Design Document

Classy Notes

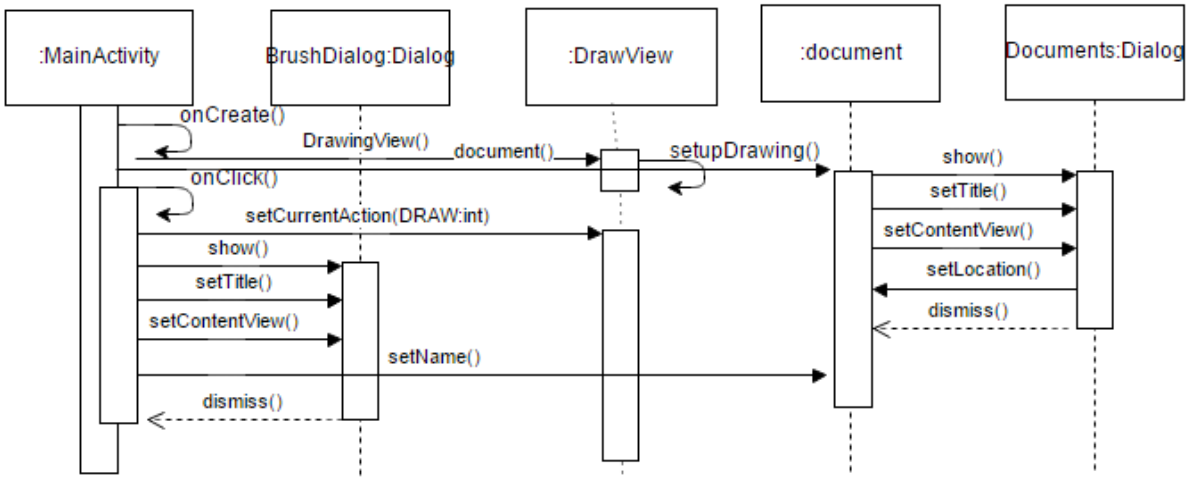
Cassie Mestayer

4/30/14

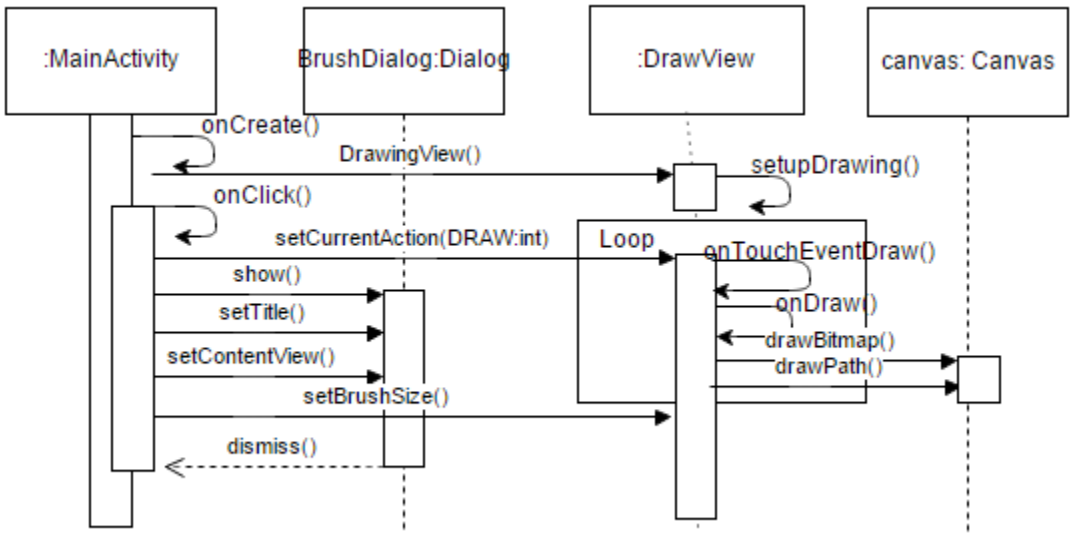
Table of Contents

Interaction Diagrams

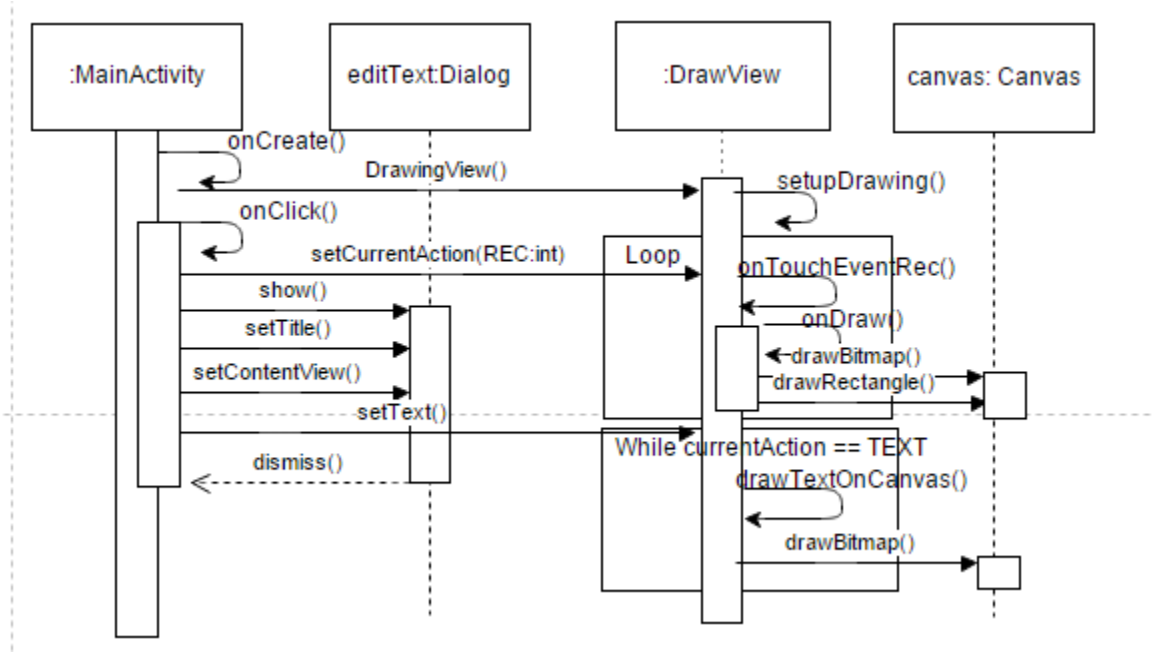
1. Use Case: createNewDocument



2. Use Case: draw

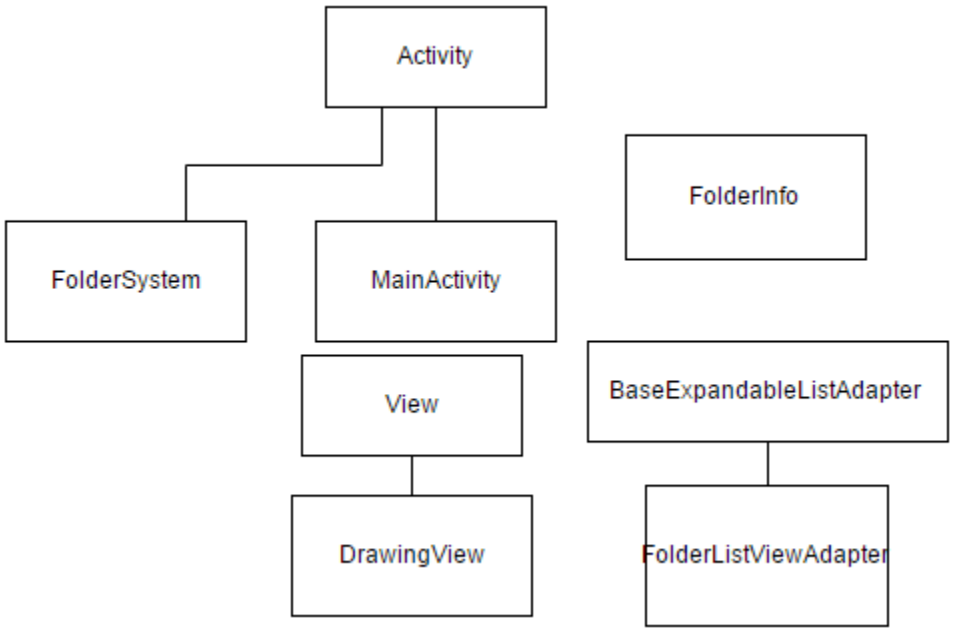


3. UseCase Type



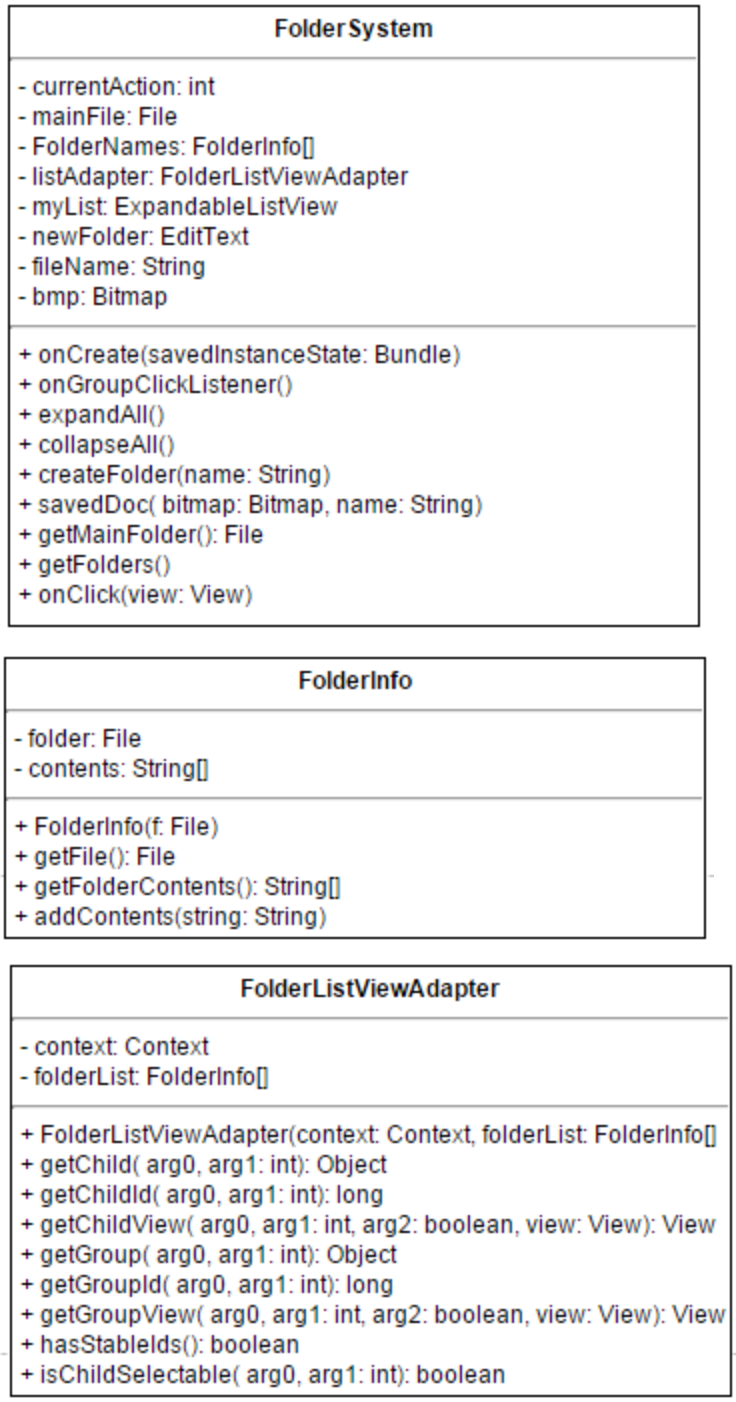
