# AppCross Interactive Object Library Specification

Version 0.0.1

內容

[AppCross Interactive Object Library Specification 1](#_Toc385437510)

[Version 0.0.1 1](#_Toc385437511)

[1. 概述 3](#_Toc385437512)

[2. Interactive Object Class 4](#_Toc385437513)

[2.1 Catalog 4](#_Toc385437514)

[2.2 Build Interactive Object Library 8](#_Toc385437515)

[3. Interactive Object Library Description 11](#_Toc385437516)

[3.1 Animation Fade 11](#_Toc385437517)

[3.2 Animation Rotate 11](#_Toc385437518)

[3.3 Animation Move 12](#_Toc385437519)

[3.4 Animation Zoom 12](#_Toc385437520)

[3.5 Audio Player 12](#_Toc385437521)

[3.6 Interactive Button 13](#_Toc385437522)

[3.7 Interactive Doodle 13](#_Toc385437523)

[3.8 Finger Paint 14](#_Toc385437524)

[3.9 Flipper Show 15](#_Toc385437525)

[3.10 Gallery View 15](#_Toc385437526)

[3.11 Global 15](#_Toc385437527)

[3.12 Group Object Handler 16](#_Toc385437528)

[3.13 Interactive Image 18](#_Toc385437529)

[3.14 JSON Object 19](#_Toc385437530)

[3.15 Interactive Google map 20](#_Toc385437531)

[3.16 Reader Page Handler 21](#_Toc385437532)

[3.17 Interactive Postcard 21](#_Toc385437533)

[3.18 Interactive Scrollable 22](#_Toc385437534)

[3.19 Interactive Slideshow 23](#_Toc385437535)

[3.20 Interactive Video 24](#_Toc385437536)

[3.21 Interactive Web View 25](#_Toc385437537)

[3.22 Interactive Youtube 25](#_Toc385437538)

## 概述

AppCross Interactive Object Library 主要提供Android AppCross Reader程式互動元件，在設計上每個互動元件設計成獨立的View延伸物件，有些物件會有群組關係，有群組關係的物件會透過群組控制模組來管理，每個互動元件都會依據JSON描述檔來顯示位置跟大小還有屬性。

目前所有的互動元件皆包含在InteractiveView Android Project裡，在此Project內亦提供互動元件會用到的Animation模組、JSON Parser模組、Handler模組、Global模組。

