Software Requirements Specification

Version 1.1 approved

Prepared by
<Team 3: Sasha Tenhumberg, Yue Bu, Nguyen Luu, and Ziyuan Ye>

<7/12/2017>

Revision History

Date	Changes	Version
6/7/2017	First draft	1.0
7/12/2017	Second draft	1.1

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to describe how the game "Animal Pop" functions and its varying features. The intended audience are the gamers, game designers, managers, and testing engineers. This document has the following goals: 1) define the scope of the project, 2) provide definitions to help better understand the document, 3) explain the functions and capabilities of the system, and 4) present a representation of the system interface.

1.2 Scope

For this project we will be making a single-player animal-themed game where the objective is to match 3 or more animal blocks of the same type while obtaining points to reach the goal points and move on the next level. The format of the game will follow a 9x9 grid, which allows for 81 elements to be displayed on the grid in the gameplay view at random. The task of the player is to match three of the same elements to score a point. After each successful match-up, those elements disappear, and everything is shifted down so new random elements can fill the grid again. As the game progresses, the levels will get gradually more difficult by decreasing the number of swipes a player gets and reformatting the grid, which will provide entertainment for players of all skill levels. Our goal for this software is to make a game that is fun, addicting, and challenging for players of all ages. We don't have any business goal at this time, but it would be easy to implement a revenue stream for this app by implementing ads at the bottom of the screen and including an option to pay to remove them.

1.3 Definition, Acronyms

Xcode - An integrated development enviroment for ios system .
Swift - A programming language for macos, iOS, and watchos.
Level - A series of stages that determine the difficulty.
Target - The number of points a player must reach to go on to the next level.
Moves - The number of swipes or exchanges of blocks a player gets for that round.
Score - The number of points a player currently possess.
Grid - A boardlike structure which contains all the element blocks for the round and changes from round to round.
Shuffle - A button at the bottom of the screen that allows for all the elements in the grid to be moved randomly to a new spot.

1.4 Overview

The following contents of this document can be classified into four parts. The first part is an overall description of this project. The second part is the specific requirement. In this part, the game will be explained in details. The third part will talk about the future improvement of this game. The last part is the reference page. It will lists the resource we cited in our document.

2. Overall Description

2.1 Product Perspective

2.1.1 System Interface

The “Animal Pop” system works independently of other systems. No data from other sources is required to operate the system.

2.1.2 User Interface

Users interact with the program through their iPhones.

2.1.3 Hardware Interface

This app can be install and run on any mobile devices that has an iOS operating system. Since the mobile application do not have any designated hardware, it does not have any direct hardware interface. The hardware connection to the database server is managed by the iOS operating system on the mobile phone.

2.1.4 Software Interface

The program will initially work on iOS mobile devices exclusively using Swift. This program can be download via the App Store which require iOS 10 and above.

2.1.5 Communications Interface

No communication between devices or a server will be necessary as this is a single-player game and high scores will be stored on the device. This app does not require any communications.

2.1.6 Memory

Roughly 30 megabyte of storage space on the device will be required.

2.1.7 Operations

Start: Program begins, displays the main menu and gaming options.

Gameplay: Users complete each level by switching the blocks to match up 3 or more of the same type of elements. As the player obtains the target score, the player will advance to the next level.

Endgame: Users are given the choice of ending the game at any given moment by exiting out of the app.

2.1.8 Site Adaptation Requirements

There is no adaptation for any installations.

2.2 Product Function

The intended function of this system is to act as an interactive form of entertainment for its user base through its addicting gameplay.

2.3 Users Classes and Characteristics

The user classes we anticipate will use this product are:

Beginner: Users who are very young and/or may not be able to solve levels very quickly or effectively but should still be able to find enjoyment in the graphics and easier levels, as well as being able to stumble on correct answers to harder levels through luck

Intermediate: Users who are able to solve easy levels quickly and find intermediate difficulty levels challenging but still possible, these users may still enjoy advanced levels but won't be able to solve them reliably

Advanced: Users who can solve all or most levels easily, these users will be the most challenging to entertain because giving them gameplay that engages them enough will be difficult

2.4 Constraints

The game is for iOS System device only. Users need to have iOS system devices and have enough space to download this application via App store.

2.5 Assumptions and dependencies

We will be developing this game for iOS mobile phones and are making the assumption that the development environment won't suddenly change. We are also making the assumption that the system the game is being downloaded via App Store and user will have enough memory space for this application.

3. Specific requirement

3.1 External interface

This application does not contain any external interfaces.

3.2 Functions

3.2.1 Phase 1

3.2.1.1 Gamescreen

3.2.1.1.1

The game displayed when the app is opened, priority is high – vital feature for the app to function.

3.2.1.1.3

REQ-1: Background that is pleasing to look at.

REQ-2: Interactive grid that displays the units for the game.

3.2.1.2 Interactive Match-3 Game

3.2.1.2.1

Grid filled with elements that can be slid 1 space switching that elements with the space adjacent to it. When 3 or more matching elements are switched into a row those matching elements disappear and the score is increased, and new elements take their place.

3.2.1.2.2

When an element is dragged into another elements space those elements switch places, if there are now 3 or more matching elements in a row those elements disappear and more elements take their place, the score is increased.

3.2.1.2.3

REQ-1: Game grid - a grid of varying shapes that contains at most 81 elements that can be dragged one space to the top, bottom, left, or right, switching those elements.

REQ-2: Elements matched together should disappear from the game board and be replaced by new randomized elements.

3.2.1.3 Sound Effects

3.2.1.3.1

Sound effects after a successful match-3 or more, priority is low – not a vital feature for the app to function but makes the game more interactive.

3.2.1.3.2

REQ-1: Sound effect for a successful match.

3.2.2 Phase 2/Future Phases

3.2.2.1 Shuffle Button

3.2.1.2.1

A button at the bottom of the screen that randomizes the elements and resets the scoreboard.

3.2.1.2.2

When the shuffle button is pressed the elements should be randomized.

3.2.1.2.3

REQ-1: Randomizes elements on shuffle button press.
REQ-2: Resets scoreboard to default state for that level.
REQ-3: Reset puzzle is still solvable.

3.2.2.2 Scoreboard

3.2.2.1.1

A section that displays the target score, player’s score, and number of moves left.

3.2.2.1.2

REQ-1: Scoreboard - a section at top of the display which displays the number of moves left as well as the current score and the required score to pass the level.
REQ-2: The score displayed at the top must increase when 3 or more elements are matched together. Each element removed from the game grid should increase the score by 20 for each. element multiplied by the number of elements matched minus 2.

3.2.2.3 Level Complete/Game Over Screen

3.2.2.1.1

If the player successfully obtain the target score with the available amount of moves, a level complete screen will pop up.

3.2.2.1.2

If the player does not obtain the target score with the available amount of moves, a game over screen will pop up.

3.2.2.1.3

REQ-1: Level complete appears after successfully completing the level.
REQ-2: Game over appears after losing the level.

3.3 Performance Requirement

The app must be able to run smoothly under several conditions:

Normal input - between 5-60 taps per minute
Slow input - less than 5 taps per minute
Heavy input - faster than 60 taps per minute

3.4 Design Constraints

The major constraint of this system is that it will be made for iOS mobile devices only. Any non iOS devices including those that run other system will not be able to use this app. Any iOS tablet will not be able to use this app.

3.5 Software System Attributes

3.5.1 Security

There is no security features(login, passwords) needed for this application

3.5.2 Availability

The system is available for iOS mobile devices only. if it has any error or crashes it can be solved by download from App store.

3.5.3 Safety

Prolonged exposure to this game may cause eyesight problems to users.

3.5.4 Testability

The app will be tested by us before we believe it is available for release.

3.5.5 Portability

The app can run on any iOS mobile devices, and the app installed through the APP store.

3.6 Response Time

The system will be able to respond to user’s input in less than 1 second.

4 Appendix