





# ActivityConstant

Gồm các hằng số dùng khi giao tiếp giữa các Activity

## Value Name

Dùng khi 01 intent.PutExtra

+VALUE\_NAME\_VAPLIST\_APINFO : String = “alarmPoint”

+VALUE\_NAME\_APINFO\_MYFILES : String = “tunePath”

+VALUE\_NAME\_APINFO\_RECOMLIST : String = “tunePath”

+VALUE\_NAME\_CTDLIST\_ATASK : String = “task”

**+ VALUE\_NAME\_RINGOFF\_GAME**

**+ VALUE\_NAME\_RINGOFF\_VLT**

## Request Code

Dùng đê quản lý các activity khi start

+REQUESTCODE\_VAPLIST\_APINFO : int = 1000

+REQUESTCODE\_APINFO\_MYFILES : int = 1001

+REQUESTCODE\_APINFO\_RECOMLIST : int = 1002

+REQUESTCODE\_CTDLIST\_ATASK : int = 1003

## Result Code

Dùng đê quản lý kết quả trả về từ Activity

+RESULTCODE\_VAPLIST\_APINFO\_SAVE : int = 2001

+RESULTCODE\_VAPLIST\_APINFO\_REMOVE : int = 2002

# CaldroidAlarmPointListFragment

Fragment dùng chung cho ViewAlarmPointListActivity và ViewRisingPlanActivity. Dùng đê quản lý việc xem danh sách AlarmPoint và các thao tác chỉnh sửa trên danh sách

## Attributes

-alarmPointList : HashMap<Date, AlarmPoint> : danh sách các điểm báo thức được trả về bởi hàm AlarmPointListHandler.GetAlarmPointList(…) khi khởi tạo

~~-calendarFragment : Object : fragment dùng để vẽ và quản lý calendar~~ => this

-keyValue : AlarmPoint : giá trị dùng trao đổi với AlarmPointInfoManagementActivity

-isRisingPlan : Boolean : == True nếu ViewRisingPlanActivity đàng dùng Fragment, == False nếu ViewAlarmPointListActivity đang dùng Fragment

## Operators

+ViewAlarmPointListFragment(isRisingPlan : Boolean): Contructor để khởi tạo biến alarmPointList và isRisingPlan

~~-loadCalendar() : Boolean : sử dụng alarmPointList để xây dựng Calendar~~ => CaldroidCustomAdapter

~~-updateCalendar() : Boolean : cập nhập Calendar sau khi nhận kết quả từ AlarmPointInfoManagementActivity~~ => hiện thực trong hàm onActivityResult

#onCreate(savedInstanceState : Bundle)

#onActivityResult(requestCode : int, resultCode : int, data : Intent)

-startAPInfoMgmActivityForResult() : dùng để start AlarmPointInfoManagementActivity for result, có thể hiện thực thành 01 hàm riêng hoặc hiện thực trong 01 hàm nào đó

# ViewAlarmPointListActivity

Giao diện chính của chương trình. Có Menu với 02 chức năng Create Rising Plan và Modify Recommended Tunes

## Attributes

-vAPListFragment : ViewAlarmPointListFragment

## Operators

+onCreateOptionsMenu(menu : Menu) : Boolean

+onOptionsItemSelected(item : MenuItem) : Boolean

-startRecommendedTunesActivity() : dùng để start RecommendedTunesActivity, có thể hiện thực thành 01 hàm riêng hoặc hiện thực trong onOptionsItemSelected

-startCreateRisingPlanActivity() : dùng để start CreateRisingPlanActivity, có thể hiện thực thành 01 hàm riêng hoặc hiện thực trong onOptionsItemSelected

# ViewRisingPlanActivity

Dùng để xem và chỉnh sửa kết quả của chức năng Create Rising Plan trước khi kết nối (chuyển thành danh sách alarm point chính thức)

## Attributes

-vAPListFragment : ViewAlarmPointListFragment

-connectBtn : Button : khi được click vào, sẽ hiển thị confirmation trước khi kích hoạt hàm RisingPlanHandler.Connect để thực hiện kết nối

## Operators

#onCreate(savedInstanceState : Bundle)

