả ngày bắt đầu một tháng âm lịch và ngày bắt đầu tháng sau đó thì tháng đó không có Trung khí và như vậy có thể là tháng nhuận.

4.2.5 Tìm ngày bắt đầu tháng 11 âm lịch

Đông chí thường nằm vào khoảng 19/12-22/12, như vậy trước hết ta tìm ngày Sóc trước ngày 31/12. Nếu tháng bắt đầu vào ngày đó không chứa Đông chí thì ta phải lùi lại 1 tháng nữa.

- Đổi ngày 31/12/yy ra ngày Julius jd
- Đổi jd ra tuần trăng thứ k kể từ tuần trăng 1/1/1900
- Tìm điểm sóc của tuần trăng k là nm
- Tính tọa độ mặt trời tại nm rồi chia 30, nếu kết quả lớn hơn hoặc bằng 9 thì nm tính lại tại k-1
- nm là ngày bắt đầu tháng 11 năm yy

Ghi chú: Xem code phụ lục 6.

4.2.6 Xác định tháng nhuận

Nếu giữa hai tháng 11 âm lịch (tức tháng có chứa Đông chí) có 13 tháng âm lịch thì năm âm lịch đó có tháng nhuận. Để xác định tháng nhuận, ta sử dụng hàm `getSunLongitude` như đã nói ở trên. Cho a11 là ngày bắt đầu tháng 11 âm lịch mà một trong 13 tháng sau đó là tháng nhuận. Thuật toán sau cho biết tháng nhuận nằm ở vị trí nào sau tháng 11 này.

- Đổi ngày bắt đầu tháng 11 năm yy ra jd
- Đổi jd sang tuần trăng k
- i=1
- Lặp: Tính tọa độ mặt trời của tuần trăng k +i và k+(++i). Nếu hai vị trí đó sau khi chia 30 mà bằng nhau thì tháng nhuận là i-1

Ghi chú: Xem code phụ lục 7.

Giả sử thuật toán trên trả lại giá trị 4, như thế tháng nhuận sẽ là tháng sau tháng 2 thường. (Tháng thứ 4 sau tháng 11 đáng ra là tháng 3, nhưng vì đó là tháng nhuận nên sẽ lấy tên của tháng trước đó tức tháng 2, và tháng thứ 5 sau tháng 11 là tháng 3).

4.2.7 Đổi ngày dương ra ngày âm

Với các phương pháp hỗ trợ trên ta có thể đổi ngày dương dd/mm/yy ra ngày âm dễ dàng. Trước hết ta xem ngày monthStart bắt đầu tháng âm lịch chứa ngày này là ngày nào (dùng hàm getNewMoonDay như trên đã nói). Sau đó, ta tìm các ngày a11 và b11 là ngày bắt đầu các tháng 11 âm lịch trước và sau ngày đang xem xét. Nếu hai ngày này cách nhau dưới 365 ngày thì ta chỉ cần xem monthStart và a11 cách nhau bao nhiêu tháng là có thể tính được dd/mm/yy nằm trong tháng mấy âm lịch. Ngược lại, nếu a11 và b11 cách nhau khoảng 13 tháng âm lịch thì ta phải tìm xem tháng nào là tháng nhuận và từ đó suy ra ngày đang tìm nằm trong tháng nào.

- Đổi ngày dương sang jd và tính xem nó thuộc tuần trăng thứ k nào.
- Tìm ngày đầu tuần trăng k là nm.
- Ngày âm tính được: jd-nm+1
- Tính ngày bắt đầu tháng 11 của năm yy là a11.
- Nếu a11 >= nm, năm âm: yy ngược lại là yy +1 và tính lại b11.
- Kiểm tra năm yy có nhuận không
- Tính tháng âm.

Ghi chú: Xem code phụ lục 8.

4.2.8 Đổi lịch âm ra lịch dương

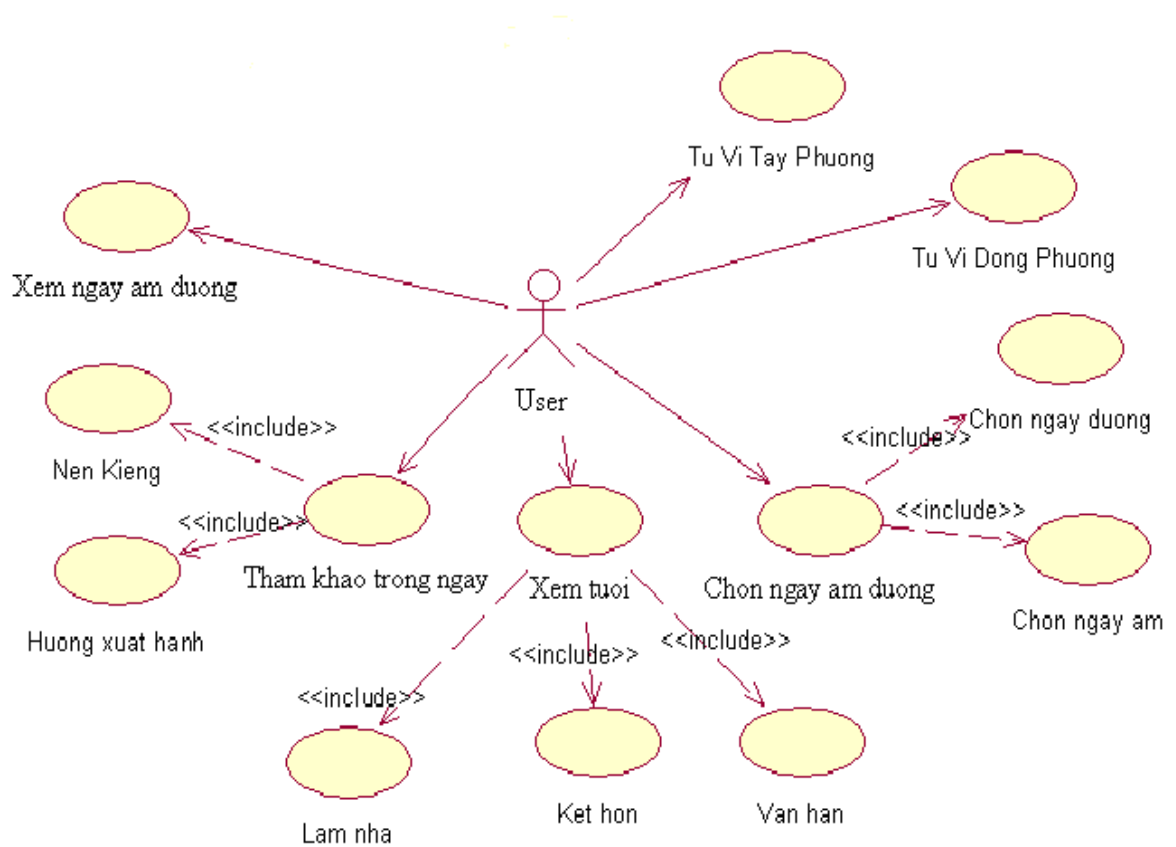
Cũng tương tự như thuật toán chuyển ngày dương sang ngày âm:

Ghi chú: Xem code phụ lục 9

CHƯƠNG 2: PHÂN TÍCH HỆ THỐNG

1. SƠ ĐỒ USE CASE

1.1 Sơ đồ use case tổng quát



Hình 2.1 Sơ đồ ca sử dụng tổng quát

1.2 Mô tả các ca sử dụng hay chức năng của nó

Tên ca sử dụng	Chức năng
Xem ngày âm dương	Người dùng có thể xem ngày dương, ngày âm, giờ hoàng đạo trong này.

