COMS 6901: Midterm Progress Report

Sanil Shah | ss4924 | Summer 2018

SAGE Scratch: Parson's Puzzles and Mission Management

1. Abstract

This paper aims to describe the progress made on the Gameful Direct Instruction as part of the SAGE project. It will focus primarily on the work done towards the Parson's Puzzle 1.1 feature. The report will cover the user stories worked on in greater detail including the architecture of the various features added, the technologies used during development and the current status of progress made towards implementing parson's puzzles creation and editing in the SAGE Scratch branch. Finally the report covers the future work to be done on this project and presents a timeline for tasks to be completed by the end of the semester.

2. Architecture

This project builds features on top of the existing SAGE Scratch project. This means that the architecture follows the existing design and methodology of SAGE Scratch. Query parameters are used to pass information from the SAGE Affinity Space (front end) to the swf object which allows Scratch to initialize with the correct data. The data tells Scratch about the puzzle being played or edited, the user ID currently using Scratch (instructor or student) as well as whether Scratch needs to be opened in PLAY or DESIGN mode. This data is also used to pass information about instructions to be displayed to the user and the expected solution which can be used in the future for scoring purposes.

This project is built entirely in ActionScript while using Gradle as a build system. The features added to this project are also completed in ActionScript by modifying files of the existing SAGE Scratch branch. The build process setup in previous years was not modified and most development was done locally by running local versions of the SAGE Affinity Space and the SAGE Assessment Server, as well as a local version of SAGE Scratch.

3. Progress

3.1. SAGE Scratch Setup

Setting up the SAGE Scratch development environment locally took up a considerable amount of time over the first few weeks. While the instructions on TFS are mostly up to date, the build process is no longer the same since the project has moved on to using Gradle. This made building the scratch.swf file locally extremely challenging. This was further complicated by a previously made bad merge into the development branch that prevented SAGE Scratch from building at all. This was fixed by rolling back to an earlier commit that was successfully building and adding the below features on top of that locally.

3.2. Parson's Creation

The Parson's Puzzle Creation user story was aimed at allowing instructors to create new parson's puzzles from the SAGE Affinity space. Whenever a game is created the instructor is redirected to the game creation page which generates a new game ID and opens in Parson's puzzle creation mode. Scratch automatically initializes in DESIGN mode, allowing the instructor to design a new parson's puzzle. Changes made to the puzzle and the desired solution are saved as they are made in the Scratch playground. Similarly the instructor can modify the "Question" and the "Hint" that will be displayed while the puzzle is being worked on by the student. Both these properties will also be saved as part of the game and displayed to the student when they attempt to work on the puzzle. This functionality is close to complete and needs to be merged back into the development branch. Some issues with this will be discussed in the limitations section.

3.3. Parson's Edit

The Parson's Puzzle Edit user story builds on the previous one. It allows instructors to update parson's puzzles that have already been created. The SAGE Affinity space already exposes this functionality through the UI. Development done on this feature ensures that SAGE opens in DESIGN mode when a instructor is attempting to update an existing puzzle. Further it ensures that the instructions are not displayed to the instructor since those are distracting and essentially redundant. Changes made to the puzzle are saved automatically, similar to the parson's puzzle creation user story. Progress has been made on this piece as well, but it is yet to be completed.

4. Limitations and Assumptions

Current development has been done entirely locally. The features worked on have yet to be tested in production with an actual database behind them. The assumption made while working on the progress made so far has been that the database integration will work seamlessly once this code is merged into the development branch.

Another limitation was that the features described above have been built on top of an old commit since the latest commit was failing to build. This may make the merge more complicated to ensure that any code between the last successful build and the failing build isn't completely lost. It is still unclear what code was added in the bad merge and why that code was submitted, so it is possible that it may be safe to ignore entirely and simply build functionality on top of an older commit as is being done currently.

5. Future Work

As described in the project proposal future work will include completing work on the Parson's Edit user story and tackling the Parson's Play Palette user story as well. Time permitting I would like to work on the Parson's Scoring piece as well to ensure that the scoring system during puzzles is offering useful information to students. The remainder of the semester will follow the following schedule if all goes according to plan:

Date	Milestone
06/16/2018	Development environment setup including IntelliJ and local testing
06/23/2018	Understanding of existing functionality and Parson's Puzzle MVP
07/14/2018	Parson's Puzzle creation implementation
07/21/2018	Parson's Puzzle editing functionality implemented
08/01/2018	Palettes hidden in Parson's Puzzle play mode
08/11/2018	Merging existing features into the development branch