Class Diagram and Interface Specification

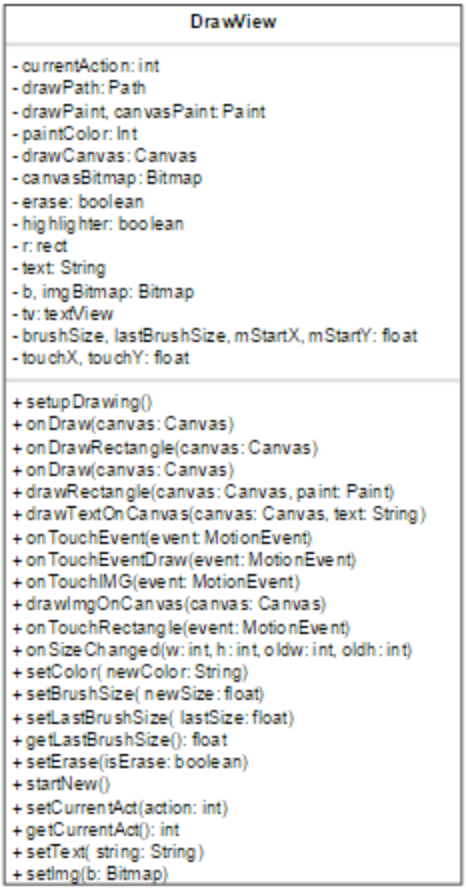
a. Class Diagram



b. Data Types

MainActivity
- title: String - currPaint, drawBtn, eraseBtn, newBtn, saveBtn: ImageButton - textBtn, camBtn, imgBtn, highBtn: ImageButton - smallBrush, mediumBrush, largeBrush: float - drawView: DrawingView
+ onCreate(savedInstanceState: Bundle) + onCreateOptionsMenu(menu: Menu) + paintClicked(view: View) + selectDrawSize(size: Int) + setColor(String newColor) + textClicked(view: View) + setName() + openCamera(view: View) + openLibrary(view: View) + onActivityResult(requestCode: int, resultCode: int, data: Intent) + setBrushSize(view: View, erase: boolean, high: boolean


