Richard Huang

Project Proposals: To make a visual programming language to help beginner get used to the idea of programming without worrying about code and syntax. Users are able to drag and drop different expressions, statements, and blocks of code, such as print statements, setting variables, if statements, for loops, etc. Users are able to use a combination of all these tools and nest them as they wish. This will allow them to create a simple function/program just by dragging and dropping different blocks. The goal is to make the programming experience more intuitive and easier for beginners to understand. Also this visual programming language can then be translated into python code, so users can see what their visual code would look like if it were written in Python. They can create new tabs or functions and combine all the functions together and translate all to Python. This code can then be ran in the program to see the results of their code. Users will also be able to save and open functions in this program as well. Error checking while writing and running your code is also available.

Features:

* Tutorial for users to understand how it works.
* Drag and drop features of different command boxes in the dashboard.
* Able to infinitely nest the command boxes and also pop out and of them. The boxes also resize and snap into place.
* Able to change function name and parameters as a comma delimited string.
* Able to convert the current function into python code and also able to convert all open functions into python code as well.
* Tabs are implemented as well to open and create multiple functions.
* Able to save and open each function later one with each visual box at the same location.
* Able to undo and redo for each function.
* Does error checking on whether or not the code is written correctly.
* Able to run the visual program with your own parameters.

User Studies: Points that people have brought up to add.

* Actually run the program the users created
* Add a return statement
* Add a place where users can implement recursion and make function calls

Fixes

* Create a "Run Python Code" button and create a new window where users can run their functions with their parameters. It will display the print result and return result.
* Add a return statement box on my dashboard.
* Add the purple expression box, where you can write any line of python code (such as function calls).

Hard Problems to Solve:

* Creating the user interface and making animations in tutorial.
* Implementing the drag and drop feature.
* Implementing a good infrastructure and design for the project, so it can be more scalable and more features can be added on later.
* Implementing converting the function to python code using recursion.
* Using recurison to drag and drop different visual command boxes to and from other nested boxes.
* Creating the run python window and do error checking on code
* When running program, recursive functions didn't work until a global dictionary was defined to exec.
* Adding windows/frames/widgets in the actual frame and make sure the x, y coordinates are right.
* Getting the files to save and open later on.

Competitive Analysis:

* Alice
  + 3D world
  + Create Visual Programs for a 3D virtual world, interact with the 3D objects
  + Very good graphics, and can be used for very complicated programs
  + Mainly used for animation and not for simple non-animation based code
  + Also very similar to code, need to actually type a lot of things
* Scratch
  + Can be used for a variety of different things, from animation, music, to simple programs
  + Easily interact with graphics
  + Like how the blocks of code snap in with each other, will implement in my program
  + Has many different functions to drag and drop
* Blockly
  + Very similar to scratch but for simpler programs
  + Can modify the draggable objects for own projects
  + Has a puzzle like feel to it
  + Open source and has many different apps built using Blockly
  + There exists a tool that allows you to translate into some programming language (Python, JavaScript, XML)

StoryBoard

