

WakeUp

"Changing the way people wake up, one alarm at a time."

Organization of this document:

1. Problem and motivation
2. Solution
3. Prototype status
4. Fiware Stack in WakeUp
5. Installation and Developer API
6. Future plans
7. Video
8. Team

1. Problem and motivation

Getting up early sucks. Everyone hates that moment when the alarm fires off and you have to get out bed in order to start your day.

Most people use the same alarm sound for everyday, without considering things like the current weather conditions, how busy their calendar is or whether you'll experience light or heavy traffic while driving to the office, so we can pretty much say that current alarm clocks are dumb and do not follow the current pace of progress other aspects of technology are following.

We also did some research on the matter and found some interesting studies, such [this paper](#) by Raymond W. Lim and Michael S. Wogalter from the Department of Psychology of the North Carolina State University. In which they present the current status of alarm clocks, what is wrong with them and how we can solve it.

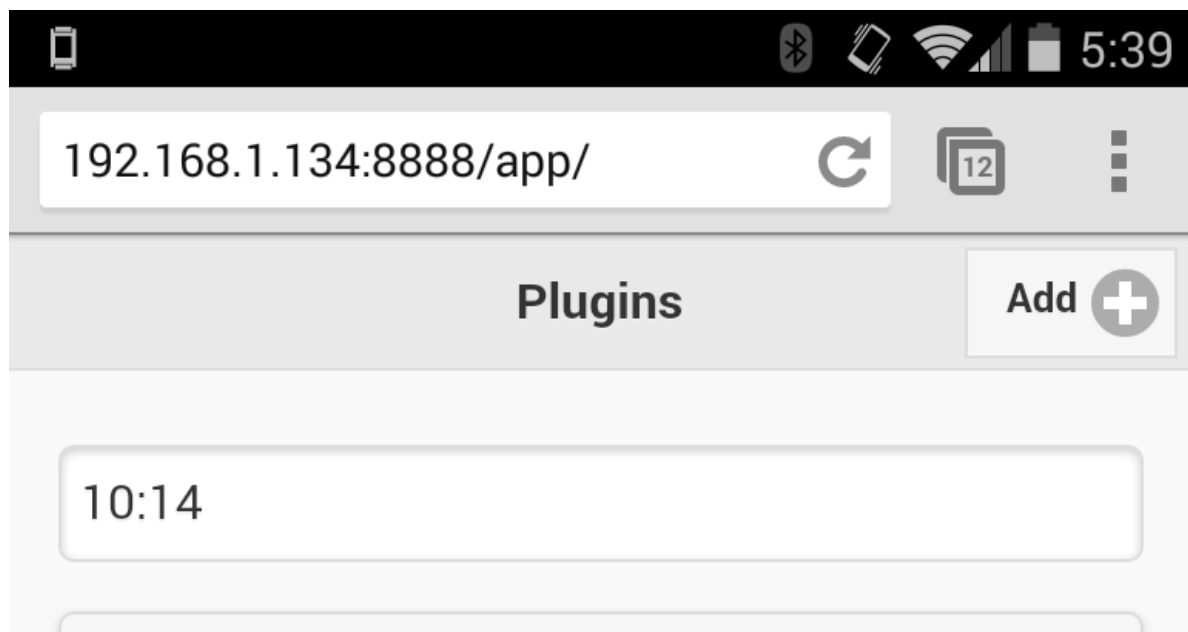
Finally, we realized that this a moment of the day in which you usually do nothing because you are still sleepy and it's uncomfortable to read. So, the best way of getting information while you wake up is by hearing it.