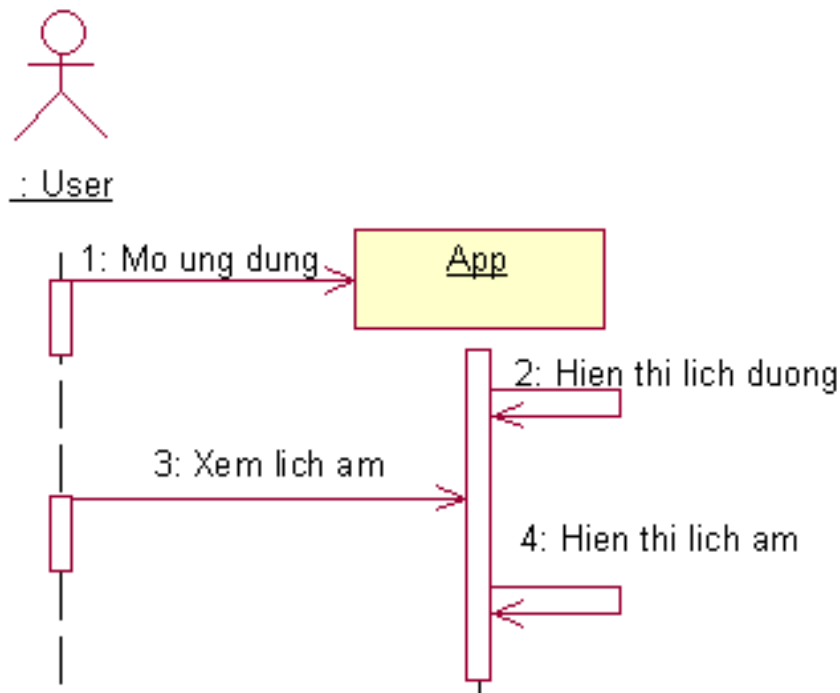
Tham khảo trong ngày	Người dùng có thể xem một số tham khảo trong ngày như nên làm gì, không nên làm gì, nên đi về hướng nào, không nên đi về hướng nào
Xem tuổi	Trong chức năng này người dùng sẽ nhập tuổi và xem tuổi của mình năm nay có thể làm nhà hay không, có thể làm nhà vào những năm nào trong tương lai, những tuổi nào có thể làm nhà trong năm nay. Ngoài ra, người dùng còn xem tuổi của mình năm nay làm ăn, danh vọng như thế nào (gọi là vận hạn trong năm). Xem tuổi của mình có thể
Tử Vi Đông Phương	Chức năng này giúp người dùng có thể khám phá được bản thân của mình qua năm sinh (Tí, Sửu, ..) về tính cách, tình yêu,
Tư Vi Tây Phương	Tương tự như chức năng trên, nhưng được giải mã theo ngày tháng sinh thông qua các chòm sao (Bảo Bình, Kim Ngưu,...).
Chọn ngày âm dương	Người dùng có thể đi đến một ngày nào đó thì có thể chọn một ngày dương hoặc một ngày âm bất kì. Chương trình sẽ tự động hiển thị ngày mà bạn muốn xem. Nếu là ngày âm thì sẽ tự động chuyển sang ngày dương rồi hiển thị

Bảng 2.1: Chức năng của từng ca sử dụng

2. SƠ ĐỒ TUẦN TỰ

2.1 Quá trình xem lịch âm dương

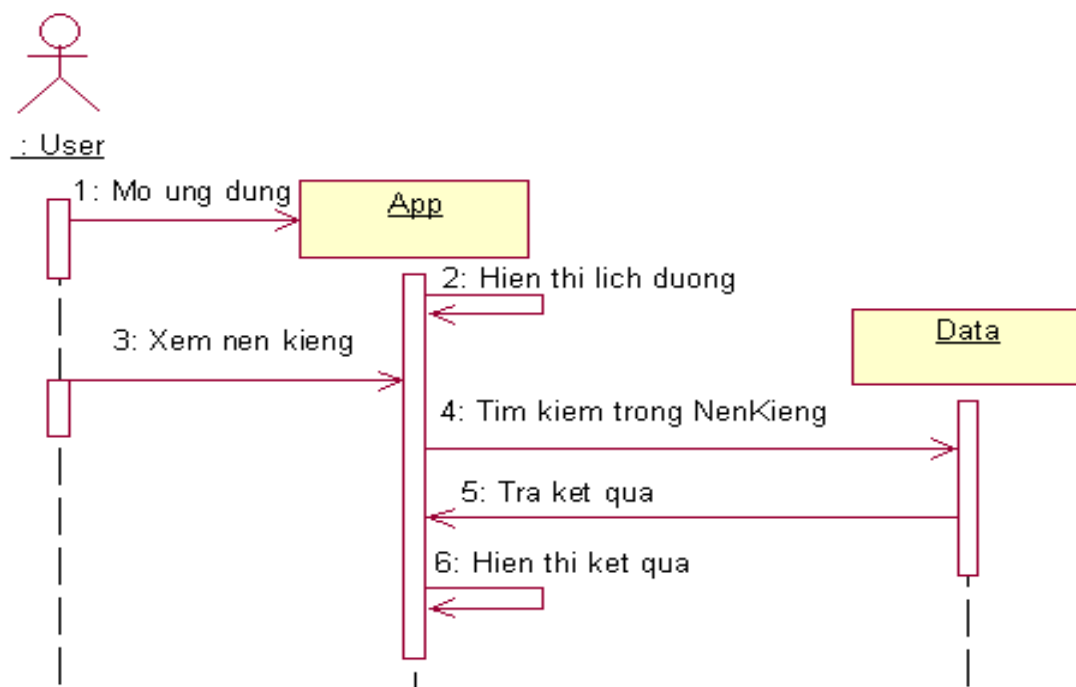
Chức năng xem lịch âm dương giúp người dùng xem hôm nay là thứ mấy, ngày, tháng, năm nào, thuộc tuần thứ mấy trong năm. Sau đó xem tiếp hôm nay là ngày, tháng, năm âm, giờ hoàng đạo trong ngày.



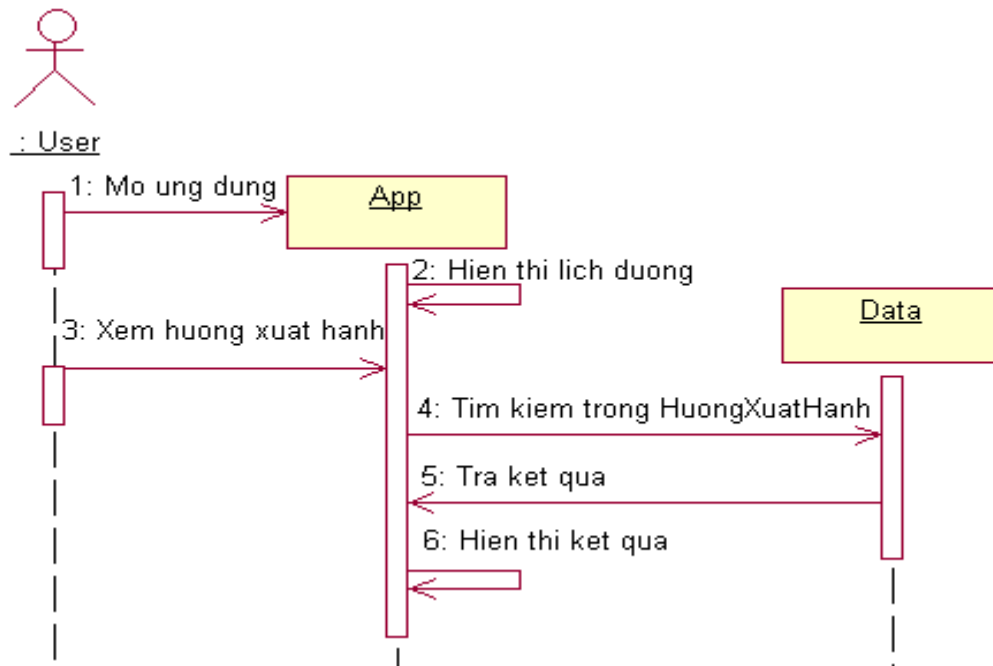
Hình 2.2 Sơ đồ tuần tự quá trình xem lịch âm dương

2.2 Quá trình xem tham khảo trong ngày

Sau khi mở ứng dụng, lịch sẽ ở ngày hiện tại và ta có thể xem những việc nên, không nên làm và hướng đi tốt xấu trong ngày đó. Ví dụ như: Nên: xuất hành, nhập học, khai trương, ... Kiêng: cúng tế, nhận chức, ... Nên đi về hướng: Đông, ...



