Android App Programming Directed Study ~ DrawingFun

Jenny Huang

December 19, 2014

Contents

1 first checkin 10/27/2014

1.1 Goal

- According to the instructor's requirements that we are going to implement an simple window's Paint like Android app for later on integrating Unicon's 2D graphics to Android app.

1.2 Course Introduction

- We have only two students, the other one is an udnergraduate exchange students with solid Java programming background and relatively slightly week problem-solving skills. For the first more than half semester, we used Sudoku as the starting point and tried several different topics to get our hands wet.

1.3 Project Introduction

- It's after middle term already, the way we were currently trying on to make it work may just work perfectly for the other classmate, but for me, I feel like it takes forever for me to be able to make any significant progress. So about half a month ago, I was motivated and thought instead of surfacing around and having fun learning by trial and error, maybe I should start from an simple GUI app as a starting point and try my best to expend/extend the APP functionality from there. And also we would be able to work to our final project slightly earlier.

- This GUI will be my very second GUI interface that I have ever created for my Computer Science major, (this first one was an Python Tkinter GUI one week short project for plotting graphics with data abstracted from backend database during an internship;). And I guess it may still be slightly difficult for me to start write Android App code of my own line by line, so I simply searched internet, and trying an tutorial to make a working starting point Android Paint GUI. I integrated the codes from the reference link all together, fixed minor compile errors, and it worked!

- This "Copied" GUI will serve as the starting point, and my functionality updates start from here, and I will update my progress for this project later on by week according to the instructor's requirements and suggestions.

1.4 References

- http://code.tutsplus.com/tutorials/android-sdk-create-a-drawing-app-interface-creation-mob
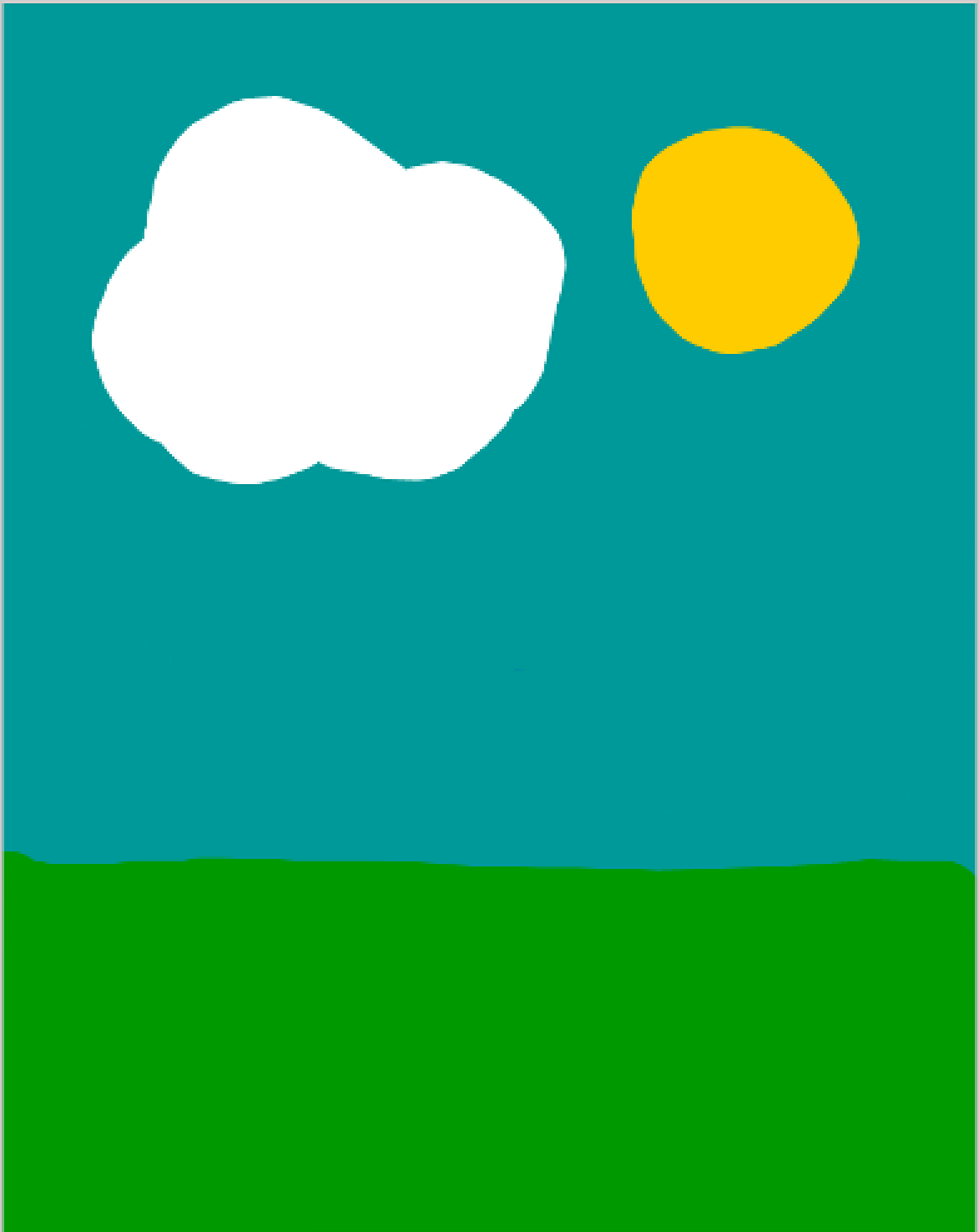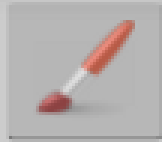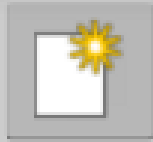
2 Checkin for 11/3/2014

2.1 Buttons I have worked on

2.1.1 Color$_{Picker}$:

2.1.2 Undo/Redo:

2.2 Functionalities and References

2.2.1 Color$_{Picker}$:

- Motivated by the Picasso Android app, seeing their multiple color choices, our starting point **12** fixed colors were too limited.

2.2.2 Undo/Redo Buttons:

- Also motivated by the Picasso app, intended to work on **Undo** button, and ended up found **Redo** button could be very convenient as well.

