jobR – A job search tracking tool

# A full-stack web app

## jobR is designed to provide project lifecycle resources to the job seeker

1. Offer a job search tool to query by job title (e.g., nurse, chemical engineer, etc.) in a geography by City, State or Zip Code
2. Enable the user to save a job to a database so they may return to the job’s URL.
3. Use AJAX calls to capture from the CareerOneStop API, provided by the US Dept of Labor:
   1. Job Essentials
      1. Job Title – API derived
      2. Company – API derived
      3. Location (City, St) – API derived
      4. Post URL – API derived
      5. Post Date – API derived
4. Store all the API data in an ORM model to extend the user’s capability to:
   1. Expand upon the job by adding
      1. Salary
      2. Job Description – user entry or paste
      3. Dates
         1. Applied Date
         2. Interview 1 Date/Time
         3. Interview 2 Date/Time
         4. Offer Date
      4. Notes – tabular with Date/Time
      5. Kanban-ize the job into boards
         1. Interested
         2. Applied
         3. Interview
         4. Offer

## User stories

| Story | Role | Goal | Motivation |
| --- | --- | --- | --- |
|  | As a… | I want to… | So I can… |
| Locate a job | User | Search for a job by title and geography | Limit the number of jobs I see at one time |
| Store a job locally | User | Capture information about each job, including: salary (if available), a description (if applicable), and any notes I might add | Help to determine if I have contacts at the company, whether to apply, and any other job-company information I would need to pursue this job |
| Add tasks associated with the job | User | Add a task related to an activity associated with this job, e.g., apply, write a cover letter, request an interview | Continue to pursue this job or to eliminate it as a possibility |
| View jobs associated with their Kanban board location | User | Move or categorize the job according to overall activities associated with it, i.e., interested, applied, interview scheduled | Isolate and readily find the jobs I am focused on pursuing |

## ER Diagram of the sequelize models

ER Diagram of models


Figure 1 - ER Diagram of jobR Models

## RESTful CRUD Routes

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CONTROLLER** | | | | | **MODEL** | **VIEW** | | |
| **Path** | **Method** | | **Action** | **EJS Page** | **Sequelize** | **Response Contents** | **View Logic** | **Links to...** |
| /widgets | 1 | GET | index |  | **findAll()** | Array of Objects [{widgets}] Status: 200 | Loop over array, show each item inserted into page | new, detail, edit\*, delete\* |
| 2\* | POST | create\* |  | **create()** | No data Status: 201 | Redirect to index | none (redirect) |
| /widgets/:id | 3 | GET | detail/show |  | **findOne(), findById()** | Single Object {widget} Status:200 | Show one object inserted into page | edit, delete |
| 4\* | PUT | update\* |  | **update()** | No data Status: 203 | Redirect to index or detail | none (redirect) |
| 5 | DELETE | delete\* |  | **destroy()** | No data Status: NONE | Redirect to index | none (redirect) |
| /widgets/new | 6(2\*) | GET | new |  | **--** | render('new') | Show empty create form | POST /widgets |
| /widgets/:id/edit | 7(4\*) | GET | edit |  | **findOne(), findById()** | Single Object {widget} render('edit') | Show update form populated with data | PUT /widgets/:id |