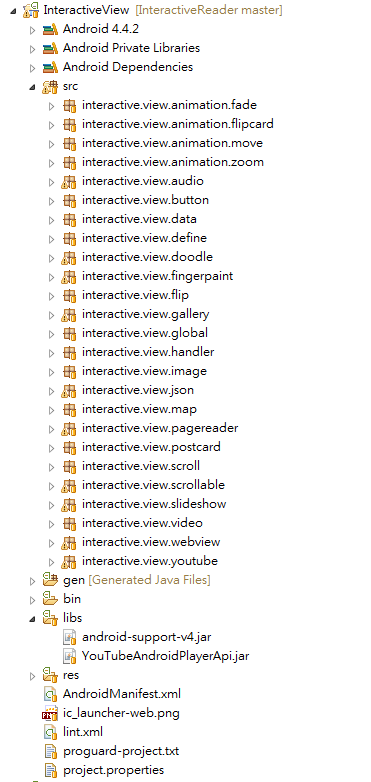
## Interactive Object Class

### Catalog

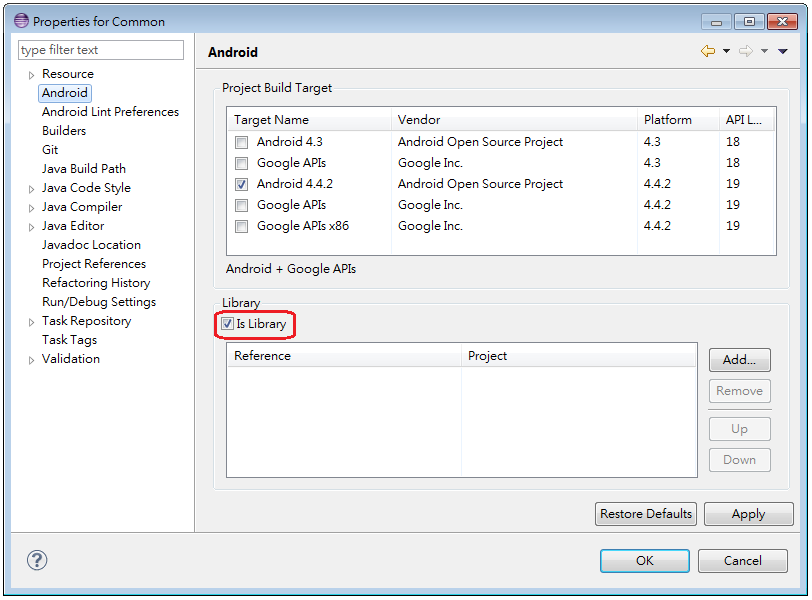
|  |  |  |
| --- | --- | --- |
| **Module Name** | **File** | **Description** |
| Animation Fade | FadeHandler.java | 控制兩個View切換時，淡入淡出的效果 |
| Animation Rotate | Rotate3dAnimation.java  Rotate3d.java | 控制視窗翻轉的動畫效果 |
| Animation Move | MoveHandler.java | 控制視窗移動的動畫效果 |
| Animation Zoom | ZoomHandler.java | 控制視窗zoom in和zoom out動畫效果 |
| Audio Player | AudioPlayer.java  MediaController.java | 聲音播放與播放控制列模組 |
| Interactive Button | ButtonView.java | Button互動元件 |
| Page Data | PageData.java | 頁面資訊模組 |
| Define | InteractiveDefine.java | 互動元件屬性定義檔 |
| Interactive Doodle | DoodleView.java | 繪圖視窗互動元件 |
| Finger Paint | FingerPaintView.java | 繪圖功能模組 |
| Flipper Show | FlipperView.java  AnimationType.java | 控制視窗滑動顯示 |
| Gallery View | GalleryView.java | Gallery視窗 |
| Global | Global.java | 全域共用函數 |
| Data type | Type.java | Define data type |
| Group Object Handler | InteractiveAudioHandler.java  InteractiveButtonData.java  InteractiveButtonHandler.java  InteractiveEventData.java  InteractiveGoogleMapData.java  InteractiveHandler.java  InteractiveImageData.java  InteractiveMediaData.java  InteractiveMediaHandler.java  InteractiveMediaLayout.java | 群組物件控制模組，控制Button元件與Image元件的行為，控管Video單一播放的行為 |
| Interactive Image | EventImageView.java  ImageViewData.java  ImageViewHandler.java | 事件圖片與手勢型為控制 |
| JSON Object | InteractiveAudio.java  InteractiveButton.java  InteractiveDoodle.java  InteractiveIframe.java  InteractiveImage.java  InteractiveMap.java  InteractiveObject.java  InteractivePostcard.java  InteractivePuzzle.java  InteractiveScrollable.java  InteractiveSlideshow.java  InteractiveVideo.java  InteractiveWebPage.java | 個互動元件的Json parser |
| Interactive Google map | GoogleMapActivity.java  GoogleMapFragment.java  GoogleMapView.java | Google map互動元件 |
| Reader Page Handler | ChapterChangeListener.java  ChaptersAdapter.java  DisplayPage.java  HorizonPageView.java  PageChangeListener.java  PageContainer.java  PagerAdapter.java  PageReader.java  PagesAdapter.java  VerticalPagerAdapter.java  VerticalPageView.java  VerticalViewPager.java  ViewPager.java | Reader的頁面控制模組，包含左右翻頁與上下翻頁 |
| Interactive Postcard | Postcard.java  PostcardMailbox.java | 明信片互動元件 |
| Scroll Handler | ScrollHandler.java | 處理視窗scroll事件行為 |
| Interactive Scrollable | AutoScrollableLayout.java  AutoScrollableView.java  HorizonScrollableView.java  ScrollableImageHandler.java  ScrollableView.java  VerticalScrollableView.java | Scrollable互動元件 |
| Interactive Slideshow | SlideshowView.java  SlideshowViewActivity.java  SlideshowViewItem.java | Slideshow互動元件 |
| Interactive Video | MediaController.java  VideoPlayer.java | Video player互動元件 |
| Interactive Web View | InteractiveWebView.java  WebBrowser.java  WebBrowserActivity.java | Web view互動元件 |
| Interactive Youtube | DeveloperKey.java  YoutubeFragment.java  YouTubePlayActivity.java  YoutubeView.java | Youtube player互動元件 |

