



Projects

Remise Zippée du projet sur <http://foad2.unicaen.fr/moodle/course/view.php?id=20911>

Attention au nommage **prenom.nom** !

-- Contexte

Vous travaillez sur un outil permettant de gérer des projets.

Voici les principales caractéristiques du système d'information :

- Chaque projet [**project**] possède un nom, un descriptif, une date de début et de fin, et un propriétaire (owner)
- L'équipe est constituée d'un ensemble de développeurs [**developer**].
- Chaque User story [**story**] a un code et un descriptif, et appartient à un projet.
- Il est possible de lui apposer des tags [**tags**], composés d'une couleur et d'un label.
- Elle peut être affectée à un développeur [**dev**] (qui a juste une identité).
- Elle peut contenir une liste de tâches [**tasks**], à réaliser ou réalisées.



-- Eléments à implémenter

- **Strict respect des normes, nommages et consignes** données ⇒ **1 point**
- Usage de **Browserify** pour créer un unique bundle ⇒ **1 point**

-- Squelette de l'application, routage (3 points)

//TODO 2.1.1

Le dossier root de votre application devra être de la forme : **prenom.nom**

Créer la structure suivante :

Emplacement	Fichier	Rôle
/	index.html	Fichier principal
/app		Fichiers js
	app.js	Fichier principal de l'application (module)
	config.js	Fichier de routage et configuration
/app/controllers		contrôleurs
	myProjects.js	Définit le contrôleur myProjectsController
	project.js	Définit le contrôleur projectController

Emplacement	Fichier	Rôle
/app/services		services
/app/views		templates et vues HTML
/public/		Feuille(s) de styles + fichiers js externes

//TODO 2.1.2

Implémenter le routage suivant :

URL	Composante	Valeur
/home	Description	Affiche la liste des projets de l'utilisateur, et les projets auxquels il participe
	template	app/views/myProjects.html
	contrôleur	myProjectsController
	alias	myProjectsCtrl
/project/:_id	Description	Affiche le projet correspondant à _id
	template	templates/project.html
	contrôleur	projectController
	alias	projectCtrl
/home	Route par défaut	

-- Service (2 points)

//TODO 2.2

Le service **daoService** simule la connexion à un web service ; il est défini de la façon suivante :

stories, **tags** et **devs** seront définis "en dur" à l'intérieur du service dans l'équivalent de variables privées :

```

var _projects={"_embedded":[
  {"_id":{"$oid":"58d038b705d0b0b35f9764da"},"name":"Open-beer","descriptif":"A
free, public database and API for beer information.,"startDate":"March 20, 2017
21:16:55"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764d9"},"name":"Boards
analysis","descriptif":"AngularJS application + REST API","startDate":"March 20,
2017 21:16:55","owner":{"identity":"Rod
Johnson","_id":{"$oid":"58d038b705d0b0b35f9764d5"}}},
  {"_id":{"$oid":"58d038b705d0b0b35f9764d8"},"name":"Boards
admin","descriptif":"Administration interface for Boards app with
javaFX","startDate":"March 20, 2017 21:16:55","owner":{"identity":"Linus
Torvalds","_id":{"$oid":"58d038b705d0b0b35f9764d7"}}}
],"_id":"Project","_returned":3};
var _stories={"_embedded":[
{"_id":{"$oid":"58d038b705d0b0b35f9764df"},"code":"Beer1","descriptif":"affichage
de la liste des bières /beers (L'affichage de la bière n'affiche pas le brasseur
associé)","project":{"name":"Open-beer","descriptif":"A free, public database and
API for beer information.,"startDate":"March 20, 2017
21:16:55","_id":{"$oid":"58d038b705d0b0b35f9764da"}},"developer":{"identity":"Linus
Torvalds","_id":{"$oid":"58d038b705d0b0b35f9764d7"}},"tags":[{"title":"bug","color"
:"#EE0701","_id":{"$oid":"58d038b705d0b0b35f9764e0"}}],"tasks":[]},
  {"_id":{"$oid":"58d038b705d0b0b35f9764de"},"code":"Dev1","descriptif":"En tant
que développeur, je peux consulter mes projets","project":{"name":"Boards
analysis","descriptif":"AngularJS application + REST API","startDate":"March 20,
2017 21:16:55","owner":{"identity":"Rod

