FullstackAcademy / **jpfp-requirements**   Private

The requirements for JPFP.

☆ **0** stars   ⑂ **10** forks

| ☆ Star | ⊙ Watch ▾ |

‹› **Code**   ⊙ Issues   ⇅ Pull requests   ▶ Actions   ▦ Projects   📖 Wiki   ⊘ Security

⑂ main ▾   ⋯

🏔 **ThompsonHarris** formating   ⋯   5 days ago   🕘 **4**

View code

README.md

JPFP

Requirements

The requirements below are broken into separate **tiers**, which model the way we **recommend you approach the project**. That is, we recommend you complete (or complete the majority of) the requirements in Tier 1 before moving on to Tier 2, and so on. Of course, if you get stuck on a particular feature, we recommend moving on and trying another feature - don't sacrifice the good for the perfect!

[See some wireframes here!](#)

Optional Boilerplates to start! 🌶

- [Spicy](#)
- [Medium](#)
- [Mild](#)

Tiers

Tier 1: All Campuses and Students (20/61)

▼ Details

Frontend

- ☐ Write a component to display a list of all campuses (at least their names and images)
- ☐ Write a component to display a list of all students (at least their names)
- ☐ Write a campuses sub-reducer to manage campuses in your Redux store
- ☐ Write a students sub-reducer to manage students in your Redux store
- ☐ Display the all-campuses component when the url matches `/campuses`
- ☐ Display the all-students component when the url matches `/students`
- ☐ Add a links to the navbar that can be used to navigate to the all-campuses view and the all-students view

Backend

- ☐ Write a route to serve up all students

- ☐ Write a route to serve up all campuses

  - Write a `campuses` model with the following information:

    - ☐ name - not empty or null
    - ☐ imageUrl – with a default value
    - ☐ address - not empty or null
    - ☐ description - extremely large text

  - Write a `students` model with the following information:

    - ☐ firstName - not empty or null
    - ☐ lastName - not empty or null
    - ☐ email - not empty or null; must be a valid email
    - ☐ imageUrl – with a default value
    - ☐ gpa - decimal between 0.0 and 4.0

- ☐ Students may be associated with at most one campus. Likewise, campuses may be associated with many students

Seed

) Running the seed file creates campuses and students for demonstration purposes

Congrats! You have completed your first vertical slice! Make sure to `commit -m "Feature: Get All Campuses and Students"` before moving on (see RUBRIC.md - points are awarded/deducted for a proper git workflow)!

Tier 2: Single Student and Single Campus (12/61)

▼ Details

Frontend

- Write a component to display a single campus with the following information:

  - The campus's name, image, address and description
  - A list of the names of all students in that campus (or a helpful message if it doesn't have any students)

  Display the appropriate campus's info when the url matches `/campuses/:campusId`

  Clicking on a campus from the all-campuses view should navigate to show that campus in the single-campus view

- Write a component to display a single student with the following information:

  - The student's full name, email, image, and gpa
  - The name of their campus (or a helpful message if they don't have one)

  Display the appropriate student when the url matches `/students/:studentId`

  Clicking on a student from the all-students view should navigate to show that student in the single-student view

  Clicking on the name of a student in the single-campus view should navigate to show that student in the single-student view

  Clicking on the name of a campus in the single-student view should navigate to show that campus in the single-campus view

Backend

  Write a route to serve up a single campus (based on its id), *including that campuses' students*

  Write a route to serve up a single student (based on their id), *including that student's campus*

Congrats! You have completed your second vertical slice! Make sure to `commit -m`
`"Feature: Get Single Campus and Student"` before moving on (see RUBRIC.md -
points are awarded/deducted for a proper git workflow)!

Tier 3: Adding a Campus and Adding a Student (10/61)

▼ Details   ●

Frontend

❚ Write a component to display a form for adding a new campus that contains inputs
for *at least* the name and address.

❚ Display this component as part of the all-campuses view, alongside the list of
campuses

- Submitting the form with a valid name/address should:

  ❚ Make an AJAX request that causes the new campus to be persisted in the
  database
  ❚ Add the new campus to the list of campuses without needing to refresh the page

☐ Write a component to display a form for adding a new student that contains inputs
for *at least* first name, last name and email

☐ Display this component as part of the all-students view, alongside the list of students

- Submitting the form with a valid first name/last name/email should:

  ☐ Make an AJAX request that causes the new student to be persisted in the
  database
  ☐ Add the new student to the list of students without needing to refresh the page

Backend

◼ Write a route to add a new campus
☐ Write a route to add a new student

Congrats! You have completed your third vertical slice! Make sure to `commit -m`
`"Feature: Add Campus and Student"` before moving on (see RUBRIC.md - points are
awarded/deducted for a proper git workflow)!

Tier 4: Removing a Campus and Removing a Student (8/61)

▶ Details

Tier 5: Updating a Campus and Updating a Student (11/61)

▶ Details

Bonus Tier: Finishing Touches (15 EC)

▶ Details

Releases

No releases published

Packages

No packages published

Contributors   2

leafoflegend Eliot Szwajkowski

ThompsonHarris Thompson Harris