



INTRODUCTION TO WEARABLE DEVELOPMENT WITH PEBBLE

THOMAS SARLANDIE
MARCH 5, 2014 - WEARABLES DEV CON



ABOUT ME



@sarfata

French, Mobile developer (in a past life) and now Pebble developer evangelist

LOGISTICS

10:45

11:30

12:00

12:45

Discover Pebble

SimplyJS

H

Pebble C SDK

iOS

Download the official Pebble app on the AppStore



Download the latest Beta app
<http://pbl.io/android>

My goal today: **Turn all of you into wearable developers!**

PLEASE DO STOP ME FOR QUESTIONS!

A BRIEF HISTORY OF PEBBLE - A WEARABLE PROJECT

THE EARLY DAYS



IN-PULSE

Notifications
Battery life
Apps
Blackberry only





KICKSTARTER

Pebble: E-Paper Watch for iPhone and Android

A Product Design project in Palo Alto, CA by Pebble Technology · [send message](#)

PROJECT HOME UPDATES 13 BACKERS 68,929 COMMENTS 6,907



68,929
BACKERS
\$10,266,845
PLEDGED OF \$100,000 GOAL
0
SECONDS TO GO

FUNDING SUCCESSFUL
This project successfully raised its funding goal 6 days ago.

PLEDGE \$1 OR MORE
2615 BACKERS



Four hardware buttons

Screen 144x168 B&W pixels

3 axis accelerometer

Vibration

Backlight

Back

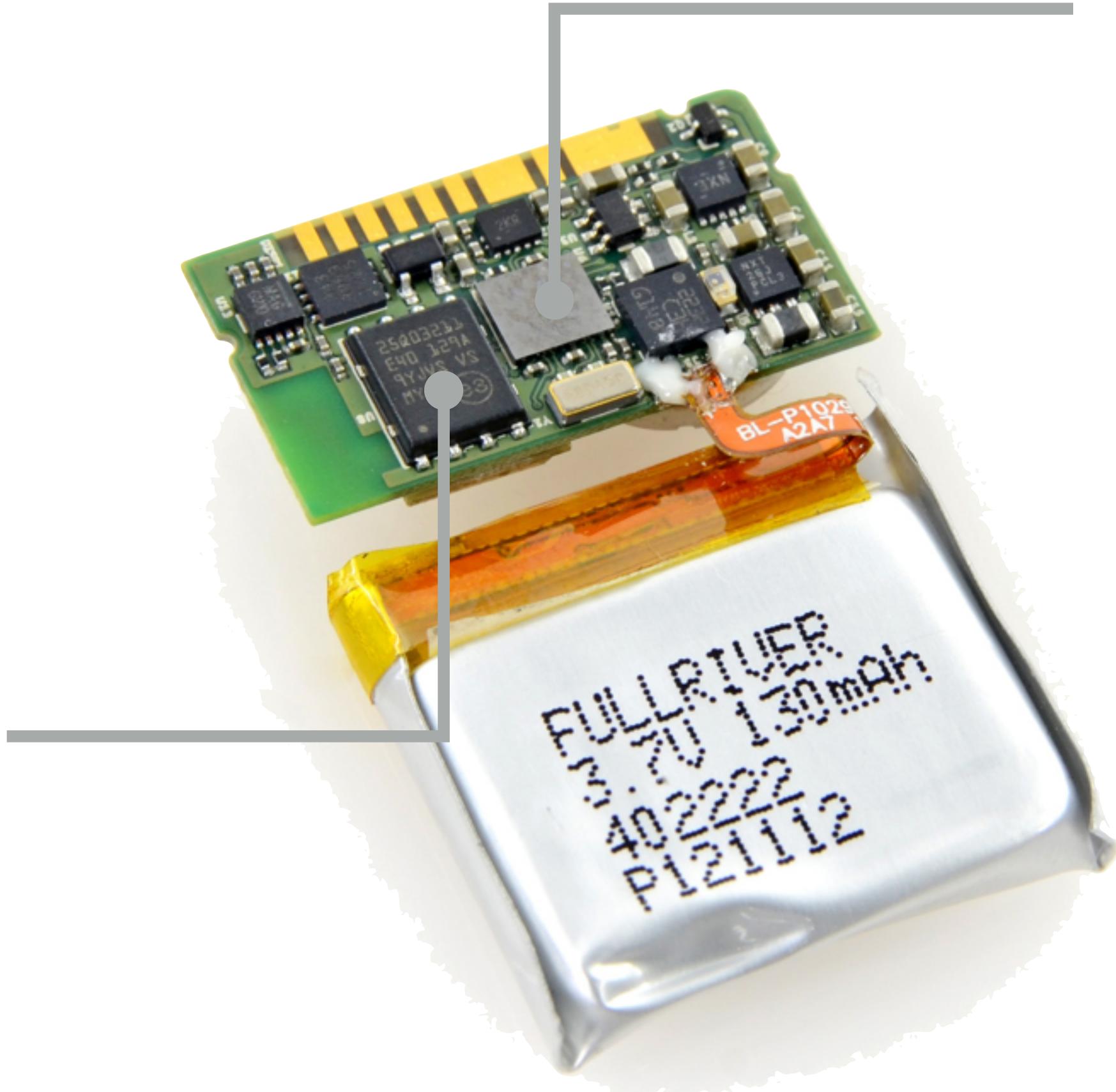


Up

Select

Down

serial flash
4 megabytes



processor
80MHz
ARM Cortex-M3 core
512 kilobytes flash
128 kilobytes RAM

bluetooth radio
(hidden on back)
Bluetooth 2 + BLE

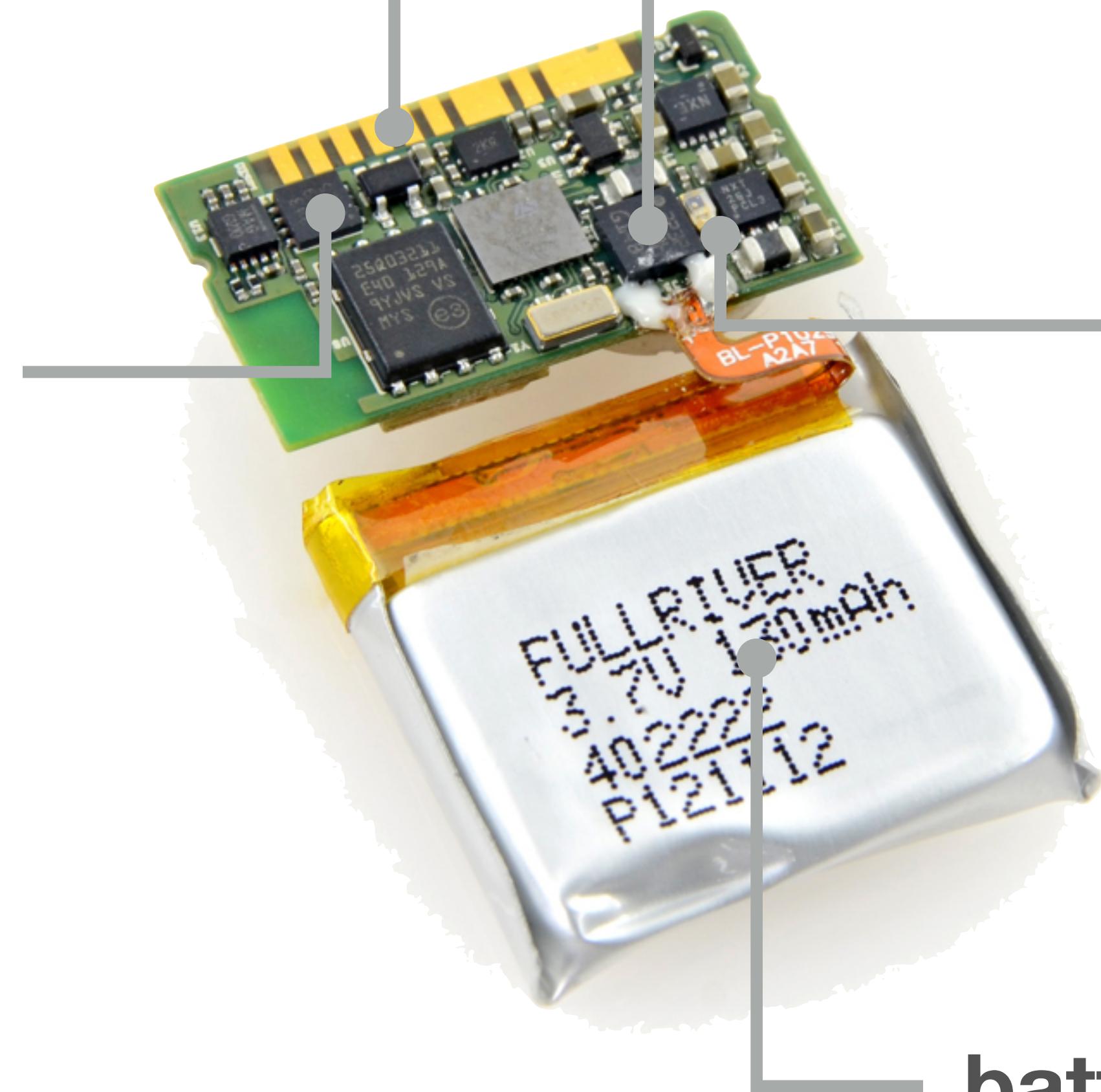
display connector

magnetometer

accelerometer

**ambient light
sensor**

battery





Pebble Steel

DISCOVER PEBBLE



Music



Alarm



Phone



Messages



Email

PEBBLE COMES WITH SEVERAL BUILT-IN APPS



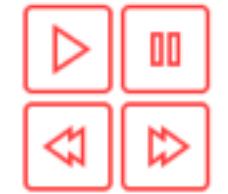
PEBBLE IS CUSTOMIZABLE - WITH WATCH FACES



DAILY

Apps in daily provide quick access to stocks, weather, news, and other timely data Pebblers check everyday.

Check Traffic, Headlines, Financial info, Weather...



REMOTES

Does your app control other products like a camera, thermostat, or automobile right from Pebble? It's a remote app.

Control music, Home appliances, Car door locks, Cameras...



FITNESS

Cyclists, swimmers, and athletes of all types use these apps to support their active lifestyles & fitness goals.

Count swim laps, Monitor calories, Guided workouts...



TOOLS & UTILITIES

Need an app to count, measure or calculate something? Or quickly find your parked car or your ride home? It's a tool.

Convert currency, Calculate, Learn, Find local services...



NOTIFICATIONS

A Pebble notifications app helps users get customized notifications from their favorite apps & web services right to Pebble.

Social media, Travel, Auctions, Events, Calendars...