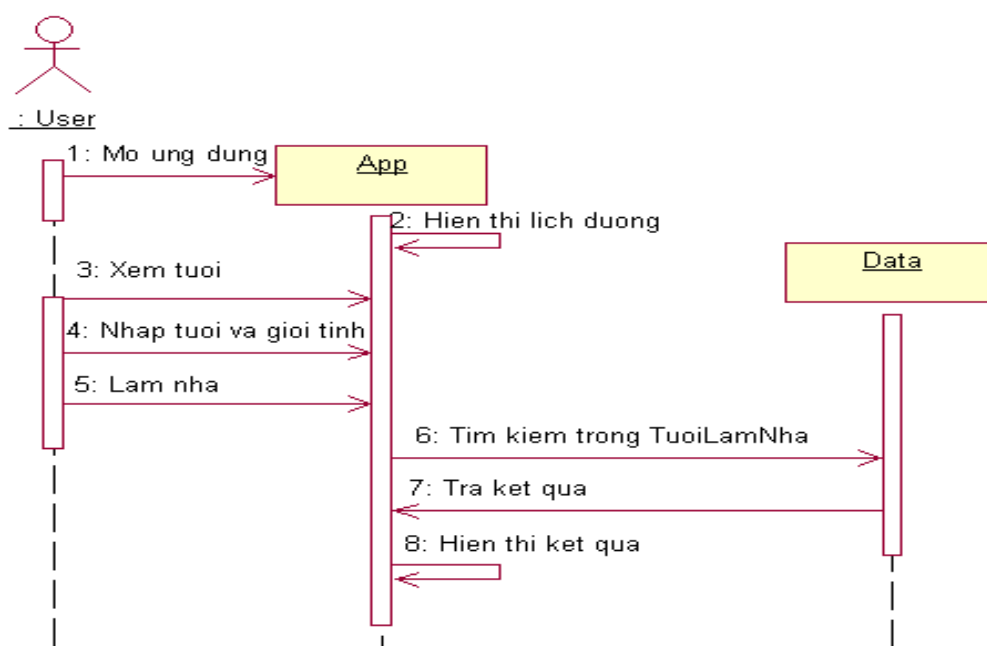
Hình 2.3 Sơ đồ tuần tự quá trình xem nên kiêng trong ngày



Hình 2.4 Sơ đồ tuần tự quá trình xem hướng xuất hành

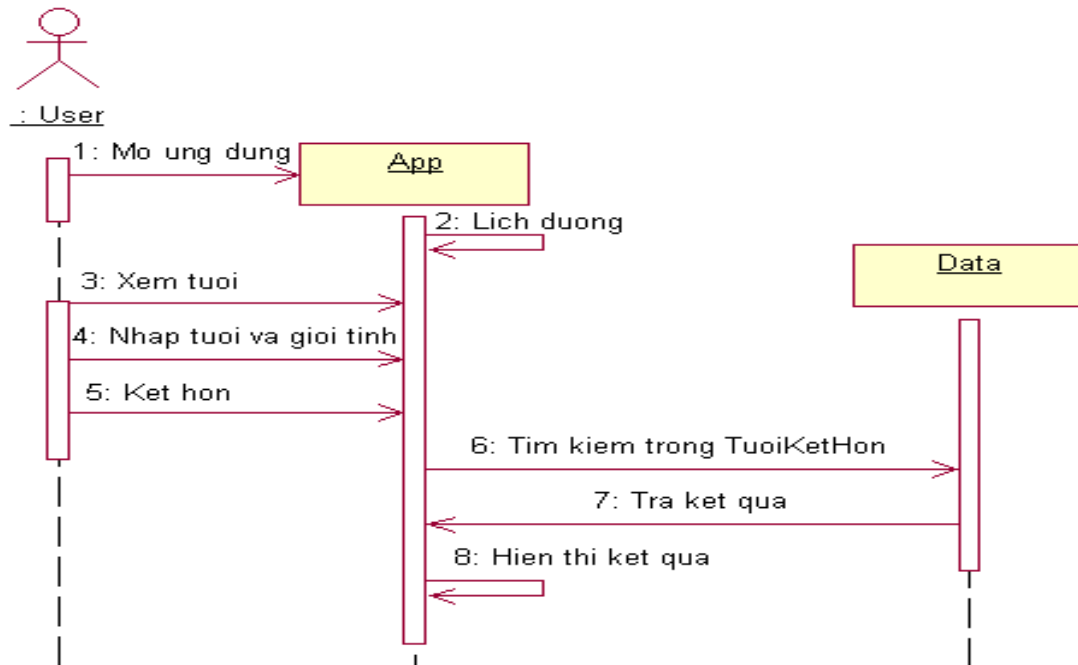
2.3 Quá trình xem tuổi

Xem tuổi để làm nhà là xem tuổi đó có phạm vào ba điều kỵ : Tam Tai, Kim Lâu và Hoang Ốc, nếu phạm vào 1 trong 3 điều kỵ đó sẽ không được làm nhà. Chương trình sẽ tự động tính và đưa ra lời khuyên bạn nên làm nhà trong năm nay không, những năm bạn có thể làm trong tương lai, những tuổi có thể làm nhà trong năm nay.



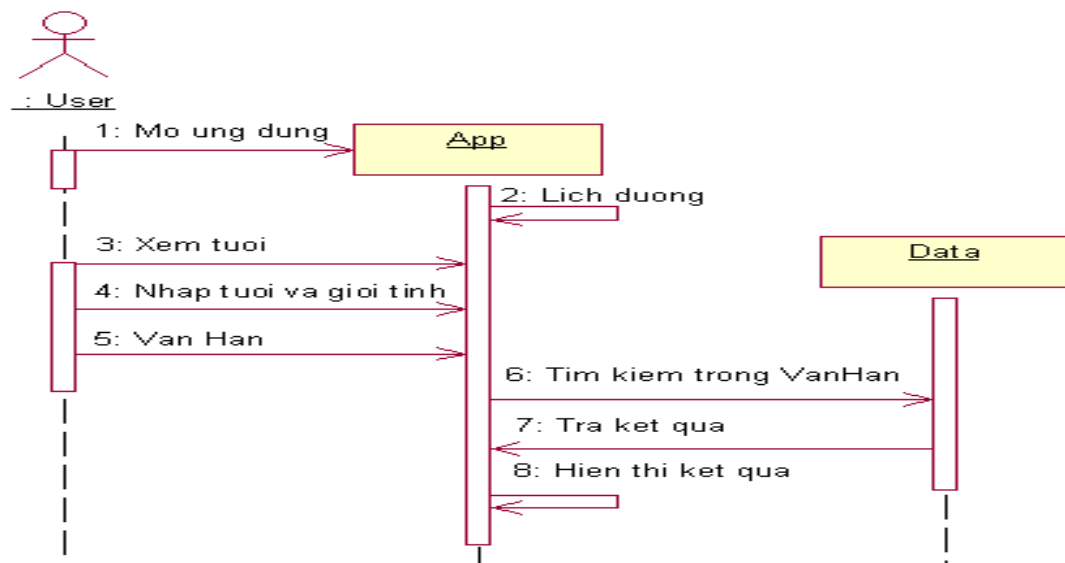
Hình 2.5 Sơ đồ tuần tự quá trình xem tuổi làm nhà

Xem tuổi để kết hôn là xem tuổi của bạn hợp với tuổi nào, không hợp với tuổi nào. Chương trình sẽ đưa ra lời khuyên mang tính tham khảo cho bạn dựa vào dữ liệu năm sinh và giới tính bạn đã nhập.



