

NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

IT2901 - PROJECT 2

Sifteo cubes, and it's usage for the Don't Panic board game

Authors:

Jens Even Blomsøy - jenseven@stud.ntnu.no
Stian Aurheim - aurheim@stud.ntnu.no
Jørgen Foss Eri - jorgeer@stud.ntnu.no
Sindre Svendsrud - indrsv@stud.ntnu.no
Adrian Arne Skogvold - adriansk@stud.ntnu.no
Jim Frode Hoff - jimfrode@stud.ntnu.no

Customer:

Ines Di Loreto - inesd@idi.ntnu.no

Supervisor:

Mohsen Anvaari - mohsena@idi.ntnu.no

May 6, 2013

Contents

1	Sifteo cubes	3
1.1	Usage for Don't Panic	3
1.2	Implementation	3
1.3	Recommended work	3
	Appendices	5
	Appendix A Description of the Sifteo Cubes	5
A.1	How do Sifteo Cubes work?	5
A.2	Who are Sifteo Cubes for?	5
A.3	How do I play games on my cubes?	5

1 Sifteo cubes

In this section, we will present the possibilities for using Sifteo Cubes as a client for the Don't Panic server.

1.1 Usage for Don't Panic

As an effort to making the game more engaging, the gestures of navigating the Sifteo cubes will give the players a wider experience. We discussed with the customer about how the cubes could be used in order to make a new client. By making a physical board with each node as a custom socket, the cubes can represent players. This way, there are more possibilities of displaying the data. For instance, you could limit information by only showing the number of people of the adjacent nodes. By doing this, you force the players to move around just to find out what zones are in panic.

For some of the other functionality, we came up with some ideas that we found feasible. For moving people from one zone to another, it could be possible to have a socket or sensor in each node, and by placing a legal cube in a node, you would begin moving people "into" your cube incrementally. When you have the desired amount of people with you, the player would move the cube to the new zone, drop off the people, and then return to its original node.

Since the server only needs to be able to send and receive JSON, we are able to make a client based on any language or any component as long as it is able to read and write JSON.

1.2 Implementation

In order for the cubes to communicate with the Node.js server, it will be necessary to make a program translating the Sifteo events into JSON, and also translate JSON from the server to the cubes to display. We will call this program *the middleman*. Since the cubes IDI got are of the first generation, the middleman needs to interpret the events from the cubes with the C# API before sending JSON to the server. The middleman must also be able to receive response from the server, and update the cubes accordingly. This could be achieved with some event-driven code.

1.3 Recommended work

We researched what was already available for communication between the cubes and a PC. As it stands at time of writing, there is no support for doing this in the official API that Sifteo provides. The communication between USB and the cubes are strictly limited to running the game on the PC, and sending display data for the screens. November 21, 2012, it was posted a feedback on the official developer boards with the title "Allow I/O through USB connection", where Sifteo said that runtime I/O requires updates in the cubes firmware.¹

However, there have been examples of people going into the code Sifteo made for their first generation cubes, and intercepting the wireless data that is not available in the official API.² We have not been successful in acquiring any frameworks that does the job for us, so in order to make the Sifteo cubes able to communicate with the Don't Panic server, somebody needs to dive into the Sifteo code and find it themselves. This is not a limitation by the server, but the API of the Sifteo cubes

¹<http://support.sifteo.com/entries/22448082-Allow-I-O-through-USB-connection>

²<http://thenxtstep.blogspot.no/2012/09/nxt-and-sifteo-cubes.html>

not including runtime I/O to PC. During time of writing, the site is down for maintenance, but the developer driven wiki for the first generation cubes might prove to be a good resource.³

It is difficult to estimate how long this will take to develop, seeing that we have not found a way for the cubes to communicate I/O in runtime. The time span required is dependant on prior knowledge to C# and wireless protocols, and how the events from the cubes are structured. We do not feel we have the competence to estimate the time required to make *the middleman* mentioned earlier.

³<http://wiki.sifteocentral.com>

Appendix A Description of the Sifteo Cubes

Taken from the Sifteo FAQ

A.1 How do Sifteo Cubes work?

Sifteo Cubes is a hands-on interactive game system. They communicate wirelessly to respond to each other and being tilted, shaken, flipped, and pressed. The Sifteo Base stores your games and plays audio. You can play with 3-12 Sifteo cubes at once!

Sifteo Cubes use 1 AAA battery each. Sifteo Bases use 2 AAA batteries. Rechargeable nickel metal hydride (NiMH) batteries, standard alkaline, or lithium (non-rechargeable) batteries will each work. Batteries last for approximately 8 hours of gameplay.

A.2 Who are Sifteo Cubes for?

Sifteo Cubes are perfect for kids and kids at heart who geek out on brainy challenging games and cool new gadgets (recommended for ages 7 to adult). Our games can be played individually and with friends and family. Sifteo Cubes are sturdy and can withstand some abuse. Still, they are sophisticated electronic devices, and very rough handling (throwing, dunking in water, and other forms of cube torture) can break a cube permanently.

A.3 How do I play games on my cubes?

Sifteo Cubes come pre-installed with 4 games that work right out of the box. You can buy more games using the Sifteo desktop software ([download here](#)) and install them by connecting the Sifteo Base to your computer with the USB cable that comes with it. Sifteo games are 80-120 credits (about 8–12) each. You can buy credits through the desktop software. View full tech specs and system requirements [here](#) .

List of Figures

List of Tables