Singularity Software

Milestone 2

September 25, 2011

By signing below, I approve the contents of the following document.

Alex Mullans ________________________________

Ruben Rodriguez ________________________________

Ethan Veatch ________________________________

Kurtis Zimmerman ________________________________

Contents

1 Executive Summary

This document elaborates the functional features of the the Siftables Emulator project. It first gives a brief background of the project before moving on to elaborate features with use cases and declarative statements, where appropriate. It also maps all covered use cases to their relevant features. Finally, it presents mockups of the Emulator's design for consideration and to solicit client feedback. Future documents will reference use cases presented in this document as features are progressively implemented, tested, and delivered.

2 Introduction

Developers of applications for the Sifteo Cubes currently must test programs they create for the platform on the Cubes themselves. With a full release of the Cubes and corresponding Application Programming Interface (API) still pending, developers unable to join the Sifteo Early Access program are left without a software-based interface within which to productively develop Sifteo programs. As such, Singularity Software will provide, in the form of the Siftables Emulator, a software-based emulator for the Sifteo Cubes that will allow any developer to try programming in the unqiue environment provided by the Cubes.

This document is the second in a series of milestone documents that will accompany the planning of the Siftables Emulator. It provides coverage of functional features via uses cases and declarative statements before presenting initial design mockups. Future milestones will present plans for change control, coding standardization, and testing. Finally, design and usability reports will make up the core of milestones near the end of the quarter as the software stabilizes.

3 Project Background

The Siftables Emulator is being developed by Singularity Software as part of the Junior Project sequence of classes at Rose-Hulman Institute of Technology. When projects were solicited for the sequence, clients Tim Ekl and Eric Stokes (both Rose-Hulman alumni) submitted a request for an emulator for Sifteo Cubes, a new platform intended for "intelligent play." After Singularity was chosen for the project, we met with Mr. Ekl to determine the three primary features of the Emulator: a Workspace where 1-6 cubes could mimic the manipulations possible with physical Cubes, an API to program those virtual cubes, and a set of example games designed to show off the first two features. Singularity's Emulator will be the first program of its kind on the market for Sifteo Cubes.

4 Use Cases

4.1 Load program

A. Name: Load program

B. Description: User selects program file and loads it into the emulator.

C. Actors: User

D. Basic flow:

1. User presses ”Load a program” button.
2. User selects *.siftem file in file dialog.
3. User presses ”Open” button.
4. Emulator loads program on cubes in emulator.

E. Alternate flows:

When the user opens an incompatible file (i.e. any file without the .siftem extension), or
When the user opens a corrupt or unloadable file, or
When the user presses ”Cancel” button:

1. An error dialog summarizing the issue is presented to the user.
2. The use case terminates and no program is loaded.

F. Pre-conditions:

1. Emulator is running.

G. Post-conditions:

1. Program is loaded or user cancelled loading.

H. Special requirements

1. Emulator should indicate when program is finished loading.

4.2 Reload program

A. Name: Reload program

B. Description: User reloads the current program file in the emulator.

C. Actors: User

D. Basic flow:

1. User presses ”Reload this program” button.
2. User presses ”Yes” on caution dialog.
3. Emulator loads program on cubes in emulator.

E. Alternate flows:
When User presses ”no” on caution dialog:

1. The use case terminates and the program is not reloaded.

F. Pre-conditions:

1. A program is loaded in the emulator.

G. Post-conditions:

1. Program is loaded.

H. Special requirements:

1. Emulator should indicate when program is finished loading.

4.3 Zoom screen

A. Name: Zoom screen

B. Description: User selects program file and loads it into the emulator.

C. Actors: User

D. Basic flow:

1. User adjusts zoom slider.
2. Emulator magnifies cubes in emulator according to zoom level.

E. Alternate flows:
None

F. Pre-conditions:

1. Emulator is running.

G. Post-conditions:

1. Program running at beginning of use case, if any, is still running.

4.4 Add/remove cubes

A. Name: Add/remove cubes

B. Description: User adjusts the number of cubes present in the emulator.

C. Actors: User

D. Basic flow:

1. User adjusts ”Number of cubes” slider or spinbox.
2. Emulator adds/removes cubes in emulator.

E. Alternate flows:
None

F. Pre-conditions:

1. Emulator is running.

G. Post-conditions:

1. Number of cubes has been adjusted to number specified.
2. Program running at beginning of use case, if any, is still running, but only on the cubes not newly added.

4.5 Snap cubes to grid

A. Name: Snap to grid

B. Description: User resets the cubes to their initial positions in the emulator.

C. Actors: User

D. Basic flow:

1. User presses ”Snap to Grid” button.
2. Emulator moves cubes to a grid orientation based on their current positions.

E. Alternate flows:
None

F. Pre-conditions:

1. Emulator is running.

G. Post-conditions:

1. Cubes are arranged in a grid.

4.6 Manipulate cube

A. Name: Manipulate cube

B. Description: User manipulates a cube by clicking buttons or the cube itself.

C. Actors: User

D. Basic flow:

1. User double clicks on cube.
2. Cube responds as if a screen click occured.

E. Alternate flows:

1. User clicks on buttons superimposed on cube edges.

2. Cube executes appropriate action (i.e. flips, rotates, tilts).

1. User drags cube next to another cube.
2. Cube communicates (”neighbors”) with the cube(s) it is adjacent to.

F. Pre-conditions:

1. Emulator is running.

G. Post-conditions:

1. If a program is running, the emulator has updated its state based on the cube’s change.

H. Special requirements: ???

5 Use Case Feature Mapping

6 Storyboards

Glossary

Application Programming Interface is an interface implemented by a software program that enables it to interact with other software. 3

Sifteo Cubes are small machines capable of loading programs and interacting with one another as well as responding to predefined movements. 3

References

1. Sifteo Inc. Online: http://www.sifteo.com

2. Tim Ekl. Client Meeting. 12 September 2011 12:45 p.m.

Index