Hình 2.6 Sơ đồ tuần tự quá trình xem tuổi kết hôn

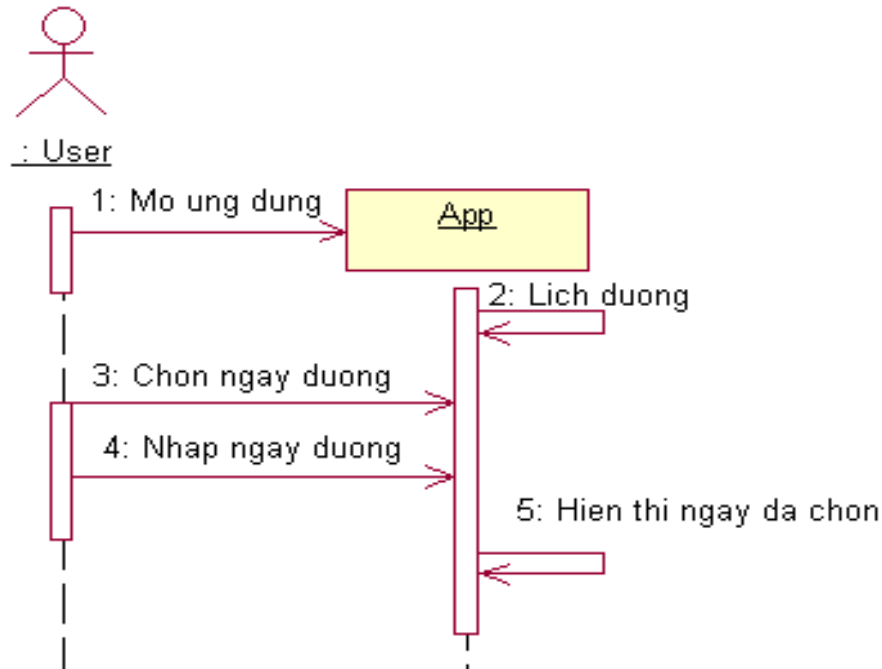
Xem vận hạn là xem tuổi đó trong năm nay có biến cố gì về tiền tài, danh vọng và bệnh tật, ... Mỗi tuổi của nam và nữ tương ứng với một sao. Mỗi sao sẽ mang một ý nghĩa khác nhau và đó cũng là vận hạn của tuổi đó trong năm. Chương trình sẽ cho biết điều đó.



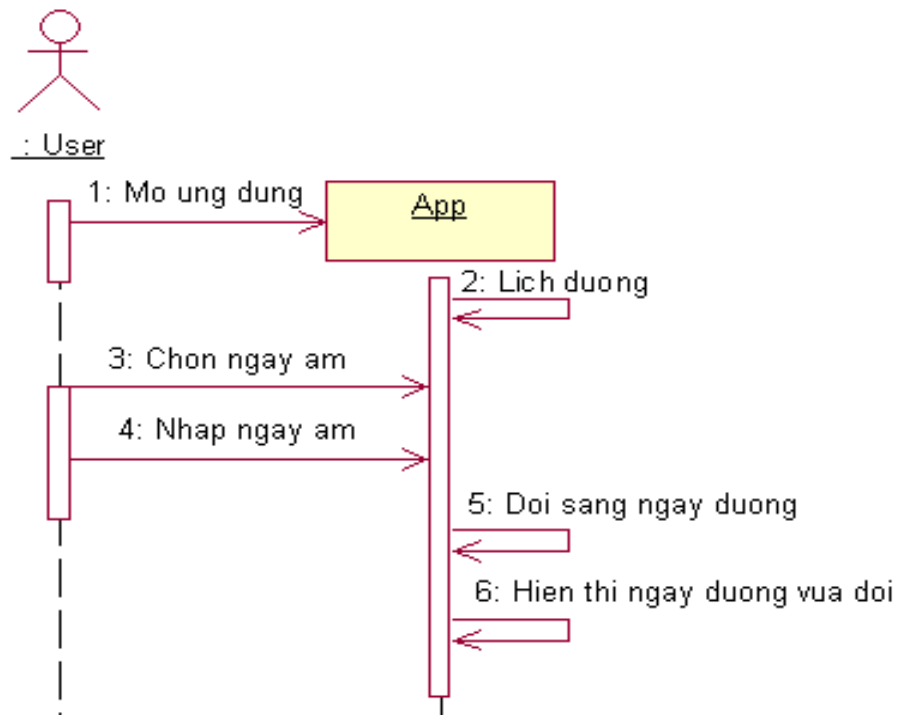
Hình 2.7 Sơ đồ tuần tự quá trình xem vận hạn của tuổi

2.4 Quá trình chọn ngày

Để đến một ngày khác, ta chỉ cần nhập ngày, tháng, năm dương lịch hoặc nhập vào ngày, tháng, năm âm lịch từ đó chương trình sẽ tự động chuyển sang ngày dương và hiển thị ra màn hình.



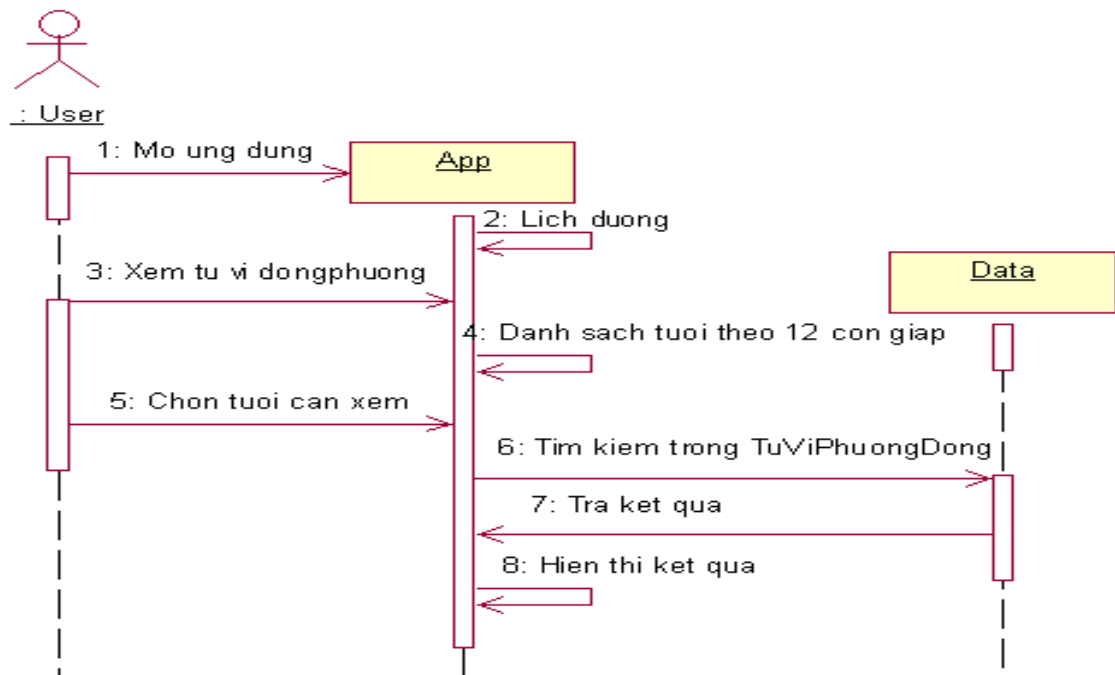
Hình 2.8 Sơ đồ tuần tự quá trình chọn ngày dương