- needs to update these Undo/Redo methods later on, this is just the starting point most basic implementation for this button set.
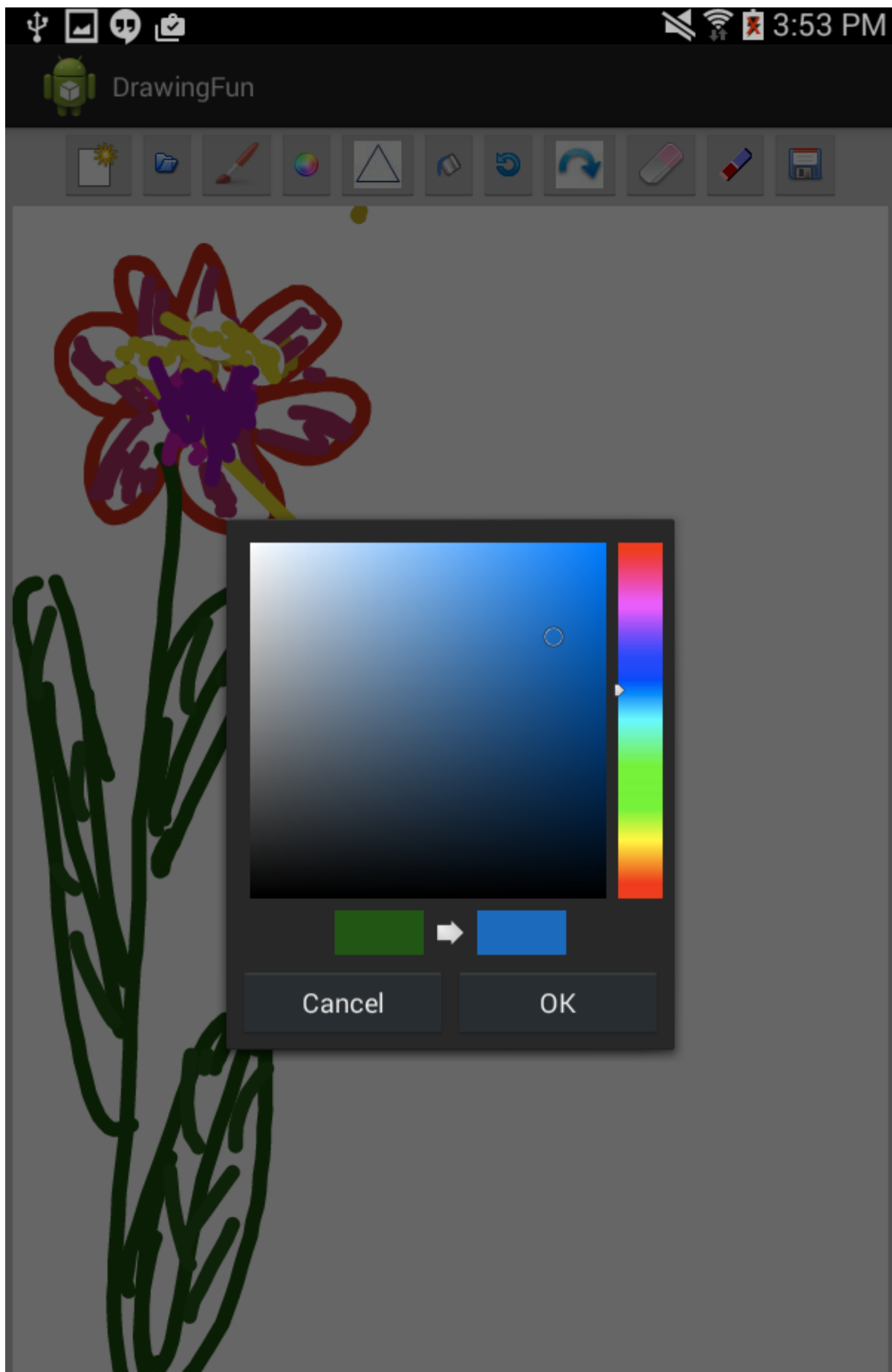
2.4 Todo

2.4.1 Drawing shapes with Finger for primitives

refer to the reference below:

- http://gmariotti.blogspot.com/2014/01/drawing-shapes-with-fingers.html

- This button will be first priority to finish

2.4.2 Load image file button

2.4.3 Erase Rectangle

2.4.4 Undo/Redo

3 Checkin for 11/10/2014

3.1 Buttons I have worked on

3.1.1 shapeBtn for primitives

3.2 Functionalities and References

3.2.1 shapeBtn for primitives: Drawing shapes with Finger for primitives

- refer to the reference below for some basic shapes: line, smooth line, circle, triangle, Rectangle, square

- http://gmariotti.blogspot.com/2014/01/drawing-shapes-with-fingers.html

- **ListView** in **Alert Dialog** is searched from online without direct reference.

- Since the erase was using draw smooth line. This button works also means that I could erase a "**Rectangle**" shape, or "**Circle**" shape.

- I have other course priority for the passed week, so I just have enough time to finish this course's priority, but I will try to work harder in order to finish all the functionalities for this course.

- It's not a good looking ListView, but yet it's a fully functional button.

- This button right now is fully functional, but to finish this project first, I have not spent any quality time to expand any primitives yet, rather than the existing six ones from the reference listed below.

