Rubric turned in on time (5 points)
Remember to get it checked out by a TA!

Languages/Frameworks used (20 points)
- 10 - Learned/Used react.js frontend
- 10 - Learned/Used PHP Laravel backend
- 0 - MySQL Database

Functionality (51 points)
- Refreshing the page does not log a user out (3 points)
- Users can register, login, and logout (10 points)
- Passwords are hashed, salted, and checked securely (2 points)
- Employer-users can post/edit/delete their job postings (10 points)
- Employee-users can see the list of the job posting (4 points)
    - can view when click a certain job posting (4 points)
- Employee-users can search job posting by categories (e.g. job types, regions) (5 points)
- Employee-users can submit resume (4 points)
    - and employer-users can open the applicants' resume. (can reach out through contact info) (4 points)
- Employee-users can create their profile and allow employer-users to open their profiles. — job searching becomes bi-directional (5 points)
    - ex) Profile includes -age, job type preferences, wage preferences, location/time preferences.

Best Practices (5 points)
- Code is readable and well formatted (3 points)
- All pages pass the html validator (2 points)

Usability (4 points)
- Site is easy to navigate and intuitive (4 points)

Creative Portion (15 points)
> ideas for now
- Employers can rate employees — only employer-users can see the ratings
- Employees can rate employers — only employee-users can see the ratings
- Employer-users also can search potential employees by searching. Employer-users can see the matches, read profiles, and reach out.

Approved by Brad!