Singularity Software

Milestone 1

September 14, 2011

Contents

1 Executive Summary

This is where the executive summary lives.

2 Introduction

This is where the introduction lives.

3 Client Background

Clients Tim Ekl and Eric Stokes are alumni of Rose-Hulman. Mr. Ekl is currently working on a M.S. degree in Engineering Management; Mr. Stokes is currently working for nãsk Signal Processing Systems in Denver, Colorado. As (former) Rose-Hulman students, the clients are avid users of technology who follow new trends in the industry. As a result of this interest in new technology, both Mr. Stokes and Mr. Ekl discovered and purchased Sifteo Cubes, a revolutionary product that consists of a set of (anywhere from 1 to 6) mini computers. While both clients have a set of 3 Sifteo Cubes, they realized that not everyone interested in the project could have the luxury of physical hardware to work with. As such, they asked Singularity Software via the Junior Project proposal process at Rose-Hulman to construct a software emulator for the cubes. The emulator is intended to make the development of Sifteo applications easier, especially in the testing phase.

4 Current System

Currently there is no way to simulate Sifteo Cubes. Mr. Ekl has developed a basic emulator in the past but asked that we not look at his code or in any way base the Siftables Emulator on his project.
Sifteo has announced that they will be releasing their API later this month; as of this printing, the API has not yet been made public.

5 User/Stakeholder Description

5.1 User/Stakeholder Profiles

5.1.1 Sifteo application developers

As the primary target audience of the system, developers targeting the Sifteo platform are assumed to have a reasonable amount of technical background; they are not novice computer users and are familiar with programming concepts like Object-Oriented Programming and Application Programming Interfaces. As developers may hail from platforms ranging from Windows to Mac to Linux, cross-platform support is an important consideration.

Possible problems for this user type include an unstable or crash-prone system; as developers are writing and testing their own code, it is essential that the emulator does not contribute to the failures that the user must debug. Developers will deem the emulator project a success when they can successfully program and test software that uses any or all of the features of the Sifteo cubes on their development platform.

5.1.2 Clients

The clients (Tim Ekl and Eric Stokes) are assumed to be a subset of the Sifteo application developers user class. However, their programming knowledge and knowledge of the Sifteo Cubes is known to be more advanced than that of the average application developer. As such, their needs require that the emulator is capable of being pushed to the same limits as the actual platform.

5.1.3 Singularity Software

Singularity Software, as the team behind Siftables Emulator, is primarily interested in the creation of a finished product that can be delivered to the clients at the conclusion of the Rose-Hulman junior project cycle. As a team, we are less familiar with the Sifteo platform and are also relatively inexperienced with the scale of project the Emulator entails.

5.1.4 Sriram Mohan (CSSE Department)

As the advisor of the Junior Project, Dr. Mohan has a vested interest in the creation of a finished, deliverable product. His key responsibility is to review documents created within the scope of the Junior Project series of courses.

5.2 User Environments

5.2.1 Sifteo application developers

The typical Sifteo application developer may be working on his own or he may be working with a team of developers; he or they will be working on workstations or powerful development laptops that have a significant amount of graphics horsepower. They may or may not be connected to the Internet during development, depending on the location in which they are developing. They may be Windows, Mac, or Linux users and will be using various Integrated Development Environments (IDEs) specific to their platform; integration between such IDEs and the Siftables Emulator, while possibly desirable, is not a requirement.

5.2.2 Clients

Mr. Stokes and Mr. Ekl are both primarily Mac users, although both clients also own and occasionally use Windows machines as well. Their environment is essentially the same as that of the typical Sifteo application developer.

5.3 Key Needs

5.3.1 Emulate Sifteo Cubes in a desktop GUI application

No emulator currently exists for the Sifteo platform; the need is currently filled by homebrew efforts like Mr. Ekls Java-based emulator, or developers simply use the Cubes themselves as a test environment. The clients envision a solution where all of the interactions possible with a set of Early Access Sifteo Cubes can be replicated in a software emulator.

