Software Requirements Specification

for

Pick-A-Path

Version 1.0 approved

Prepared by

Group 2

September 7, 2018

Table of Contents

1. Preface

This SRS is designed to describe the requirements and specifications of Pick-A-Path, a game-development engine. It explains the functional features of the software, along with interface details, design constraints and related considerations such as performance characteristics. This SRS is intended for interested users, stakeholders, and developers of the software.

2. Introduction

The main goal of Pick-A-Path is to have an adaptive environment molded by user decisions in order to experience basic story game development. This software will be a GUI for a user to make logical decisions within a base environment given. This can be then used to make a number of different possible outcomes for their game. The user will have two different modes to develop their game, a developer mode and a game mode, to make this software as user-friendly as possible.

~~To create this idea, we went to the drawing board several times to decide how it would look and how it would actually function.~~ Deciding to make two separate modes and to broaden the outcomes a user can make are two of the main functions that we will focus on in this project. The user will be able to create their file, save it, and come back to it later if they choose. This will allow users to change their ideas without starting from scratch.

While this program may seem similar to other game-developing engines, this program is set apart by its ability to use a logic model to direct users to their desired outcome. ~~Going back and forth with the team and the professor, our requirements are not only goals we would like to achieve, but requirements we believe are able to be met with the ability of team.~~

This program will be challenging to develop, but with ample teamwork and dedication, our team will be able to piece together this software by the end of the given semester.

3. Glossary

SRS – Software Requirements Specification. Document that captures complete description about how the system is expected to perform.

GUI – Graphical User Interface. A type of user interface that allows users to interact with electronic devices through graphical icons and visual indicators.

Environment – The area where a user will work on and run their program. Synonym to a platform.

Developer – Person or persons involved in creating, designing, testing, etc. a given software.

IDE – Integrated Development Environment. A software application that provides facilities to computer programmers.

N-Amount – N in this sense is used as a variable to indicate any number of possible amounts.

CPU – Central Processing Unit. The brains of the computer where most calculations take place

GPU – Graphics Processing Unit. A programmable logic chip specialized for display functions.

RAM – Random Access Memory. The hardware in a computer where the operating system, application programs, and data in current use are kept so they can be quickly reached by the device's processor.

HDD – A computer Hard Disk Drive. The mechanism that controls the positioning, reading, and writing of the hard disk.

4. User Requirements Definition

Functional Requirements:

- Pick-A-Path will provide a platform for the users to create their own adventure games.
- The system shall provide two environments for the user: Developer Mode and Game Mode. The user can use the developer mode to create their own games according to their logic, and the Game mode is where the created game can be accessed to play.
- In the Developer Mode, the user shall be provided with boxes and arrows, which could be designed together to make a logical flow of the story that the user creates by inputting text into each of them.
- Each of these boxes has attributes and functionalities of its own, which would further branch the story of the game.
- The user can start a new game or should be able access a previously started game and edit it further whenever they need to.
- The user should be able to go back to the previous parts of the game to edit or delete a particular branch that they would want to.
- The user would not be able to edit any other games that were created using a different game development platform other than Pick-A-Path.
- The system's interface shall be made user-friendly to use by providing drop down menus, radio buttons and dialogue boxes.
- In case of power cut the system shall save the current work automatically, which can be later accessed by the user.

Non-Functional Requirements:

Performance:

- Every action that would take place in the system shall have a response time less than 3 seconds.
- The user can add as many boxes and arrows that their story would need.

Development:

- The project is estimated to be completed by December 2018.
- The resources used for the development of Pick-A-Path include five staff members (a Project Manager, Design Lead, two Software Developers and a Tester), working 5+ hours/week.

5. System Architecture

Our software will be comprised of three major “blocks” or pieces. These blocks are:

Main menu – This will be the first GUI the end user interacts with. At this point, the end user can either choose to load a previously saved game file to play or they can choose to create a new game or edit an existing one. If they choose to play a saved game, this will take them to "game mode”, which is essentially the software’s IDE or console. If the user chooses to edit or create a new game, then this will take them to “developer” mode.

