

**Problem 1.1.7 Scratch Game or Story**

|  |  |
| --- | --- |
| Introduction  Software developers sometimes use their skills to create a product for a client. At other times, developers create a product for themselves. Whether meeting their own needs or the needs of a client, collaboration allows developers to work on bigger projects and have more fun.  The way that team approaches a problem is called their methodology. In this problem, you will apply the Agile methodology.  What do you want to create? How will you work with others to define the problem and create the solution? | artists,children,easels,girls,instructors,painters,painting,persons,teachers,women |

Materials

* Computer with Scratch™, Internet access, and camera
* Project notebook

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. Using diagrams, lists, and flowcharts, explain to another team of developers what you want the product to do. 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. You will now serve as the software developer for your own project. Plan the first sprint. The planning session is normally **timeboxed**, meaning it is limited to 3 minutes of planning for each hour of the sprint. Since this is your first sprint, you might take longer.

Your plan should include two bullet point lists, as follows:

* A **product backlog** provides the most important **user stories** at the top. User stories lower on the list can be large and poorly defined.
* A **sprint task list**. Sprint backlog **items** should be specific and broken into small **tasks**.

1. Strategize, code, and test in small increments.
   1. Switch driver and navigator roles every 10 minutes or so.
   2. Include Scratch comment balloons as you develop your solution.
2. Write a reflection on how well your product meets the clients’ needs. Include descriptions on strengths and weaknesses of the solution as well as potential plans for the next sprint.
3. Prepare to present your project at the end of the first sprint. Your teacher will describe the presentation format they would like to use.

**Conclusion**

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

**The useful part of the creative process was being able to shoot ideas off of each other, accept the possible or creative ones and set aside the more impossible and less creative ones allowing only the best possible ideas to make it through.**

1. Reflect on the team dynamic. What helped the team work well together? Discuss your reflection with your partner and then write a reflection individually. **Communication helped make the cogs turn a little easier in the creative process and the production process. Creating more ideas, pointing out solutions, and problems were all apart of communication. Thus communication in this sort of project is of the upmost importance.**

**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 * 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 |
| Other comments: |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Presentation | Effective presentation techniques:  Posture  Gestures  Voice  Eye Contact | Mostly effective presentation techniques:  Posture  Gestures  Voice  Eye Contact | Mostly adequate presentation techniques:  Posture  Gestures  Voice  Eye Contact | Inadequate presentation techniques:  Posture  Gestures  Voice  Eye Contact |
| 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 |
| Explanation of Problem Solution | Prose clearly and thoroughly explain how the solution meets the need  Prose clearly explains the solution’s strengths and weaknesses and strategizes for improvement | Prose explains how the solution meets the need  Prose mentions a strength or weakness and ideas for improvement | Prose explains how the solution meets the need  Prose mentions a strength or weakness | Prose does not address how the solution is connected to the need |
| Planning | Product backlog shows intent  Sprint task list subdivides to simpler subproblems | Product backlog shows intent  Sprint task list shows subdivision to simpler subproblems, but some tasks are too big | Product backlog and sprint task list show inadequate attempt to indicate long-term intent and an attempt to break down problem into simpler tasks | Product backlog or sprint task list are missing. |