### Pre-employment checklist

- [] Add the deployed URL to repo settings

- [] Add a readme with:

- [] An embedded screenshot of the app

- [] Explanations of the \*\*technologies\*\* used

- [] \*\*Installation instructions\*\* for any dependencies

- [] Link to your \*\*user stories\*\* – who are your users, what do they want, and why?

- [] Link to your \*\*wireframes\*\* – sketches of major views / interfaces in your application

- [] Descriptions of any \*\*unsolved problems\*\* or \*\*major hurdles\*\* you had to overcome

User Stories

## Weigh-In

User will be able to create a weigh-in {OBJECT}

User will be able to add a new participant [CREATE a record]

## Participants

User will be able to conduct a weigh-in [adding participants’ weight using WEIGH-IN OBJECT]

User will be able to record the weight of a program participant [CREATE new weight]

User will be able to view a list of all program participants [Participants’ INDEX]

User will be able to add a new participant [CREATE new participant]

User will be able to show the weigh-in history of a program participant [SHOW method]

## Statistics

User will be able to conduct computations on weigh-in data

### Hints and Suggestions

\* \*\*Don’t get too caught up in too many awesome features\*\* – simple is always better. Build something impressive that does one thing well.

\* \*\*Design first.\*\* Planning with user stories & wireframes before writing code means you won't get distracted changing your mind – you'll know what to build, and you can spend your time wisely by just building it.

\* \*\*Don’t hesitate to write throwaway code\*\* to solve short term problems.

\* \*\*Read the docs for whatever technologies / frameworks / API’s you use\*\*.

\* \*\*Write your code DRY\*\* and \*\*build your APIs RESTful\*\*.

\* \*\*Commit early, commit often.\*\* Don’t be afraid to break something because you can always go back in time to a previous version.

\* \*\*Keep user stories small and well-defined\*\*, and remember – user stories focus on what a user needs, not what development tasks need accomplishing.

\* \*\*Write code another developer wouldn't have to ask you about\*\*. Do your naming conventions make sense? Would another developer be able to look at your app and understand what everything is?

\* \*\*Make it all well-formatted.\*\* Are you indenting, consistently? Can we find the start and end of every div, curly brace, etc?

\* \*\*Comment your code.\*\* Will someone understand what is going on in each block or function? Even if it's obvious, explaining the what & why means someone else can pick it up and get it.

\* \*\*Write pseudocode before you write actual code.\*\* Thinking through the logic of something helps.

---