Developer mode – If the end user chooses the developer mode option, they will then be prompted what they would like to do (create a new file or fetch a saved file) and create or edit their game as described in this document.

Game mode – In this final stage, the end user can test their different choices, go back to the developer mode, save their game to a file for later use, and run their game.

6. System Requirements Specification

Purpose

The Pick-A-Path will be a computer-based software system that aids a user into creating their own conditional environment in which they insert their own objects and decisions.

System Description

Pick-A-Path will operate only on Desktop or Laptop computers, it currently will not support mobile devices. It will be coded and ran using primarily Java. The system in its first state will be a box with the option to create arrows and more boxes.

How it works

Pick-A-Path will open with a GUI that displays a box. The user can insert content into this box, and then branch off from the first box with arrows. These arrows represent options, so if there are three arrows, there are three options that the user can choose. From here, they will have to fill in what each option is, and then those are the choices the player can make within the software. Each box can contain n-amount of arrows, and thus an n-amount of boxes. There will also be several arrows that lead back to other boxes instead of creating a new box. The system state will end when the user creates a box that has no arrows. There will also be objects the user can put into the game, which they can specify to interact with certain arrows.

Logical Structure

All logic applied by the user. We will provide various conditional statements in the form of arrows such as *IF*, *THEN*, *ELSE*, along with others, to guide the user through the possible conditions they wish to make

Hardware Requirements

The hardware needed for the game should not be very demanding. The user will need to meet the minimum hardware requirements to run our software:

- CPU: Intel Core i3-3210 3.2 GHz / AMD A8-7600 APU 3.1 GHz or equivalent
- RAM: 2GB
- GPU (Integrated): Intel HD Graphics 4000 (Ivy Bridge) or AMD Radeon R5 series (Kaveri line) with OpenGL 4.4*
- HDD: At least 1GB for game core and other files

7. System Model

Nonfunctional Requirements

- Create two separate systems including a developer mode and a game player.
- Basic easy to use GUI when developing.
- Implement evaluation boxes that both contain the scenario and evaluate the state of the game.
- There are arrows that lead to other boxes which represent choices.
- Allow for the creation of objects that can change the state of the boxes.

Organizational Requirements

- The GUI prompts users when booting the program if they would like to create a new file or to edit an existing file.
- The ability to retrieve a file or to go to game player.
- Allow for the programs multiple environments to run in a java IDE.
- There is development environment that allows for the creation of the story.
- There is a run-time environment that plays the game.

Product Requirements

- The game player must be able to read and play the saved game files.
- Files may be saved, edited, and deleted.
- State must be correctly evaluated to maintain functional arrow and box structure.
- Files will be text and object based so memory and storage are not currently a concern.
- The creation of state altering objects will be supported to enrich the gaming experience.
- There are two kinds of boxes one that evaluates state and one that maintains the functionality of the original box with the addition of being a “game ending box” that allows no further choices.

External Requirements

- Allow for any safety, ethical, legal or any other requirements to be addressed and implemented appropriately.

Ethical Requirements

- Provide a content checker that asks the user if the wish to change words or phrases deemed controversial before completing a file.

8. System Evolution

Additional possible features

- Graphics
- Built-in tutorials
- Export ability
- Mobile interface

9. Product Designers Approval

The undersigned acknowledge they have reviewed the Pick-A-Path Product Design Specification document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Role:	Project Manager	Date:	9/7/2018
Print Name:	Olivia Langley		
Signature:			
Role:	Design Lead	Date:	9/7/2018
Print Name:	Lucia Ristea		
Signature:			
Role:	Developer	Date:	9/7/2018
Print Name:	Logan Murphy		
Signature:			
Role:	Developer	Date:	9/7/2018
Print Name:	James Erpenbeck		
Signature:			
Role:	Tester	Date:	9/7/2018
Print Name:	Pranaya Kalidindi		
Signature: