

# **Gameful Direct Instruction Midterm Progress Report Fall 2018**

Veronica Woldehanna

## **1. Abstract**

This report will describe the progress made in the Gameful Direct Instruction, parson's coaching feature. The items discussed are the user stories completed and how they were implemented. This report will also discuss Items to be completed before the end of the semester.

## **2. Architecture**

The features being discussed were implemented by adding and editing functions in Scratch.as, ScratchObj.as, ScratchSprite.as and related classes in Sage-Scratch.

## **3. Implementation**

So far, the focus has been on incorporating visual programming to the extent that it can be incorporated into Scratch. As mentioned in the proposal, having a visual representation of their solutions can be very helpful in helping students spot errors they are making. Especially if they are able to see how that representation changes with changes they make to their solution. The user stories designed to do this are Auto-execution and Auto-initialization.

### **3.1 Auto-execution**

This task was motivated by wanting to show students how the changes they make to their solution, dropping a block in the scripts pane, affects the intended outcome of their solution more immediately. So now as soon as a student drops a block on to the scripts pane, their new solution would run automatically instead of them having to press the Green Flag button themselves. Following convention, this requires they have a "When green flag is clicked" block in the scripts pane. Consequently, this block has been added as a default block in the first sprite that the teacher creates when designing the game and that block will also be visible in the students' view in the scripts pane.

### **3.2 Auto-initialization**

Each time that auto-execute runs their solution, if their solutions actually involve moving the direction or position of sprites, at the end of the execution, their sprites will potentially be in a new location with new direction. If they continued to build/edit their script and auto-execute runs their solution again, the sprites will start with their updated location. So auto-initialization, meaning initializing sprites to an initial position that they reset to at the beginning of each execution, was done because the goal was to show them how the history, in this case the path

of the sprite, and final result of their variables change with changes they make to their solutions. Auto-initialization makes it easy for them to compare the execution of the code they saw with and without the change allowing them to see if they are heading in the right direction. This initial sprite location is just the initial random location the sprite gets during creation but teachers are able to change this initial position.

#### 4. Limitations and Assumptions

For auto-execution, it assumed that students incorporate the “when green flag is clicked” button in their solution. And even though, it is now there as a default, I have yet to make sure that they are unable to remove this block from the scripts pane.

For auto-initialization, it is assumed that the teacher will set an initial position for the sprites and that these sprites are what the students will be working with. At the moment, if they create a new sprite, it will create a default initial position they can’t change.

#### 5. Future Work

For the rest of the semester, I plan on finding more ways to incorporate visualization into parson’s puzzle and scratch. I will also be working on displaying the history of actual variables in one run of students’ scripts. Depending on what I find and if time permits, I also plan on working on error feedback for students. As mentioned in the proposal, a better feedback system was requested by the majority of students in Dale Parson’s first paper on parson’s programming puzzles. For now, this involves collecting explanations from students that drop a wrong block onto the scripts pane but then immediately remove it to add the correct block. This leverages the scoring system that is in place by assuming that a student knows when they have dropped the wrong block as they’ll get point deductions. These explanations can then be used to help students in the same situation that are unable to find the correct block.