FolderSystem
- currentAction: int
- mainFile: File
- FolderNames: FolderInfo[]
- listAdapter: FolderListViewAdapter
- myList: ExpandableListView
- newFolder: EditText
- fileName: String
- bmp: Bitmap
+ onCreate(savedInstanceState: Bundle)
+ onGroupClickListener()
+ expandAll()
+ collapseAll()
+ createFolder(name: String)
+ savedDoc(bitmap: Bitmap, name: String)
+ getMainFolder(): File
+ getFolders()
+ onClick(view: View)
FolderInfo
- folder: File
- contents: String[]
+ FolderInfo(f: File)
+ getFile(): File
+ getFolderContents(): String[]
+ addContents(string: String)
FolderListViewAdapter
- context: Context
- folderList: FolderInfo[]
+ FolderListViewAdapter(context: Context, folderList: FolderInfo[]
+ getChild(arg0, arg1: int): Object
+ getChildId(arg0, arg1: int): long
+ getChildView(arg0, arg1: int, arg2: boolean, view: View): View
+ getGroup(arg0, arg1: int): Object
+ getGroupId(arg0, arg1: int): long
+ getGroupView(arg0, arg1: int, arg2: boolean, view: View): View
+ hasStableIds(): boolean
+ isChildSelectable(arg0, arg1: int): boolean

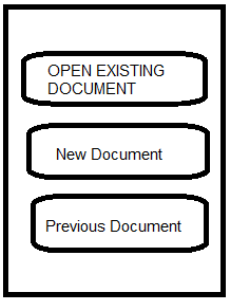

DrawView
- currentAction: int
- drawPath: Path
- drawPaint, canvasPaint: Paint
- paintColor: Int
- drawCanvas: Canvas
- canvasBitmap: Bitmap
- erase: boolean
- highlighter: boolean
- r: rect
- text: String
- b, imgBitmap: Bitmap
- tv: textView
- brushSize, lastBrushSize, mStartX, mStartY: float
- touchX, touchY: float
+ setupDrawing()
+ onDraw(canvas: Canvas)
+ onDrawRectangle(canvas: Canvas)
+ onDraw(canvas: Canvas)
+ drawRectangle(canvas: Canvas, paint: Paint)
+ drawTextOnCanvas(canvas: Canvas, text: String)
+ onTouchEvent(event: MotionEvent)
+ onTouchEventDraw(event: MotionEvent)
+ onTouchIMG(event: MotionEvent)
+ drawImgOnCanvas(canvas: Canvas)
+ onTouchRectangle(event: MotionEvent)
+ onSizeChanged(w: int, h: int, oldw: int, oldh: int)
+ setColor(newColor: String)
+ setBrushSize(newSize: float)
+ setLastBrushSize(lastSize: float)
+ getLastBrushSize(): float
+ setErase(isErase: boolean)
+ startNew()
+ setCurrentAct(action: int)
+ getCurrentAct(): int
+ setText(string: String)
+ setImg(b: Bitmap)

System Architecture and System Design

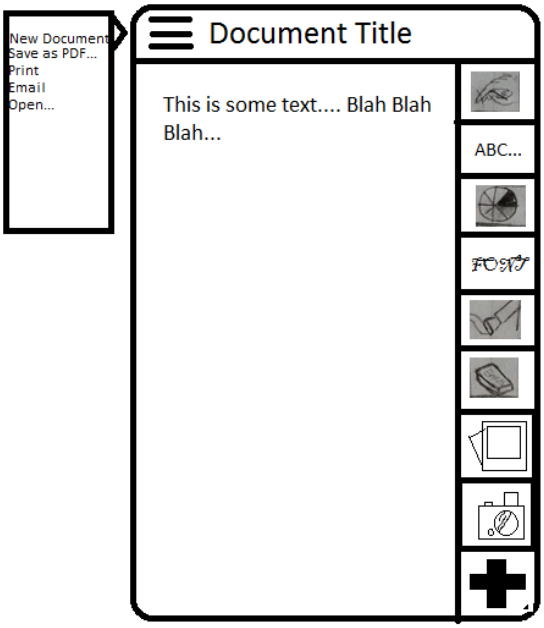
Hardware Requirements

The minimum requirements for this application is an Android device with version 3.0 Honeycomb or higher. To use some of the features of the app like importing pictures from the camera would require the device to have a camera but not limit the device from being able to use the app. It would also be more useful if the user owned a thin point stylus to use for the drawing feature, but again it would not limit the user from being able to use the app. Another feature that benefit from other hardware is being able to type on a separate keyboard rather than the built in on screen keyboard.

