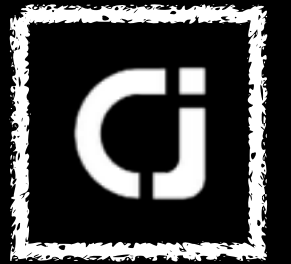


CAKEFEST 2016

CAKEPHP 3

AS A POWERFUL *BACK-END* API



INTRODUCTION

- Avinash Joshi
- From Bengaluru, India
- Currently live in Dallas, Texas, United States
- Software Engineer @ 2M Companies
- Passionate about Free & OpenSource Software
- Outside of work & code:
 - Spend time with my wife (who also codes at work)
 - Volunteer at Embracing the World, a global network of charitable and social service organizations

TALK DESCRIPTION

[IN CASE YOU ♥ TO READ]

Many are looking to build beautiful mobile and interactive web applications. We all love CakePHP for what it is - a clean, simplistic, and powerful MVC framework!

But with technologies evolving and people moving to build API based applications, we need to leverage the batteries that CakePHP provides. In this talk / workshop, I will dive in to how you can enable CakePHP to be used as a powerful REST API, various ways of structuring your project to enable the use of CakePHP with a front-end web application like AngularJS.

THE NEXT HOUR

- *back-end* \Rightarrow *front-end* separation & integration
- CakePHP 3 as a powerful backend (API) framework
- ideas & experiences on going modular
- example separation/folder structure(s)

WHAT I WON'T BE TALKING ABOUT

1. Implementing API in CakePHP
2. What's the best front-end technology i.e.,
Angular vs Ember vs Backbone vs React vs other
front-end out there

“Always remember that you are absolutely unique.
Just like everyone else.”

–MARGARET MEAD

WHAT DO I MEAN BY

FRONT-END

- Part user sees, touches and experiences
- Not just Photoshop anymore!

BACK-END

- Supports the front-end: Server, Application, Database

THE RIGHT WAY

MONOLITHIC

CakePHP is your MVC guru — handles *back-end* business logic & generates *front-end* user interface that may have Javascript plugins

MODULAR / MICRO-WEB

You have a separation of *back-end* & *front-end* (two completely different technologies)



No! Try not!

DO *or* DO NOT,

there is no *try*.

GO MODULAR

- Light-weight
- A better API-driven architecture
- Token based Authentication (JWT)
- {Multi,Re}-use in Web, Mobile, and Desktop Apps
- *And many more that I may have missed out on...*

THE SEPARATION

MVC becomes **MC | V**

CAKEPHP

ANGULARJS

Types:

- A. Complete Separation
- B. Integrated Separation

COMPLETE SEPARATION

- Two separate repositories
- Independently code, build, test & deploy
- Host your front-end on a CDN for faster delivery
- Spin multiple EC2 / EB instances for back-end
- Prepare for CORS (Cross-Origin Resource Sharing)

REPOSITORIES

`myGitHost.ext/username/project-frontend`

- Hosts your front-end code
- `yourApp.com`

`myGitHost.ext/username/project-backend`

- Hosts your back-end CakePHP API code
- `api.yourApp.com`

FRONT-END REPO

frontend

- ├─ bower.json
- ├─ bower_components/
- ├─ components
 - ├─ home
 - ├─ homeController.js
 - ├─ homeView.html
- ├─ index.html
- ├─ scripts
 - ├─ app.controller.js
 - ├─ app.factory.js
 - ├─ app.module.js
 - ├─ config.router.js
- ├─ styles
 - ├─ style.css
- └─ tpl
 - ├─ footer.html
 - ├─ header.html

BACK-END REPO

backend (your CakePHP Code)

```
├─ ...
├─ src
│   ├─ ...
│   ├─ Controller
│   │   ├─ AppController.php
│   │   └─ BlobsController.php
│   └─ Model
│       ├─ Behavior
│       │   └─ empty
│       ├─ Entity
│       │   └─ Blob.php
│       └─ Table
│           └─ BlobsTable.php
├─ Routing
│   └─ Filter
│       └─ CorsFilter.php
└─ ...
```

INTEGRATED SEPARATION

Front-end

- `yourApp.com`

Back-end

- `yourApp.com/api` OR
- `yourApp.com/_api` OR
- `yourApp.com/_/api` OR
- Whatever you prefer!

INTEGRATED REPOSITORY

```
.
├── _api (CakePHP)
│   ├── ...
│   ├── src
│   │   ├── Console/
│   │   ├── Controller
│   │   │   ├── ApplicationController.php
│   │   │   ├── Component/
│   │   │   │   └── V1
│   │   │       └── BlobsController.php
│   │   └── ...
│   └── ...
├── _ui (AngularJS)
│   ├── bower.json
│   ├── bower_components
│   ├── components/
│   ├── scripts/
│   ├── styles/
│   └── tpl/
└── index.html
```

“Talk is cheap. Show me the code.”

–LINUS TORVALDS

TIPS

- Think about using AWS S3 to store files
- AngularJS lazyload
- If SEO is important, use prerender.io
- Use CRUD - github.com/ADmad/cakephp-jwt-auth
- Use CRUD - github.com/friendsofcake/crud

QUESTIONS?

`github.com/avinashjoshi`

`twitter.com/avinashjoshi`

`a10i.me`

`irc: nashj`