-showComfirmation() : hiển thị yêu cầu xác nhận với người dùng trước khi kết nối

-showSuccessfully() : hiển thị thông báo kết nối thành công

# AlarmPointListHandler

Quản lý các thao tác hỗ trợ Alarm Point List

## Operators

+GetAlarmPointList(isRisingPlan : Boolean) : HashMap<Date, AlarmPoint> : đọc dữ liệu và trả về Alarm Point List

+SaveAlarmPoint(isRisingPlan : Boolean, newAlarmPoint : AlarmPoint) : int : lưu 01 Alarm Point mới vào dữ liệu. Chú ý khi có sự thay đổi của repeat với alarm point cũ

+RemoveAlarmPoint(isRisingPlan : Boolean, date : Date) : int : xoá 01 Alarm Point khởi dữ liệu

+OverwriteAlarmPointList(isRisingPlan : Boolean, newAlarmPointList : HashMap<Date, AlarmPoint>) : int : ghi đè dữ liệu. Nếu danh sách đè là AlarmPointList, thì không ghi lên các AlarmPoint đã được protected. Nếu danh sách đè là 01 RisingPlan cũ, thì xoá và cập nhập lại toàn bộ.

+IsMorning(systemAlarm : DateTime, alarmPointList : HashMap<Date, AlarmPoint>) : Boolean: kiểm tra 01 alarm được gởi từ hệ thống với danh sách alarm point từ dữ liệu xem đó là alarm đó sáng hay chiều

-registerAlarmPointToSystem(alarmPoint : AlarmPoint) : int : đăng ký với hệ thống để alarm có thể reo. Sử dụng RingOffActivity class khi đăng ký.

-removeAlarmPointFromSystem(alarmPoint : AlarmPoint) : int : remove alarmPoint khỏi hệ thống.

# AlarmPoint

Kiểu dữ liệu AlarmPoint

## Attributes

-date : Date : **thêm vào khi khởi tạo, không được thay đổi**

-time01 : Time

-time02 : Time

-tune : String : **khởi tạo là đường dẫn của default tune trong recommended list**

-fadeIn : Boolean : **default = true**

-repeat : Boolean[] : danh sách 7 boolean tương ứng 7 ngày trong tuần, **default = false**

-protected : Boolean : không thể xoá hay ghi đè/chỉnh sửa, **default = false**

-color : Boolean : khi hiển thị trên calendar có tô màu hay không, **default = false**

## Operators

+getDate() : Date

+getTimePoint(position : int) : Time

+getTunePath() : String

+isFadeIn() : Boolean

+getRepeatList() : Boolean []

+isProtected() : Boolean

+isColor() : Boolean

~~+setDate(newDate : Date) : int~~ => AlarmPoint(Date date)

+setTimePoint(newTime : Time, position : int) : int

+setTune(tunePath : String) : int

+setProperties(fadeIn : Boolean, protected : Boolean, **color: Boolean**) : int

+setRepeat(repeatList : Boolean []) : int

**+removeProtect()**

~~+setColor(color : Boolean) : int~~

# CalendarLibrary

Một thư viện để hiện thị và quản lý Calendar View

---------------^ Thịnh ^----------------

# CreateToDoListActivity

Tạo danh sách công việc cho ngày mai trước khi đi ngủ. Có nút Add trên Menu

## Attributes

-todoLV : ListView

-createToDoListAdpt : CreateToDoListAdapter

-todoList : Task[] : danh sách công việc được load bởi hàm ToDoListHandler.GetToDoList. Trường hợp mới tạo lần đầu, danh sách này sẽ rỗng. Trường hợp vào tạo lại lần 02 từ VLTCheckList, danh sách này sẽ có giá trị.

-keyValue : Task : giá trị dùng trao đổi với AddTaskActivity

-saveBtn : Button : lưu danh sách với hàm ToDoListHandler.SaveToDoList(…) và trở về Activity trước

## Operators

#onCreate(savedInstanceState : Bundle)

#onActivityResult(requestCode : int, resultCode : int, data : Intent)

+onCreateOptionsMenu(menu : Menu) : Boolean

+onOptionsItemSelected(item : MenuItem) : Boolean

-startAddTaskActivityForResult() : dùng để start AddTaskActivity for result. Có thể được hiện thực thành 01 hàm riêng hoặc hiện thực trong onOptionsItemSelected

# AddTaskActivity

Hỗ trợ CreateToDoListActivity thêm task mới

## Attributes

-taskTV : TextView

-timeTV : TextView

-addBtn : Button : gởi keyValue về cho CreateToDoListActivity

-keyValue : Task : giá trị dùng trao đổi với CreateToDoListActivity

## Operators

#onCreate(savedInstanceState : Bundle)

# ViewToDoListActivity

Xem ToDo List của ngày hôm nay

## Attributes

-todoLV : ListView

-todoAdpt : SimpleAdapter : SimpleAdapter là 01 Adapter nào đó dùng để quản lý danh sách với item là 01 chuỗi string

-todoList : Task[] : danh sách công việc được load bởi hàm ToDoListHandler.GetToDoList. Trường hợp danh sách rỗng, hiện danh sách rỗng và thông báo “hôm nay không có việc để làm”.

-viewVFTCheckListBtn : Button : start ViewVFTCheckListActivity

## Operators

#onCreate(savedInstanceState : Bundle)

-~~showNotification()~~

# ToDoListHandler

## Operators

+GetToDoList(toView : Boolean) : Task [] : Get ToDo List từ dữ liệu. nếu toView là True, thì dữ liệu với có Date == ToDay mới trả về. Nếu toView là False, thì dữ liệu phải có Date == Tomorrow mới trả về

+SaveToDoList(todoList : Task []) : int : lưu danh sách vào dữ liệu

# CreateToDoListAdapter

View của Adapter này gồm 01 textView và 01 Remove Button.

## Attributes

-ctx : Context

-todoList : Task[]

## Operators

+CreateToDoListAdapter(ctx : Context, todoList : Task [])

+getView(position : int &, convertView : View, parent : ViewGroup)

-removeTask(position : int &) : hàm xử lý khi click vào Remove Button. Có thể hiện thực thành hàm riêng hoặc hiện thực trong getView(…)

# Task

Kiểu dữ liệu Task

## Attributes

-taskName : String

-dateTime : DateTime

## Operators

+Task(date : Date) : tạo Task với taskName rỗng và dateTime chỉ set ngày, không có giờ

+getTask() : String : trả về chuỗi gồm dateTime + taskName

+getName() : String

+getDateTime() : DateTime

+setName(taskName : String) : int

+setDateTime(dateTime : DateTime) : int

# ViewVFTCheckListActivity

## Attributes

-vftLV : ListView

-chkLstAdpt : CheckListAdapter

-vftList : String[] : tạo mặc định trong class luôn hoặc load lên từ dữ liệu có sẵn

## Operators

#onCreate(savedInstanceState : Bundle)

+onAllItemsChecked() : hiện thực từ interface CheckListAdapter.OnCheckListener: khi các items được check hết thì back về View-To-Do-Activity

# ViewVLTCheckListActivity

## Attributes

-vltLV : ListView

-chkLstAdpt : CheckListAdapter

-vltList : String[] : tạo mặc định trong class luôn hoặc load lên từ dữ liệu có sẵn

-turnOffAlarmBtn : Button : tắt chuông bằng hàm RingOffHandler. TurnOffAlarm(alarmTune : int)

-alarmTune : int : keyValue để tắt nhạc chuông

-createToDoListBtn : Button : start CreateTodoListActivity

## Operators

#onCreate(savedInstanceState : Bundle)

+onAllItemsChecked() : hiện thực từ interface CheckListAdapter.OnCheckListener: khi các items được check hết thì back về Activity trước (= tắt app, vì trước đó không có activity nào)

# CheckListAdapter

View của Adapter này gồm 1 TextView và 1 CheckBox

## Attributes

-ctx : Context

-items : String[]

-checkedCount : int : biến đếm để biết khi nào check hết các items

## Operators

+CheckListAdapter(ctx : Context, items : String [])

+getView(position : int &, convertView : View, parent : ViewGroup)

-checkItem(position : int &) : hàm được gọi khi 1 CheckBox được check. Có thể hiện thực trong getView

---------------^ Ẩn ^----------------

# AlarmPointInfoManagementActivity

Xem, chỉnh sửa, tạo mới, xoá bỏ các điểm báo thức trong 01 ngày

## Attributes

-dateTV : TextView

-time01TV : TextView

-time02TV : TextView

-tuneTV : TextView

-fadeInVolumeCBx : CheckBox

-repeatCBx : CheckBox[]