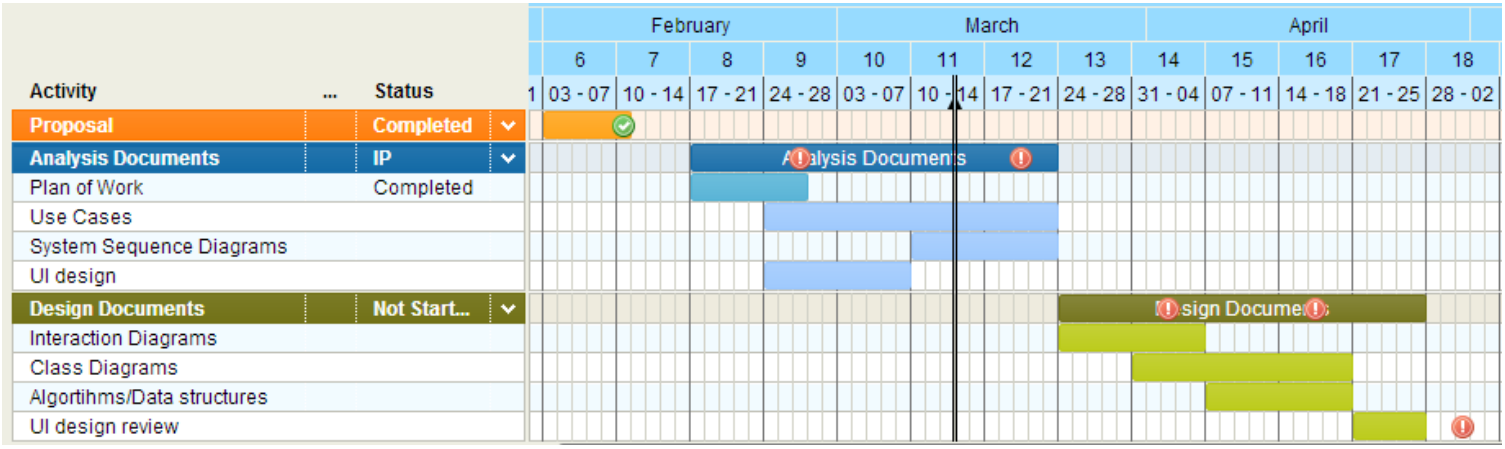
User Interface Design and Implementation



When the user opens the app, they will be shown a menu to either 'Open an Existing Document', 'Create a New Document', or 'Open a Previous Document'. Once they have selected their option they will eventually end up at the main screen. The main screen has a menu option to the top left that will include options to save the document, create a new document, or open an old document. Along the right side of the screen will be several options that the user can choose from while editing their document, like changing to the draw feature, importing a picture, or changing the color of the text or pen. Along the top of the screen will be the name of the document, where the user can easily change the name any time.



Progress Report and Plan of Work



- **Proposal (2/03 – 2/10)**
- **Analysis Documents (2/17 – 3/21)**
    a. Plan of work (2/17 – 2/26)
    b. Progress Report #1 (2/26)
    c. Use Cases (2/24 – 3/21)
    d. System Sequence Diagrams (3/10 – 3/21)
    e. UI design (2/24 – 3/07)
    f. Progress Report #2 (3/19)
- **Design Documents (3/24 – 4/25)**
    a. Interaction Diagrams (3/24 – 4/04)
    b. Progress Report #3 and Midterm Review (4/2)
    c. Class Diagrams (3/31 – 4/18)
    d. Algorithms/Data structures (4/07 – 4/18)
    e. Progress Report #4 (4/16)
    f. UI design review (4/21 – 4/25)
- **Progress Report #5 and Final Review (4/30)**

Resources

- Gantt Chart – www.tomsplanner.com/
- Android Pdf API - http://developer.android.com/reference/android/print/pdf/PrintedPdfDocument.html
- Android Camera API - http://developer.android.com/reference/android/hardware/Camera.html
- Text Editor - http://developer.android.com/guide/topics/text/index.html
- Data Storage - http://developer.android.com/guide/topics/data/index.html
- Definitions/Glossary of terms - Wikipedia