### Build Interactive Object Library

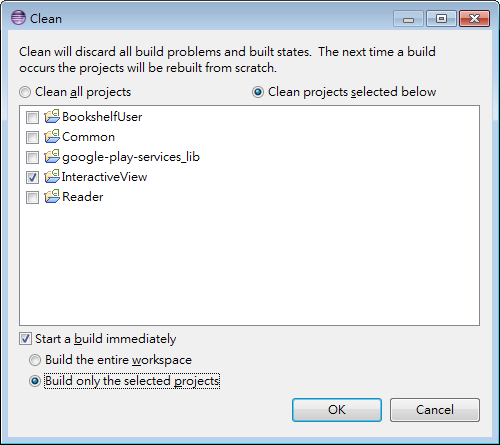
所有Interactive Object Library放在同一個Android Project，Project name是InteractiveView，目錄結構：



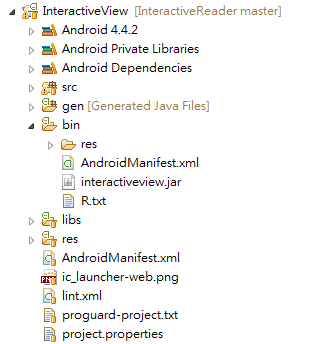
在Build Interactive Object Library時，記得要先確定Project properties內的”Is Library”要被選取



直接執行Eclipse的Clean and Build



Build complete後，會在bin資料夾產生interactiveview.jar即完成。



## Interactive Object Library Description

### Animation Fade

控制視窗切換時，使用淡入淡出來顯示切換。

|  |  |
| --- | --- |
| Function name | Description |
| void crossFade(final View showView, final View hideView, final Handler handler) | 將顯示與隱藏視窗傳入函數做視窗切換，handler是完成後接收message |

### Animation Rotate

控制視窗翻轉效果。

|  |  |
| --- | --- |
| Function name | Description |
| void applyRotation(View view, float start, float end, int nDirection) | 指定起始角度與結束角度來翻轉指定的視窗 |
| void setlistOnRotateEndListener(Rotate3d.OnRotateEndListener listener) | 設定call back，當翻轉結束會被通知 |

### Animation Move

控制視窗動畫移動。

|  |  |
| --- | --- |
| Function name | Description |
| void move(final View view, int nX1, int nX2, int nY1, int nY2, int nDuration, final Handler notifyHandler) | 指定起始的X Y位置與定位的X Y位置來移動指定的視窗，handler為完成後會收到完成的message通知 |

### Animation Zoom

控制視窗zoom in和zoom out動畫效果。

|  |  |
| --- | --- |
| Function name | Description |
| void zoom(View view, float fScaleSize) | 指定scale size去zoom指定的視窗，scale size小於1則zoom out，大於1則zoom in |

### Audio Player

互動元件Audio播放，包含播放控制列。

|  |  |
| --- | --- |
| Function name | Description |
| void init() | 新增播放元件與初始化屬性 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void setPosition(int nChapter, int nPage) | 設定元件位於哪一章節和頁面 |
| void setPlayerControls(boolean bShow) | 是否顯示控制列 |
| void setOption(boolean bAutoPlay, boolean bAutoStop, boolean bLoop, int nStart, int nEnd) | 設定自動播放、自動停止、連續播放與播放起始點屬性 |
| void setImage(String strImagePath, int nWidth, int nHeight) | 設定視窗圖片 |
| void setAudioFile(String strFile) | 設定音樂檔路徑 |
| void play() | 播放音樂 |
| void stop() | 停播音樂 |
| void release() | 回收播放元件 |

### Interactive Button

按鈕互動元件。

|  |  |
| --- | --- |
| Function name | Description |
| void initButton() | 初始背景色、畫面屬性 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void setPosition(int nChapter, int nPage) | 設定元件位於哪一章節和頁面 |
| void setImageSrc(String strSrc, String strTouchDown, String strTouchUp, int nWidth, int nHeight) | 設定視窗normal、down、up三種狀態圖片與大小。 |
| void setNotifyHandler(Handler handler) | 設定要接收通知的Handler |
| void setButtonClickType(int nClickType) | 設定按鈕點擊型態  BUTTON\_TYPE\_TAP：Click  BUTTON\_TYPE\_LONG\_PRESS：Long Click |
| void reset() | 回復按鈕一般狀態 |