-protectedCBx : CheckBox

-saveBtn : Button

-removeBtn : Button

-keyValue : AlarmPoint : dùng trao đổi với ViewAlarmPointListFragment

-keyValueTune : String : dùng trao đổi với RecommendedTunesActivity và MusicFilesManagement

## Operators

-updateTune() : cập nhập tuneTV sau khi nhận giá trị trả về từ RecommendedTunesActivity hoặc MusicFilesManagement

#onCreate(savedInstanceState : Bundle)

#onActivityResult(requestCode : int, resultCode : int, data : Intent)

-onSaveAlarmPoint() : thực hiện 01 số tác vụ, trả keyValue cùng ActivityConstant.RESULTCODE\_VAPLIST\_APINFO\_SAVE về cho ViewAlarmPointListFragment

-onRemoveAlarmPoint() : thực hiện 01 số tác vụ, trả ActivityConstant.RESULTCODE\_VAPLIST\_APINFO\_REMOVE về cho ViewAlarmPointListFragment

-showPopup() : hiển thị và điều hướng lựa chọn giữa RecommendedTunes và MyFiles

-addTuneToRecommendedList() : thêm tune vào Recommended List sau khi nhận trả về từ MusicFilesManagement

# RecommendedTunesActivity

Xem, chọn và chỉnh sửa RecommendedTunes

## Attributes

-tunes : Tune[]

-tunesLV : ListView

-rTAdpt : RecommendedTunesAdapter

-keyValueTune : String : dùng trao đổi với AlarmPointInfoManagementActivity

## Operators

-setDefault(position : int) : hàm được gọi khi có longClick trên RecommendedTunesAdapter. Chú ý, chỉ có 01 tune trong list là default.

-returnResultToAPInfoMgmActivity() : hàm được gọi khi có shortClick trên RecommendedTunesAdapter trong trường hợp RecommendedTunesActivity được start bởi AlarmPointInfoManagementActivity

# RecommendedTunesHandler

Hỗ trợ RecommendedTunes

## Operators

+GetTunes() : Tune [] : từ dữ liệu

+AddTune(newTune : Tune) : int

+RemoveTune(position : int) : int

+ModifyTune(position : int, tuneInfo : Tune) : int

+FindDefault(tunes : Tune []) : int

+CreateFadeInVolumeSound(path : String) : String : trong trường hợp hệ thống không hỗ trợ tăng dần âm lượng cho alarm, dùng hàm này để tạo ra 01 file âm thanh có hiệu ứng FadeIn và trả về đường dẫn của file đó

# RecommendedTunesAdapter

View của adapter này gồm 02 TextView (tên tune + default or not) và 01 remove button

## Attributes

-ctx : Context

-tunes : Tune[]

## Operators

+RecommendedTunesAdapter(ctx : Context, tunes : Tune [])

+getView(position : int &, convertView : View, parent : ViewGroup)

-removeTune(position : int &) : xử lý khi nhấn vào nút remove. Có thể hiện thực trong getView

# Tune

Kiểu dữ liệu Tune

## Attributes

-path : String

-default : Boolean

## Operators

+getTunePaht() : String

+getTuneName() : String : sinh ra từ path

+isDefault() : Boolean

+setTunePath(path : String) : int

+setDefault(default : Boolean) : int

# MusicFilesManagement

Activity nào đó của hệ thống dùng để hỗ trợ chọn tune cho alarm point từ myfiles

# GameActivity

Giao diện Freaking Math.