```

```
Johnson", "_id":{"$oid":"58d038b705d0b0b35f9764d5"}}, "_id":{"$oid":"58d038b705d0b0b35f9764d9"}}, "developer":{"identity":"Linus Torvalds", "_id":{"$oid":"58d038b705d0b0b35f9764d7"}}, "tags":[{"title":"duplicate", "color":"#CCCCCC", "_id":{"$oid":"58d038b705d0b0b35f9764e1"}}], "tasks":[{"content":"Rien", "closed":true}, {"content":"Rien non plus", "closed":true}], {"_id":{"$oid":"58d038b705d0b0b35f9764dd"},"code":"E140","descriptif":"En tant que créateur, je souhaite gérer les utilisateurs [methods]","project":{"name":"Boards admin","descriptif":"Administration interface for Boards app with javaFX","startDate":"March 20, 2017 21:16:55","owner":{"identity":"Linus Torvalds", "_id":{"$oid":"58d038b705d0b0b35f9764d7"}}, "_id":{"$oid":"58d038b705d0b0b35f9764d8"}}, "tags":[], "tasks":[]}, {"_id":{"$oid":"58d038b705d0b0b35f9764dc"},"code":"E120","descriptif":"En tant que créateur, je veux créer / Modifier des quiz [methods]","project":{"name":"Boards admin","descriptif":"Administration interface for Boards app with javaFX","startDate":"March 20, 2017 21:16:55","owner":{"identity":"Linus Torvalds", "_id":{"$oid":"58d038b705d0b0b35f9764d7"}}, "_id":{"$oid":"58d038b705d0b0b35f9764d8"}}, "tags":[], "tasks":[]}, {"_id":{"$oid":"58d038b705d0b0b35f9764db"},"code":"B22","descriptif":"En tant que créateur, je veux ajouter et gérer les réponses d'une question [methods]","project":{"name":"Boards admin","descriptif":"Administration interface for Boards app with javaFX","startDate":"March 20, 2017 21:16:55","owner":{"identity":"Linus Torvalds", "_id":{"$oid":"58d038b705d0b0b35f9764d7"}}, "_id":{"$oid":"58d038b705d0b0b35f9764d8"}}, "developer":{"identity":"Rod Johnson", "_id":{"$oid":"58d038b705d0b0b35f9764d5"}}, "tags":[], "tasks":[]}], "_id":"Story", "_returned":5};

var _devs={"_embedded": [
  {"_id":{"$oid":"58d038b705d0b0b35f9764d7"},"identity":"Linus Torvalds"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764d6"},"identity":"James Gosling"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764d5"},"identity":"Rod Johnson"}
], "_id":"Developer", "_returned":3};
var _tags={"_embedded": [
  {"_id":{"$oid":"58d038b705d0b0b35f9764e6"},"title":"wont fix", "color":"#FFFFFF"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764e5"},"title":"question", "color":"#CC317C"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764e4"},"title":"invalid", "color":"#E6E6E6"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764e3"},"title":"help wanted", "color":"#128A0C"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764e2"},"title":"enhancement", "color":"#84B6EB"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764e1"},"title":"duplicate", "color":"#CCCCCC"},
  {"_id":{"$oid":"58d038b705d0b0b35f9764e0"},"title":"bug", "color":"#EE0701"}
], "_id":"Tag", "_returned":7};
var _tasks={"_embedded": [
], "_id":"Task", "_returned":0};
```

Service	datas.js (dataService)
Variables privées	stories Tableau des user stories
	tags Tableau des tags disponibles
	devs Tableau des développeurs

Méthodes publiques	getStories() retourne stories
	getTags() retourne tags
	getDevs() Retourne devs
	getStory(code) Retourne la user story correspondant au code donné, null si le code est inexistant
	setStoryAvancement(story) ajoute et valorise le membre avancement de la story passée en paramètre (avancement=nb tâches réalisées/ nb tâches). Une tâche est réalisée si son membre done=true. A l'issue du passage story.avancement est existant et correctement valorisé

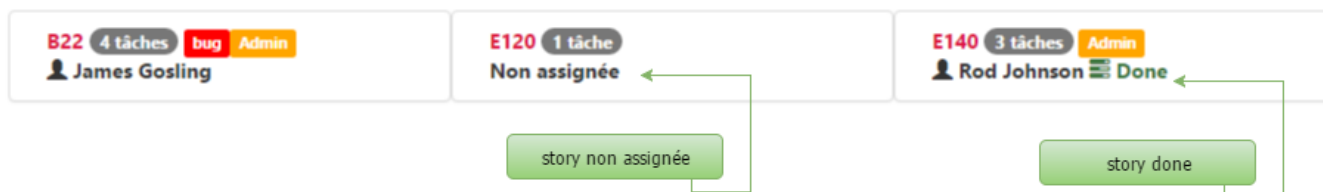
Créez dataService, implémentez les méthodes publiques.
Il faudra ensuite injecter **dataService** aux 2 contrôleurs **storiesController** et **StoryController**

-- Url /stories ou / (3 points)

//TODO 2.3

Affiche la liste des user stories.

