Scratch Game or Story

PLTW | Computer Science

Procedure

1. Form pairs as directed by your teacher. Meet or greet each other to practice professional skills. Set team norms.
2. **Review the criteria and rubric for the project on the following pages.**
3. Brainstorm ideas using tag lines and thumbnail sketches. Follow the guidelines for brainstorming: never criticize ideas during brainstorming, but "piling on" is welcome.
4. Develop one or two of your ideas with further discussion and documentation.
5. Decide on one game or story that you will develop into a product. When your instructor gives you the cue you will:
    a. Use diagrams, lists, and flowcharts, to explain to another team of developers what you want your game or story to do.
    b. **Note**: *Up to now, you have been (mostly) playing the role of the client. Now you will transition to the role of developers. When another team tells you what they want their product to do, pretend they are your client and you will create their product. If they are not giving you enough information to create the product to their satisfaction, ask questions.*
6. Now that you have feedback on your game or story, strategize, code, and test in small increments until you meet your goal.
    a. Switch driver and navigator roles every 15 minutes or so.
    b. Include Scratch comment balloons as you develop your solution.

Conclusion

1. Reflect on the creative process you used. What was useful? Discuss your reflection with your partner and then write a reflection individually.

2. Reflect on the team dynamic. What helped the team work well together? Discuss your reflection with your partner and then write a reflection individually.

Game Criteria

- User Interaction
  - The user should be able to use keyboard and/or mouse input in a way that fundamentally affects what happens.
- Objective
  - The game should have an objective with several or as many degree of progress toward the objective as possible. A score would be sufficient, but many alternatives exist.
- Multiple States
  - The game should include different states in order for the user to experience variety. Levels during which the difficulty changes or bonus stages appear would be sufficient, but many alternatives exist.

Story Criteria

- Multiple Acts
  - The story should have different sections, like multiple acts in a play. For example, the story might occur on different stages, but many alternatives exist.
- User interaction can affect story line
  - The user should be able to use keyboard and/or mouse input in a way that fundamentally affects what happens.
- User interaction between story line branch points
  - The user should be able to use keyboard and/or mouse input in a way that controls what is occurring within at least one of the acts.
- The project at **http://scratch.mit.edu/projects/12586146/** can be used as an example for structuring a story with a state map.

Problem 1.1.7 Scratch Game or Story

	4	3	2	1
Solves Problem	Artifact fully addresses personal, practical, or societal intent posed by problem statement	Artifact addresses the personal, practical, or societal intent posed by problem statement	Artifact mostly addresses the personal, practical, or societal intent posed by problem statement	Artifact does not adequately address the personal, practical, or societal intent posed by problem statement
Documentation	Uses appropriate documentation of work. The three formats for documenting work: • Scratch comments • Project Design Notebook *(Has Planning Materials)* • Named versions of project	Uses appropriate techniques in 2 forms for documenting work	Often uses appropriate techniques for documenting work	Does not usually use appropriate techniques for documenting work
Collaboration	Provides helpful original input to others Promotes positive, productive, and respectful team dynamic Encourages and incorporates input from others Promotes equitable workload	Provides adequate original input to others Maintains positive, productive, and respectful team dynamic Positively incorporates input from others Maintains equitable workload	Significant but limited input Usually maintains positive, productive, and respectful team dynamic Receives input from others Shares workload somewhat equitably	Limited input Is not promoting positive, respectful, or productive team dynamic Discourages or is unresponsive to input from others Does not promote equitable workload

Presentation (Gallery Walk)	Easy to follow instructions with perfect user interaction. Instructions & Notes/Credits sections are easy to read and completely filled out on Scratch	Easy to follow instructions with okay user interaction. Instructions & Notes/Credits sections are easy to read and mostly filled out on Scratch	Okay instructions with minor glitches in user interaction. Instructions & Notes/Credits sections are easy to read and somewhat filled out on Scratch	Inadequate instructions with glitch user interaction. Inadequate Instructions & Notes/Credits
Appropriate Algorithm	Code demonstrates use of appropriate algorithms	Code mostly uses appropriate algorithms	Code often uses appropriate algorithms	Code does not use appropriate algorithms
Explanation of Algorithm	Documentation (**comments**) clearly and thoroughly explains the algorithm(s)	Documentation explains the algorithm(s)	Documentation insufficiently explains algorithm	No documentation
Total:				