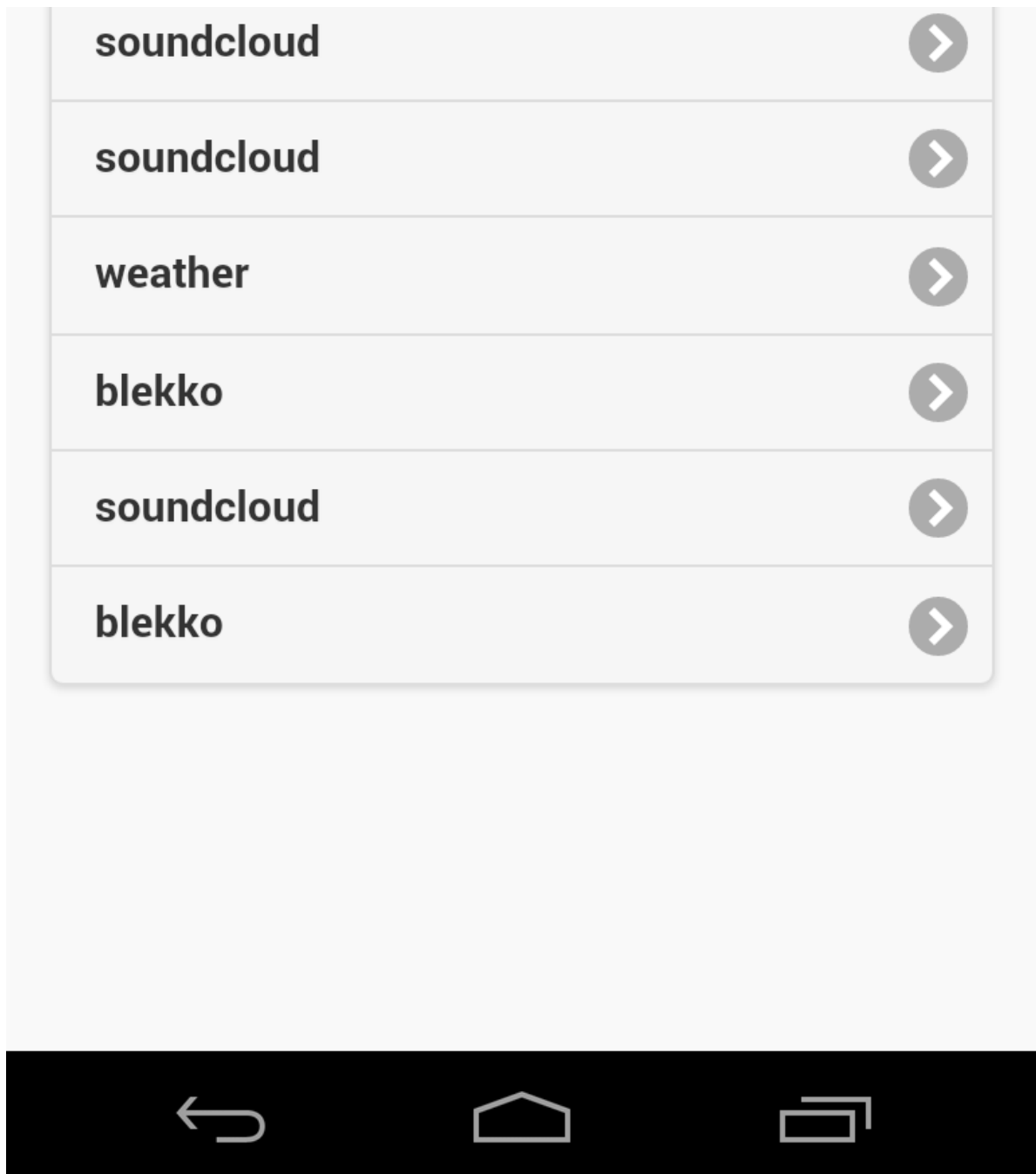
Given these three facts, we observed that current alarm clocks are outdated and need a major rethink and redesign.

2. Solution

We wanted to create a new concept of alarm clock. So the first decision we made is to let the user decide how they wanted to be woken up, and the second decision was to build a product separated from the smartphone.

In order to do this, we realized that we needed to allow developers create their own modules to wake up. We provide modules for waking up with News, weather and music, and an open platform for creating new ones (explained further in point 5). We are also open source and (if we get to build it) open hardware. The user can choose how they can be waken up by choosing among their installed modules and set the hour they want to wake up.





We are building a product separated from the smartphone to make the alarm bullet proof. If you use your phone as alarm clock, if the battery dies during the night, you won't be woken up.

3. Prototype status

Given the Fiware Hackathon, we saw it as a good opportunity to make an early prototype of WakeUp and present it in society to get feedback. Right now, to make prototyping quicker, we are using the RaspberryPi platform to build the hardware prototype. We attached a pair of speakers to it and an ethernet cable that at the moment has already been replaced by a wifi connection.