## Attributes

-volumeIcon : Button : hiển thị âm lượng nhạc chuông đang tăng dần

-countTV : TextView

-mathTV : TextView

-correctBtn : Button : gọi hàm processInput(true) từ GameHandler, nếu hàm trả về Win\_code, thì gọi winGame, nếu không gọi updateGameWorld

-wrongBtn : Button : gọi hàm processInput(false) từ GameHandler, nếu hàm trả về Win\_code, thì gọi winGame, nếu không gọi updateGameWorld

-currentGame : GameHandler

-alarmTune : int : keyValue để tắt nhạc chuông khi dùng hàm RingOffHandler. TurnOffAlarm(alarmTune : int)

## Operators

-updateVolumeIcon() : cập nhập âm lượng cho giao lượng

-updateGameWorld() : cập nhập tất cả các View từ dữ liệu trong biến currentGame

-winGame() : một số thao tác khi thắng + start ViewToDoList Activity, rồi finish

#onCreate(savedInstanceState : Bundle)

# GameHandler

BackEnd của Freaking Math

## Attributes

-count : int : số phép toán cần trả lời đúng còn lại. Khởi tạo tại Contructor.

-oldMathList : Math[] : danh sách các bài toán đã tạo (để tránh việc sinh 02 bài toán giống nhau trong 1 game)

-currentMath : Math : bài toán hiện tại

## Operators

~~+GameHandler(count : int)~~

**+ getInstance():GameHandler**

**+ get\_count(): int**

**+ get\_currentMath():Math**

~~+getCurrentStep() : HashMap<Math, int> : trả về currentMath + count~~

+processInput(userChoice : **boolean**) : int : **nếu trả lời sai sẽ return wrong code, nếu đúng sẽ tạo math mới.**

**+randomMath()** : Math : trả về 01 bài toán mới khác với các bài toán trong oldMathList

~~-isCorrect(math : Math, userChoice : Boolean) : Boolean : hỗ trợ proceesInput()~~ đã được hiện thực trong Math

~~-generateNextStep() : int : cập nhập oldMathList => cập nhập currentMath với randomizeMath().~~ **Đã được hiện thực trong processInput(…)**

~~-checkWinCondition() : Boolean : win khi count = 0~~

# Math

## Attributes

-factor01 : int

-factor02 : int

-**isAddOperation**: Boolean : True => Add. False => Minus.

-fakeResult : int

## Operators

**+ get\_factor\_1() : String**

**+ get\_factor\_2() : String**

**+ isAddOperation() : boolean**

**+ get\_fakeResult() : String**

**+ getMathResult() : boolean**

**+ toString() : String**

# GameConstant

## Attributes

+WIN\_CODE : int

**+NUMBER\_OF\_MATCHES**

**+WRONG\_ANSWER**

**+MIN\_NUMBER**

**+MAX\_NUMBER**

**+ZEN\_RIGHT\_ANS\_RATE**

**+FAKE\_ANS\_RANGE**

---------------^ Đạt + Ẩn (Giao Diện) ^----------------

# DataHelper

Class dùng để Tạo, Truy xuất hoặc Thay đổi Dữ liệu. Khuyến khích sử dụng cấu trúc file XML để hiện thực thay vì dùng Database.

## Operators

+GetAlarmPointListFromData(isRisingPlan : Boolean) : HashMap<Date, AlarmPoint> : Nếu isRisingPlan == True, đọc dữ liệu RisingPlan, ngược lại đọc dữ liệu AlarmPointList và trả về danh sách alarm point.

+SaveAlarmPointListToData(isRisingPlan : Boolean, newAlarmPointList : HashMap<Date, AlarmPoint>) : int : Nếu isRisingPlan == True, xoá dữ liệu cũ, lưu dữ liệu RisingPlan mới. Ngược lại, thao tác trên dữ liệu AlarmPointList.

+SaveAlarmPointToData(isRisingPlan : Boolean, newAlarmPoint : AlarmPoint) : int : Tìm điểm AlarmPoint từ dữ liệu (RisingPlan hoặc AlarmPointList), sửa đổi và lưu lại

+RemoveAlarmPointFromData(isRisingPlan : Boolean, date : Date) : int : Xoá điểm AlarmPoint từ dữ liệu (RisingPlan hoặc AlarmPointList)

+GetCurrentToDoListFromData() : Task [] : Đọc dữ liệu ToDoList và trả về danh sách Task

+SaveToDoListToData(todoList : Task []) : int : Xoá dữ liệu cũ, ghi danh sách Task mới xuống dữ liệu ToDoList

+GetTunes() : Tune [] : Đọc dữ liệu RecommendedTunes và trả về danh sách Tune

+AddTuneToData(newTune : Tune) : int : Thêm Tune vào dữ liệu RecommendedTunes

+RemoveTuneFromData(position : int) : int : Xoá Tune tại vị trí position khởi dữ liệu RecommendedTunes