按鈕的Event行為定義在onTouchListener內。當點擊時會發出通知給群組管理物件。

### Interactive Doodle

繪圖互動元件。

|  |  |
| --- | --- |
| Function name | Description |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void setPosition(int nChapter, int nPage) | 設定元件位於哪一章節和頁面 |
| void setImageSrc(String strImagePath, int nWidth, int nHeight) | 設定視窗圖片與大小。 |
| void setBrushes(int nBrushes) | 設定筆刷 |
| void setEraseWidth(int nWidth) | 設定橡皮擦寬度 |
| void setPalette(int nPalette) | 設定調色值 |
| void setEraserBtn(float nX, float nY, int nWidth, int nHeight, String strImagePath) | 設定橡皮擦位置大小與圖片 |
| void setPaletteBtn(float nX, float nY, int nWidth, int nHeight, String strImagePath) | 設定調色盤位置大小與圖片 |
| void setPenBtn(float nX, float nY, int nWidth, int nHeight, String strImagePath) | 設定繪筆位置大小與圖片 |
| void setResetBtn(float nX, float nY, int nWidth, int nHeight, String strImagePath) | 設定清除按鈕位置大小與圖片 |
| void setSaveBtn(float nX, float nY, int nWidth, int nHeight, String strImagePath) | 設定儲存按鈕位置大小與圖片 |

Doodle元件主要是透過FingerPaintView去做繪圖的處理，會新增一個FingerPainView在自己的視窗上面。

### Finger Paint

手繪功能視窗，視窗實做Draw繪圖功能。

|  |  |
| --- | --- |
| Function name | Description |
| void init() | 初始化筆刷、橡皮擦、畫布物件。 |
| void onDraw(Canvas canvas) | Call back function，當視窗呼叫invalidate函數時，會觸發onDraw，繪圖將在canvas物件上執行 |
| void touch\_start(float x, float y) | 繪圖起始點 |
| void touch\_move(float x, float y) | 手指移動繪圖，記錄移動路徑 |
| void touch\_up() | 手繪結束，保留手繪線條 |
| void processEvent(MotionEvent event) | 判斷手勢，執行手繪 |
| void clear() | 清除手繪畫面 |
| boolean isCapturing() | 是否執行手繪 |
| void setIsCapturing(boolean mCapturing) | 設定是否執行手繪 |
| void setEraser(boolean bEraser) | 設定是否執行橡皮擦功能 |
| void setBackground(String strImagePath) | 設定手繪視窗背景圖 |
| boolean exportBitmap(String strPath, int nWidth, int nHeight) | 將手繪視窗畫面匯出到指定的檔案路徑，匯出格式為bitmap物件 |
| void setTextToDraw(String strText) | 輸入文字到手繪視窗 |
| void setPenStrokeWidth(int nWidth) | 設定手繪線條寬度 |
| void setEraseStrokeWidth(int nWidth) | 設定橡皮擦寬度 |

### Flipper Show

控制多個視窗從上、下、左、右方向以滑動方式切換顯示。

|  |  |
| --- | --- |
| Function name | Description |
| void init(Context context) | 初始化Flipper物件與Animation屬性 |
| void setNotifyHandler(Handler handler) | 設定訊息通知接收端的Handler |
| int addChild(int nResId) | 新增視窗到Flipper Group，函數會透過Resource id將layout resource轉變為View物件，然後新增到Flipper內 |
| void showView(int nIndex) | 依據index顯示指定的視窗 |

Flipper show物件是一個視窗集合物件，控制多個視窗以滑動方式切換顯示。

### Gallery View

Gallery視窗物件，主要用於顯示書本所有頁面時用，主要是延伸Android HorizontalScrollView。

|  |  |
| --- | --- |
| Function name | Description |
| void init(Context context) | 初始化LinearLayout到HorizontalScrollView。 |
| void setCurrentItem(int nItem) | 將視窗scroll到指定的item位置 |
| void setOnScrollStopListner(GalleryView.onScrollStopListner listner) | 設定Scroll停止時的call back |
| void setItem(SparseArray<SparseArray<String>> listImage, int nWidth, int nHeight) | 新增view item到Gallery視窗 |
| void setViewCenter() | Scroll view to center |
| void scrollToStart() | Scroll到最左邊 |
| void scrollToEnd() | Scroll到最右邊 |

### Global

Global物件主要提供程式全域變數跟函數，變數與韓數皆為static資料型態。

|  |  |
| --- | --- |
| Function name | Description |
| static int getUserId() | 動態取得id，id值從2048開始累加。 |
| static class ActiveNotify | Current page資料結構 |
| static int getResourceId(Context context, String name, String defType) | 取得程式resource id |
| static int ScaleSize(int nSize) | 運算螢幕解析度，取得scale size |
| static void addActiveNotify(int nChapter, int nPage, Handler handler) | 設定指定的章節跟頁面為current page時需發通知 |
| static void addUnActiveNotify(int nChapter, int nPage, Handler handler) | 設定指定的章節跟頁面不為current page時需發通知 |
| static void notifyActive(int nChapter, int nPage) | 通知指定的章節跟頁面是否為current page |
| static void showToast(String strMsg) | 顯示Android toast警示視窗 |

### Group Object Handler

主要負責當互動元件有group功能或互動元件只能執行唯一時的管理

* class InteractiveHandler

主要的互動管理介面模組，包含了Button、Media、Audio的群組跟單執行的管理

|  |  |
| --- | --- |
| Function name | Description |
| void addButton(Context context, String strTag, String strGroupId, ViewGroup viewParent, Handler handler) | 新增button互動元件到管理介面 |
| void addButtonImage(String strButtonTag, String strImageTag, int nWidth, int nHeight, int nX, int nY,  String strSrc, String strGroupId, boolean bIsVisible) | 新增button互動元件會觸發的圖檔 |
| void addButtonMap(String strButtonTag, String strMapTag, int nMapType, double dLatitude, double dLongitude,  int nZoomLevel, String strMarker, int nX, int nY, int nWidth, int nHeight, boolean bIsVisible) | 新增button互動元件會觸發的Google map |
| addButtonEvent(String strButtonTag, int nType, String strTypeName, int nEvent, String strEventName,  int nTargetType, String strTargetID, int nDisplay) | 新增button互動元件會觸發的Event |
| int addPostcard(Postcard postcard) | 新增postcard互動元件到管理介面 |
| Postcard getPostcard(String strTag) | 透過postcard互動元件的tag name取得postcard物件 |
| void setPostcardDragTag(String strTag) | 指定拖曳時的postcard tag name |
| String getPostcardDragTag() | 取得拖曳時的postcard tag name |
| void clearPostcardDragTag() | 清除拖曳時指定的postcard tag name |
| void setDraging(boolean bDraging) | 是否執行拖曳 |
| boolean getDraging() | 取得拖曳狀態 |
| void sendPostcard() | Postcard拖曳到mail box後執行Android的share intent |
| void DragEnd(int nObject) | 結束拖曳 |
| void playVideo(String strTag) | 播放media |
| void stopMedia(int nMediaType) | 停止播放指定的media type播放  Media type：  MEDIA\_TYPE\_LOCAL  MEDIA\_TYPE\_URL  MEDIA\_TYPE\_YOUTUBE |
| void stopMedia(String strTag) | 停止播放指定的media互動元件播放 |
| void addMediaData(String strName, int nWidth, int nHeight, int nX, int nY, String strSrc, int nMediaType,  String strMediaSrc, int nStart, int nEnd, boolean bAutoplay, boolean bLoop, boolean bPlayerControls,  boolean bIsVisible, ViewGroup viewParent, boolean bIsCurrentPlayer) | 新增media互動元件資訊到管理介面 |
| boolean getMediaData(String strName, InteractiveMediaData mediaData) | 取得media互動元件資訊 |
| void initMediaView(Activity activity) | 初始化media player，包含video player跟youtube player |
| void setMediaContainer(String strTagName, ViewGroup viewContainer) | 設定media player的container |
| void addAudioPlayer(AudioPlayer audioPlayer) | 新增audio player到管理介面 |
| void releaseAllAudio() | Release all audio player |
| Handler getNotifyHandler() | 取得管理介面的Handler，互動元件透過此handler發通知給管理介面 |
| void handleReset(int nObjectType, String strTag) | 初始化指定的互動元件 |

互動元件發通知給管理介面的selfHandler，管理介面收到通知後再判斷訊息做後續的處理。

### Interactive Image

* class EventImageView

主要延伸Android ImageView物件，實做手勢行為控制ImageView的縮放與顯示

|  |  |
| --- | --- |
| Function name | Description |
| void init(Context context) | 初始化zoom功能，圖片顯示屬性與手勢的偵測 |
| void setContainer(ViewGroup viewParent) | 設定物件的上層容器 |
| void setBitmap(String strPath, int nWidth, int nHeight) | 設定圖檔 |
| void addGesture(int nDisplay, int nEvent, String strTargetID, int nTargetType, int nType) | 新增手勢類型  IMAGE\_GESTURE\_LONG\_PRESS：長按  IMAGE\_GESTURE\_PINCH：縮小  IMAGE\_GESTURE\_SPREAD：放大 |
| void addEventTargetImage(String strName, String strSrc, String strGroupID, int nX, int nY, int nWidth,  int nHeight, boolean bIsVisible) | 新增手勢Event觸發後要呼叫顯示的圖檔 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| class ScaleGestureListener | 縮放手式的實作類別 |
| boolean createScaleImage() | 產生縮放用的圖檔 |
| void removeScaleImage() | 清除縮放用的圖檔 |
| void createEventImage(InteractiveImageData imageData) | 產生手勢Event觸發後要顯示的圖檔 |
| void removeEventImage() | 清除Event觸發後產生的圖檔 |
| void handleEvent() | 實做Event行為，顯示圖檔或拖曳 |
| void showEventImage(String strImageTag) | 顯示Event觸發後要顯示的圖檔 |

### JSON Object

* class InteractiveObject

InteractiveObject主要定義了Reader要讀取的互動元件描述檔(JSON format file)與各互動元件的資料結構，InteractiveObject是JSON parser父類別，各互動元件有其對應的JSON parser，各互動元件的JSON parser除了parser json外還負責依據描述檔去配置互動元件。

|  |  |
| --- | --- |
| Function name | Description |
| abstract boolean createInteractive(ViewGroup container, String strBookPath, JSONObject jsonAll,  int nChapter, int nPage) | 抽象函數，子類別皆需依據負責的互動元件去複寫實做此函數。 |
| boolean parseJsonHeader(JSONObject jsonObject, JsonHeader jsonHeader) | 讀取JSON file內”Name”、” Width”、 “Height”、”X”、”Y”、”Src”、” IsVisible”、” IsVisible”這些互動元件共通的屬性值。 |
| boolean parseJsonEvent(JSONObject jsonObject, SparseArray<Event> listEvent) | 讀取JSON file內” Event”Tag下的所有屬性值。 |
| boolean parseJsonGesture(JSONObject jsonObject, JsonGesture jsonGesture) | 讀取JSON file內” Gesture”Tag下的所有屬性值。 |
| boolean parseJsonWebPage(JSONObject jsonObject, JsonWebPage jsonWebPage) | 讀取JSON file內” WebPage”Tag下的所有屬性值。 |
| boolean parseJsonIframe(JSONObject jsonObject, JsonIframe jsonIframe) | 讀取JSON file內” Iframe”Tag下的所有屬性值。 |
| boolean parseJsonSlideshow(JSONObject jsonObject, JsonSlideshow jsonSlideshow) | 讀取JSON file內” Slideshow”Tag下的所有屬性值。 |
| boolean parseJsonScrollable(JSONObject jsonObject, JsonScrollable jsonScrollable) | 讀取JSON file內” Scrollable”Tag下的所有屬性值。 |
| boolean parseJsonVideo(JSONObject jsonObject, JsonVideo jsonVideo) | 讀取JSON file內”Media”Tag下的所有屬性值。 |
| boolean parseJsonAudio(JSONObject jsonObject, JsonAudio jsonAudio) | 讀取JSON file內”Audio”Tag下的所有屬性值。 |
| boolean parseJsonMap(JSONObject jsonObject, JsonMap jsonMap) | 讀取JSON file內”Map”Tag下的所有屬性值。 |
| boolean parseJsonButton(JSONObject jsonObject, JsonButton jsonButton) | 讀取JSON file內”Button”Tag下的所有屬性值。 |
| boolean parseJsonPostcard(JSONObject jsonObject, JsonPostcard jsonPostcard) | 讀取JSON file內”Postcard”Tag下的所有屬性值。 |
| boolean parseJsonDoodle(JSONObject jsonObject, JsonDoodle jsonDoodle) boolean parseJsonDoodle(JSONObject jsonObject, JsonDoodle jsonDoodle) | 讀取JSON file內”Doodle”Tag下的所有屬性值。 |
| boolean parseJsonPuzzle(JSONObject jsonObject, JsonPuzzle jsonPuzzle) | 讀取JSON file內”Puzzle”Tag下的所有屬性值。 |