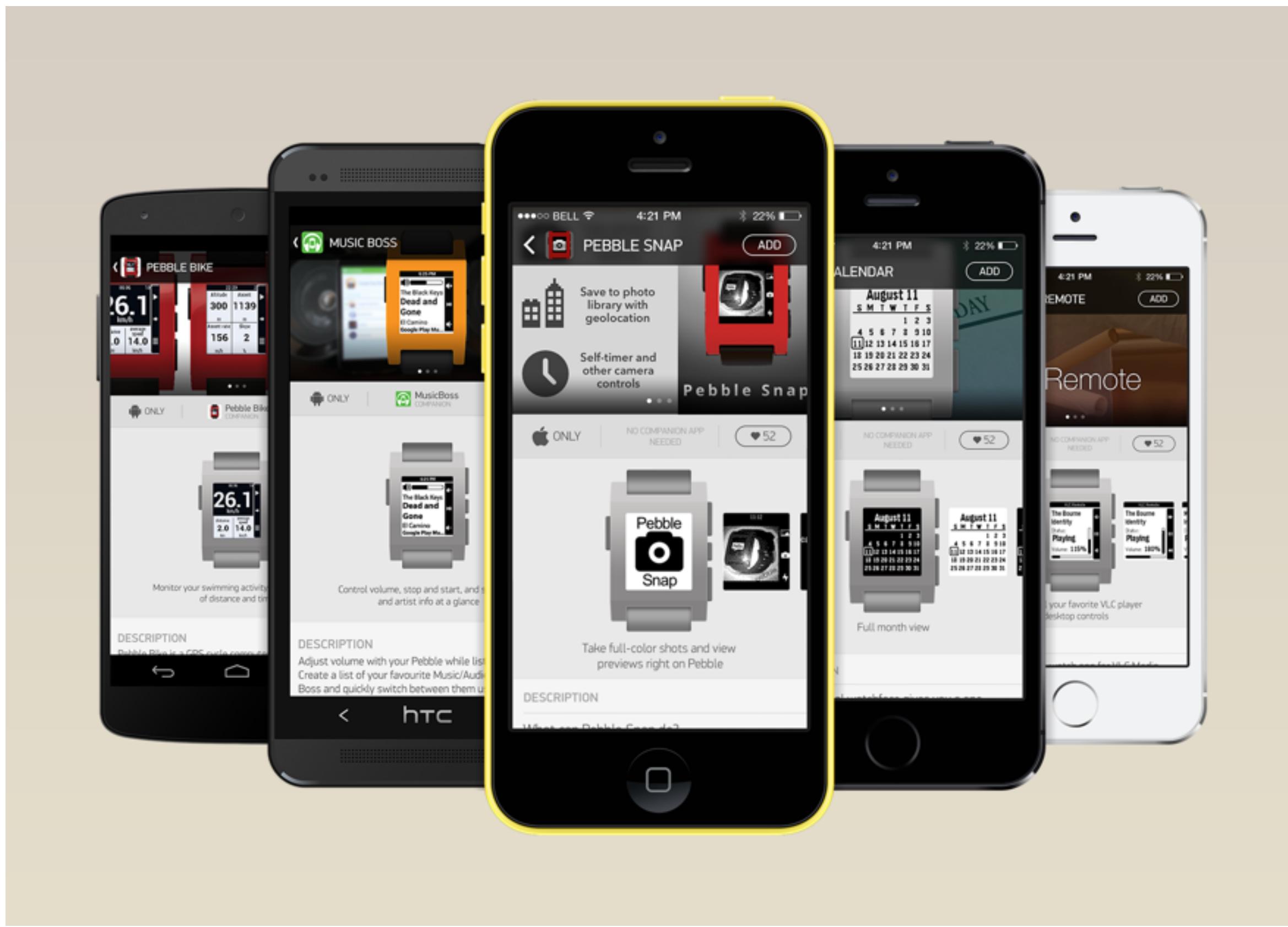


GAMES

Play On

Classic games, Puzzles, Turn-based, Action...

AND APPS



AVAILABLE ON THE PEBBLE APP STORE

SOME POPULAR APPS ON PEBBLE TODAY



POWERED BY
ESPN

Notifications

Glance

Get information and SMS messaging on your Pebble



Contextual Information



Contextual Information

Pebble Canvas

Design a watchface on your phone



Customize Pebble

Sleep as Android

Track your sleep to find the optimal moment for your wake up



Fitness and Activity Monitoring

PebbleCam

Preview and remote control your iPhone's Camera



Remote Control



Remote Control



DREAM TEAM

Remote Control

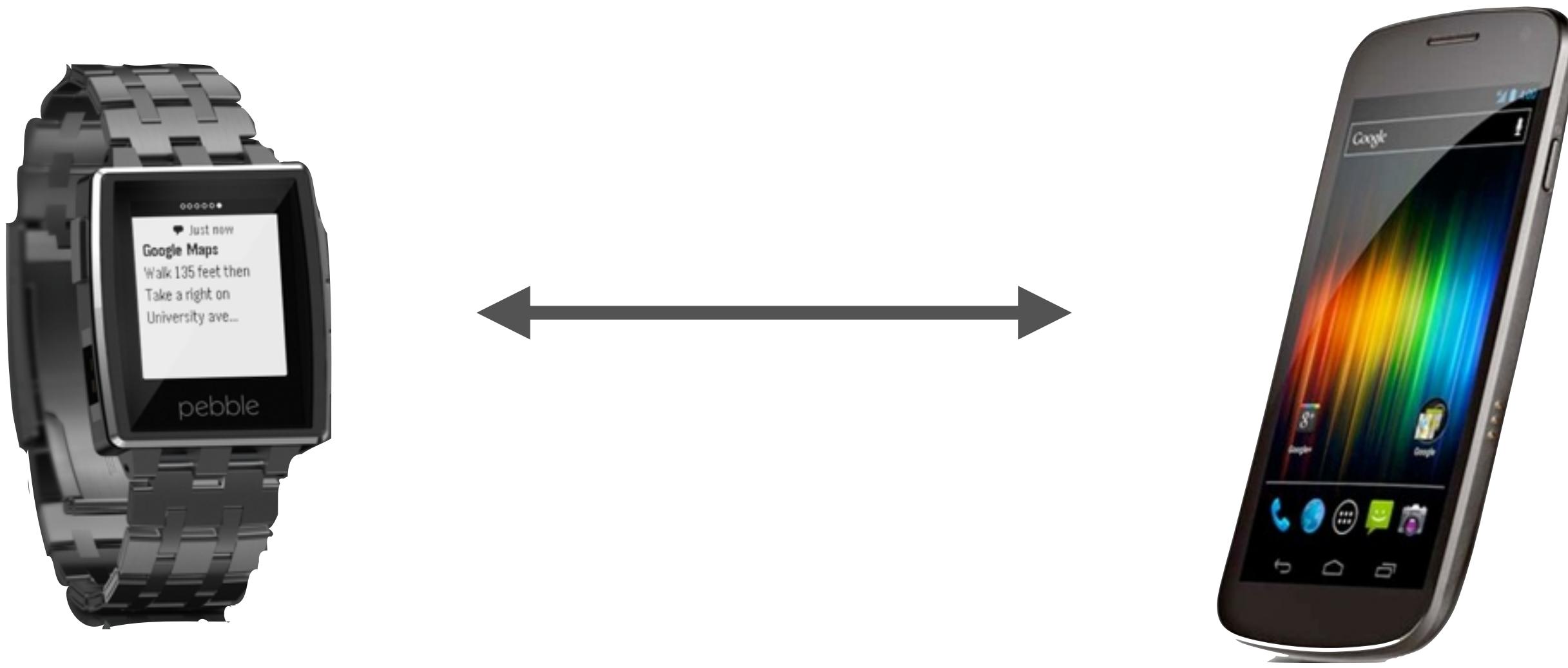
DEMO: INSTALLING AN APPLICATION FROM THE PEBBLE APPSTORE

PEBBLE AS A PLATFORM



C SDK

Developers can write applications that run on Pebble with the native C SDK



C SDK

JavaScript

Pebble applications can embed JavaScript that is executed on the phone.

Apps can show a configuration window on the phone, store settings, use the GPS, ...



C SDK



JavaScript



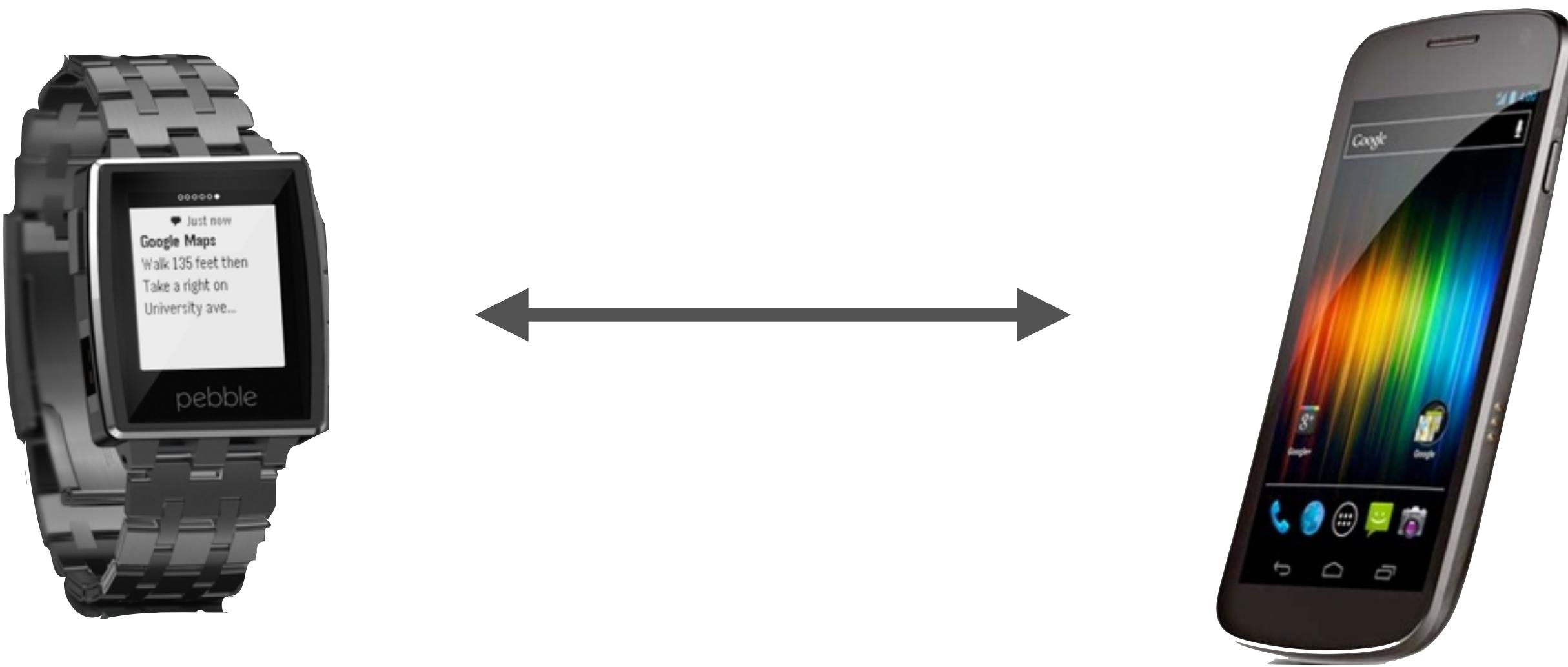
Internet

Through the phone, Pebble app have access to the Internet.