Right now you can set the alarm using the webapp and choose and order the modules with which you want to be woken up. You can choose to hear the weather, then your favorite song and then some news related to your stock market investments. It is totally customizable, you even can configure each module with your preferred settings, like the number of news you want to hear or the topic.



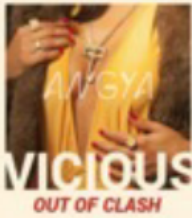
192.168.1.134:8888/app/#plugin



Back

soundcloud

angya mashup



Ibranovski vs Bruno M...





We have created a strong and reliable core, that allows us and other developers to create modules on top of WakeUp and create new experiences.

At the moment WakeUp uses the following technologies:

- NodeJS as the core platform, given its versatility and ease to create new modules for WakeUp.
- LevelDB as database, this key-value database allows us to have a schemaless database that support the different modules configuration.
- PicoTTSLibrary, as the Text to Speech library we use to read text through the speakers. It is the same library Android uses to read text.

4. Fiware Stack in WakeUp

We plan to add WakeUp some intelligence using FiWare Big Data solutions. Our idea is to use the algorithm we used in [Smads](#) to check the weather conditions or the traffic, so we can try to predict the animic state of the individual and customize their alarm to fit their animic state, and make him happier.

If we get selected to the next round, we use Campus Party week to discuss with Fiware engineers how can we get this done, due that this AI features are difficult to implement.

We are also developing a server that will be hosted in FiLab Cloud to make all the text to audio conversions, they are a bit slow in the RaspberryPi and we could use Fiware's computing power to get this done in a more efficient way. We are thinking of also using some kind of caching so we don't have to process the same sentence twice.

5. Installation and Developer API

The source code of the project can be found on [Github](#)

Installation

We assume you are already running nodeJS v0.10.x or greater and use Raspbian as OS.

To install WakeUp in your RaspberryPi you need to install first PicoTTS.

```
$ sudo nano /etc/apt/sources.list
// Add this lines at the bottom
deb http://archive.raspbian.org/raspbian wheezy main contrib non-free rpi
deb-src http://archive.raspbian.org/raspbian wheezy main contrib non-free rpi
// Ctr-X

$ apt-get update
$ apt-get install fakeroot
$ apt-get build-dep libtts-pico-utils
$ cd
$ mkdir my_build/
$ cd /my_build/
$ apt-get source libtts-pico-utils
$ ls
$ cd svox-1.0-<version>/
$ dpkg-buildpackage -rfakeroot -us -uc

$ cd my_build

$ sudo dpkg -i libtts-pico0..armhf.deb
$ sudo dpkg -i libtts-pico-data..all.deb
$ sudo dpkg -i libtts-pico-utils...armhf.deb
```

Then we will download WakeUp and install its dependencies:

```
$ git clone https://github.com/CampusHacks/WakeUp.git
$ cd WakeUp/
$ npm install
```

And run: `$ node index.js`

Developer API

It is very easy to create a module for WakeUp. You can start by copying one of the existent in `plugins/` and writing your own code. A module has to export a function that will get the `attributes` saved in the database and has to call a `callback` when it finishes to continue. You can use the library in `../../core/helpers/tts` to translate text to voice, and whenever we outsource it to Fiware, your module will update itself.

We will improve the Developer API documentation.

6. Future plans

If we get a prize in the Hackathon we would love to make the product, WakeUp, a reality. We'd love to be able to finance the first batch of devices and sell them. Given the cheap components we use, we could sell it for less than \$50, so it could be a mass product.

We think it is a product people will love and use every day. We would love to get feedback in the Hackathon and make it a reality.

7. Video

We made a short video to try to show how it works and how the prototype looks like.

It can be found in [Youtube](#)

8. Team

WakeUp is made by part of the team that built Smads, the winner of the first Fiware Hackathon in Campus Party London.

- **Alejandro Perezpayá (@alexdev_)**: He's in charge of the backend. He's got a lot of experience working with Fiware technologies. He also won the 3rd price in the Fiware Hackathon in Santander.
- **Jorge Izquierdo (@izqui9)**: He does iOS and Web Development. He attended to ICT 2013 in Lithuania with Alejandro to present their Fiware project, Smads, in front of the highest European personalities.
- **Luis Iván Cuende (@licuende)**: He's the founder of Asturix, a Linux distro based on Ubuntu. He is very good at Web and Low Level System development. He is 18, so he will be the representant of the team.