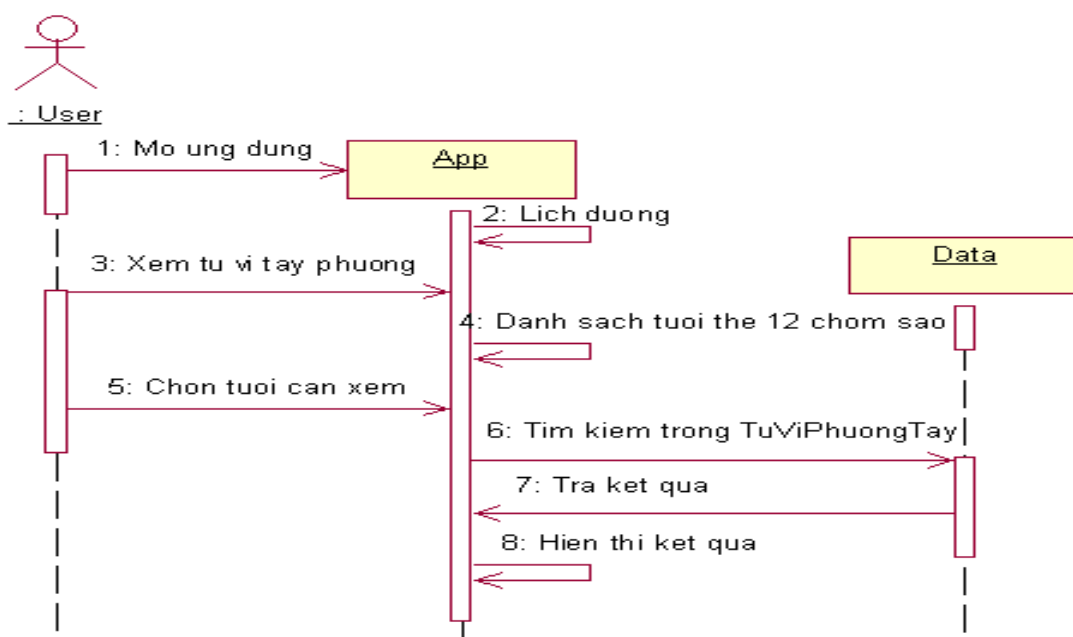
Hình 2.9 Sơ đồ tuần tự quá trình chọn ngày âm

2.5 Quá trình xem tử vi

Xem tử vi để khám phá bản thân của bạn thông qua năm sinh hoặc ngày tháng năm sinh. Điều đó góp phần tìm được ưu điểm và nhược điểm để có cách phát huy và khắc phục, giúp bạn thành công dễ dàng hơn trong cuộc sống. Có 2 cách xem: Tử vi đông phương theo 12 con giáp, Tử vi tây phương theo 12 chòm sao.



Hình 2.10 Sơ đồ tuần tự quá trình xem tử vi đông phương



Hình 2.11 Sơ đồ tuần tự xem tử vi tây phương

3. TỔ CHỨC CÁC LỚP

3.1 Lớp ReadFileText

Lớp ReadFileText dùng để đọc các file dữ liệu dạng text (.txt). Nó có các phương thức hỗ trợ đọc từng từ, từng dòng và nhảy đến dòng bất kỳ.

Các phương thức quan trọng của ReadFileText:

Phương thức	Chức năng
ReadFileText (String FilePath)	Khởi tạo ReadFileText với đường dẫn của file cần đọc
String readLine ()	Đọc một dòng trong file.
String[] readWordsLine ()	Đọc một dòng nhưng trả về các từ trong dòng đó
boolean seek (String s)	Di chuyển con trỏ đến dòng kế tiếp sau dòng s

Bảng 2.2: Các phương thức lớp ReadFileText

3.2 Lớp DateProcessing

Đây là lớp chứa các phương thức xử lý ngày tháng năm dương lịch âm lịch. Dùng để chuyển đổi ngày âm sang ngày dương và ngược lại. Ngoài ra còn có các phương thức như đổi ngày, tháng, năm ra can chi, cộng trừ ngày, kiểm tra giờ hoàng đạo, ...

Các phương thức quan trọng trong lớp ProcessDate:

Phương thức	Chức năng
int convertDateToJulius (int dd, int mm, int yy)	Đổi ngày dương sang số ngày
MyDate convertJuliusToDate (int jd)	Đổi số ngày sang ngày
MyDate convertSolarToLunar (int dd, int mm, int yy, double timeZone)	Đổi ngày dương sang ngày âm
MyDate convertLunarToSolar (int lunarDay, int lunarMonth, int lunarYear, int lunarLeap, double timeZone)	Đổi ngày âm sang ngày dương
String getCanChi (int yy, boolean CanChi)	Đổi năm yy ra can chi

String getCanChi (int mm, int yy,boolean CanChi)	Đổi tháng mm của năm yy ra can chi
String getCanChi (int dd, int mm, int yy,boolean CanChi)	Đổi ngày đ của tháng mm của năm yy ra can chi
MyDate addMyDate (int k, int dd, int mm, int yy)	Cộng thêm k ngày
int getLeapMonthOfYear (int yy, int timeZone)	Lấy tháng nhuận của năm âm lịch
int getNumberDayOfLunarMonth (int mm, int leapMonth, int yy)	Lấy số ngày của tháng âm
String getCanChiGio (String CanChi,int gio, boolean Chuoi)	Lấy can chi của giờ theo ngày CanChi
String getHoangDao (String CanChi)	Lấy giờ hoàng đạo trong ngày.
int getDayBetweenTwoDate (int dd1, int mm1, int yy1, int dd2, int mm2, int yy2)	Lấy khoảng cách ngày giữa 2 ngày dương

Bảng 2.3: Các phương thức lớp ProcessDate

3.3 Lớp CheckingForBuilding

Lớp này dùng để tính toán một tuổi nào đó có phạm vào các điều kị không, có thể làm nhà được không. Có các phương thức sau:

Phương thức	Chức năng
boolean checkAge (int yy1, int yy2)	Kiểm tra người có năm sinh yy1 có thể làm nhà vào năm yy2 được không
boolean checkTamTai (int yy1, int yy2)	Kiểm tra người có năm sinh yy1 có phạm vào điều kị Tam Tại của năm yy2 không
boolean checkKimLau (int yy1, int yy2)	Kiểm tra người có năm sinh yy1 có phạm vào điều kị Kim Lâu của năm yy2 không

boolean checkHoangOc (int yy1, int yy2)	Kiểm tra người có năm sinh yy1 có phạm vào điều kị Hoang Ốc của năm yy2 không
--	---

Bảng 2.4: Các phương thức lớp CheckingForBuilding

3.4 Lớp MyDate

Lớp MyDate được dùng để lưu trữ một đối tượng ngày gồm cả ngày âm và ngày dương. Lớp này chứa các phương thức set và get các thuộc tính của lớp.

Phương thức	Chức năng
MyDate (int day, int month, int year)	Khởi tạo đối tượng ngày với
int getDay ()	Lấy ngày
setDay (int day)	Thiết lập ngày
int getMonth ()	Lấy tháng
setMonth (int month)	Thiết lập tháng
int getYear ()	Lấy năm
setYear (int year)	Thiết lập năm
int getLeapMonth ()	Lấy tháng nhuận nếu 1 nhuận
setLeapMonth (int leapMonth)	Thiết lập tháng nhuận

