

# 1. Rest API calls with grid and form

- ExtJS documentation: <https://docs.sencha.com/extjs/6.5.3/index.html>
- Quick start: [https://docs.sencha.com/extjs/6.5.3/guides/quick\\_start/introduction.html](https://docs.sencha.com/extjs/6.5.3/guides/quick_start/introduction.html)
- ExtJS examples: <http://examples.sencha.com/extjs/6.5.3/examples/>

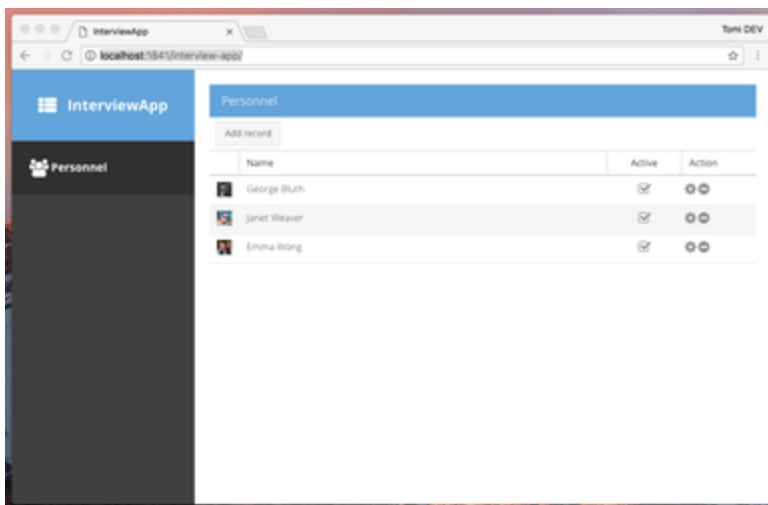
Setting up the development environment:

- Download ExtJS (<https://www.sencha.com/products/extjs/> > Try for free)
- Download Sencha Cmd (<https://www.sencha.com/products/extjs/cmd-download/>)
- Use sencha cmd to generate a simple app ([https://docs.sencha.com/cmd/guides/extjs/cmd\\_app.html](https://docs.sencha.com/cmd/guides/extjs/cmd_app.html)) and follow the instructions for Development Mode

## Assignment

Use the in previous step created demo app and modify it in a way it will only display a grid which fetches the data from a public REST API.

The result should look like this:



## Detailed requirements

### Show only Personnel grid

- Modify the created app, to only have one item in left navigation, called Personnel

### Model definition

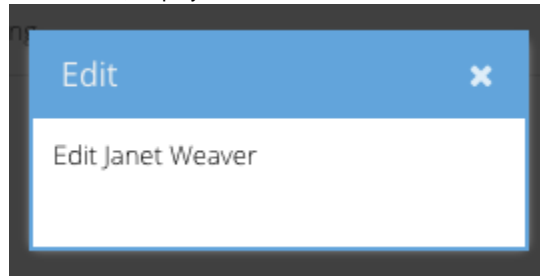
- Modify the already by default created Personnel.js model to reflect the following data:
  - id (int)
  - first\_name (string)
  - last\_name (string)
  - fullname (string), convert from first\_name and last\_name
  - avatar (string)
  - active (boolean), default value 1

### Store definition

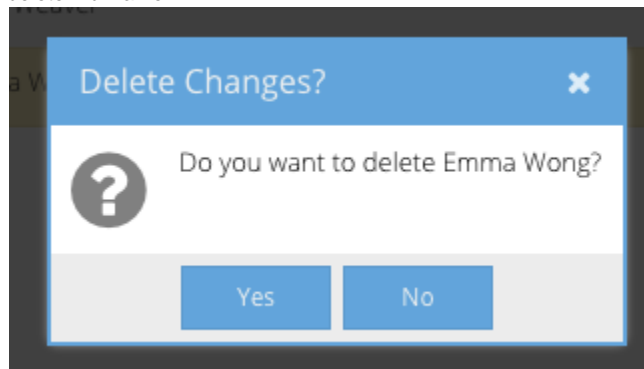
- Modify the already by default created Personnel.js store
- Use the REST proxy
- Public API endpoint <https://reqres.in/api/users> to read the users
  - A reader has to be defined

### Grid Definition

- Modify the already by default created List.js grid
- Needed columns:
  - Avatar
    - render the picture from url
  - Name
    - Normal column
    - Flex 1
  - Active
    - Check column
  - Action
    - Action column with two items
      - "Edit"
        - Click on edit, display an Alert with text "Edit <fullname>"



- "Delete"
  - Click on delete, display a yes/no window with ? Icon, "Yes" and "No" buttons and text "Do you want to delete <fullname>?"

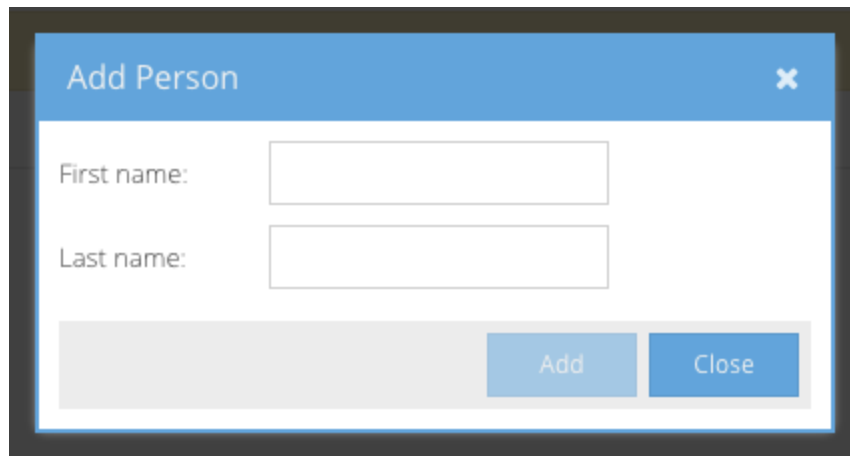


- Clicking on "Yes", should remove the record and close the window
  - Clicking on "No", should close the window
- Clicking anywhere in the grids row should display a Toast with the text: "You clicked on <fullname>"



## Add record to grid definition

- Add "Add record" button in the top toolbar of the grid
  - On click, window should be shown with a form inside
- Form has two fields:
  - First name (String)
  - Last name (String)

A screenshot of a web application dialog box titled "Add Person". The dialog has a blue header bar with the title and a close button (X). Below the header, there are two text input fields: "First name:" and "Last name:". At the bottom of the dialog, there is a light gray bar containing two buttons: "Add" and "Close". The "Add" button is blue and the "Close" button is also blue. The dialog is set against a dark gray background.

Add Person

First name:

Last name:

Add Close

- Two buttons: "Add" and "Close"
  - "Add"
    - Only enabled if both fields are filled out
    - Should add the new record to the store, close the window and new record needs to be visible in the grid
  - "Close"
    - Should close the window