pebble
+  **RunKeeper**



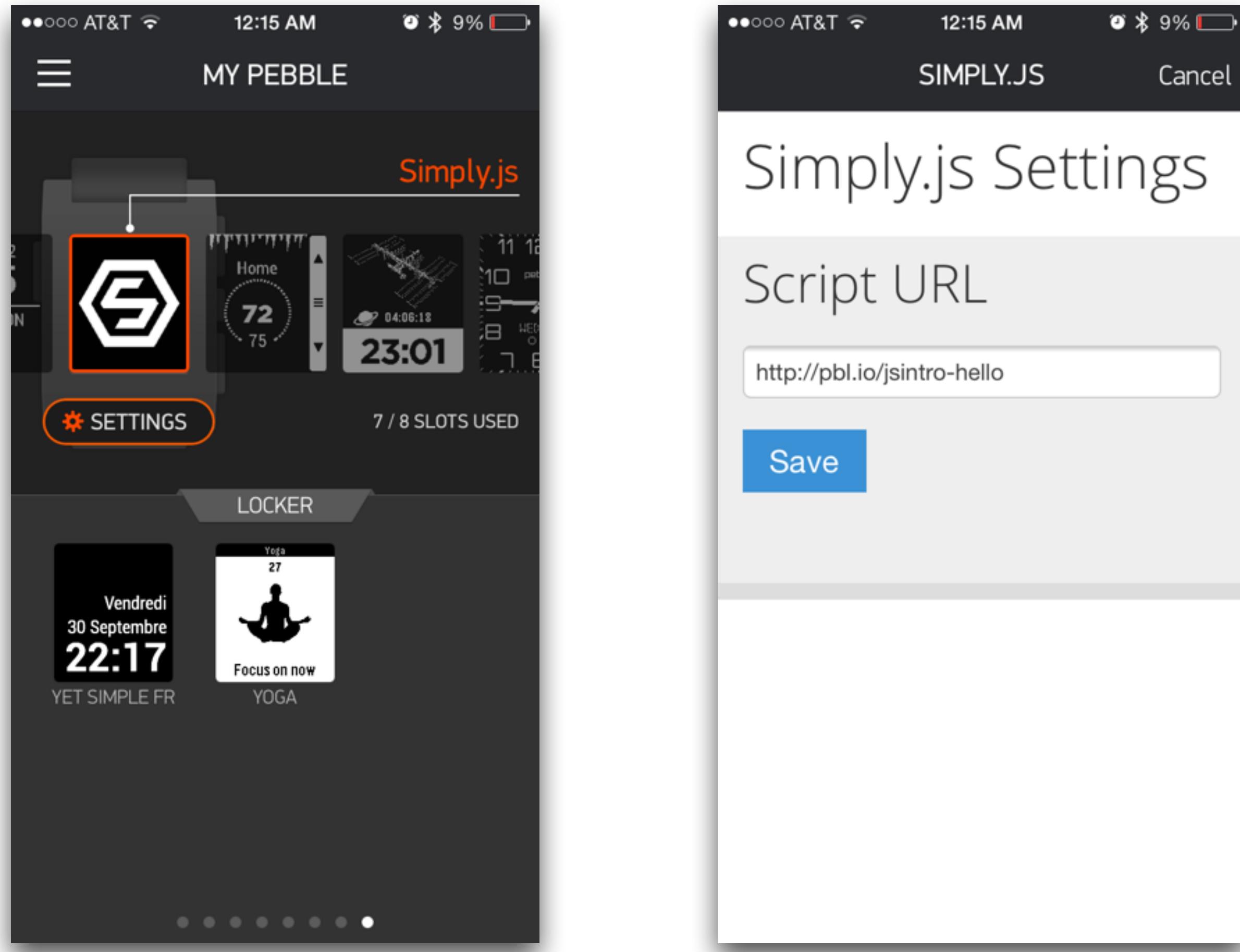
Extend existing app with PebbleKit iOS and PebbleKit Android



SimplyJS is a JavaScript framework to write Pebble applications. The JavaScript is running on the phone and remote-controls the Pebble.

<http://simplyjs.io>

GETTING STARTED WITH SIMPLY JS



1. **Download Simply JS** from the Pebble appstore
2. Configure Simply JS to load **pbl.io/jsintro-hello**
3. **Run** your first SimplyJS app!

SIMPLY.JS: HELLO WORLD



pbl.io/jsintro-hello



SIMPLY.JS: BUILDING A CLOCK

pbl.io/jsintro-clock

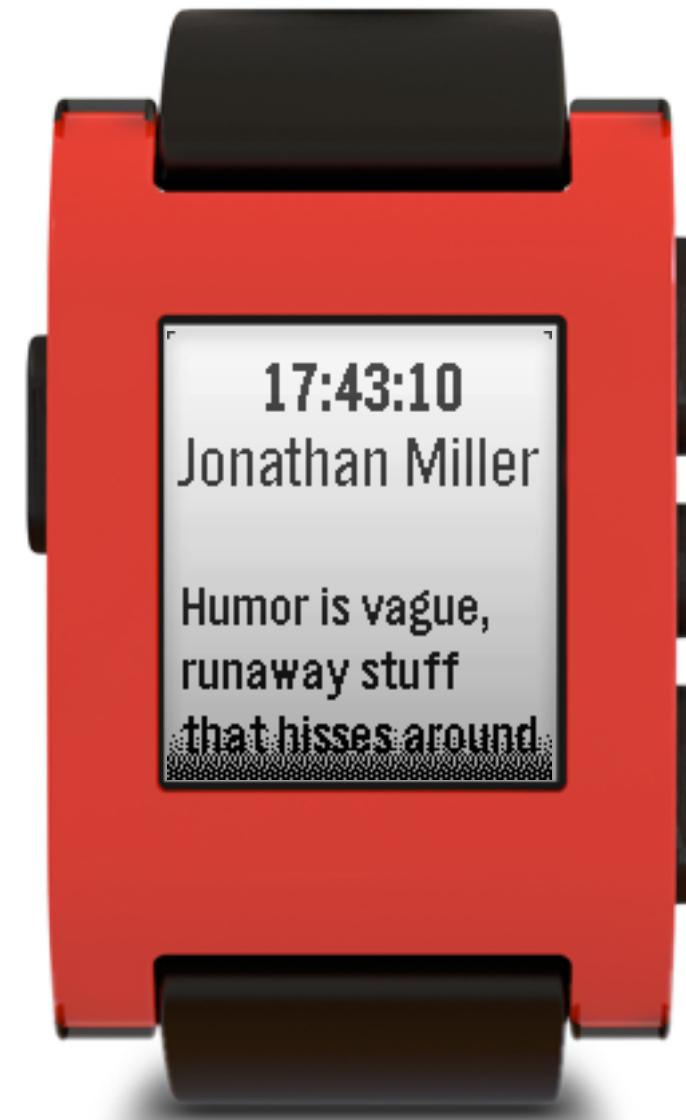
```
function displayTime() {  
    var now = new Date();  
    simply.text({  
        title: ""  
        + now.getHours() + ":"  
        + now.getMinutes()  
        + ":" + now.getSeconds()  
    }, true);  
}  
  
simply.fullscreen(true);  
displayTime();  
setInterval(displayTime, 1000);
```



SIMPLY.JS: ADDING SOME CONTENT

pbl.io/jsintro-quote

```
// Fetch a quote
ajax({
  url: 'http://api.theysaidso.com/qod.json',
  type: 'json'
},
function(quote) {
  simply.text({
    subtitle: quote.contents.author,
    body: quote.contents.quote });
}
);
simply.scrollable(true);
```



SIMPLY.JS - APIs OVERVIEW

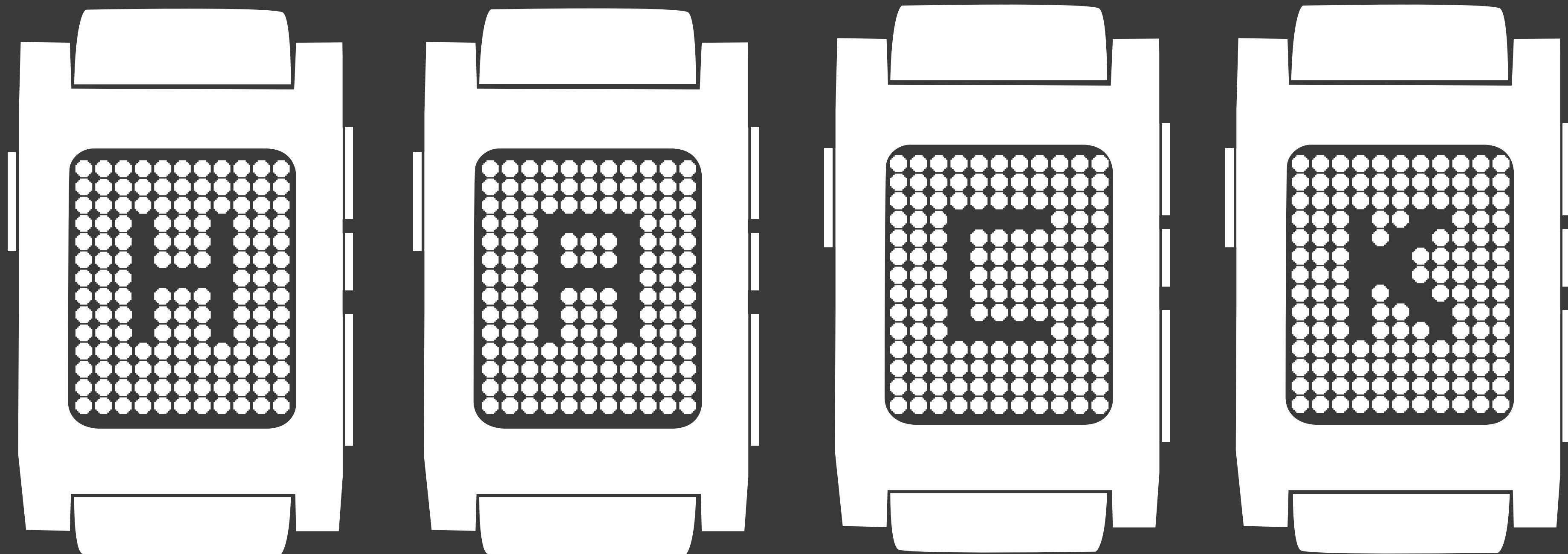
Display text on screen. Support three different styles and you can make the text scrollable.

Button input: Send an http request every time a button is pressed.

GPS and Accelerometer are supported too.

SimplyJS is great to quickly connect Pebble to the Internet!

<http://simplyjs.io>



Tip: Use dropbox to save a JS file and quickly share it.
rawgithub.com is another great option.

GETTING STARTED WITH PEBBLE NATIVE SDK

USING PEBBLE NATIVE SDK

Develop for Pebble. It's an open platform.

DELIGHT USERS WITH YOUR APP
Pebble applications run on your wristwatch and are always a glance away.

FITNESS, REMOTE CONTROL OR NOTIFICATIONS
Pebble provides a rich set of APIs to seamlessly integrate into everyday life

[DISCOVER PEBBLE SDK](#) [INSTALL PEBBLE SDK](#)

CloudPebble

Online Pebble Development
Write apps without dealing with Linux, virtual machines, compilers, or Python.

[Get started](#)

Nothing to install
Write apps and watchfaces directly in your browser. Skip all the setup instructions!

Cloudy
It's on the internet! Accessible from anywhere! That's

No JSON to edit
Resources are managed through a soon-to-be friendly user interface; no need to deal with `resource_map.json`.