Stories



Créer le template **/templates/stories.html**, associé au contrôleur **/js/controllers/stories.js**

Contrôleur	stories.js (storiesController)
Variables privées	stories Tableau des user stories
Méthodes publiques	goto(story) Accède à l'url story:code en utilisant le service \$location

Remarque :

Une **story** est marquée **Done** si son avancement est égal à 1 (100%)

-- Directive storyHeader (2 points)

//TODO 2.4



On souhaite intégrer l'entête de chaque story affichée à l'url /stories dans une directive, pour la réutiliser plus facilement dans la page suivante, pour présenter le détail de chaque story.

La directive **storyHeader**, accessible en tant qu'élément, ou attribut, permettra d'afficher le contenu suivant d'une story. Il sera nécessaire de lui passer la variable story (qu'elle est en charge d'afficher) dans son **scope**.

- Déclarer la directive **storyHeader**,
- intégrer la dans l'application,
- créer le template correspondant,
- utilisez la à l'url **/stories**

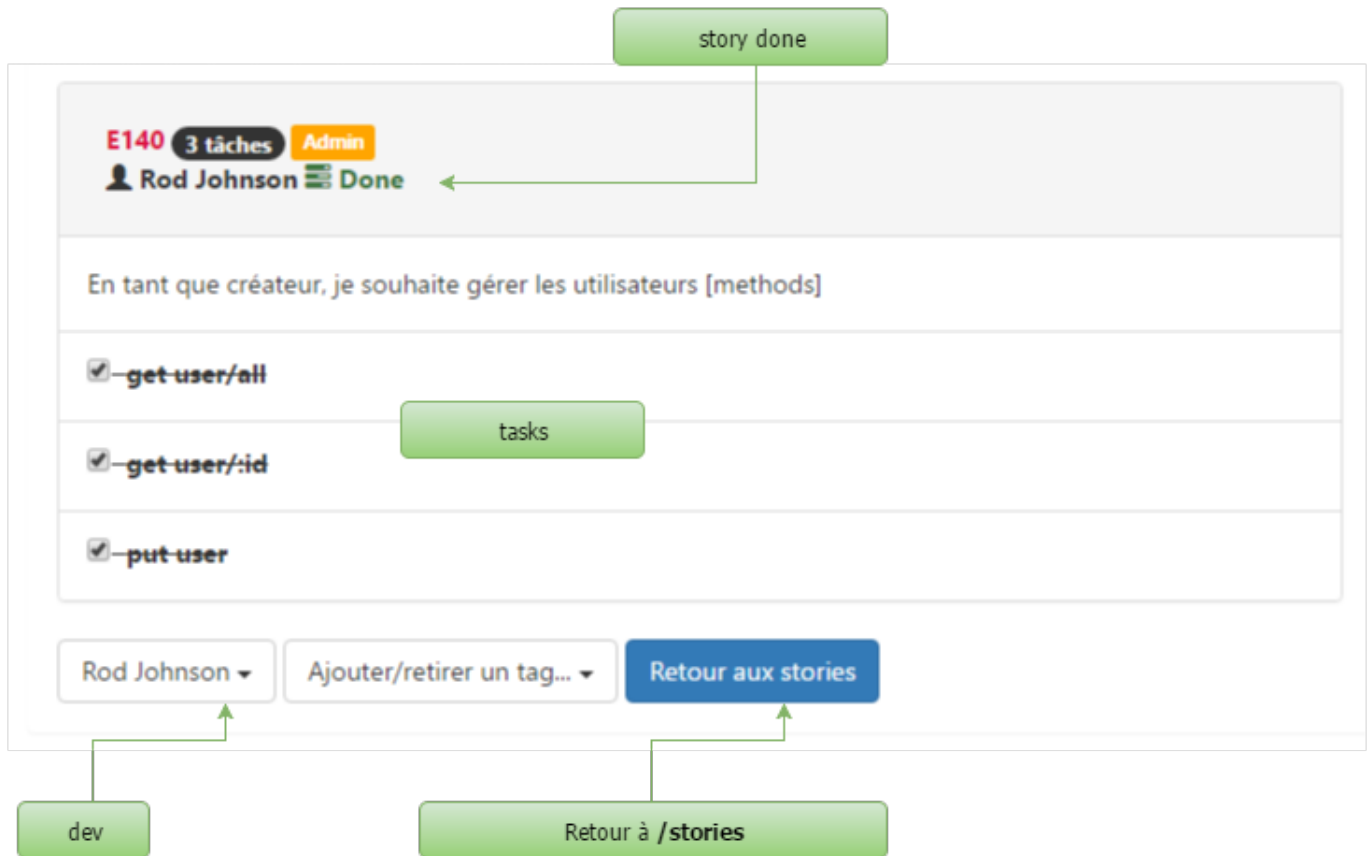
On respectera la structure suivante :

Emplacement	Fichier	Rôle
/js/directives		Dossier définissant les directives
	storyHeader.js	Fichier définissant la directive storyHeader
/js/directives/templates		Fichiers templates
	storyHeader.html	Fichier template de la directive storyHeader

-- Url /story/:code (5 points)

//TODO 2.5

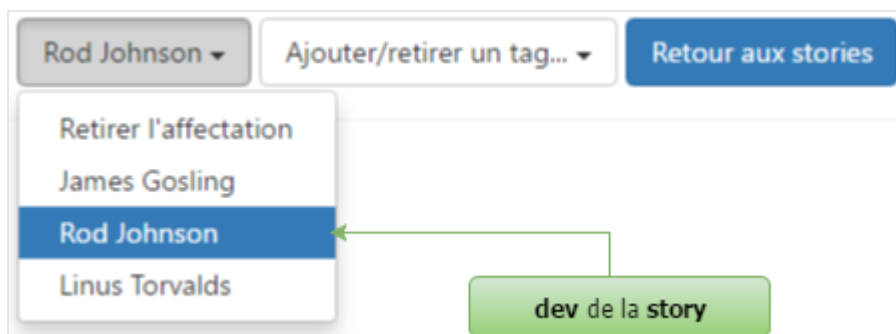
Affiche la **story** correspondant au **code** passé dans l'url.



Comportement de l'interface :

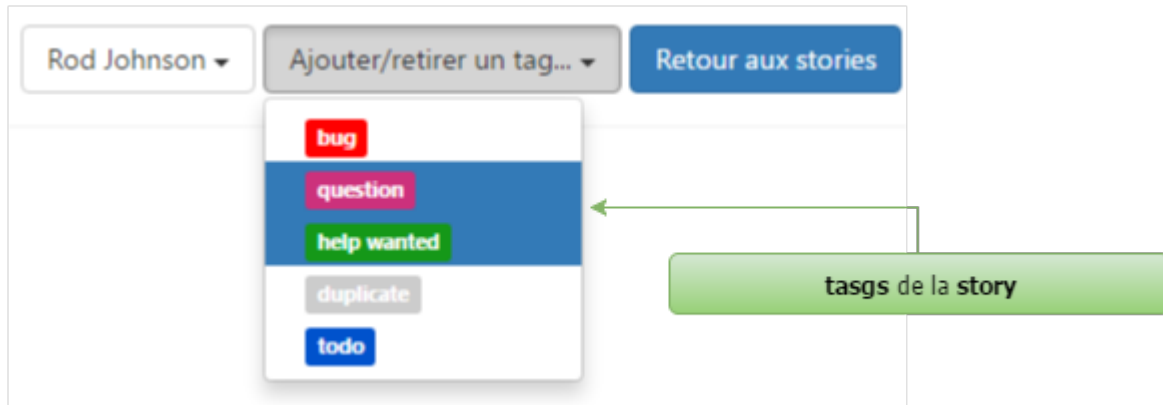
- Une task **done** est barrée
- Les cases à cocher permettent de faire passer une tâche de non réalisée (done=false) à réalisée (done=true) et inversement

Menu Dev :



- Le dev actif est sélectionné (classe bs **active**)
- Le clic sur un autre dev change l'affectation
- L'élément **Retirer l'affectation** passe le dev de la story à **null**

Menu Tags :



- Les tags présents dans la story sont sélectionnés (classe bs **active**)
- Le clic sur un tag l'ajoute ou le retire à la liste des tags de la story