DrawingFun

Rectangle

Square

Circle

Line

Smooth Line

Triangle

Cancel   OK
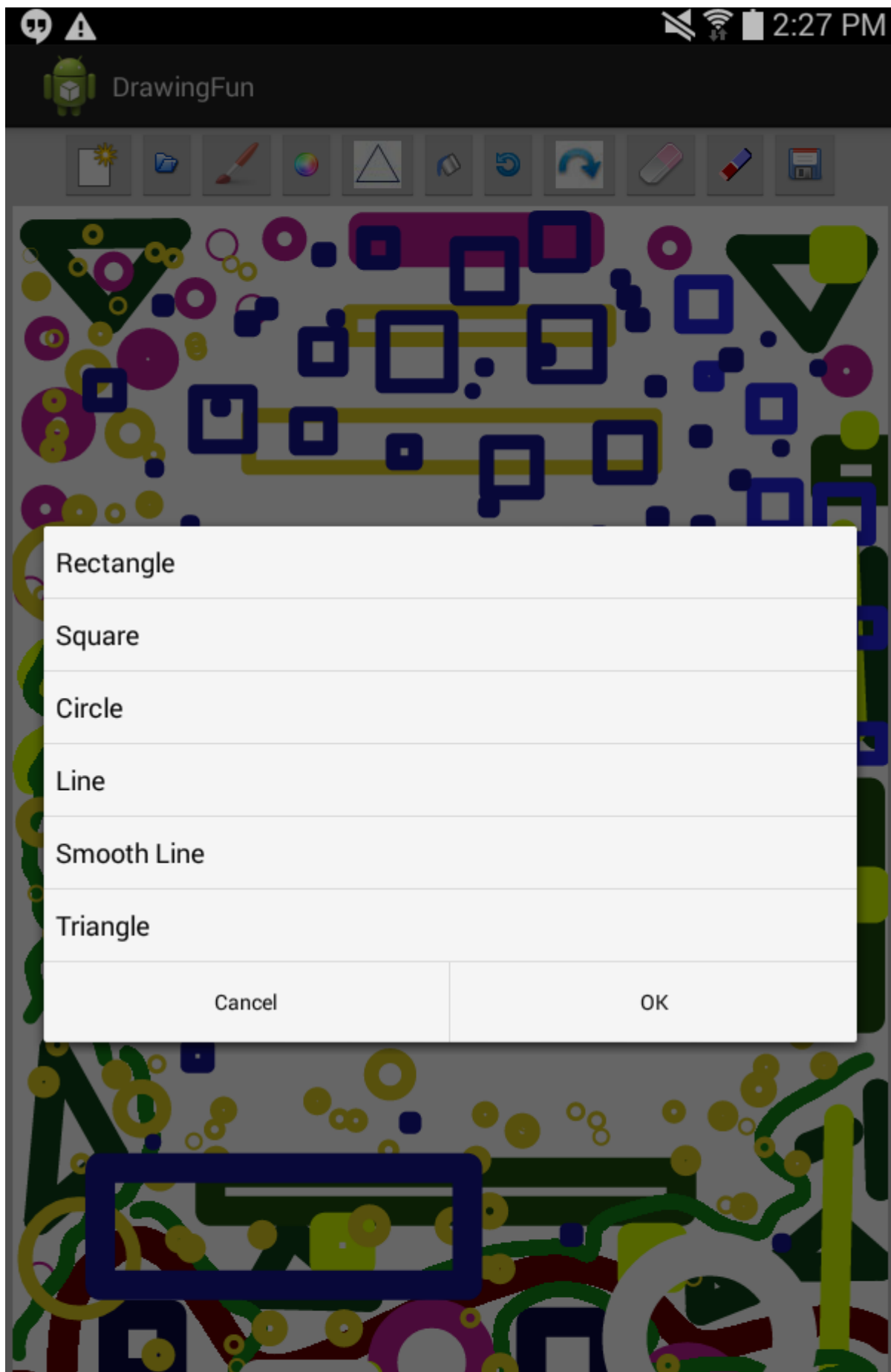
3.4 Special Situation

- There were too many students piled/lined up in front of Dr. Jeffery's door, and he didn't break the line by stating that it's our direct study time. So the other classmate and I just stepped away from his office, and we didn't really meet during last week.

- The other classmate and I have talked, and we happened to have worked on the same shapeBtn, I applied ListView in a dialog box with all six drawing shapes applied, and he created a (ListView? not sure) with a clickable button as one element with four shapes applied. And he agreeed my ListView looked way prettier than his buttons did.

- But I am willing to and more than happy to think that he could have worked on something else important for him that I actually didn't have time to work on during the passed week.

3.5 Todo

- Load image file button

- Erase Rectangle

- Undo/Redo

- Fill paint

4 Checkin for 11/17/2014

4.1 Buttons I have worked on

- openBtn for loading an image file as an ImageView

- Undo/Redo

4.2 Functionalities and References

4.2.1 openBtn for loading an image file as an ImageView

- The method I applied is memory saving for AsyncTask, which is better than load images directly, which could potentially block UI for couple of seconds;

- Loaded an image from online, but would like to try load internal images from device later on, like a drawing which I saved earlier onto my internal device;

- Potentially apply layer oncepts to produce multiple layer drawing, needs suggestions to organize my idea how to implement this feature.

- **Question**: Right now, my image is an ImageView in my layout, what ideas that I could use to set/change/-transfer my ImageView to be my draw view background?

- References:

  http://www.learn2crack.com/2014/06/android-load-image-from-internet.html

  http://stackoverflow.com/questions/5776851/load-image-from-url

  https://github.com/koush/UrlImageViewHelper

4.2.2 Undo/Redo

- After implemented subclass SuperActivity class which extends Activity on week checkin for 11/10/2014 for my ListView implementation, subclass of Path() was very difficult for me to think about implement before, but after my trial on ListView, super/sub class in Java all made sense to me now. It's a piece of cake, and I know I can wrap whatever material I need in order to paint nice and neat.

- Implemented by developing a subclass myPath to wrap the super Path(), drawPaint color, and drawPaint strokesize together as an object.

- Based on previous progress that I can undo/redo only with all the drawCanvas with the same paint color, now my updo/redo paths could be colorful and with various strokesizes.

- References:

  Path() library:

  http://grepcode.com/file/repository.grepcode.com/java/ext/com.google.android/android/2.3.1_r1/android/graphics/Path.java

  Bitmap cacheing:

  http://stackoverflow.com/questions/3406910/efficient-2d-drawing-in-android/3408641#3408641

- **Questions**:

  1. Undo/Redo for simple path seem to behavor fairly ok, but instead of lineTo wired line, how do I implement smooth line? How could I differentiate different strokesizes more clear with lines I have so far?

  2. One little detail though, I dras after touch up, my paint color change delayed, how do I implement **real time**?

     Answered: drawPath.reset() produced all the trouble.

  3. About previous ListView six different shapes, with Undo/Redo properly functionaing, I realize I just lost my siz shapes again cause I need to rewrite/implement methods in order for them to be able to Undo/Redo ~? (My subclass works perfectly for this propose, just that I lost my internal link to primitives, which means I probably should rewrite my primitives draw methods according to undo/redo prerequirements. I don't think it will be difficult, but I don't have enough time for this for the pass week, and I need to organize my ideas about these implementation clear.)

  4. I prioritize undo/redo to be more important than any other buttons cause I know they would give me great practise together with primitives implementation methods rewrite. So I have not touch "Erase Rectangle" button and "paint fill" button yet. According to these idea, I would prioritize Rectangle rewrite with the highest priority, so that later on I can follow up with erase Rectangle (which means draw Rectangle first, fill with background color, and undo could remove this erase step). Correct me if I am wrong.

