CakePHP-Push Documentation

Release 0.0.1

Romain Monteil

Aug 06, 2018

Getting Started

CHAPTER 1

Introduction

1.1 CakePHP Push

Push is a plugin for CakePHP to send downstream message through services like Firebase Cloud Messaging.

1.2 Requirements

- PHP >= 5.6
- CakePHP >= 3.4.*

CHAPTER 2

Installation

2.1 Using Composer

To install this plugin, run `composer require ker0x/cakephp-push` or add this snippet in your project's `composer.json`.

```
{
    "require": {
        "ker0x/cakephp-push": "~2.0"
    }
}
```

2.2 Enable plugin

You can load the plugin by typing the following command in a terminal:

```
bin/cake plugin load Kerox/Push -b
```

or by adding the following line inside your `config/bootstrap.php` file:

```
Plugin::load('Kerox/Push', ['bootstrap' => true]);
```

CHAPTER 3

Configuration

Make a config file `config/push.php`:

```
return [
    'Push' => [
        'adapters' => [
            'Fcm' => [
                'api' => [
                    'key' => '<api-key>',
                    'url' => 'https://fcm.googleapis.com/fcm/send',
                ],
            ],
        ],
    ],
];
```

CHAPTER 4

Firebase Cloud Messaging Adapter

4.1 Introduction

It currently only supports HTTP protocol for :

- sending a downstream message to one or multiple devices

4.2 Configuration

First, you have to get an API key. Go to https://console.firebase.google.com/ , create a project then in your project's settings you will see your Web API Key.

Update your push.php config file to set Fcm to the adapters's array:

```
return [
    'Push' => [
        'adapters' => [
            'Fcm' => [
                'api' => [
                    'key' => 'your_web_API_key',
                    'url' => 'https://fcm.googleapis.com/fcm/send',
                ],
            ],
        ],
    ],
];
```

4.3 Usage

```
use Kerox\Push\Adapter\Fcm;

$adapter = new Fcm();
$adapter
    ->setTokens($tokens)
    ->setNotification($notification)
    ->setDatas($datas)
    ->setParameters($parameters);
```

where:

- `$tokens` is an array of device's token. (required)
- `$notification` is an array containing the notification. (optional)
- `datas` is an array with some datas that will be passed. (optional)
- `$paramaters` is an array of parameters for the notification. (optional)

4.4 Example

```
use Kerox\Push\Adapter\Fcm;
use Kerox\Push\Push;

$adapter = new Fcm();
$adapter
    ->setTokens(['1', '2', '3', '4'])
    ->setNotification([
        'title' => 'Hello World',
        'body' => 'My awesome Hello World!'
    ])
    ->setDatas([
        'data-1' => 'Lorem ipsum',
        'data-2' => 1234,
        'data-3' => true
    ])
    ->setParameters([
        'dry_run' => true
    ]);

$push = new Push($adapter);

// Make the push
$result = $push->send();

// Get the response
$response = $push->response();
```

4.5 Appendice

FCM HTTP Protocol reference

CHAPTER 5

Indices and tables

- genindex
- modindex
- search