Story Creator Requirements Specification

Brandon Roberts, Nate Olderman, Billy Rathje, DJ Maguddayao, Kyle Dybdal

September 25, 2013

1 Summary

This SRS describes the requirements and specifications for the Story Creator module of the edith software, a system to help younger students learn to program.

2 Introduction

2.1 Edith

Edith will be a web-based educational system designed to help younger students develop an interest in and learn about programming. The user will be able to create a "story" by dragging and dropping objects, giving the objects animations, and creating ways for the user or other users to interact with their story. By doing this students will be able to learn how relationships among objects work in programming.

2.2 Scope

The Story Creator module is intended to function as a display that integrates the work of the Animation Systems module and Visual Editor module in order to allow the user to create an animated story. This module will employ the Visual Editor and Animation Systems module to interact with objects that are given by the Object Creator. This module will provide a user interface that combines the displays of the Animation System and Visual Editor module. Story creator will add extra functionality to the Visual Editor and Animation Systems module and put together the interaction between the Object Creator module in order to specify actions performed by the objects given by the Object Creator module.

2.3 Purpose

The purpose of the Story Creator section of the Edith software is to ensure that each other piece of the software can work together as a cohesive whole. This will be done through the user interface by providing a way for the user to select objects and choose actions and animations for each object. From there, the user will be able to interact in different ways with the animated objects they have created and then finalize the "story" they have created

so they can keep it or share it among friends. Story creator will construct these interactions between the pieces as well as a pleasant display view for the user.

3 Functional Requirements/Use Cases

3.1 "Create a new story"

- Actor: The user
- Preconditions/Assumptions: No preconditions.
- Flow of Events:
  - User will open program.
  - User wants to create a new story.
  - User will select an option to create a new story.
- Alternatives: A previously created story may already be opened, in that case the user will close the story and proceed with the second event.
- Postconditions: The program will be ready for the user to create a story.

3.2 "Save a story"

- Actor: The user
- Preconditions/Assumptions: None.
- Flow of Events: - User selects a "save story" option.
  - A representation of the story and its objects, scripts, and settings is saved.
- Alternatives: If the story has never been saved before, the story editor will prompt the user to "save as."
- Postconditions: The story is saved to the local computer.

3.3 "Close a story"

- Actor: The user
- Preconditions/Assumptions: None.
- Flow of Events: - The user selects an option to close the story.
  - The story is removed from the story editor.
  - Animations, scripts, and objects associated with the story are cleared from their respective editors.
- Alternatives: If the story is not saved, the story editor will prompt the user to save the story. If new objects, animations, or scripts have been created specifically for this story and have not been saved, the story editor will also prompt the user to save these new components.
- Postconditions: None.

3.4 "Create an object"

- Actor: The user
- Preconditions/Assumptions: A scene is open.
- Flow of Events:
  - User wants to create an object in their story.
  - User will select an option to create an object.
  - User will use the mouse to drag the object.
  - User will choose where to place the object.
  - User will drop the object in that place.
- Alternatives: Object may not be compatable with some other objects/animations already in story, in that case user will choose whether to delete the old object or not create the new one.
- Postconditions: The program will display one additional object than it did before.

3.5 "Remove an object"

- Actor: The user
- Preconditions/Assumptions: At least one object is in the scene.
- Flow of Events: - User selects an option to delete objects.
  - User selects the object to delete.
  - The object dissapears from the scene.
- Alternatives: None.
- Postconditions: The program will display one fewer object than it did before.

3.6 "Move an object"

- Actor: The user
- Preconditions/Assumptions: At least one object is in the scene.
- Flow of Events: - User selects an option to move objects.
  - User drags the object from its original location to a new one.
- Alternatives: If the user attempts to drag the object outside of the scene, the object will stay within the scene frame's boundaries.
- Postconditions: None.

3.7 "Develop a script"

- Actor: The user
- Preconditions/Assumptions: None.
- Flow of Events: - User interacts with the visual editor to develop a new script.
- Alternatives: None.

- Postconditions: A script is added to the visual editor's list of scripts.

3.8 "Edit a script"

- Actor: The user
- Preconditions/Assumptions: None.
- Flow of Events: - User interacts with the visual editor to edit a script.
- Alternatives: None.
- Postconditions: A script in the visual editor's list of scripts is changed.

3.9 "Delete a script"

- Actor: The user
- Preconditions/Assumptions: None.
- Flow of Events: - User interacts with the visual editor to delete a script.
- Alternatives: None.
- Postconditions: A script in the visual editor's list of scripts is removed.

3.10 "Save an object or script"

- Actor: The user
- Preconditions/Assumptions: The user has created a new object or script.
- Flow of Events: - The user wants to save an object or script for reuse. - A representation of the object or script is saved.
- Alternatives: If the story has never been saved before, the story editor will prompt the user to "save as."

- Postconditions: The object or script is available for reuse in the scene. It can also be made available for reuse in other scenes.

3.11 "Animate an object"

- Actor: The user
- Preconditions/Assumptions: At least one object is in the scene and one script is
  in the visual editor.
- Flow of Events: - User selects a script from the visual editor.
  - User drags or otherwise attaches the script to an object in the scene.
- Alternatives: None.
- Postconditions: The object has an associated script/animation.

3.12 "Play back animation for a single object"

- Actor: The user
- Preconditions/Assumptions: At least one object is in the scene.
- Flow of Events: - User selects an option to play back object animations.
  - User selects a "play" option.
- Alternatives: None.
- Postconditions: None.

3.13 "Playback a story"

- Actor: The user
- Preconditions/Assumptions: None.

- Flow of Events: - User selects a "play scene" option.
  - The story plays for the user in the animation viewer.

- Alternatives: The user may interrupt playback with a variety of options, including stop, fast-forward, or rewind. The playback system will respond to these options. In the case of "stop," the scene playback will cancel. In the case of fast-forward or rewind, the scene will progress forward or backward more rapidly as long as the user continues to select the fast-forward or rewind option.

- Postconditions: None.

3.14 "Share a story"

- Actor: The user

- Preconditions/Assumptions: None.

- Flow of Events: - User selects a "publish story option."
  - The story is converted to a representation that the sharing framework can read.
  - The story representation is sent to the sharing framework so that it can share the scene.

- Alternatives: If the story is not saved, or changes have been made since the last save, the story editor will prompt the user to save the story before sharing it.

- Postconditions: None.

4 Nonfunctional Requirements

4.1 "Ease of Use"

The system should be easy for novices to use. One test of the system's usability is the number of errors made per hour of use. An "error" is defined as the user selecting the wrong options for:

a) adding, editing, or deleting an object
b) attaching a script/animation to an object
c) saving and sharing the story
d) playing back the story.

The number of errors made by experienced users may differ significantly from those made by first-time or novice user. An experienced user should make no more than 3-5 errors per hour. While there is some complexity to the story editor, an expereinced user should not have trouble knowing which options to choose to add objects or play the scene, for example. In comparison to the rest of the system, the story editor should be relatively similar to navigate, like a programming IDE or video editing system. Novice users may make many more errors when initially learning the system, but the system should not be frustrating to learn because it is designed to engage novice programmers and to make the task of programming feel simple and intuitive. Anything beyond 5-15 errors within 1-3 hours of use would be too many.