4.2.3 References: all about Android

- https://github.com/kesenhoo/android-training-course-in-chinese
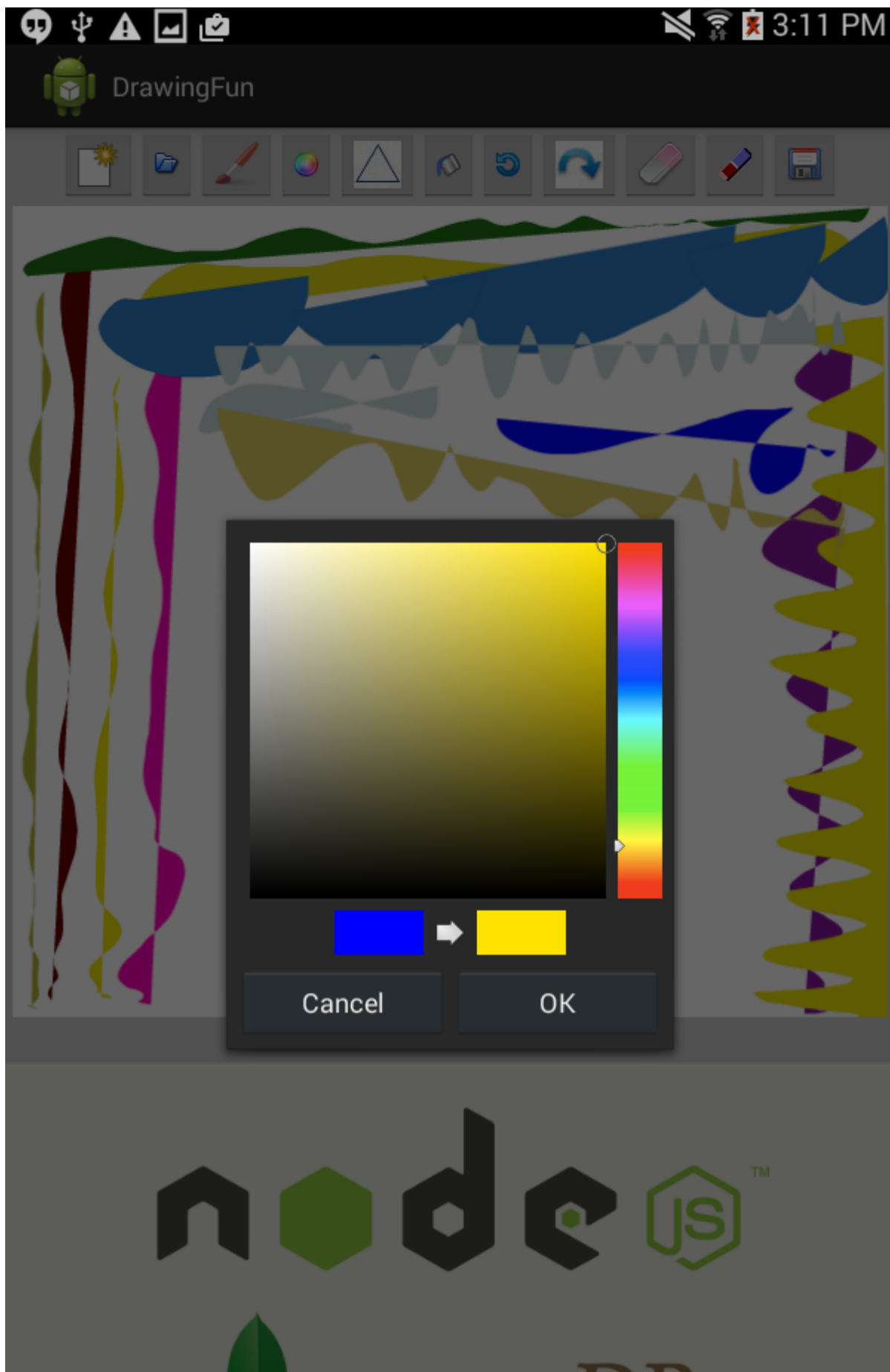
4.4 Todo

Only two button left untouched, could do the following or anything I am interested to implement.

- Erase Rectangle

- Fill paint

May try to **save** into Galaxy⋯ as Dr. Jeffery mentioned it last time when we meet during class;
Potential interests: may implement depends on how I spend thanksgiving ~

- touch ImageView Activities: zoomin, zoomout, rotate, fading, etc

- SurfaceView rotate images through new thread

- canvas save() and restore()

- OpenGL spinning circle

- widely used draw methods

- Easy draw operations

5 Checkin for 12/01/2014

5.1 Buttons I have worked on

- ImageView to Bitmap

- start newBtn

- Undo/Redo

5.2 Functionalities and References

5.2.1 ImageView to Bitmap

- Worked on Bitmap so that I can load a picture as my drawView background;

- This could be considered to be a trial, and could try to add user options to different background pictures later on;

5.2.2 start newBtn

- Realized that my newBtn lost its functionality during last checkin because of different mechanisms, and I fixed it after having implemented undo/redo for paths;

- The wired drawing path shapes (like the dramatic curves in previous Snapshot) got corrected as well by writing to Bitmap;

- References: mutuable immutable bitmaps

  http://stackoverflow.com/questions/13119582/android-immutable-bitmap-crash-error

- But I still failed to start new because some minor error about implementation. I uninstalled the app and restart, the error was still there;

- I was so focused on the mview thing that I completely lost focus on the true reason. Once I asked and the instuctor helped explain that invalidate() simply calls onDraw() function, I could immediately realize that I forgot to clean my undo/redo arraylist paths and undonePaths!

- It was the invalidate() function confused and prevented me from relaxing on the mview, and I was stubborn there for about one hour this afternoon. Realizing that I felt so sorry for myself for the one hour being so stupid! And right now I am on my way following the good habit reading Qt creator documents systematically before googling the correct answer only when I try to solve my technical difficulties, which is good.

- While it still worths a minutes to rewind and rethink about what happened during that one hour, how I trusted myself so much and suspected on low probability corner cases situations, rather then double check and confirm that all the steps/processes I had made were correct and reliable. I wished I spend the hour with a scientific attitude the latter.

5.2.3 Undo/Redo

- If I really don't want to separate/pack my ListView items into objects, will it be possible for me to use command pattern instead, and how difficult could command pattern to be comparatively spearking?

- References: List:

    - <http://stackoverflow.com/questions/11114625/android-canvas-redo-and-undo-operation>

    Command Pattern:

    - <http://www.28im.com/android/a141932.html>
    - <http://www.javaworld.com/article/2077569/core-java/java-tip-68--learn-how-to-implement> html
    - <http://www.28im.com/android/a141932.html>
    - <http://blog.csdn.net/lovingprince/article/details/1532869>
    - <http://www.2cto.com/kf/201409/333267.html>
    - <http://www.2cto.com/kf/201406/309574.html>
    - <http://blog.csdn.net/rhljiayou/article/details/7212620>

- Answers:

    - We didn't really talk about command patterns at all this afternoon, but rether to solve both the other classmate's and my technical difficulties, and also discussions about the questions we raised, for example, my interested ones including multiple layers Potentials when using Bitmap and removing any layers afterwards, and autosave nsapshots if we save bitmap every 20 minutes, and Potential values we could apply with those save displays in paths & undonePaths during each 20 minoutes interval.