### Interactive Google map

Google Map互動元件。

|  |  |
| --- | --- |
| Function name | Description |
| void init(Context context) | 初始化map click事件 |
| void init(String strTag, Activity activity, int nMapType, double dLatitude, double dLongitude,  int nZoomLevel, String strMarker) | 初始化map經緯度與顯示類型 |
| void setMarker(double latitude, double longitude, String title, String snippet) | 設定地標 |
| void drawPolyline(SparseArray<LatLng> listLatLng) | 顯示路徑規劃 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void initOption() | 設定地標屬性 |

Google Map互動元件主要是透過Android Google Map API 來顯示地圖資訊。

### Reader Page Handler

主要負責Reader翻頁功能，透過Android ViewPager實做翻頁功能。

* class PageReader

|  |  |
| --- | --- |
| Function name | Description |
| void initPageReader(Context context) | 建構與初始化ViewPager物件 |
| void setOnPageSwitchedListener(PageReader.OnPageSwitchedListener listener) | 設定頁面切換時的監聽器 |
| void notifyPageSwitched() | 通知頁面切換 |
| void createBook(SparseArray<SparseArray<DisplayPage>> book) | 新增與配置各頁面內容 |
| void jumpPage(int nChapter, int nPage, boolean bFade) | 跳頁 |
| void crossFade(int nChapter, int nPage) | 頁面以淡入淡出切換顯示 |
| void removeCrossFade() | 移除淡入淡出切換顯示 |
| void goForward() | 跳到上一頁 |
| void addHistory(int nChapter, int nPage) | 紀錄頁面切換 |
| void setCurrentPosition() | 紀錄current page到Global物件，通知互動元件current page |
| int getTotalPage() | 取得總頁數 |
| void lockScroll(boolean bLock) | 鎖定翻頁 |
| void lockHorizonScroll(boolean bLock) | 鎖定橫向翻頁 |
| void lockVerticalScroll(boolean bLock) | 鎖定直向翻頁 |

### Interactive Postcard

明信片互動元件

|  |  |
| --- | --- |
| Function name | Description |
| void initPostcardFrame(String strName, int nX, int nY, int nWidth, int nHeight, String strFront,String strBack) | 初始化postcard畫面與手繪跟翻頁手勢 |
| void initPostcard(String strFront, String strBack) | 設定明信片正面與背面圖片 |
| void switchCard() | 正反面切換 |
| void showFront() | 顯示正面 |
| void showBack() | 顯示背面 |
| void showEdit(boolean bShow) | 顯示編輯圖案 |
| void showImageFrom(boolean bShow) | 顯示拍照與開啟圖片檔案圖示 |
| void initPen(int nWidth, int nHeight, int nX, int nY, String strImagePath) | 設定手繪筆刷 |
| void initEraser(int nWidth, int nHeight, int nX, int nY, String strImagePath) | 設定手繪橡皮擦 |
| void initTextArea(int nWidth, int nHeight, int nX, int nY, String strImagePath) | 設定文字輸入 |
| void initCamera(int nWidth, int nHeight, int nX, int nY, String strImagePath) | 設定拍照 |
| void initOpenButton(int nWidth, int nHeight, int nX, int nY, String strImagePath) | 設定開啟圖片檔案 |
| void sendPostcard() | 呼叫share物件傳送明信片 |

### Interactive Scrollable

Scrollable互動元件

* class ScrollableView

|  |  |
| --- | --- |
| Function name | Description |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void createHorizonScrollView(String strTag, String strPath, int nWidth, int nHeight, int nScrollType, int nOffsetX, int nOffsetY, ViewGroup container) | 建構橫向捲動元件 |
| void createVerticalScrollView(String strTag, String strPath, int nWidth, int nHeight, int nScrollType, int nOffsetX, int nOffsetY, ViewGroup container) | 建構直向捲動元件 |
| void createAutoScrollView(String strTag, String strPath, int nWidth, int nHeight, int nScrollType,int nOffsetX, int nOffsetY, ViewGroup container) | 建構直橫捲向物件 |
| void setImage(String strTag, String strPath, int nWidth, int nHeight, int nScrollType, int nOffsetX,int nOffsetY, ViewGroup container) | 設定圖檔 |