Créer le template **/templates/story.html**, associé au contrôleur **/js/controllers/story.js**

Contrôleur	story.js (storyController)
Variables privées	story user story à afficher
	devs Tableau de tous les développeurs
Méthodes publiques	tags Tableau de tous les tags
	toggleDone(task) Bascule de vrai à faux et inversement le membre done de la tâche task passée en paramètre
	assignDev(dev) Assigne le développeur dev à la story
	indexOfTag(tag) Retourne l'index du tag passé en paramètre dans la liste des tags de la story (-1 s'il n'est pas trouvé)
	toggleTag(tag) Ajoute ou retire le tag passé en paramètre de la liste des tags de la story

-- A poursuivre...

//TODO 2.6

Fonctionnalités supplémentaires à implémenter :

Dans la page story/:code :

- Permettre l'ajout/modification/suppression de tasks (2 points)
- Permettre l'ajout/modification/suppression de nouveaux tags (2 points)

Dans la page /stories ou / :

- Filtrer les stories à afficher par sélection de devs et/ou par la présence de tags et/ou par la mention **done** (2 points)

-- Ressources

HTML/CSS

```
<!DOCTYPE html>
<html>
<head>
  <base href="http://127.0.0.1/yoursite/">
  <meta charset="UTF-8">
  <link rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">

  <script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.2/jquery.min.js"></script>
  <script async type="text/javascript"
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>

  <script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.9/angular.min.js"></script>
  <script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.9/angular-route.min.js"></s
cript>

  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
<div class="container">
  <div class="panel panel default">
    <div class="panel-body">

    </div>
  </div>
</div>
</body>
</html>
```

```
<IfModule mod_rewrite.c>
  Options +FollowSymlinks
  RewriteEngine On
  RewriteBase /yoursite/
  RewriteCond %{REQUEST_FILENAME} !-f
  RewriteCond %{REQUEST_FILENAME} !-d
  RewriteCond %{REQUEST_URI} !.*\.(css|js|html|png|jpg|jpeg|gif|txt|ttf|woff)
  RewriteRule (.*) index.html [L]
</IfModule>
```

```
body{
  font-family: 'Segoe UI', Frutiger, 'Frutiger Linotype', 'Dejavu Sans',
```

```
'Helvetica Neue', Arial, sans-serif;
}
.story-panel{
  float: left;
  width: 300px;
}
.story{
  padding: 10px;
  cursor: pointer;
}
.story-code{
  font-weight: bold;
  color: crimson;
}

.story-panel span{
  vertical-align: inherit;
}
.assign-to{
  font-weight: bold;
}
```

Bootstrap



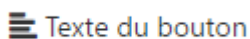
```
<span class="glyphicon glyphicon-align-left" aria-hidden="true"></span>&nbsp;Done
```

3 tâches

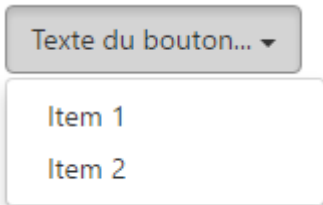
```
<span class="badge">3 tâches </span>
```

Admin

```
<span class="label" style="color: orange">Admin</span>
```



```
<button type="button" class="btn btn-default" aria-label="Left Align">
  <span class="glyphicon glyphicon-align-left" aria-
  hidden="true"></span>&nbsp;Texte du bouton
</button>
```



```
<div class="btn-group">
  <button type="button" class="btn btn-default dropdown-toggle" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    Texte du bouton... <span class="caret"></span>
  </button>
  <ul class="dropdown-menu">
    <li>
      <a href="#">
        Item 1
      </a>
    </li>
    <li>
      <a href="#">
        Item 2
      </a>
    </li>
  </ul>
</div>
```

En-tête
Body
Item 1
Item 2

```
<div class="panel panel-default">
  <div class="panel-heading">
    En-tête
  </div>
  <div class="panel-body">Body</div>
  <ul class="list-group">
    <li class="list-group-item">
      Item 1
    </li>
    <li class="list-group-item">
      Item 2
    </li>
  </ul>
</div>
```

Last update:
2019/08/31 14:40

slam4:richclient:angularjs:projects <http://slamwiki2.kobject.net/slam4/richclient/angularjs/projects?rev=1490041991>

From:
<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:
<http://slamwiki2.kobject.net/slam4/richclient/angularjs/projects?rev=1490041991>

Last update: **2019/08/31 14:40**