    - I began to realize that I COULD have my own little brain-turning/intuitive ideas when I began to understand things, like I spent hours today just to understand Bitmap~
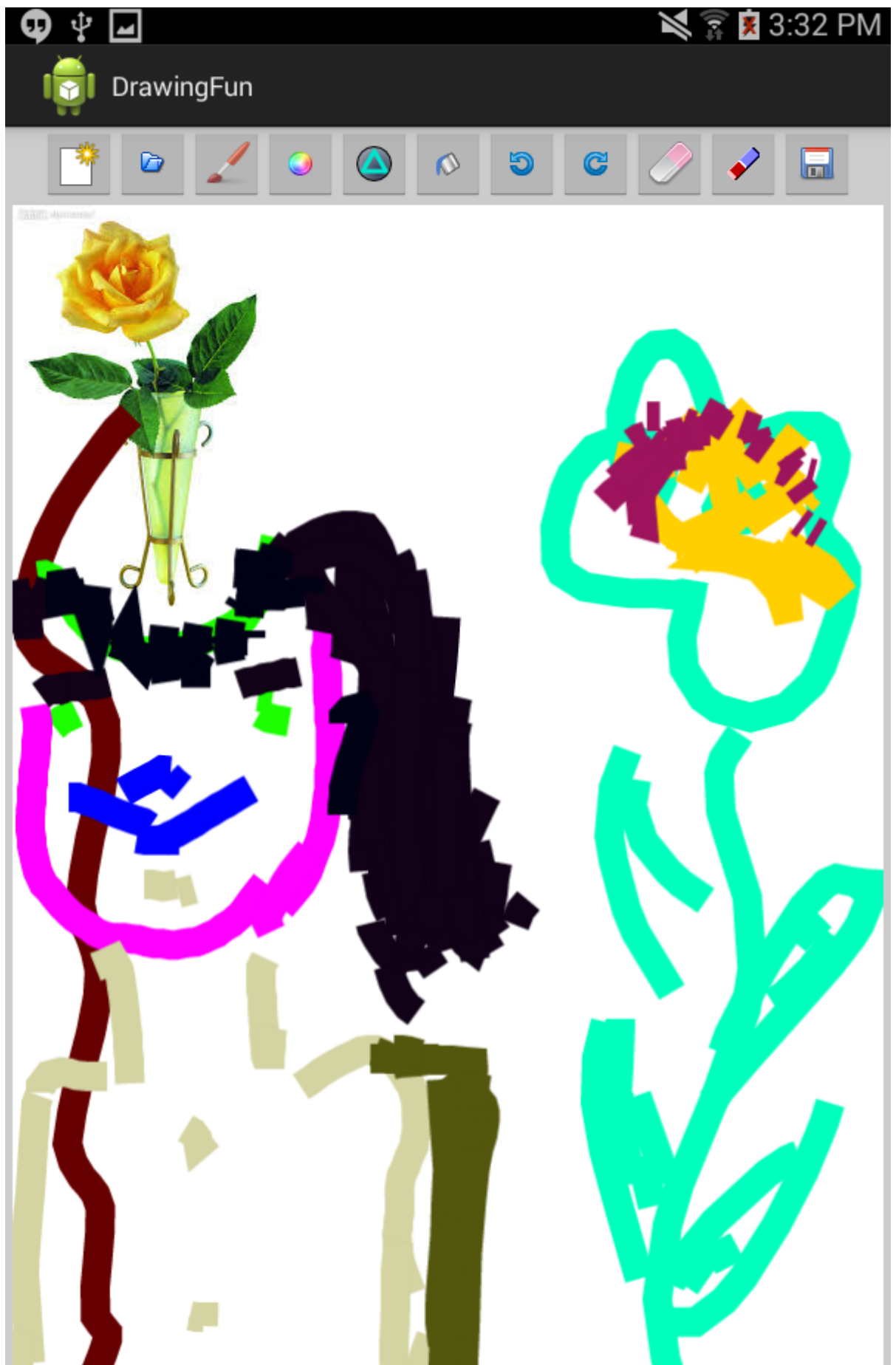
5.4 Todo

- finish the undone functions and wrap up project and do basic demo on coming Monday;

- short about one page summary, could at most to be 2 pages;

6 Checkin for 12/08/2014

6.1 Buttons I have worked on

- Undo/Redo

- Erase Button

- FloodFill

6.2 Functionalities and References

6.2.1 Undo/Redo

- **Cleaned** my contamination or origial bitmap in DrawView **clear()** method by replacing "new2Bitmap = originalBitmap; " with "new2Bitmap = bridgeBitmap.copy(Bitmap.Config.ARGB$_{8888}$, true);"

- I used bitmap only for the propose of adding the Yellow Rose which I liked it too much and wanted to keep it as a corner backgroung; But for the rest of drawings, they are all drawn on canvas instead of bitmap;

- I could draw all the contents in bitmap, but my **Straight Line** looked really wired on bitmap when I draw in progress, but otherwise I don't have any clear idea how to remember the start and end points and draw a straight line during onDraw. I need some idea here, if I continued to use bitmap instead of canvas;

- Fixed minor issue for smooth line undo when I tried the function onSizeChange, which was originally working before;

6.2.2 Erase button

- Becuase I liked the Yellow Rose too much, I had to conpensate and rewrite the erase function to draw shapes using background color, because the old method doesn't work any more when I used bitmap; The function itself was not difficult at all though.

6.2.3 FloodFill

- Implemented on Bitmap instead of canvas

- I used bitmap only for the propose of adding the background image and do the FloodFill on the background image. But for the rest of drawings, they are all drawn on canvas instead of bitmap;

- Applied the following method, but it was way too slow, and look urgyly

- http://stackoverflow.com/questions/12669740/android-using-flood-fill-algorithm-getting-out

- http://stackoverflow.com/questions/8070401/android-flood-fill-algorithm

- http://www.codeproject.com/Articles/364413/Queue-Linear-Flood-Fill-A-Fast-Flood-Fill-Algor

- http://stackoverflow.com/questions/8723590/fill-the-complete-canvas-but-keep-the-bound-fil 12777805#12777805

- http://blog.csdn.net/jia20003/article/details/8908464

6.2.4 Other Issues

- setBrushSize

  - Issue: the setBrushSize option always changed the drawView color back to initial default color, which is Color.BLUE, and which is not convenient;

  - In **MainActivity**, when $draw_{btn}$ I setBrushSize, I have to do **drawView.setColor(mColor);**, otherwise, it always set drawView's paintColor to be default Color.BLUE;

  - But I don't think the above implementation is logical. I don't think click on $draw_{btn}$ need to do anything about color when I suppose to set the brushSize; But I have difficulties to understand the process and find a better "logical" solution for it.

- onSizeChange

  - I think I have bug on **onSizeChange** function, because whenever I changed my device from vertically to horizontally, all the contents on my canvas just went away.

  - I can think the reason is because I didn't really draw anything on the bitmap yet, and that's the reason whenever I changed from vertical to horizontal, I have only my fresh loaded background bitmap;

  - But, if I want to draw on bitmap, I will have to reimplement my undo/redo differently, the current ArrayList method won't work any more.

  - If I reimplement on bitmap, what will be the good idea to implement it?

  - I would be happy if the instructor helps Introduce a little bit more 0about bitmap, undo/redo on it, and its utilities.

6.2.5 report

- Course report is in **report.org** file, and main sections are also copied into the followed section for the reader's convenience.

