VUESS

@cheddam

Support Developer at `this`

What is Vue?

"A library for creating web interfaces. Together with some other tools you can also call it a 'framework', although it's more like a set of optional tools that work together really well."

–Evan You

- Super simple to get started
- Supports wide range of methodologies
- Stable (A SUPER UNIQUE FEATURE)

Relevance?

- SilverStripe 4 Core is built with React
- React knowledge will be integral for working on the backend

- The front-end is still a wide open field
- Some alternatives are easier to get up and running

A Quick Overvue

```
app = new Vue({
   el: '#app',
   data: {
      firstName: 'Eris',
      lastName: 'Morn',
      counter: 0
   },
   computed: {
      fullName: function() {
         return this.firstName + ' ' + this.lastName;
      }
   },
   methods: {
      incrementCounter: function() {
         this.$set('counter', this.counter + 1);
      }
   },
   created: function() {
      // Do something at boot
   }
});
```

```
<div id="app">
   <h1>Your name is {{ fullName }} and your number is {{ counter }}</h1>

   <p>First Name: <input type="text" v-model="firstName"></p>
   <p>Last Name: <input type="text" v-model="lastName"></p>

   <p><button @click="incrementCounter">Increment Number</button></p>
</div>
```

Bridging to SilverStripe

The Secret Sauce:

RestfulServer

- Super simple to get data flowing
- Handles GET / POST / PUT / DELETE

```
composer require silverstripe/restfulserver
```

```php
class Ticket extends DataObject
{
    $db = [
        'Title' => 'Varchar(250)'
    ];

    $has_one = [
        'Author' => 'Member'
    ];

    $has_many = [
        'Comments' => 'Comment'
    ];

    private static $api_access = true;
}
```

```
GET /api/v1/Ticket/1
```

```
<Ticket href="http://venndesk.d/api/v1/Ticket/1.xml">
   <Title>First Test Ticket</Title>
   <AuthorID>1</AuthorID>
   <ID>1</ID>
   <Comments linktype="has_many" href="http://venndesk.d/api/v1/Ticket/1/
Comments.xml">
      <Comment href="http://venndesk.d/api/v1/Comment/1.xml" id="1"/>
      <Comment href="http://venndesk.d/api/v1/Comment/2.xml" id="2"/>
   </Comments>
</Ticket>
```

```
<Ticket href="http://venndesk.d/api/v1/Ti                    >
   <Title>First Test Ticket</Title>
   <AuthorID>1</AuthorID>
   <ID>1</ID>
   <Comments linktype="has_many" h                  //venndesk.d/api/v1/Ticket/1/
Comments.xml">
      <Comment href="http://ve                  pi/v1/Comment/1.xml" id="1"/>
      <Comment href="http:/                    /api/v1/Comment/2.xml" id="2"/>
   </Comments>
</Ticket>
```

```
GET /api/v1/Ticket/1.json
```

```
{
   "Title": "First Test Ticket",
   "AuthorID": "1",
   "AssigneeID": "1",
   "ID": "1",
   "Comments": [
      {
         "className": "Comment",
         "href": "http://venndesk.d/api/v1/Comment/1.json",
         "id": 1
      },
      {
         "className": "Comment",
         "href": "http://venndesk.d/api/v1/Comment/2.json",
         "id": 2
      }
   ]
}
```

- Uses `canView()`, etc. for permissions
- You can limit the properties sent over the wire

All sorted, right?


ACTUALLY NO
SEX BOB OMB

- PHP 5.6 is an asshole
  - `;always_populate_raw_post_data=-1`
- No context of the request in permission methods
  - Adding tons of logic breaks other interfaces

The Secreter Sauce:
RestfulAPI

```
composer require colymba/silverstripe-restfulapi
```

- Cleaner output (JSON by default)
- Same base configuration as RestfulServer
  - `$api_access = ['GET', 'POST'];`
- Default Permission Manager uses `can*()` *but* you can override this very simply

Basically, use this one.

Bonus Points

vueify

.vue

```
<template>
   <div class="ticket">
      <h1 class="title">{{ ticket.Title }}</h1>
      <Comment v-for="commentsForTicket(ticket.ID) as comment" @comment="comment"></Comment>
   </div>
</template>

<script>
   export default {
      props: ['ticket']
   }
</script>

<style lang="scss">
   .ticket {
      .title {
         color: fuchia;
      }
   }
</style>
```

- Scoped CSS
- Props ensure only relevant data is available
- Events or `$root` methods for interactions

vuex

vue-router

Summary

- Vue is swell
- RestfulAPI is also swell
- Consider dropping in Vue for widgets (or SPAs)
- There are great examples of Vue projects around
  - Facebook even used it (!!!)

@cheddam

on Twitter + Github