



# The Powerpuff Girls

Time: 3 to 4 hours

## Intro

At RTL we mainly develop *Single Page Applications* (SPA's) for web facing applications. This allows us to create responsive web sites with code completely focused on the UI, and separated from business logic on the server side. In this assignment we would like you to develop a simple SPA as a showcase of your frontend skills.

Since React is currently the UI framework of choice for RTL, we require that you use React to render UI components in this project. We realize however that React is not the only framework out there. You might have extensive experience with another framework like, for example, Vue. In that case you will probably pick up the basics of working with React quickly. However, getting started from scratch can be bit of a pain.

To have a level playing field for developers with and without React experience, we ask that you bootstrap your project with "Create React App" (<https://github.com/facebook/create-react-app>). This a starter kit, developed by Facebook, that does all scaffolding for a basic React application for you.

Default setup of a project created with Create React App is very basic. The documentation contains a lot of information on how to expand from there, with additional topics like unit tests and build tools. It is up to you to decide how to expand the basic project setup with extra functionality.

## Goal

We would like to see a SPA with two pages (routes): a details page for the TV show "Powerpuff Girls", and an episode detail page with information about specific episodes. Note that since this is an SPA, there should be no full page refresh when navigating between those routes.

The TV show page should **at least** display the following information (you're free to add more):

- Show title
- Show Description
- Show cover image
- Episode list. Every episode in the list should link to a details page for that specific episode

The episode detail page should contain at least:

- Episode title
- Episode Summary
- Episode cover image



We expect your application to have at least a page for one specific show: "The Powerpuff Girls". You are free to make other shows available as well.

As an example of how a TV show page might look, you can look at the TV Maze show pages here: <http://www.tvmaze.com/shows/6771/the-powerpuff-girls>.

## Data

Data in your application should be retrieved from the TV Maze API (<http://www.tvmaze.com/api>). TV Maze provides a free Rest API with data that strongly resembles the data we use internally to render our VOD platforms like Videoland and RTL XL. We expect that you use some kind of state management in you app. We encourage you to use an existing state management library, like Redux.

## Design

The "look and feel" (design) of the application is completely up to you. Rest assured, we will not judge you on your design skills. But there are some things we expect to see:

- Use CSS to style your components. You're encouraged to use preprocessor such as Sass or Less.
- The site should be responsive and should be usable both on desktop and (small) mobile screens. Define at least one breakpoint in your CSS.

## Deliverables

**Try to spend a maximum of 3-4 hours on this assignment. You are free to use more time, but we don't expect you to.**

Check in all your source code into a **public** Git repository (Bitbucket.org for example offers free unlimited private repositories) and make it available for the duration of the interview process.

Use branching at your discretion, but the end result has to be available in the master branch.

Your app should run from local web server (localhost). After checkout of the code, the only things we should do in order to view the end result is:

- Install dependencies
- Start local web server
- Open web browser

Documentation, especially on building and running the app, is appreciated. It is up to you to decide how comprehensive this documentation should be.



## Other considerations

- We think adding automatic tests is always a good idea, but not a requirement in this assignment.
- Your application should work in all modern browsers.
- We prefer code written in ES6 (use a transpiler for browser compatibility).
- English is the preferred language in code comments, documents, variable names, etc.

In general, choose a development methodology and tool set you feel most comfortable with, as long as you adhere to the requirements below.

## Summary of all requirements

- Use **React** to render UI components.
- Use "**Create React App**" to bootstrap your project.
- Setup **two routes** in your app, one for TV shows and one for episodes.
- Retrieve data from **TV Maze REST API**
- Setup **state management** and **data flow**
- Style your components with **CSS**, make sure that the end result is **responsive** and has **at least one breakpoint**.
- All code should be **checked in into a Git repository**
- **Documentation is optional but appreciated**, the same goes for comments in your code

For any technical questions, don't hesitate to contact us!