- The demo was reviewed today and the course is done. I did pretty good job for this DrawingFun project and the course, but I need to prioritize my tasks, and I won't be able to update this repository for a while. Thanks for looking.
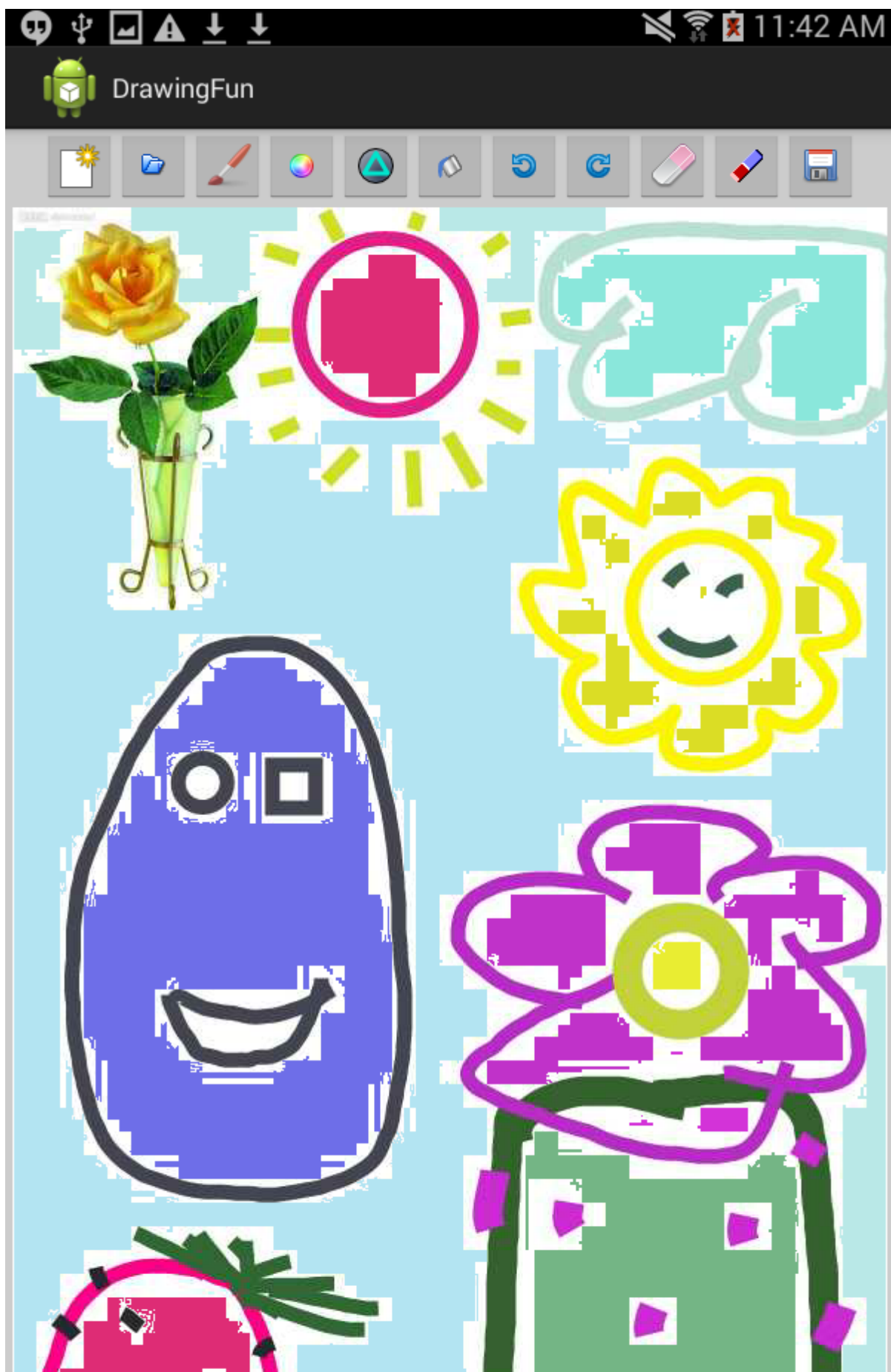
7 Course Review

7.1 Course Goal and General Review

- Taking this course, I wanted to help myself stay on schedule and learn some cutting-edge knowledge as a starting point.

- I never had any "new" knowledge like "Android" learned before, this is the first time, and I enjoy it;

- I enjoyed two modules the most: the color popup dialog and undo/redo functionalities. And in the middle, ListView helped a little bit as well;

  - The color picker was not my original work, but for me at that time, it was very complicated and it forced me to understand all the Android framework for an App to function, the manifest, layout, value etc;

  - To implement a fully functional ListView together with the rest functionalities, I figured out my own way of creating a bridge SuperActivity class, which in term of Java-programming, created a start point of confidence that I can implement my ideas (any idears) in Java as far as I **Think** it through, clear. **It is always the ideas that matter, instead of any implementation.**

  - For undo/redo interface/implementation, I had thought to skip around by implementing Command Patterns, but now I am glad that Dr. Jeffery insisted us to apply interface/implementation. And I had been frustrated yet more than happy take my own effort to try, step by step, implement and see eventually it is working~! And during this process, I felt I began to be exposed to Java OOD, Android canvas, bitmap, drawing primitives, and I understood the theory behind them now, even only the parts that I implemented.

7.2 Course Benefits

- The latter half semester of implementing DrawingFun Paint project helped me realize that I can perform great in concentrated topics, which helps me focus.

- It has been a challanging and interesting learning experience during this Android App Programming, and it successfully reached the target which I expected from this course. I learned the basic necessary knowledge to build my Android App and Java Programming background, and I practised and cultivated the necessary and usefull skills to think logically, solve problems and debug my codes.

- The course built knowledge, practised skills, as well as built confidence in programming and problem-solving, and help cultivated my **I CAN DO** attitude towards projects.

- After taking this course, I have a sufficient starting point to self-learn and practise Android App Programming. And now I am ready to prefer Java over c++ as my primary and first choice programming language, and I will try to conduct more practise on Java programming so I can be proficient on it in not far future.

7.3 Report Feedback from Instructor - 12/12/2014

- I got the report feedback from the course instructor today that he doesn't require any coding work any more as was concluded on Monday's demo and review already, but he would still expect a slightly better version of report from me with the marked requirements;

- A scann copy of my origial report and the course instructor's marks are attached below, and I will make necessary modifications for it so it satisfies his requirements.