Mostly unfinished

Download and install the SDK



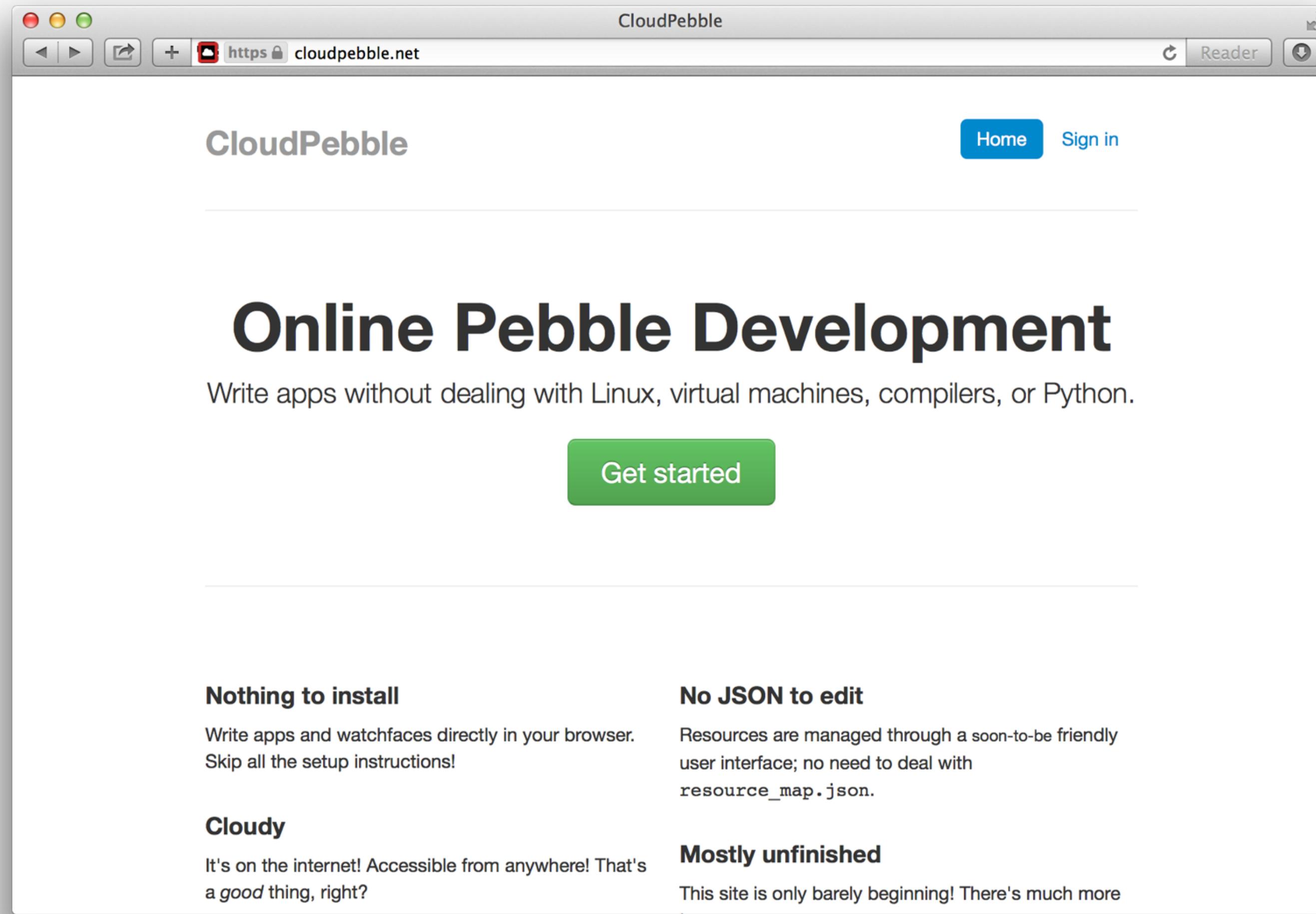
Use CloudPebble



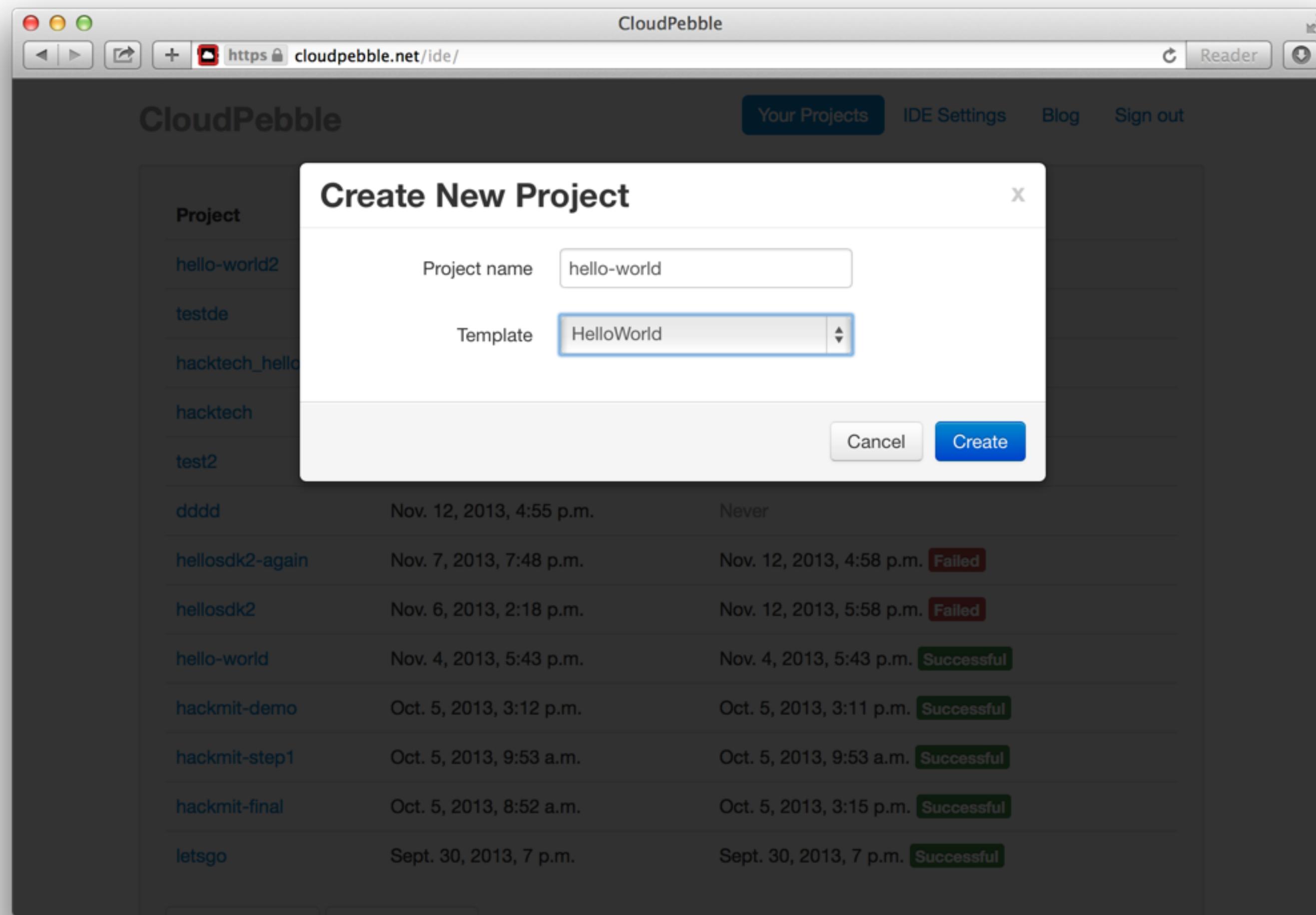
WALKTHROUGH - BUILD AND INSTALL YOUR FIRST NATIVE APP

1. Log into **Cloud Pebble**
2. **Create** a new project with the ‘helloworld’ template
3. **Build** the project in the “Build tab”
4. Enable the **developer connection** on your phone
5. **Install and run** your first application

LOG INTO CLOUDPEBBLE.NET



CREATE A NEW PEBBLE PROJECT



BUILD YOUR FIRST PROJECT

The screenshot shows the CloudPebble IDE interface for the 'hello-world' project. The left sidebar contains navigation links: PROJECT, Settings, Compilation (which is selected), GitHub, SOURCE FILES (with 'hello_world.c'), New C file, New JavaScript file, RESOURCES, and New resource. The main content area displays the 'Last build' details:

- Started: February 25, 2014 at 8:46:50 PM PST
- Build time: 2 seconds
- Status: Succeeded
- Size: 6 KiB (4 KiB resources, 1 KiB binary)
- Available RAM: 23908 / 24576 bytes (97%)
- [Download compiled PBW \(get short link\)](#)
- [Build log](#)