Bảng 2.5: Các phương thức lớp MyDate

3.5 Các lớp giao diện

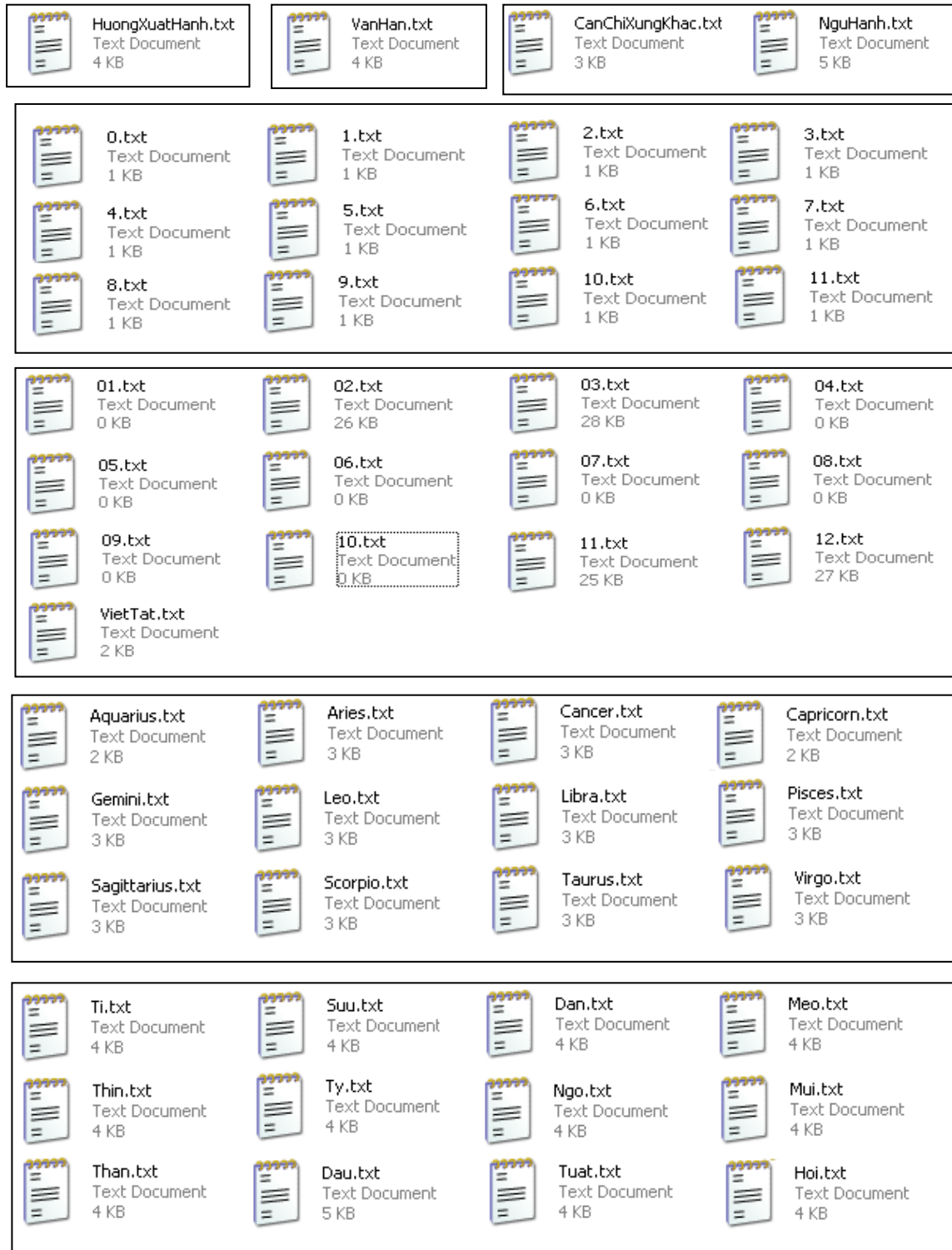
Các lớp giao diện dưới đây sẽ sử dụng các lớp trên để truy vấn dữ liệu, tính toán và hiển thị các thông tin cần thiết cho người dùng. Các lớp này được thừa kế từ các lớp khác trong J2ME như Canvas, Form, List,... Trong chương trình này, ta sử dụng các lớp giao diện cơ bản sau:

Tên lớp	Chức năng
Start	Hiển thị menu chọn lựa các thao tác với ứng dụng. Các dialog thao tác và hiển thị thông tin: Chọn ngày, xem tuổi, xem tử vi, ...
SolarCalendar	Dùng để hiển thị tất cả các ngày dương trong một tháng, những ngày cuối cùng của tháng trước , những ngày đầu của tháng sau. Hiển thị các dialog thao tác liên quan đến ngày, tháng năm: Xem ngày âm dương, Hướng xuất hành, Nên kiêng trong ngày.

Bảng 2.6: Các lớp giao diện cơ bản

4. TỔ CHỨC CƠ SỞ DỮ LIỆU

Dữ liệu của chương trình là các file text. Các file này được lớp ReadFileText đọc và đổ dữ liệu vào các lớp giao diện ở trên như: XuatHanh, VanHan, LamNha, KetHon, NenKiem, TuViTayPhuong, TuViDongPhuong .



Hình 2.12 Sơ đồ dữ liệu

CHƯƠNG 3: TRIỂN KHAI VÀ KẾT QUẢ CHẠY DEMO

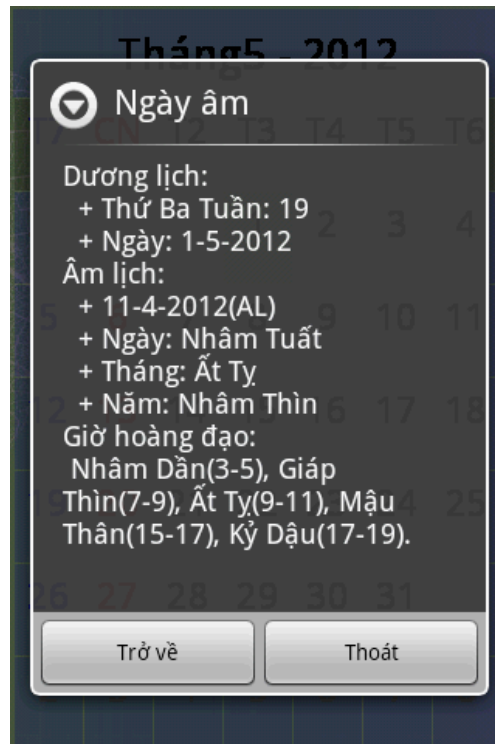
1. MÀN HÌNH HIỂN THỊ LỊCH ÂM, LỊCH DƯƠNG

Khi chạy ứng dụng, màn hình hiển thị lịch dương luôn được hiển thị đầu tiên. Đây cũng là màn hình chính để đi đến các màn hình khác. Từ màn hình này, ta chọn ngày bất kỳ và touch vào màn hình. Sẽ có 1 dialog hiển thị và cho lựa chọn các tính năng xem ngày âm, hướng xuất hành, nên kiêng.



Hình 3.1. Màn hình xem lịch dương lịch âm

Nếu chọn xem ngày âm, thông tin ngày âm sẽ được hiển thị như hình bên dưới bao gồm: Thông tin ngày dương, ngày âm, giờ hoàng đạo trong ngày.



Hình 3.2. Màn hình xem lịch âm

2. MÀN HÌNH XEM THẢO TRONG NGÀY

Tương tự như màn hình trên, cũng từ màn hình dialog chọn “Nên-Kiêng” hoặc “Hướng xuất hành”.



Hình 3.3. Màn hình nên kiêng và hướng xuất hành trong ngày

3. MÀN HÌNH XEM TUỔI

Tương tự như trên, ta chọn “Menu”. Màn hình chọn năm sinh âm lịch sẽ xuất hiện. Trong màn hình này ta chọn năm sinh. Sau đó 1 màn hình sẽ xuất hiện. Từ màn hình này ta cũng có thể chọn hạng mục muốn xem và chọn “Xem tuổi Nam” hoặc “Xem tuổi Nữ”. Sẽ có các dialog để hiển thị các thông tin cho ta biết tuổi của bạn có thể làm nhà trong năm nay không, có thể làm vào những năm nào trong những năm đến, vận hạn trong năm đó ra sao và tuổi của bạn nên kết hôn với tuổi nào, ...



Hình 3.4. Màn hình chọn tuổi, xem tuổi làm nhà, kết hôn và vận hạn



Hình 3.5. Màn hình thông tin về kết hôn, vận hạn, làm nhà

4. MÀN HÌNH CHỌN NGÀY ÂM, NGÀY DƯƠNG

Từ màn hình hiển thị lịch dương, chọn “Menu “ chọn “Chọn ngày” , xuất hiện màn hình cho phép ta chọn ngày dương muốn xem. Chọn “Đồng ý” để hiển thị ngày mà ta đã chọn.