Android App Programming Directed Study ~ DrawingFun

Jenny Huang

December 8, 2014

Contents

rather than a table of contents, this report deserves a sentence or two of introduction that defines the goal. You could pull section 5 up here into the introduction

1 Buttons Implemented

rename   *→Functionality and User Interface*

This can just be stated on a line or two in a single sentence, allowing room for a screenshot

- New
- Open
- Draw
- Color
- Shape
- Fill
- Undo
- Redo
- Erase
- EraseRect
- Save

2 Data Structures Used

- ArrayList
- LinkedList

combine 2-4 into an Implementation section
write short sentences, not just lists of words

how much code wrote by you
" " " from other sources

3 Java OOD

- Inheritance
  - SuperActivity, MainActivity
- Interface & Implementation

bigger margin, don't cut off please

  - for myPath, myLine, myRect, mySqar, myCirc, myTria classes which implement DrawElement interface:

4 Android Libraries Used

- ListView for shapes
- Dialog boxes for all the popups
- Fragment for color selection
- Thread for FloodFill implementation
- OnTouchListener, onClickListener for implementing and conducting all the choices

5 Course Goal and General Review

- Taking this course, I wanted to help myself stay on schedule and learn some cutting-edge knowledge as a starting point.
- I never had any "new" knowledge like "Android" learned before, this is the first time, and I enjoy it;
- I enjoyed two modules the most: the color popup dialog and undo/redo functionalities. And in the middle, ListView helped a little bit as well; *cite everything that you used (websites, books)*
  - The color picker was not my original work, but for me at that time, it was very complicated and it forced me to understand all the Android framework for an App to function, the manifest, layout, value etc;
  - To implement a fully functional ListView together with the rest *of the* functionalities, I figured out my own way of creating a bridge SuperActivity class, which in terms of Java-programming, created a start*ing* point of confidence that I can implement my ideas (any idea*s*) in Java as far as I **Think** it through, clear. It is always the ideas that matter, instead of any implementation.
  - For undo/redo interface/implementation, I had thought to skip around by implementing Command Patterns, but now I am glad that Dr. Jeffery insisted us to apply interface/implementation. And I had been frustrated yet more than happy take my own effort to try, step by step, implement and see eventually it is working~! And during this process, I felt I began to be exposed to Java OOD, Android canvas, bitmap, drawing primitives, and I understood the theory behind them now, even only the parts that I implemented.

6 Course Benefits

- The latter half semester of implementing DrawFun Paint project helped me realize that I can perform great in concentrated topics, which helps me focus.
- It has been a challenging and interesting learning experience during this Android App Programming, and it successfully reached the target which I expected from this course. I learned the basic necessary knowledge to build my Android App and Java Programming background, and I practised and cultivated the necessary and usefull skills to think logically, solve problems and debug my codes.
- The course built knowledge, practised skills, as well as built confidence in programming and problem-solving, and help cultivated my **I CAN DO** attitude towards projects.
- After taking this course, I have a sufficient starting point to self-learn and practise Android App Programming. And now I am ready to prefer Java over c++ as my primary and first choice programming language, and I will try to conduct more practise on Java programming so I can be proficient on it in not far future.

practice

- The final report deadline for me is Friday, coming week. Of course I would be able to finish on time. I will update this repository when I finish my updated report as well.

7.4 practice vs practise

- When I knocked the door of my instructor's office yesterday (12/12/2014, **now I am modifying on 12/13/2014**) at noon, the instructor said he was right on working on my report, and if I could wait several minutes, he would be able to finish. So I waited in his office, and he searched internet on propose that the **practise** in my last section should be written as **practice**, and that was the reason he wrote **practice** on my second page.

- And after he searched the word using google, he emphasized by talking to me that in bratish english, people may spell **practice** as **practise**, but in America, they use to write as **practice**. He wrote the word down on my report without circling the original wrong-spelled word.

8 Personal Conclusion

- I had minor difficulties setting up my Android environment at the beginning of the semester, like my window's SDK manager never worked; And at the beginning of the semester my Linux Mint 17 Eclipse kept crashing ⋯ It was from time to time, I searched and googled, and get my Linux version stable; And I did have some help from the other classmate as well;

- The first half semester kind of, the course contents were slightly distributed, and I felt I didn't really know what to focus, and I don't like that half of semester;

- The rest about half semester I worked on this DrawingFun project, and I am confident that I did pretty good job, comparatively spearking, compared with the other classmate.

  - I applied Color-Picker functionality, while he applied mine;
  - I applied ListView for drawing primitives, while he applied the same original setting-brush-size methods - a popup dialog with button choices included;
  - I initiated to include the undo/redo button in our app ideas orginally came from Picasso app motivation; We independently implemented undo/redo functionalities while mine is fully functioning and his some primitives cannot conduct undo/redo yet;
  - I found and debugged my setBrushSize() function to remember last applied color, the other classmate didn't seem to be able to notice this, or he hadn't have time to look into it yet;
  - I spent some time on the onSizeChange() function tried to make my program work then I hold it horizontally, while the other student directly set his App to be applied vertically FIXED so that he didn't have any onSizeChange() issues at all;
  - I used bitmap and reimplemented my erase function, while he kept my default first version method. And his undo/redo/erase design supports only erasing smoothline (one of my six primitives, he had four or five), while mine supports earsing all my six primitive styles, and I liked this implementation;
  - I googled and applied FloodFill function in my App on bitmap level, and I tried two implementations, one ASyncTask idea (which was slow and sometimes my App main UI froze), and one Thread implementation (and the UI never freeze when I floodfills my bitmap); while the other classmate floodfilled the whole bitmap background with one color while loosing all other App functionalities, and I guess he didn't really understand the difference between canvas and bitmap because otherwise he should have loaded a background bitmap which is adapted for floodfilling somewhere;

- I completely understand that the other classmate could have his other priorities, and he is undergrad while I am the graduate level, just like cs480 Senior Design the course instructor didn't require me to do any further design, but it is my first time to be able to handle such big and interesting project, and I want to dive into it and get myself well-practised, and I will insist this idea by implementing at least one of project great (senior design project, or midi Controller project, most probably the latter) so that I learn and understand.

- As the course instructor has claimed at the beginning of semester too, he was also trying to learn Android for his Unicon graphics implementation later on, we were NOT a great nor efficient team yet, but it forms a great learning experience, at least for me. And later on, I should set higher standards on myself than now, and I will **practice** more for my own good.

- This became another experience for me that **Self-adaption, self-motivation and passion are very important for projects and career success.**

9 MS Computer Science and Potential Career Opportunities

9.1 updated 12/13/2014, Saturday

- During the spring 2012 when I was seeking sugestions for a MS Computer Science from close relatives, they as the most close relatives here in US, didn't offer any enough reasonable sugggestions but rather leave me to make the decision.

- I had a Statics MS background and have used up my OPT, which means I **won't have OPT** for Computer Science if I get the degree at the Master level;

- I hoped an opportunity and chance to make my own effort and survive here in US.

- I have only one cs120 with **B** as the grade as all the my applying background, but I got admitted by the department;

- I was allowed to register cs121 and cs150 7 credits in total for my first semester, I talked to advisor and another professor in the department to target a 2 year MS plan; And I would leave for China at that time if I can only register 7 credits, and won't be able to finish in 2 years; It ended up by allowing me to regitster more credits, and targeted for 2 years;

- China's Elite, **Soho's** Chief Executive **Xin Zhang** and her husband, **Shiyi Pan** launched the initiative Tuesday by signing a **$15 million** gift agreement with **Harvard University**.

