

Exercises: Routing and Architecture

Team Manager

Create a JS application for managing teams. Use [Handlebars](#) for rendering, [Sammy.js](#) for routing and [Kinvey](#) as a backend provider. Structure your work so that it is easy to manage. The example is styled using [Bootstrap](#).

1. App Structure

- **Home Page** - Show relevant info, depending on the status of the user
- **Catalog** - A list of all registered teams
- **About** - Page that would hold information about the app
- **Register User**
- **Create Team**
- **Edit Team**
- **View Team Details** - A detailed page that shows all members of the team and management controls

Create a header that is shared across all pages and place links to the relevant sections in it.

2. CRUD Operations

The app must support user **registration**, **login** and **logout**. Store the user credential in **session storage**. Once logged in, the user is free to browse all registered teams and **join** or **create** a new team. At any point, the user should be able to **leave** the team he is a member of. The user can only join **one** team at a time. Also when a user **creates** a team he **automatically** joins it. He **can NOT** create a team **again** unless he leaves the newly created team.

3. Entity Structure

A team has a **name** and a **comment** that are displayed while browsing. A user has a **username**. You may create databases and entries as you see fit. A sample collection structure is as follows:

```
teams {  
  name,  
  comment  
}
```

Add a column **teamId** to the default **users** collection, showing which team they have joined currently. When determining whether a person is the owner of a team, look at the property **_acl.creator** of the team record.

Screenshots

Use this information as a guideline. You may style and structure your solution differently, so long as the required functionality is present.

[Home page] when the user is logged in and a header, that is shared across all pages.



User registration form

A screenshot of the 'Register Page' in the application. The title 'Register Page' is centered at the top. Below it, the label 'Username:' is followed by a text input field. Below that, the label 'Password:' is followed by a text input field. Then, the label 'Repeat Password:' is followed by another text input field. At the bottom, there is a 'Register' button. The footer text 'Team Manager SPA © 2017' is centered at the very bottom.

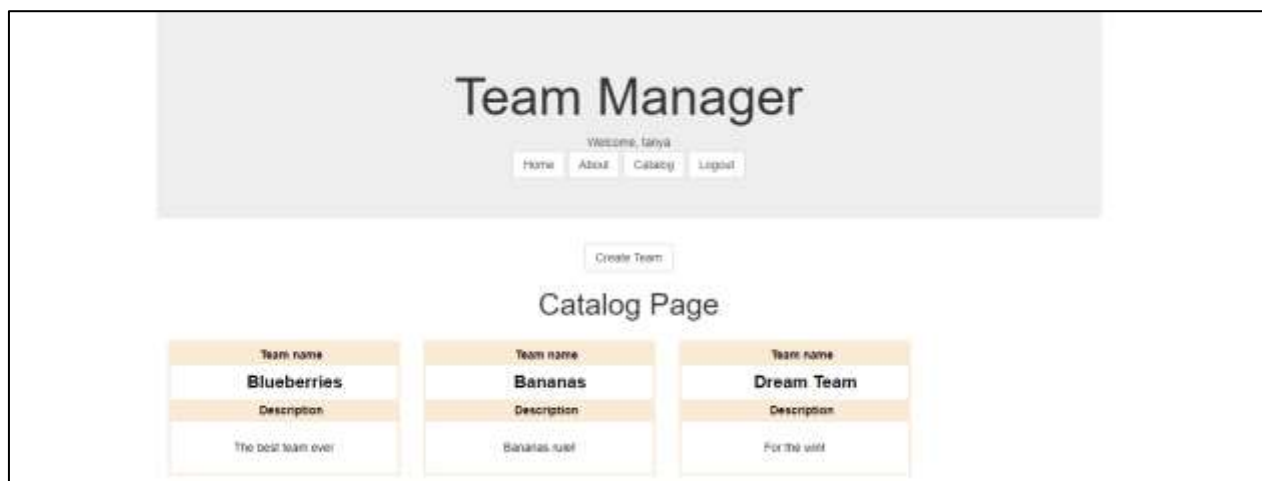
Login form

A screenshot of the 'Login Page' in the application. The title 'Login Page' is centered at the top. Below it, the label 'Username:' is followed by a text input field. Below that, the label 'Password:' is followed by a text input field. At the bottom, there is a 'Login' button. The footer text 'Team Manager SPA © 2017' is centered at the very bottom.

Home page view for a registered user. Note the header navigation has changed to reflect that.



Team browser.



Create team and edit team forms are identical.



Team details with option to join the team and a list of all current members.



Team management. If the user is a member, they can leave the team. If the user is the creator, they can edit it.