**Project 6 -- Final Project (87/100)**

**Documentation (10/10)**

· Readme is very thorough, easy to follow.

o Forgot to explain the ‘Controllers’ code

· In-code documentation is phenomenal.

o Especially the connections between routes.rb and your controller ruby files

· Git commit messages are adequate.

**Correctness + Testing (24/30)**

Positives.

· 23 tests all passed

· Everything else not in negatives works as expected.

Negatives. **-6**

· Ability to add the same student to a group multiple times.

· Ability to add the same group to a project multiple times.

· When logged in as a student, allows for negative ratings.

· Viewing evaluations when there are multiple groups leads to a **NoMethodError.**

· If a student is a member of multiple teams in different projects, it shows all their open evaluations regardless of project, instead of only the evaluations for the selected project.

· For a student, if a project is past its due date, it still shows the project in “Evaluations To-Do” and in “Past Evaluations”.

**Functionality/Features (20/25) -5**

· Only one class.

· Students can not edit their own information.

· A user can submit scores and comments for all their teammates.

· Admin can populate the class, and groups, and projects.

· Admin interface does give a useful view of the scores assigned within a team.

· There is login for admins and students.

· Admins and can add and delete users

· Users can be in multiple groups, and apart of multiple projects.

**Aesthetics (8/10) -2**

· Text is easy to read, and webpages are intuitive to navigate.

· Example text in text boxes is nice.

· CSS highlights on links are nice, accordion in evaluations page is nice.

· CSS for buttons and links within pages is not consistent.

· The header is not on every page, but it could be. Also, the header looks unprofessional.

**Style – HTML/CSS/JS (8/8)**

· Good

**Style - Ruby (7/7)**

· Good

**Design - HTML/CSS/JS (5/5)**

· Good

**Design - Ruby/Rails (5/5)**

· Good