Hình 3.6. Màn hình chọn ngày dương và hiển thị



Hình 3.7. Màn hình chọn ngày âm và hiển thị

5. MÀN HÌNH XEM TỬ VI

Trong “Menu” của màn hình hiển thị lịch dương ta bấm chọn “Tử vi Đông phương” sẽ xuất hiện list chứa danh sách tuổi Tí, Sửu, ... Sau đó ta chọn tuổi của mình trong danh sách để xem tính cách của mình, các tuổi hợp, tuổi kỵ



Hình 3.8. Màn hình chọn tuổi và xem tính cách

Tương tự như trên, nhưng ta chọn “Tử vi Tây phương” để xem tính cách, cuộc đời, các tuổi hợp, tuổi kỵ của bạn theo ngày sinh thông qua các chòm sao.



Hình 3.9. Màn hình chọn chòm sao và xem tính cách

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. KẾT QUẢ ĐẠT ĐƯỢC

1.1. Về mặt lý thuyết

Có nhiều vấn đề cần nghiên cứu để xây dựng được một chương trình chạy trên thiết bị di động hoàn chỉnh. Tuy vậy, trong đồ án tốt nghiệp em chỉ đặt ra một số yêu cầu cần giải quyết. Cụ thể là hiển thị được lịch âm dương, đưa một số nội dung mang tính chất tham khảo trong cuộc sống hằng ngày. Với yêu cầu đặt ra như thế, tôi cũng đã nghiên cứu khá nhiều các lý thuyết liên quan đến việc xây dựng ứng dụng trên điện thoại di động, ngôn ngữ lập trình và dữ liệu về phong tục xem ngày truyền thống. Ứng với mỗi chức năng em nghiên cứu các thuật toán thực hiện chức năng đó để từ đó rút ra nhận xét, so sánh rồi mới chọn lựa thuật toán phù hợp cho việc cài đặt ứng dụng. Chi tiết về những lý thuyết nghiên cứu được em đã trình bày khá rõ ràng trong chương II. Đó chính là kết quả đạt được về mặt lý thuyết của đồ án.

1.2 Về thực nghiệm

Dựa vào yêu cầu đặt ra ban đầu của đồ án tốt nghiệp, tôi đã xây dựng thành công và chạy thử nghiệm trên điện thoại di động và đạt được kết quả sau:

- Xây dựng thành công ứng dụng chạy thật sự trên thiết bị di động điển hình là điện thoại di động.
- Giao diện dễ sử dụng.
- Số năm mà ứng dụng có thể chạy được khá lớn.
- Đưa được tiện ích xử lý ngày và một số tham khảo hằng ngày trong cuộc sống.

2. HẠN CHẾ

Mặc dù luận văn đã đạt được một số kết quả khả quan như trên nhưng vẫn còn một số hạn chế:

- Giao diện chưa được hoàn thiện, cố định nên không thể thay đổi được, gây nhàm chán cho người dùng.
- Chưa phát triển cho các dòng điện thoại có kích thước màn hình khác.
- Dữ liệu phải load lên từ file text nên còn chậm.
- Còn nhiều dữ liệu tham khảo hằng ngày chưa tích hợp được hết.

3. HƯỚNG PHÁT TRIỂN ĐỀ TÀI

Trong luận văn này ngoài việc xem ngày âm dương bạn còn có thể xem các tham khảo trong ngày, nên hướng phát triển là tích hợp thêm nhiều tiện ích khác như tử vi hàng ngày, tử vi hàng tuần, tử vi cả năm, xem bói, ...

Ngoài ra, phải thiết kế giao diện sử dụng đẹp hơn, mềm dẻo hơn, có thể thay đổi theo tùy thích người sử dụng.

Tìm ra các giải pháp để tăng tốc load dữ liệu nhanh hơn, đồng thời giải quyết vấn đề dữ liệu nhiều nhưng dung lượng phải nhỏ để thiết bị di động có thể chạy được.

Xây dựng chương trình tự động upload về thiết bị mỗi khi có phiên bản mới

PHỤ LỤC

Phụ lục 1

```
public int convertDateToJulius(int dd, int mm, int yy) {  
    int a = (14 - mm) / 12;  
    int y = yy + 4800 - a;  
    int m = mm + 12 * a - 3;  
    // Lich Gregory  
    int jd = dd + (153 * m + 2) / 5 + 365 * y + y / 4 - y / 100  
        + y / 400 - 32045;  
    // Lich Julius  
    if (jd < 2299161) {  
        jd = dd + (153 * m + 2) / 5 + 365 * y + y / 4 - 32083;  
    }  
    return jd;  
}
```

Phụ lục 2

```
public MyDate convertJuliusToDate(int jd) {  
    int a, b, c;  
  
    // Julius lon hon 2299160 la lich Gregory (Sau 5/10/1582  
    if (jd > 2299160) {  
        a = jd + 32044;  
        b = (4 * a + 3) / 146097;  
        c = a - (b * 146097) / 4;  
    } else { // Lich Julius  
        b = 0;  
        c = jd + 32082;  
    }  
    int d = (4 * c + 3) / 1461;  
    int e = c - (1461 * d) / 4;
```

```
int m = (5 * e + 2) / 153;
int day = e - (153 * m + 2) / 5 + 1;
int month = m + 3 - 12 * (m / 10);
int year = b * 100 + d - 4800 + m / 10;
return new MyDate(day, month, year);
}
```

Phụ lục 3 :

```
private int INT(double d) {
    return (int) Math.floor(d);
}
```

Phụ lục 4

```
private double getNewMoon(int k, double timeZone) {
    double T = k / 1236.85;
    double T2 = T * T;
    double T3 = T2 * T;
    double dr = Math.PI / 180;
    double Jd1 = 2415020.75933 + 29.53058868 * k + 0.0001178 * T2
        - 0.000000155 * T3;
    Jd1 = Jd1 + 0.00033 * Math.sin((166.56 + 132.87 * T - 0.009173 * T2) *
dr);
    double M = 359.2242 + 29.10535608 * k - 0.0000333 * T2 - 0.00000347
        * T3; // Sun's mean anomaly
    double Mpr = 306.0253 + 385.81691806 * k + 0.0107306 * T2 +
0.00001236 * T3; // Moon's mean anomaly
    double F = 21.2964 + 390.67050646 * k - 0.0016528 * T2 - 0.00000239
        * T3; // Moon's argument of latitude
    double C1 = (0.1734 - 0.000393 * T) * Math.sin(M * dr) + 0.0021 *
Math.sin(2 * dr * M);
    C1 = C1 - 0.4068 * Math.sin(Mpr * dr) + 0.0161 * Math.sin(dr * 2 * Mpr);
    C1 = C1 - 0.0004 * Math.sin(dr * 3 * Mpr);
}
```

```
C1 = C1 + 0.0104 * Math.sin(dr * 2 * F) - 0.0051 * Math.sin(dr * (M +
Mpr));
C1 = C1 - 0.0074 * Math.sin(dr * (M - Mpr)) + 0.0004 * Math.sin(dr * (2 *
F + M));
C1 = C1 - 0.0004 * Math.sin(dr * (2 * F - M)) - 0.0006 * Math.sin(dr * (2 *
F + Mpr));
C1 = C1 + 0.0010 * Math.sin(dr * (2 * F - Mpr)) + 0.0005 * Math.sin(dr *
(2 * Mpr + M));
double deltat;
if (T < -11) {
    deltat = 0.001 + 0.000839 * T + 0.0002261 * T2 - 0.00000845 *
T3
    - 0.000000081 * T * T3;
} else {
    deltat = -0.000278 + 0.000265 * T + 0.000262 * T2;
}
double JdNew = Jd1 + C1 - deltat;
return INT(JdNew + 0.5 + timeZone/24);
}
```