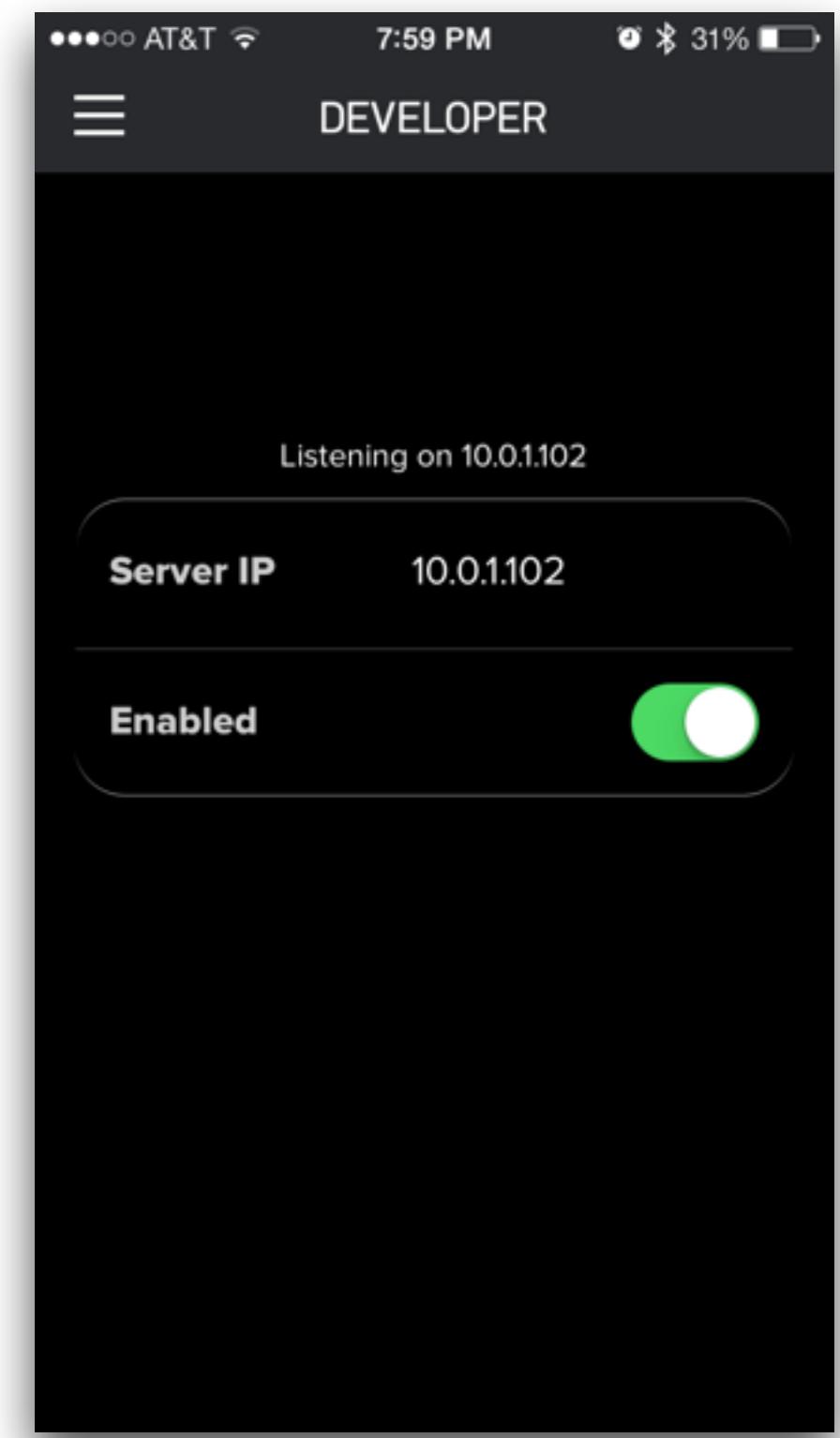
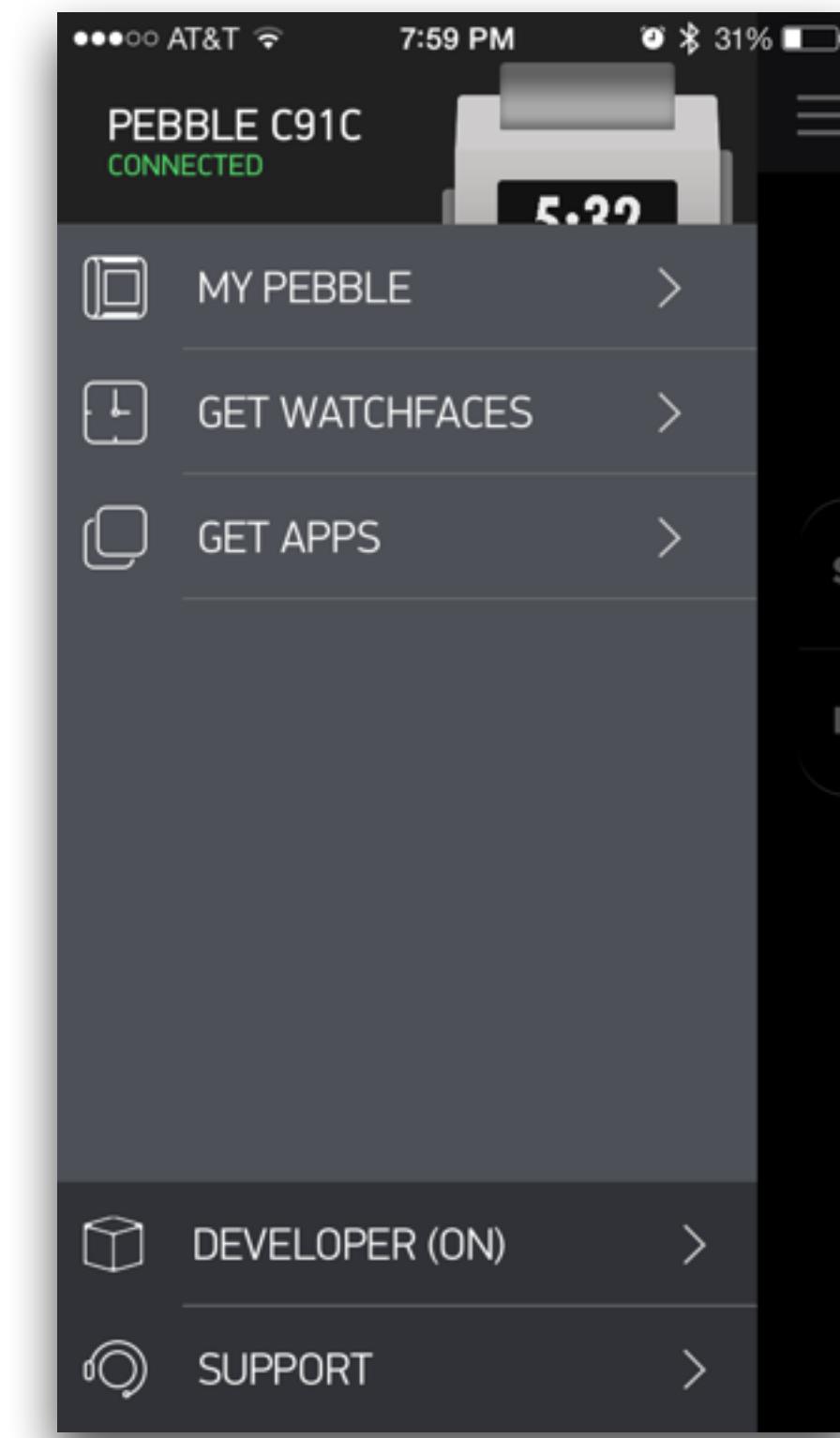
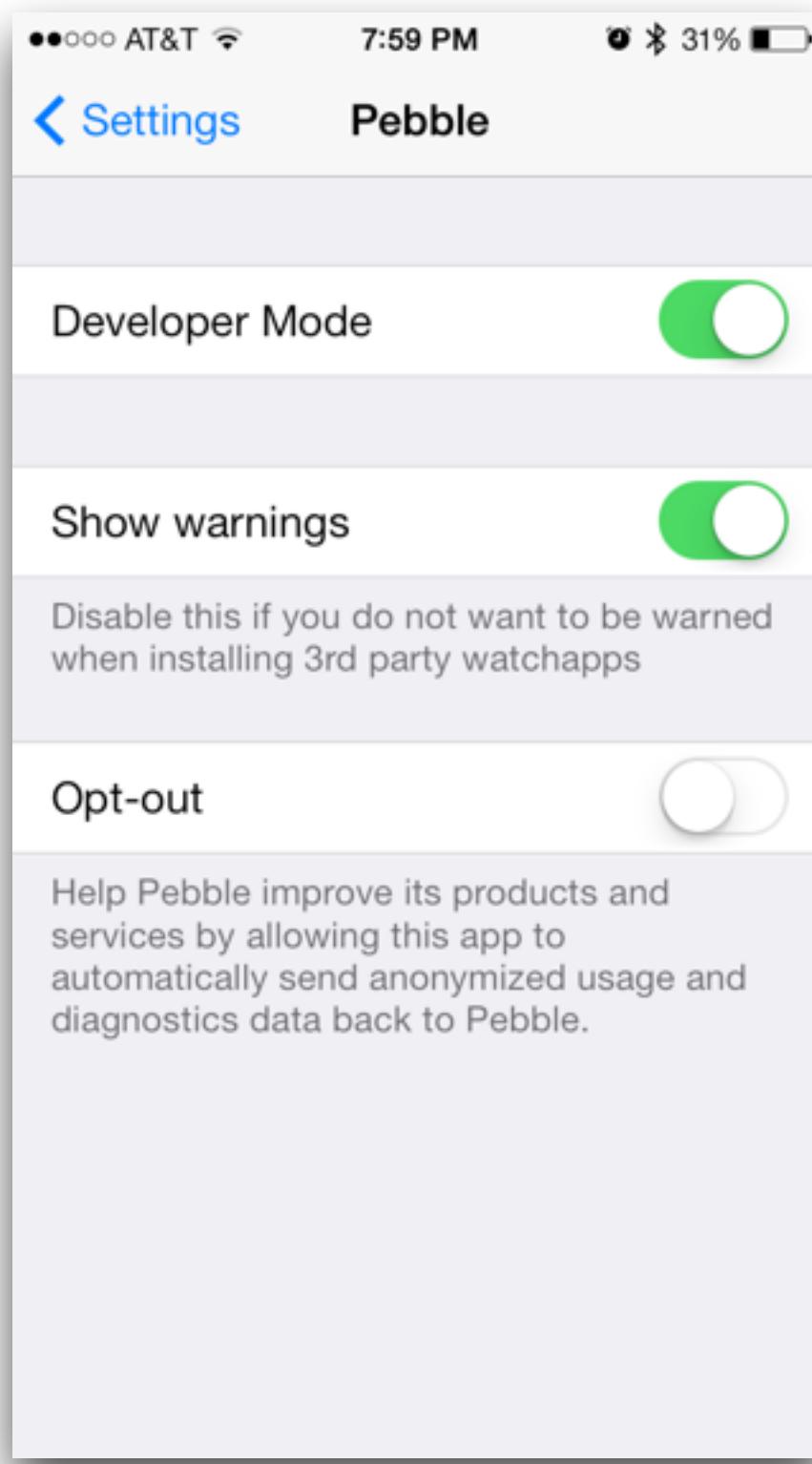
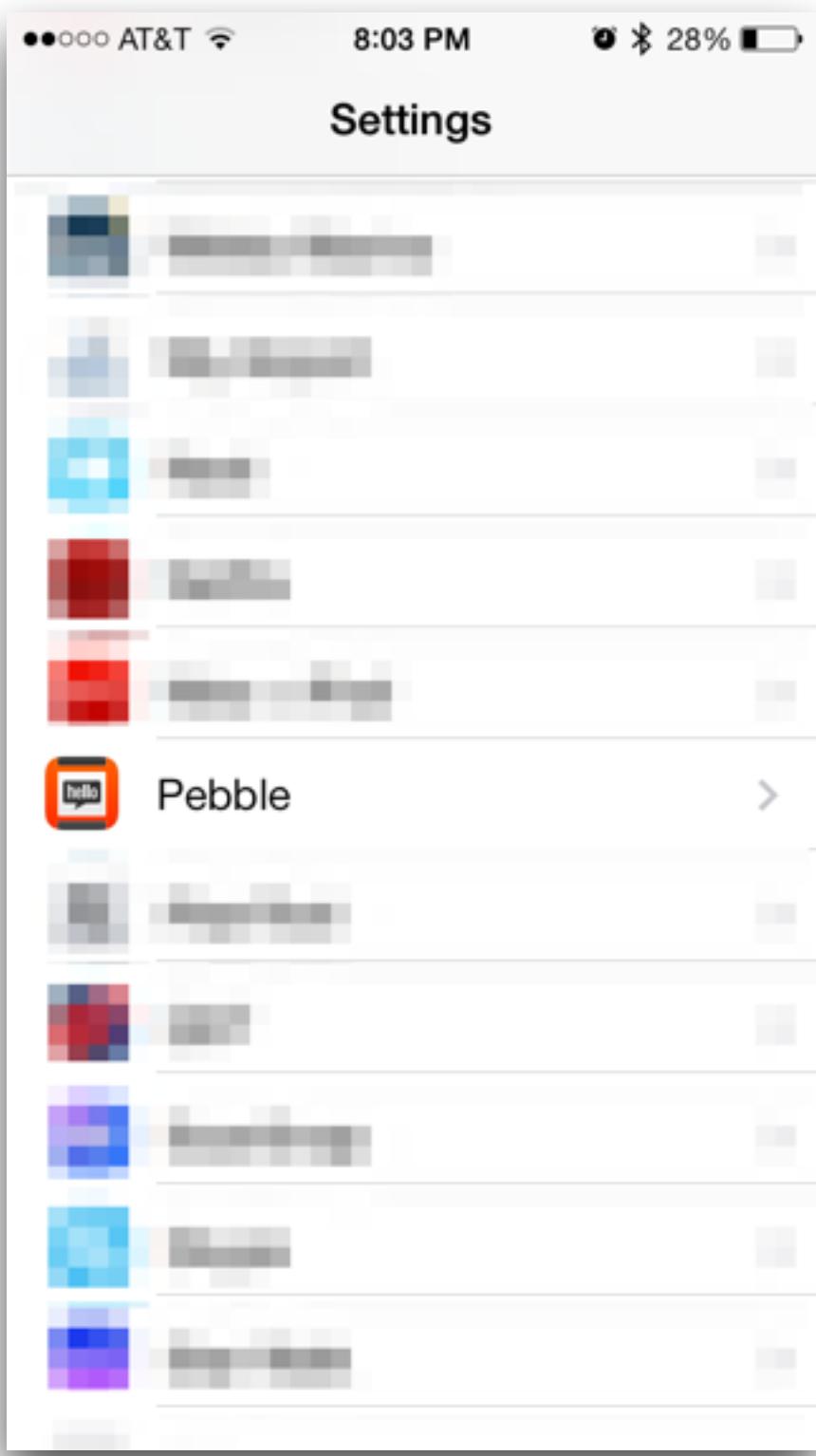
A form on the right allows entering a Phone IP address (10.0.0.153 is shown) and provides options to Install & Run, View app logs, and take a Screenshot. A note states: "Requires that the Pebble app be running on your phone with Developer Mode enabled."

