**Taylor Xia**

**Software Engineering Challenge**

**Requirements**

The goal of this engineering challenge is to create a graphic animated interface using Java. Because this engineering challenge is related to graphics, the Applet class as well as the Applet interface. The project will probably take about 1 week, with extra time after school if needed.

**Research**

The project will, of course, require skill in Java. It may also require some skills in digital art in the planning process. The most common example of this kind of project online would be from 8 bit pictures created online by artists, though this one will be more like a gif. Even though these pictures are in an 8 bit style, the color usage and dynamic of the picture is still amazing. I could use similar methods to create a very good picture. The only problem I see is that the picture I have in mind might be too difficult to make and take more time than I have. The book also contains a lot of examples of graphics and animating graphics as well. I think I do have the resources to make this program, though time is questionable. Any knowledge I need to complete the program I should know.

**Design**

See paper

**Construction – included in summary doc**

Document your construction progress with date and progress made (build a chart/table in word or excel), also, include at least three screen shots of your applet (beginning, middle, and end)

**Testing/Results/Delivery—summary doc not required to start project**

Preliminary testing

Does your program meet the needs of the initial challenge or problem? How do you know?

Are any changes or modifications needed, why or why not? Document any changes.

Summarize the project - what worked, what didn’t work, note your successes and/or failures.

Looking back, would you have done anything differently, why or why not

Software Summary Document

Daily Progress Report

|  |  |
| --- | --- |
| Date | Summary of Work Done |
| 2/26 | Started project today. Just set up the basics like the applet class, the background, and one object (the moon in the picture). |
| 2/27 | Faced my first problem when I tried drawing the closest building. I wanted the city picture to be in 3D, and so the rectangle of the window has to be to tilted. I’ve been looking into some solutions. |
| 2/28 | One of the solutions I think I could do is use a class called AffineTransform from the Graphics2D library. It’s a bit complicated, though, and I the exact way of using it aren’t exactly clear online. I tried one method from Stack Overflow, but it didn’t work and instead all the other objects just disappeared. I guess I’ll have to find a different way to do it. Because of this, work has been pretty slow. |
| 2/29 | So I’ve found a much better way to do this. There actually is a method in Graphics called fillPolygon() and it takes two arrays of X and Y values and makes a polygon out of it. I used this to finish the closer building, and also finished building\_1 (see drawing). I just got a couple of things left to draw and then I’ll move on to animating. |
| 3/1 | Added building\_2, the ground, and started on just animating the TV screen. I need to add furniture and maybe put something in the background. |
| 3/5 | Finished furnishing the inside of the close building and the background city. I think to finish it all, I’ll need to animate the moon slowly moving across the sky and it’ll tie it all together. |
| 3/6 | Alright, I think it’s done. I’ll turn this in as soon as possible. |

Summary

The program fits the challenge and worked very well. The final product came out different from what I had originally planned, though. I wanted to animate a car driving across the road, but that would’ve taken way too much time. I would’ve needed first to draw about 3 different polygons and 1 rectangle for the car itself. I’d then need to move each one of the shapes across the screen and at a relatively high fps or it’d look weird. I also originally wanted to replace building\_2 with a restaurant, but then decided another apartment building would maybe look better. I would’ve liked to done this and more, but it really would’ve taken too much time. I think everything worked out pretty well, nonetheless. Animations are good, though a bit choppy, the effect of having the picture be in 3d worked really well, and overall I achieved the picture and effect I was looking for. Looking back, I might’ve went ahead and leaned about the .fillPolygon() method so I wouldn’t have needed to look for it for 2 days.