- I have been working hard on my major, and now would still work hard and look forward to see if these 3 years study here in US would just end up with a degree, leaving no working opportunities for me at all ~ !!

- So far the university seems to fully support that students look for new opportunities, a new post was updated yesterday (12/12/2014) at around 4:20pm right after I left CSAC from campus. A snapshot of LinkedIn homepage is attched, and it was snapshot today, and the post was created yesterday. I was just noticed back to LinkedIn during past about one week, otherwise I would have sync added persons together with one special person for me.
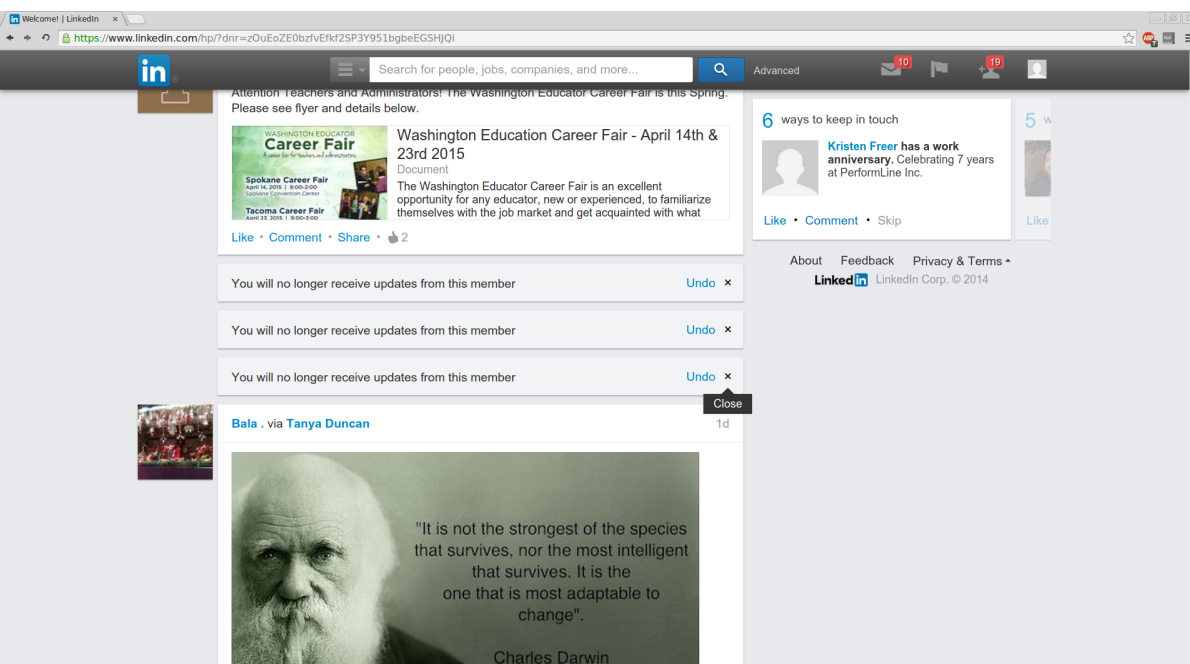
9.2 updated 12/16/2014, Tuesday

- Fact: As a student spent eight years here in US, I want to say that I am pround of my country, and I am pround of the culture my country has. And now as a grown up and educated person, I am pround that I was born in China and I feel pround on the good manner/behavior the elite individuals have done just like **Xin Zhang** and **Shiyi Pan**;

- Fact: On Monday afternoon the university in LinkedIn updated another posts seems to clarify that it had nothing to do with me on last Friday's post, though this has been the closest several posts ever during recent university LinkedIn posts;

- At this point, it is good news for me, and for the university as well. And I feel **sorry** that I did worry that the university will block me from seeking for working opportunities.

- This is a person who had special childhood sufferings. I had slightly damaged hearing and I have difficulty differentiating subtle sounds, and I have difficulty pronouce them correctly as well. And I have my physical medical problems needs to pay close attention to. The current defensive and slightly sensitive characters are the results of those sufferings as well.

- Everything happens with a reason. While I felt sorry, at the same time, I expect the reader, the audience, or the US culture could also help understand that as a slightly sensitive person as I was grown up to be now, I have every reason to suspect and be panic at that time as well. The "practised" were not circled, and I have sufffered some here already, and that was the first time I noticed and felt I was focused by the university on LinkedIn cuase I had been too busy on my projects and were not able to notice back into LinkedIn for quite a while;

- I agree with the famous saying that "A team is a group of people who trust each other. " As a sensitive person, especially when I had not-so-good feeling on some teammates already, for example, as teammate being especially cold and impolite and never replies email, or a instuctor exposes students' performance publicly, it's difficult for me to trust them any more. And I guess that was the main reason I got panic at that moment. People trust a person when they like, or have love on them, just as I love a person for years. I wonder why "somebody" would try so hard, by all means, to separate a beloved couple? I need think deep into this.

- I intended to snapshot the post the university posted in LinkedIn on Monday, but since it has the university name on the picture, I won't do that.

- But even I don't have project updates for this repository for today, I figured that I should still get this part updated as soon as possible to reduce the hurt.

9.3   updated 12/19/2014, Friday

- The final report was in **report.pdf**. Both the hard copy and electronic copy was turned in yesterday (12/18/2014) to the course instructor; I regenerated the .tex and .pdf files today for Github updates because I miss-deleted them today;

- I didn't want to include in the **Personal Conclusion** section in my report, but when I got feedback from course instructor on 12/12/2014 and when he used the word "deserve", I felt the instructor was slightly picky on me compared with the other classmate even when I performed better than the other. I felt pushed to include that part in, same happened to later section of **practise vs practice**. I just expected that education helps build our confidence as students rather than the hardly built confidence got destroyed by any reason.

- The town is conservative, yet is highly influenced by in-town university; A 19 years old undergraduate girl was gunshot by one university professor who suicide afterwards in summer 2011;

- My life were almost contaminated by some professor in university but I prevented it from happening from my side because I have one already the fall semester about one year ago;

- The special person is 13 years elder than me, and we are four-generation far relatives. His grandpa from my uncle's (his dad's) side and my grandpa from my mom's side were siblings sharing same parent. He looks very young though. I would love to get married when environment gets ready.

- I like my Computer Science major the most. I prefer San Francisco Bay Area job opportunities but I am willing to relocate anywhere in US as far as the company will sponsor **H1B** work visa for me **before 4/1/2015** because I won't have any OPT.

- With semester ended today 5:00pm, finally I am entering into self-learning mode this evening.