A 'Run build' button is located below the build history section.

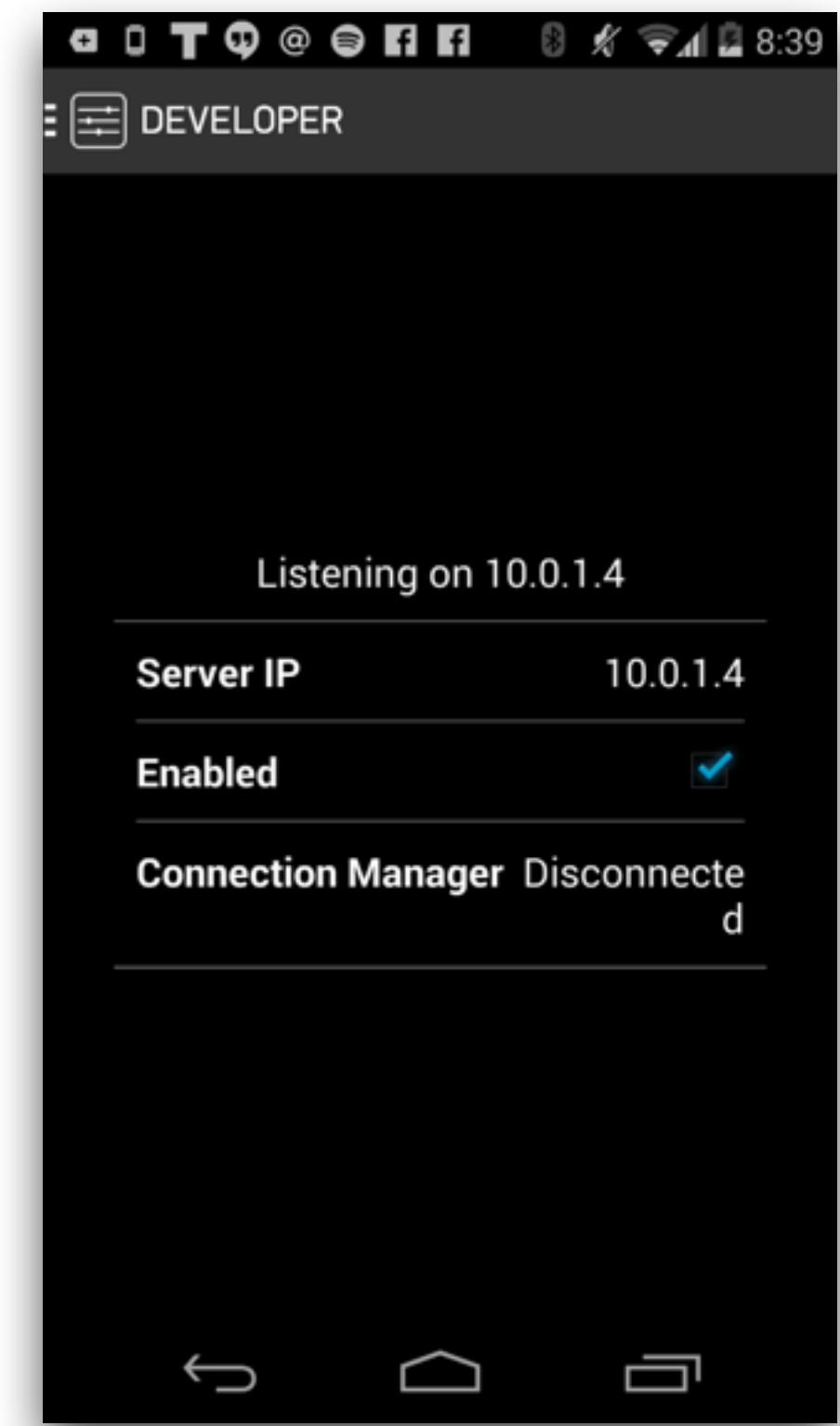
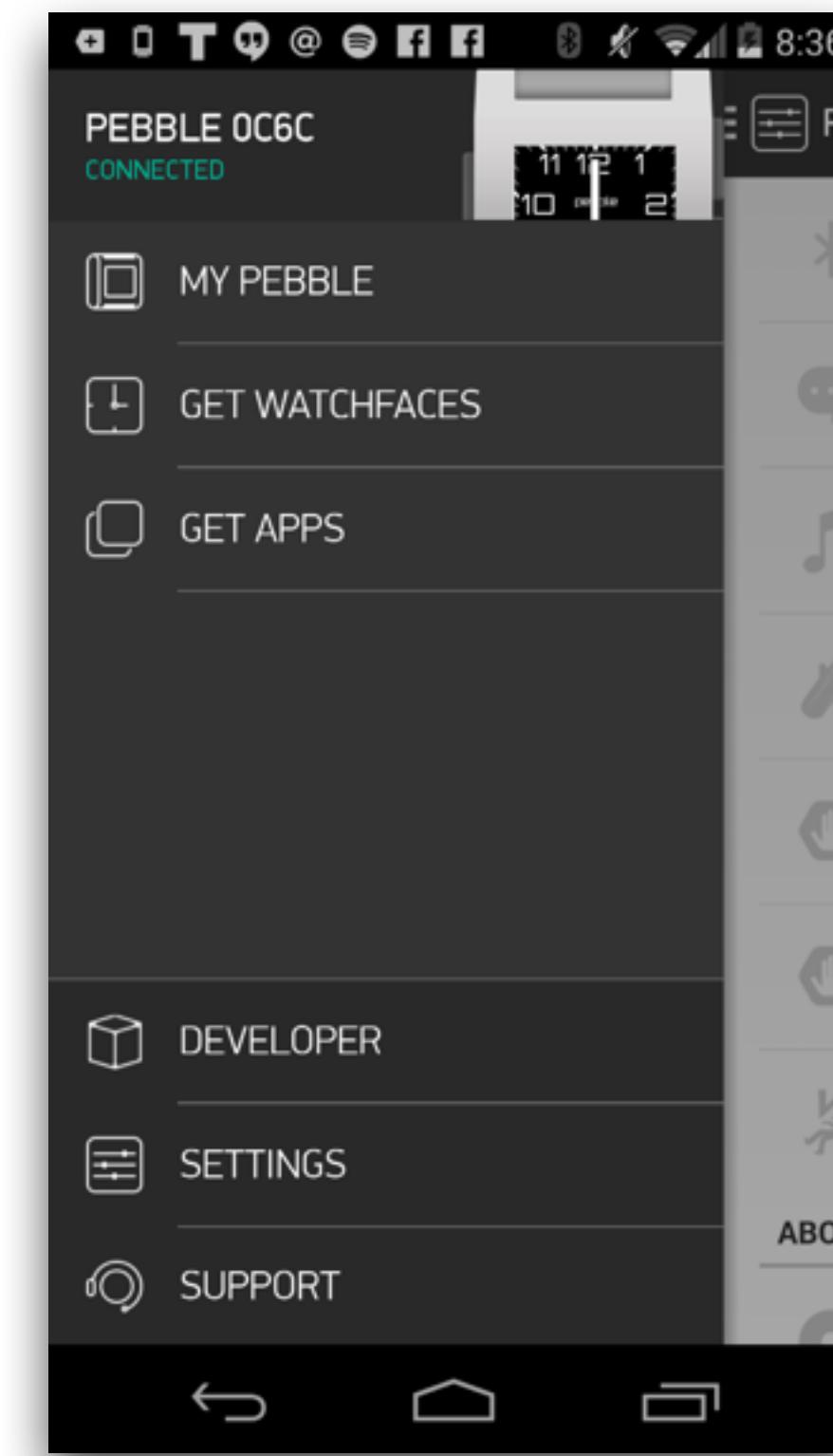
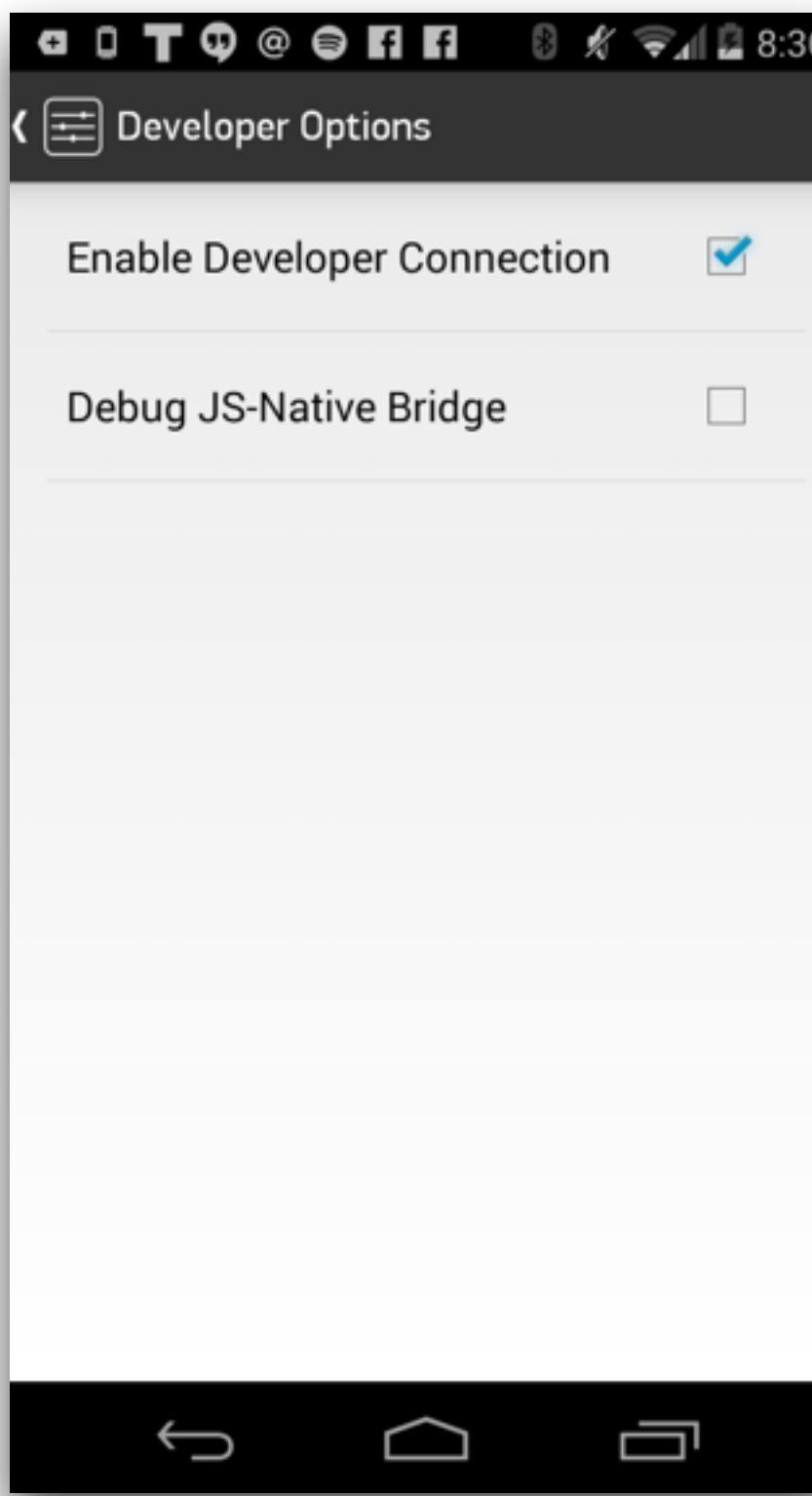
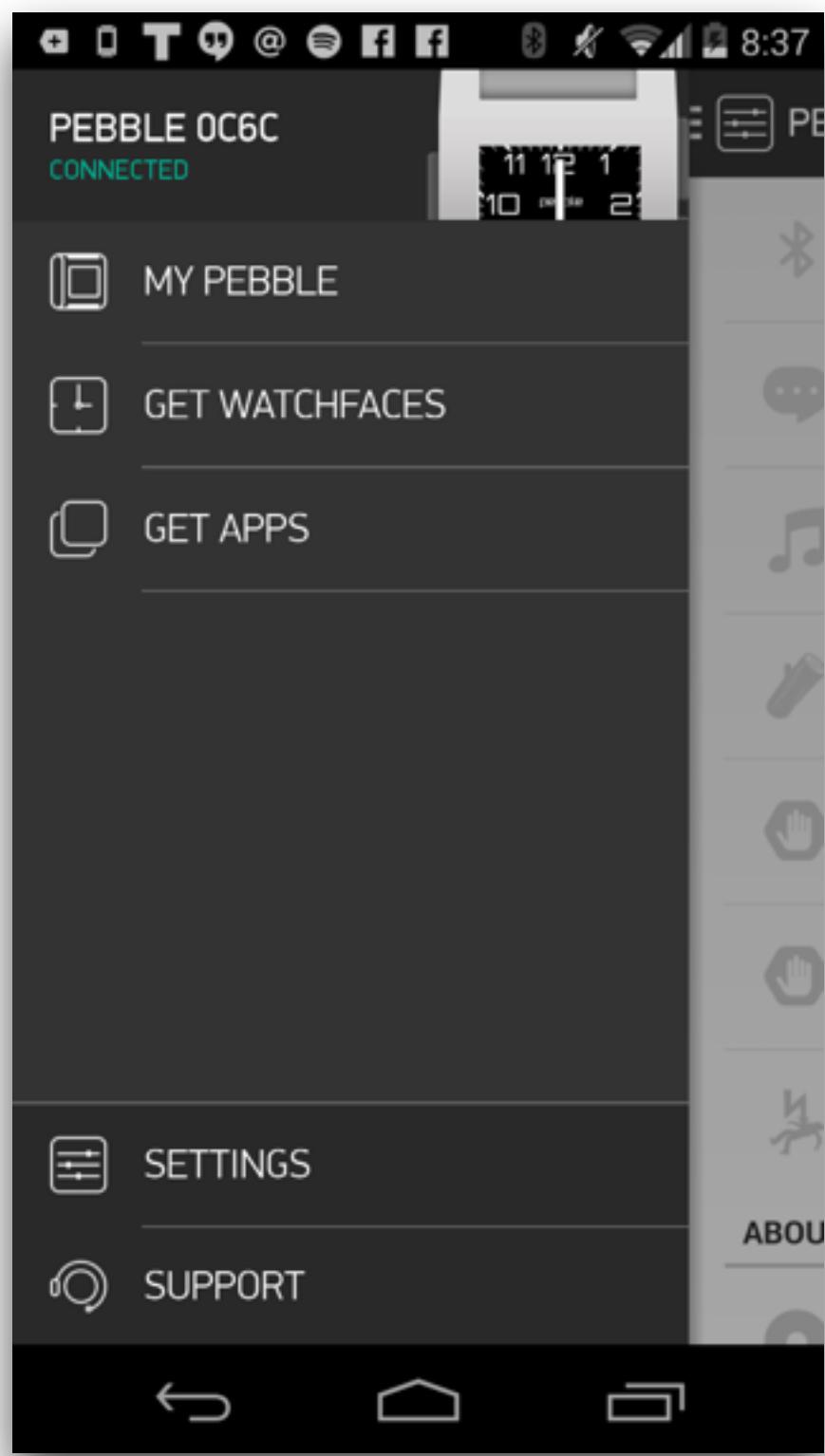
Build history