### Interactive Slideshow

Slideshow互動元件

|  |  |
| --- | --- |
| Function name | Description |
| void init(Context context) | 初始化橫向捲動物件 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void setPosition(int nChapter, int nPage) | 設定元件位於哪一章節和頁面，接收頁面切換通知 |
| void setViewCenter() | 子畫面置中 |
| void setThumbnailCenter(int nItem) | 設定thumbnail置中 |
| void setItemTitle(int nCurrentItem) | 設定子畫面標題 |
| int getCenter() | 取得畫面中心位置 |
| void scrollToEnd() | 捲至最右 |
| void setCurrentItem(int nItem) | 捲至指定的子畫面 |
| void setOnScrollStopListner(SlideshowView.onScrollStopListner listner) | 設定捲動停止時的監聽器 |
| void setOnItemSwitchedListener(SlideshowView.OnSlideshowItemSwitched listener) | 設定子化面切換實的通知監聽器 |
| void notifyItemSwitched() | 通知子畫面切換 |
| int getDisplayWidth() | 取得螢幕寬度 |
| int getDisplayHeight() | 取得螢幕高度 |
| ImageView getImageView(int nResId, String strPath) | 建構子畫面圖片視窗 |
| void setItem(SparseArray<SlideshowViewItem> listItem, int nItemWidth, int nItemHeight, boolean bFullScreen) | 設定Slideshow所有的子畫面 |
| InteractiveMediaLayout initMedia(SlideshowViewItem viewItem, int nWidth, int nHeight) | 建構video子畫面 |
| void initIndicator(int nCount) | 初始化頁面指標 |
| void setIndicator(boolean bSelected, int nPosition) | 設定頁面指標狀態 |
| void setIndicatorShow(boolean bShow) | 顯示頁面指標 |
| int getCurrentItem() | 取得當前子畫面的index |
| void initGestureDetector() | 初始化手勢偵測 |
| void setFlingOnePage(boolean bOnePage) | 設定是否捲動一頁 |
| void goOnePage(MotionEvent e1, MotionEvent e2, float velocityX, float velocityY) | 跳頁 |
| void showThumbnail() | 顯示thumbnail |
| void setShowThumbnail(boolean bShow) | 是否顯示thumbnail |
| void initThumbnail() | 初始化thumbnail |
| void setThumbnail(ImageView imageView, String strImagePath) | 設定thumbnail圖片 |

### Interactive Video

Video互動元件，Video Layout的管理Local video播放主要透過Android的VideoView物件。

|  |  |
| --- | --- |
| Function name | Description |
| void init(Context context) | 初始化手勢與建構VideoView |
| void setVideo(String strVideoPath) | 設定要播放的檔案 |
| void setVideo(Uri uri) | 設定要播放的URL |
| void play() | 播放影片 |
| void play(String strVideoPath) | 播放指定的影音檔 |
| void pause() | 暫停播放 |
| void stop() | 停止播放 |
| void seekTo(int nMsec) | 設定播放位置 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void setLoop(boolean bLoop) | 設定重複播放 |
| void showController(boolean bShow) | 設定是否顯示播放控制列 |

### Interactive Web View

主要延伸Android WebView物件。

|  |  |
| --- | --- |
| Function name | Description |
| void init() | 設定WebView屬性 |
| boolean shouldOverrideUrlLoading(WebView view, String url) | Call back function，點擊Web page畫面上的連結時會觸發。 |
| Intent newEmailIntent(Context context, String address, String subject, String body, String cc) | 呼叫Android mail send intent。 |
| void setPosition(int nChapter, int nPage) | 設定元件位於哪一章節和頁面，接收頁面切換通知 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void setAutoPlay(boolean bAutoPlay) | 設定是否自動reload |

### Interactive Youtube

主要引用Google Youtube API，實做YoutubeFragment與YouTubePlayer兩個物件。

* class YoutubeView

|  |  |
| --- | --- |
| Function name | Description |
| void init(String strTag, Activity activity) | 建構youtube fragment |
| void initVideo(String strVideoId, boolean bLoop, boolean bShowControl, final int nStart, final int nEnd) | 設定youtube video屬性 |
| void setLoop(boolean bLoop) | 設定重複播放 |
| void showController(boolean bShow) | 設定是否顯示播放控制列 |
| void setDisplay(int nX, int nY, int nWidth, int nHeight) | 設定視窗位置與大小 |
| void play(String strVideoId) | 播放指定的影音id |
| void pause() | 暫停播放 |
| void seekTo(int nSec) | 設定播放位置 |