COMS 4115-Project Proposal
Fall 2014
Language Name: Photoshop--

Group Members:
1)Alana Ramjit
UNI: amr2235
Role: Language Design/Systems

2) Gil Feig
UNI: grf2108
Role: Manager

3) David Figueroa
UNI: df2442
Role: Test Suites/Testing

Description:
The language that we plan to implement is an educational language with the intention of representing coding concepts visually. Because it is targeted to both a younger audience in terms of age and an inexperienced audience in terms of coding knowledge, our language will purposefully be stripped down to essential coding concepts. However, we intend on implementing a dynamic, graphic component to our language. Users will run their code and be greeted with a visual representation of what their code produced; for example, if they ask to draw a small green circle with a width of 30 pixels in a for loop crossing the middle of the screen, then that is what will be produced. This will allow users to see what it means to have a for loop, to change attributes of a type or object, and even conditionals.

Types of Programs: This language is intended to draw small artistic scenes that can be static or animated. It is similar to and inspired by other teaching languages such as Scratch by MIT and Processsing.js in conception. While it can be used to just draw a simple, unmoving scene, the fact that the main loop will run as a drawing loop would allow people to create animated scenes or even interactive video games.

Language features

-Users will be able to create primitive types such as lines, circles, squares and give them attributes such as color and position that can be accessed by dot notation

-they will also have standard primitive types for ints and booleans and possibly some other standard features

-they will have keywords such as “move” that will allow them to perform operations on the x-y coordinate grid to visually manipulate the primitive types

-the main block is the execution entry point and needs to contain a loop block that will continually redraw whatever is inside of it which will enable animation

-can have functions that are denoted by block

-also would like to implement small event-handling UI capabilities such as tracking and finding the cursor location and using those as keywords in the language

-a goal is to have naturalistic, English sounding language so that users can discuss their programs in our language but still have it sound familiar. For example, since functions are called “block” and are executed by the keyword “run”, then users can say “run the testHit block.”

Sample Source Code

```
~this is a comment~
~this is a program that will draw two squares on the left and right side of the drawing canvas and
```

move them across the screen until they collide at which point they change colors and move away from each other off the screen into infinity~

```
rect s1 10 10 green;
rect s2 10 10 blue;

bool isHit = false;

block testHit {
    if s1.right >= s2.left {
        isHit = true;
    }
    else {
        isHit = false;
    }
}

main {
    put s1 at  0 50;
    put s2 at 100 50;
    loop {
        run testHit;
        if isHit{
            move s1 right 5;
            move s2 left 5;
            s1.color = blue;
            s2.color = green;
        }
        else {
            move s1 left 5;
            move s2 right 5;
            s1.color = green;
            s2.color = blue;
        }
    }
}
```