#	Date	Status	Size
286800	February 25, 2014 at 8:46:50 PM PST	Succeeded	6 KiB

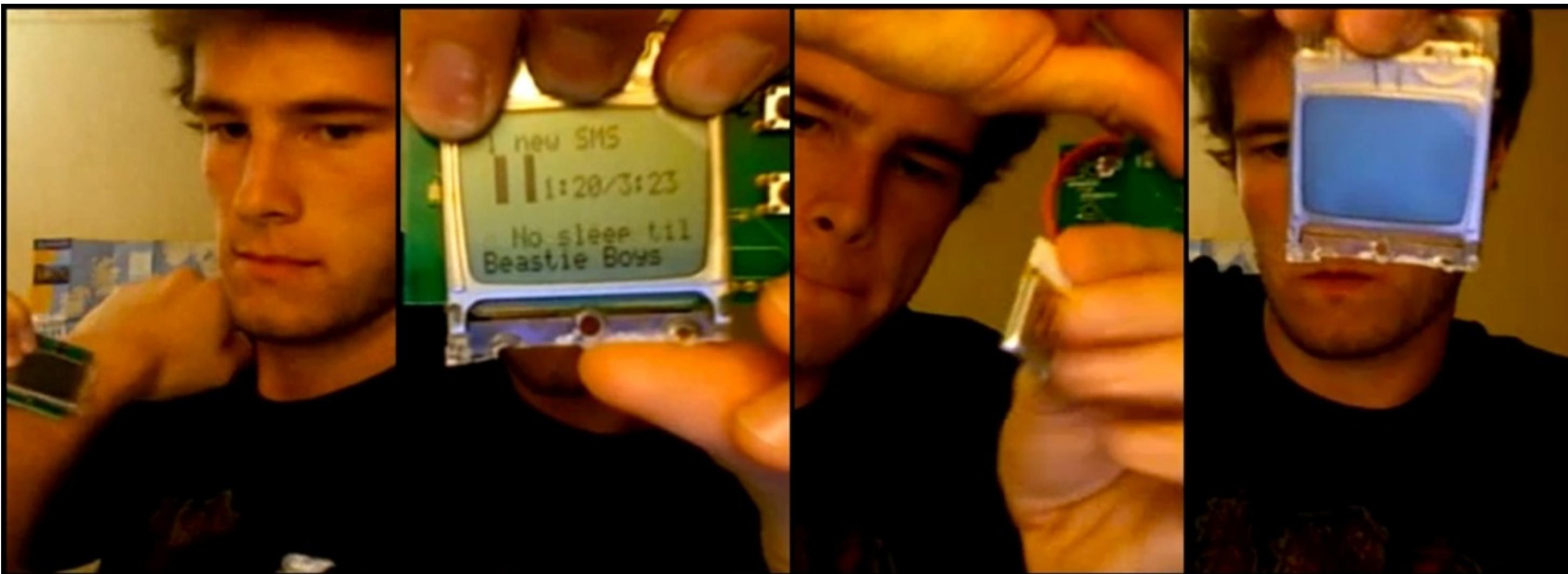
ENABLE THE DEVELOPER CONNECTION ON IOS



ENABLE THE DEVELOPER CONNECTION ON ANDROID



INSTALL AND RUN YOUR APPLICATION



INSIDE YOUR FIRST PEBBLE APP

UNDERSTAND YOUR FIRST PEBBLE C APP

```
int main(void) {  
    handle_init();  
    app_event_loop();  
    handle_deinit();  
}
```

You provide a function to initialize your app

The system will run an event loop and send events to your app

The event loop returns when the app is ready to exit.
Clean up!

CloudPebble - hello-world

https://cloudpebble.net/ide/project/37421#

CloudPebble – hello-world

hello-world Your Projects IDE Settings Blog Sign out

PROJECT

Settings Compilation GitHub

SOURCE FILES

hello_world.c New C file New JavaScript file

RESOURCES

New resource

```
1 #include <pebble.h>
2
3 Window *window;
4 TextLayer *text_layer;
5
6 void handle_init(void) {
7     // Create a window and text layer
8     window = window_create();
9     text_layer = text_layer_create(GRect(0, 0, 144, 154));
10
11    // Set the text, font, and text alignment
12    text_layer_set_text(text_layer, "Hi, I'm a Pebble!");
13    text_layer_set_font(text_layer, fonts_get_system_font(FONT_KEY_GOTHIC_28_BOLD));
14    text_layer_set_text_alignment(text_layer, GTextAlignmentCenter);
15
16    // Add the text layer to the window
17    layer_add_child(window_get_root_layer(window), text_layer_get_layer(text_layer));
18
19    // Push the window
20    window_stack_push(window, true);
21
22    // App Logging!
23    APP_LOG(APP_LOG_LEVEL_DEBUG, "Just pushed a window!");
24 }
```

Delete Reload file Save

```
#include <pebble.h>
```

```
Window *window;  
TextLayer *text_layer;
```

Declare our two UI elements

```
#include <pebble.h>  
  
Window *window;  
TextLayer *text_layer;  
  
void handle_init(void) {  
    // Create a window and text layer  
    window = window_create();  
    text_layer = text_layer_create(GRect(0, 0, 144, 154));  
  
    // Set the text, font, and text alignment  
    text_layer_set_text(text_layer, "Hi, I'm a Pebble!");  
    text_layer_set_font(text_layer, fonts_get_system_font(FONT_KEY_GOTHIC_28_BOLD));  
    text_layer_set_text_alignment(text_layer, GTextAlignmentCenter);  
  
    // Add the text layer to the window  
    layer_add_child(window_get_root_layer(window), text_layer_get_layer(text_layer));  
  
    // Push the window  
    window_stack_push(window, true);  
  
    // App Logging!  
    APP_LOG(APP_LOG_LEVEL_DEBUG, "Just pushed a window!");  
}
```

```
void handle_init(void) {  
    window = window_create();  
    text_layer = text_layer_create(  
        GRect(0, 0, 144, 154)  
);
```

```
#include <pebble.h>  
  
Window *window;  
TextLayer *text_layer;  
  
void handle_init(void) {  
    // Create a window and text layer  
    window = window_create();  
    text_layer = text_layer_create(GRect(0, 0, 144, 154));  
  
    // Set the text, font, and text alignment  
    text_layer_set_text(text_layer, "Hi, I'm a Pebble!");  
    text_layer_set_font(text_layer, fonts_get_system_font(FONT_KEY_GOTHIC_28_BOLD));  
    text_layer_set_text_alignment(text_layer, GTextAlignmentCenter);  
  
    // Add the text layer to the window  
    layer_add_child(window_get_root_layer(window), text_layer_get_layer(text_layer));  
  
    // Push the window  
    window_stack_push(window, true);  
  
    // App Logging!  
    APP_LOG(APP_LOG_LEVEL_DEBUG, "Just pushed a window!");  
}
```