5.3.2 Develop an API for creating applications in the emulators Cubes

An Application Programming Interface (API) is necessary to facilitate interaction with the virtual Sifteo Cubes. Currently, no API is made available by Sifteo for the physical cubes, and no emulator API exists because no emulator exists. The clients would like an API (that may or may not resemble the to-be-released Sifteo API) with which the virtual Cubes can be programmed.

5.3.3 Provide several example games that showcase Cube/emulator functionalities

Sifteo currently provides example games that run on the Cubes as a showcase of what can be achieved with the Cubes and what the Cubes unique futures offer to the programmer and to the user. The clients would like to have a similar showcase available for the emulator; such examples would aid in understanding both the emulator platform and the larger Sifteo Cubes programming platform.

5.4 Alternatives & Competition

Singularity Softwares Siftables Emulator will be the first software of its kind for the Cubes. The primary competition, then, is the Sifteo Cubes themselves. The Cubes have the advantage of physicality as tactile objects, they will always be superior in terms of user experience when compared to an emulator. However, they are expensive and only manufactured in limited quantities at the moment; the Siftables Emulator is, by contrast, infinitely available as an open source piece of software.

6 Product Overview

6.1 Product perspective

Siftables Emulator is a free independent system used to emulate the way Sifteo Cubes handle motions and interactions.

6.2 Elevator statement

Due to the limited availability of Sifteo Cubes, developers unable to obtain a set of Cubes have no good way to test the programs they create for the platform. At Singularity Software, our goal is to develop an emulator for the Cubes. The emulator, which will consist of the emulator platform, an API, and example games, will be able to emulate an arbitrary number of Sifteo Cubes and the way they handle physical motions and interactions.

6.3 Summary of capabilities

The main features of our emulator allow developers to quickly start Sifteo application development by making a virtual edition of the Cubes available for emulation and testing.

Feature	**Benefit**
Workspace where multiple cubes can be emulated	A user-friendly way to develop for multiple cubes
Buttons to control the cubes	An easier way to control the basic movements of the cubes in place of physical manipulations
Ability to load programs into the cubes	Makes the emulator functional for other than basic arrangement of cubes
Example games (requirement)	Gets new emulator users started with the platform
Open source (requirement)	Allows the community to contribute improvements to the emulator
API (requirement)	A standard way of interacting with the virtual Cubes

6.4 Assumptions and dependencies

Sifteo has plans to release an API of their own for the Cubes; Singularity will attempt to make our API shadow much of the language and functionality of the official Sifteo API.

6.5 Estimate of cost

7 Features

Two attributes accompany each feature described below: **Status**, which is a measure of the feature's progress duing the project definition period, and **Priority**, which indicates the relative importance of each feature.

Feature	Description	Status	Priority
An individual, manipulable Cube	The user will be able to manipulate a virtual Sifteo Cube with his mouse	Approved	Critical
A workspace with multiple Cubes	The user will be able to place a number (between 1 to 6) of Cubes on the workspace and manipulate them	Approved	Critical
Buttons to control the cubes	The user will be able to click buttons that correspond to each manipulation possible with a physical Cube	Approved	Critical
Cubes interact with each other	Each Cube on the workspace will interact and communicate with neighboring Cubes	Approved	Critical
The ability to load programs into the cubes	The user will be able to load a program of his creation onto the virtual Cubes in the workspace	Approved	Critical
Auto-align	The user can activate a mechanism that aligns all Cubes on the workspace to a grid or other order	Proposed	Useful
Zoom ability	The user will be able to zoom the workspace to focus on an individual cube, a group of cubes, or the whole space	Proposed	Useful

8 Constraints

While much of this project is open-ended, there are a few basic constraints. At the direction of the clients, all code should be open source and version-controlled. Mr. Ekl requested that the emulator run easily on Mac as well as Windows, with the stipulation that Linux compatibility would satisfy the Mac requirement for Singularity's testing purposes. In addition, the clients requested that an issue tracking system be put in place and used throughout the development process. Finally, the emulator must be completed by the end of the year to satisfy the requirements of the clients and of Dr. Mohan.