+ModifyTuneFromData(position : int, tuneInfo : Tune) : int : Sửa Tune tại vị trí position của dữ liệu RecommendedTunes

+~~SaveFadeInVolumeTuneToData~~(name : String) : String : Lưu FadeInVolumeTune với tên name, và trả về đường dẫn của tune

# CreateRisingPlanActivity

Tạo ra lịch tập dậy sớm cho người dùng từ những thoái quen và mục tiêu của họ

## Attributes

-avgTimeTV : TextView : thời gian thức dậy trung bình

-expTimeTV : TextView : thời gian thức dậy mong muốn

-startDateTV : TextView : ngày dự định bắt đầu kế hoạch

-endDateTV : TextView : ngày dự định kết thúc kế hoạch

-expSleepHoursTV : TextView : dự tính ngủ 1 ngày mấy tiếng

-planItBtn : Button : chạy hàm RisingPlanHandler.CheckParameters. Nếu kQ trả về == 1, start ViewRisingPlan Activity. Nếu kết quả trả về khác 1, thì chạy hàm showNotification(notification : String) với tham số lấy về từ RisingPlanHandler.GetNotificationFromErrorCode.

## Operators

-showNotification(notification : String)

#onCreate(savedInstanceState : Bundle)

# RisingPlanHandler

Hỗ trợ RisingPlan

## Operators

+CheckParameters(avgTime : Time, expTime : Time, startDate : Date, endDate : Date, expSleepHours : int) : int

+GetNotificationFromErrorCode(int : errorCode) : String : errorCode là kết qua trả về của CheckParameters. Notification là những lí do tại sao tham số không hợp lệ

+CreateRisingPlan(avgTime : Time, expTime : Time, startDate : Date, endDate : Date, expSleepHours : int) : int : tạo RisingPlan từ tham số và lưu xuống dữ liệu.

+Connect() : int

# RingOffActivity

Activity này sẽ được start khi chuông reo

## Attributes

-alarmTune : int : keyValue dùng để tắt chuông mà SystemAlarmManagement gởi tới

-alarm : DateTime : keyValue dùng để xác định alarm mà SystemAlarmManagement gởi tới

-isMorning : Boolean : là kết quả trả về của RingOffHandler.IsMorning

## Operators

-startActivityAfterRingOff() : nếu isMorning == True, start GameActivity. Ngược lại start ViewVLTCheckListActivity. Sau khi start Activity mới, RingOffActivity sẽ finish

#onCreate(savedInstanceState : Bundle)

# RingOffHandler

Hỗ trợ khi chuông reo

## Operators

+TurnOffAlarm(alarmTune : int)

+IsMorning(alarm : DateTime) : Boolean : dùng AlarmPointListHandler.GetAlarmPointList để load danh sách alarm, rồi dùng AlarmPointListHandler. IsMorning() để kiểm tra

# SystemAlarmManagement

Class dùng quản lý các alarm

# Weather manager

+ setWeatherIcon(int actualId, long sunrise, long sunset) : dùng để set icon dự báo thời tiết, tùy vào các ngày khác nhau mà có những icon khác nhau

+ getWeatherJSON(String lat, String lon) : trả về thời tiết dự trên tọa độ (Sử dụng google api)

# FaceGraphic

+ Class dùng để định dạng khuôn mặt

# FaceTrackerActivity

+ Class dùng để nhận diện nụ cười

+ startCameraSource(): khởi động camera

+ GraphicFaceTrackerFactory(): Theo dõi khuôn mặt, phát hiện nụ cười (sử dụng google api)

# Một số lưu ý

* DateTime: là 01 kiểu dữ liệu dữ liệu được trả về từ SystemAlarmManagement (hiện chưa xác định)
  + Sẽ dùng cho Task, RingOffActivity và RingOffHandler
* Time: sử dụng java.sql.Time
* Date: dùng chung với thử viện CalendarLibrary : java.util.Date;
  + <http://javarevisited.blogspot.com/2012/10/difference-between-javasqltime-date-timestamp-jdbc-interview-question.html>
  + <https://docs.oracle.com/javase/7/docs/api/java/sql/Time.html>