Initialize both UI elements

```
text_layer_set_text(text_layer,
    "Hi, I'm a Pebble!");
text_layer_set_font(text_layer,
    fonts_get_system_font(
        FONT_KEY_GOTHIC_28_BOLD));
text_layer_set_text_alignment(text_layer,
    GTextAlignmentCenter);
```

```
#include <pebble.h>

Window **window;
TextLayer *text_layer;

void handle_init(void) {
    // Create a window and text layer
    window = window_create();
    text_layer = text_layer_create(GRect(0, 0, 144, 154));

    // Set the text, font, and text alignment
    text_layer_set_text(text_layer, "Hi, I'm a Pebble!");
    text_layer_set_font(text_layer, fonts_get_system_font(FONT_KEY_GOTHIC_28_BOLD));
    text_layer_set_text_alignment(text_layer, GTextAlignmentCenter);

    // Add the text layer to the window
    layer_add_child(window_get_root_layer(window), text_layer_get_layer(text_layer));

    // Push the window
    window_stack_push(window, true);

    // App Logging!
    APP_LOG(APP_LOG_LEVEL_DEBUG, "Just pushed a window!");
}
```

Define properties of our text layer

```
#include <pebble.h>

Window *window;
TextLayer *text_layer;

void handle_init(void) {
    // Create a window and text layer
    window = window_create();
    text_layer = text_layer_create(GRect(0, 0, 144, 154));

    // Set the text, font, and text alignment
    text_layer_set_text(text_layer, "Hi, I'm a Pebble!");
    text_layer_set_font(text_layer, fonts_get_system_font(FONT_KEY_GOTHIC_28_BOLD));
    text_layer_set_text_alignment(text_layer, GTextAlignmentCenter);

    // Add the text layer to the window
    layer_add_child(window_get_root_layer(window), text_layer_get_layer(text_layer));

    // Push the window
    window_stack_push(window, true);

    // App Logging!
    APP_LOG(APP_LOG_LEVEL_DEBUG, "Just pushed a window!");
}
```

```
layer_add_child(window_get_root_layer(window),
text_layer_get_layer(text_layer));  
  
window_stack_push(window, true);
```

Add the text layer to the window and push your window

Pebble WindowStack provide a uniformed way to organize navigation inside apps



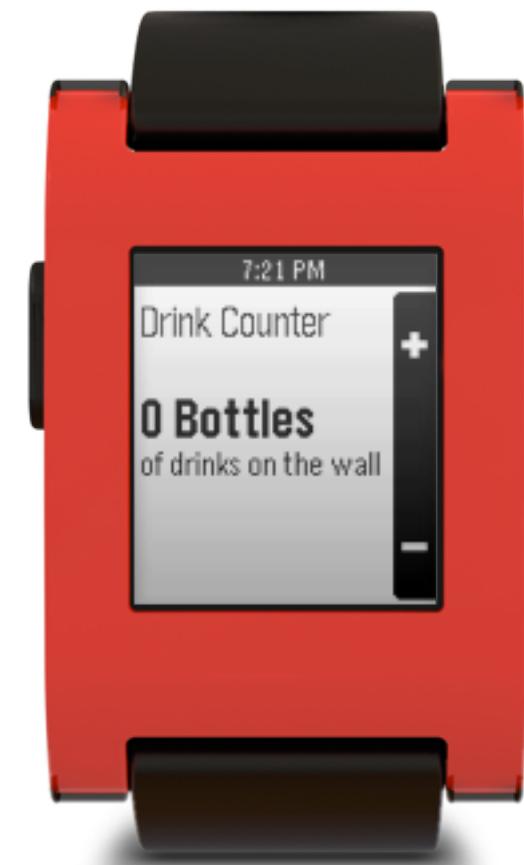
You can add different type of Layers to a Window.
Pebble SDK provides ready-to-use Layers to build beautiful apps.



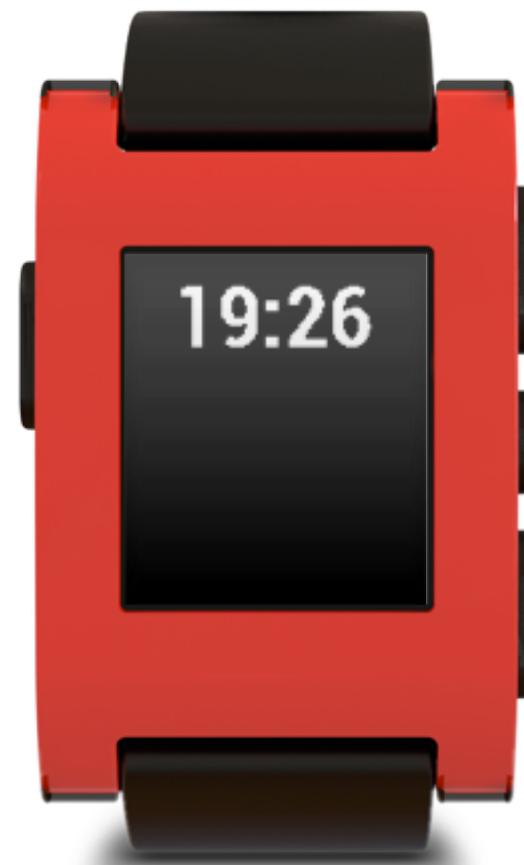
Scroll Layer
`feature_scroll_layer`



Menu Layer
`feature_menu_layer`



Action Bar
`feature_persist_counter`



Animation
`drop_zone`

```
#include <pebble.h>

Window *window;
TextLayer *text_layer;

void handle_init(void) {
    // Create a window and text layer
    window = window_create();
    text_layer = text_layer_create(GRect(0, 0, 144, 154));

    // Set the text, font, and text alignment
    text_layer_set_text(text_layer, "Hi, I'm a Pebble!");
    text_layer_set_font(text_layer, fonts_get_system_font(FONT_KEY_GOTHIC_28_BOLD));
    text_layer_set_text_alignment(text_layer, GTextAlignmentCenter);

    // Add the text layer to the window
    layer_add_child(window_get_root_layer(window), text_layer_get_layer(text_layer));

    // Push the window
    window_stack_push(window, true);

    // App Logging!
    APP_LOG(APP_LOG_LEVEL_DEBUG, "Just pushed a window!");
}
```

APP_LOG(APP_LOG_LEVEL_DEBUG,
"Just pushed a window!");

You can send logs message from your Pebble app.

[View app logs](#)

CloudPebble - hello-world

cloudpbebble.net/ide/project/37421#

CloudPebble – hello-world

hello-world Your Projects IDE Settings Blog Sign out

PROJECT

Settings

Compilation

GitHub

SOURCE FILES

hello_world.c

New C file

New JavaScript file

RESOURCES

New resource

```
[INFO] app_manager.c:143: Heap Usage for <Simply.js>: Available <19412B> Used <5016B> Still allocated <32B>
[DEBUG] hello_world.c:23: Just pushed a window!
[PHONE] pebble-app.js:?: {'runhost client uuid' = e5cd9f3a-2123-4bd8-bc88-6490731bf6c9}: {'webapp uuid' = 133215f0-cf20-4c05-997b-3c9be5a64e5b}: ++_JS_LIFECYCLE_++:KILLED
```

DEMO - ADD ANOTHER TEXT LAYER IN OUR APPLICATION



ADD SUPPORT FOR EVENTS

Pebble provides a uniformed way to deal with events based on a subscribe mechanism. You can use it for time changes, timers, app-messages, bluetooth connection changes and accelerometer detection.

```
char message[40];

void accel_handler(AccelData *data, uint32_t num_samples) {
    /* ... */
}

void handle_init() {
    /* ... */
    accel_data_service_subscribe(1, accel_handler);
}
```

DEMO - ADD LIVE ACCELEROMETER DATA



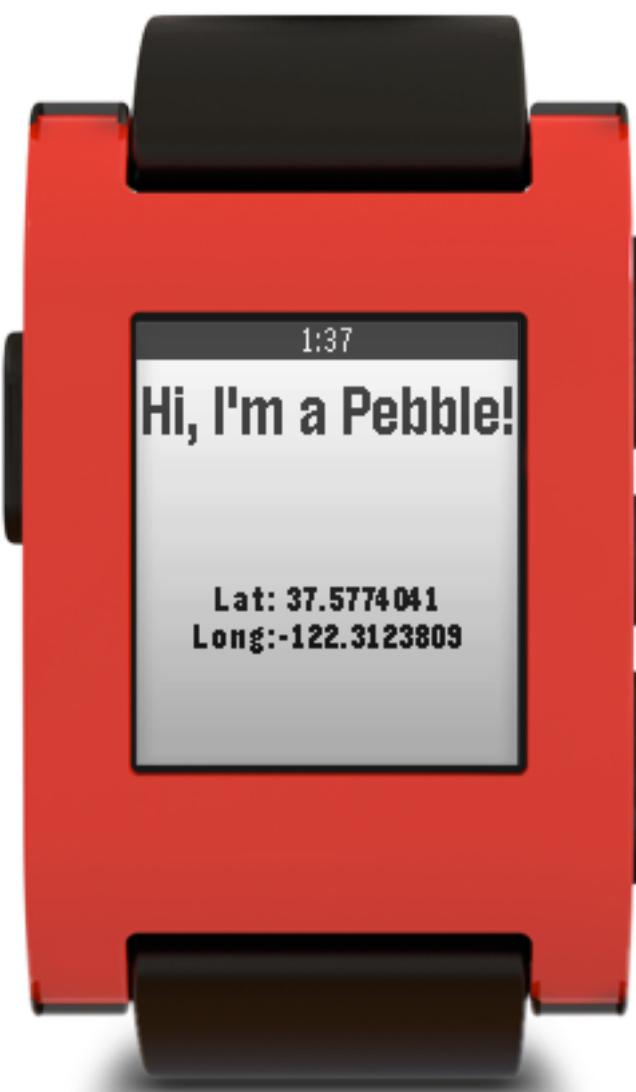
APP MESSAGE

Pebble apps can receive and send messages to a JavaScript program running in the Pebble mobile app, to an iOS app or to an Android app.

The protocol used is called App Message and is based on very simple Dictionaries.

DEMO - RECEIVING MESSAGES WITH APP MESSAGE

DEMO - USING PEBBLEKIT JS TO GET CURRENT GPS COORDINATES



OTHER APIs

There are more Pebble APIs for specific applications:

Data Logging - To collect and transfer large amount of data.

Graphics API - To draw directly on screen

App Sync - A simplified framework on top of App Message

Persistent Storage - Save data on Pebble flash storage

Vibration - User interaction



TinyBird



RayTracer



PebbGPS

RESOURCES

<http://developer.getpebble.com>

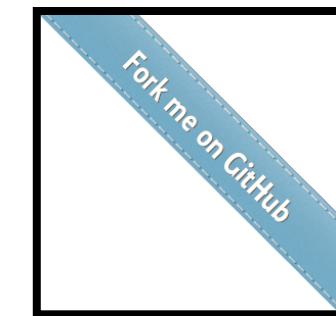
The developer web site is the main hub for Pebble development:

[Design Guide](#)

[Developer Guides](#)

[API Reference](#)

[Examples](#)



Thomas Sarlandie
@sarfata / @pebbledev

PEBBLE FORUMS

GitHub

[pebble](#)

[pebble-hacks](#)

[Search for window stack push](#)



#MAKEAWESOMEHAPPEN

APPENDIX: INSTALLING PEBBLE SDK ON YOUR COMPUTER

INSTALLING THE PEBBLE SDK ON YOUR COMPUTER

OS X

```
curl -sSL http://developer.getpebble.com/install.sh | sh
```

Linux

Follow instructions on developer.getpebble.com

Windows

We recommend CloudPebble.net or Linux in VirtualBox

USING PEBBLE SDK ON YOUR COMPUTER