Phụ lục 5

```
private double getSunLongitude(double jdn, double timeZone) {
    double T = (jdn - 2451545.0 - 0.5 - timeZone / 24) / 36525; // Time in
Julian centuries from 2000-01-01 12:00:00 GMT
    double T2 = T * T;
    double dr = Math.PI / 180; // degree to radian
    double M = 357.52910 + 35999.05030 * T - 0.0001559 * T2 - 0.00000048
* T * T2; // mean anomaly, degree
    double L0 = 280.46645 + 36000.76983 * T + 0.0003032 * T2; // mean
longitude, degree
    double DL = (1.914600 - 0.004817 * T - 0.000014 * T2) *
Math.sin(dr * M);
}
```

```
DL = DL + (0.019993 - 0.000101 * T) * Math.sin(dr * 2 * M) + 0.000290 *  
Math.sin(dr * 3 * M);  
double L = L0 + DL; // true longitude, degree  
L = L - 360 * (INT(L / 360)); // Normalize to (0, 360)  
return L;  
}
```

Phụ lục 6

```
private int getLunarMonth11(int yy, double timeZone) {  
    double off = convertDateToJulius(31, 12, yy) - 2415021.076998695;  
    int k = INT(off / 29.530588853);  
    int nm = getNewMoonDay(k, timeZone);  
    int sunLong = INT(getSunLongitude(nm, timeZone) / 30);  
    if (sunLong >= 9) {  
        nm = getNewMoonDay(k - 1, timeZone);  
    }  
    return nm;  
}
```

Phụ lục 7

```
private int getLeapMonth(int a11, double timeZone) {  
    int k = INT(0.5 + (a11 - 2415021.076998695) / 29.530588853);  
    int last; // Month 11 contains point of sun longitude 3*PI/2 (December  
            // solstice)  
    int i = 1; // We start with the month following lunar month 11  
    int arc = INT(getSunLongitude(getNewMoonDay(k + i, timeZone),  
timeZone) / 30);  
    do {  
        last = arc;  
        i++;  
    }
```

```
        arc = INT(getSunLongitude(getNewMoonDay(k + i, timeZone),
timeZone) / 30);
    } while (arc != last && i < 14);
    return i - 1;
}
```

Phụ lục 8

```
public MyDate convertSolarToLunar(int dd, int mm, int yy, double timeZone)
{
    MyDate obj = new MyDate();
    int lunarDay, lunarMonth, lunarYear, leapMonth;
    int dayNumber = convertDateToJulius(dd, mm, yy);
    int k = INT((dayNumber - 2415021.076998695) / 29.530588853);
    int monthStart = getNewMoonDay(k + 1, timeZone);
    if (monthStart > dayNumber) {
        monthStart = getNewMoonDay(k, timeZone);
    }
    int a11 = getLunarMonth11(yy, timeZone);
    int b11 = a11;
    if (a11 >= monthStart) {
        lunarYear = yy;
        a11 = getLunarMonth11(yy - 1, timeZone);
    } else {
        lunarYear = yy + 1;
        b11 = getLunarMonth11(yy + 1, timeZone);
    }
    lunarDay = dayNumber - monthStart + 1;
    int diff = INT((monthStart - a11) / 29);
    leapMonth = 0;
    lunarMonth = diff + 11;
    if (b11 - a11 > 365) {
        int leapMonthDiff = getLeapMonth(a11, timeZone);
        if (diff >= leapMonthDiff) {
```

```
        lunarMonth = diff + 10;
        if (diff == leapMonthDiff) {
            leapMonth = 1;
        }
    }
}

if (lunarMonth > 12) {
    lunarMonth = lunarMonth - 12;
}

if (lunarMonth >= 11 && diff < 4) {
    lunarYear -= 1;
}

obj.setDay(lunarDay);
obj.setMonth(lunarMonth);
obj.setYear(lunarYear);
obj.setLeapMonth(leapMonth);
return obj;
}
```

Phụ lục 9

```
public MyDate convertLunarToSolar(int lunarDay, int lunarMonth,
    int lunarYear, int lunarLeap, double timeZone) {
    int a11, b11;
    if (lunarMonth < 11) {
        a11 = getLunarMonth11(lunarYear - 1, timeZone);
        b11 = getLunarMonth11(lunarYear, timeZone);
    } else {
        a11 = getLunarMonth11(lunarYear, timeZone);
        b11 = getLunarMonth11(lunarYear + 1, timeZone);
    }
    int k = INT(0.5 + (a11 - 2415021.076998695) / 29.530588853);
    int off = lunarMonth - 11;
```

```
    if (off < 0) {
        off += 12;
    }
    if (b11 - a11 > 365) {
        int leapOff = getLeapMonth(a11, timeZone);
        int leapMonth = leapOff - 2;
        if (leapMonth < 0) {
            leapMonth += 12;
        }
        // System.out.println(leapMonth);
        if (lunarLeap != 0 && lunarMonth != leapMonth) {
            System.out.println("Invalid input!");
            return new MyDate(0, 0, 0);
        } else if (lunarLeap != 0 || off >= leapOff) {
            off += 1;
        }
    }
    int monthStart = getNewMoonDay(k + off, timeZone);
    return convertJuliusToDate(monthStart + lunarDay - 1);
}
```

TÀI LIỆU THAM KHẢO

Tiếng Việt

- [1] Chu Văn Khánh. *Lịch Vạn Niên Việt Nam Thế Kỷ XXI (2001-2100)*. Nhà xuất bản Văn Hóa - Thông Tin 2008.
- [2] Đỗ Quang Thái. *Tự học ngôn ngữ lập trình Java*. Nhà xuất bản thống kê 1999.

Trang web

- [3] <http://www.informatik.uni-leipzig.de/~duc/amlich/>
- [4] <http://developer.android.com/guide/appendix/g-app-intents.html>
- [5] <http://developer.android.com/training/basics/firstapp/index.html>
- [6] http://www.i2tuvi.com/Xem_tu_vi_26/Tu_vi_Tay_Phuong/
- [7] http://www.i2tuvi.com/Xem_tu_vi_10/Tu_vi_Dong_Phuong/
- [8] <http://www.tinhte.vn/threads/785470/>

TÓM TẮT LUẬN VĂN TỐT NGHIỆP

Cùng với sự phát triển của nền công nghệ thông tin thế giới, sự phát triển công nghệ di động cũng không hề thua kém. Con người sử dụng những thiết bị di động hiện đại ngày càng nhiều. Vì vậy, việc xây dựng những ứng dụng cho những thiết bị này là cần thiết. Mọi công việc hằng ngày có thể được đưa hết vào trong các thiết bị. Lúc đó các thiết bị này giống như người bạn luôn đưa ra những lời khuyên cần thiết cho cuộc sống hằng ngày.

Đề tài ***“Tìm hiểu công nghệ Android - Xây dựng ứng dụng Lịch Vạn Sự Vạn Niên trên mobile”*** nhằm khám phá công nghệ Android và xây dựng một ứng dụng cụ thể góp phần phong phú hơn thế giới các ứng dụng di động. Người dùng có thể nhanh chóng tìm được những gì mong muốn trong việc xem ngày âm lịch và các tham khảo trong ngày, xem tuổi, xem tính cách của mỗi cá nhân.

Ứng dụng mà tôi xây dựng thực hiện được một số chức năng cơ bản về lịch âm dương và xem các tham khảo trong ngày cũng như đoán tính cách con người qua năm sinh hay qua ngày tháng năm sinh: xem ngày dương, ngày âm; hướng xuất hành, nên kiêng trong ngày; xem tuổi để làm nhà, kết hôn, vận hạn; xem tính cách theo tử vi đông phương, tây phương; tính toán ngày; xem các ngày lễ.

Đề tài tôi thực hiện cơ bản đã đạt được những mục tiêu đề ra ban đầu về mặt lý thuyết cũng như thực tiễn. Ngoài ra, tôi cũng đã đưa ra một số phương hướng để phát triển đề tài lớn hơn sau này.