

Sprint 3 Retrospective Review Report

1. Links

- [Pivotal Tracker](#)
- [GitHub Repository](#)
- [Heroku Deployment](#)

2. Dates of the Sprint

- **Start Date:** 2/24/2025
- **End Date:** 3/7/2025

3. Team Contribution

Name	Role	Points Completed
Leo Gonzalez	Product Owner (PO)	4
Owen Schultz	Scrum Master (SM)	4
Olivia Lee	Developer (Dev)	5
Casey Kung	Developer (Dev)	4
Sam Lightfoot	Developer (Dev)	4
Ethan Nguyen	Developer (Dev)	4

4. Sprint Goal

Students can track their strengths and weaknesses in their report card, receive answer explanations, and view a leaderboard ranking based on their correct answers. Instructors can view the overall progress of the students in the class, and all users can interact easily with styled user interfaces and navigate easily with a navigation bar.

5. Sprint Achievements

This sprint, we were able to fully meet our sprint goal and increase the functionality of our project. Students are now able to see their strengths and weaknesses on their report card, see an explanation of correctness

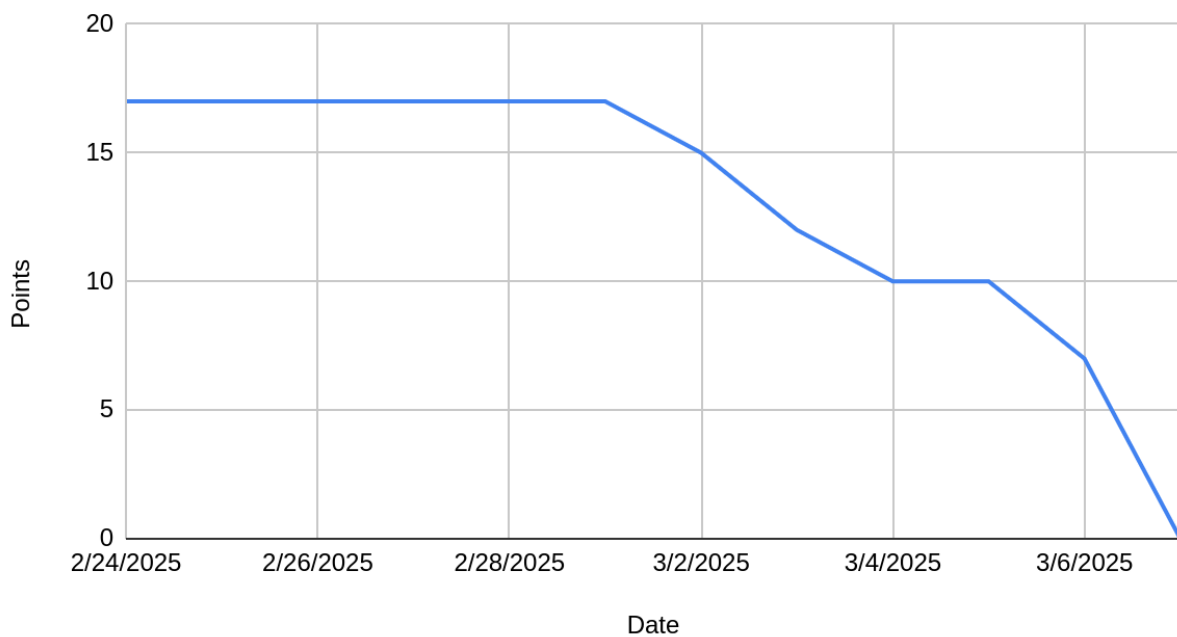
when they answer a question, answer a definition question with the term rather than the whole definition, and view a leaderboard that compares their correct answers with other students in the class. Instructors can view a summary report of their students progress, and all users are able to navigate with a navigation bar and view styled webpages that make the website more pleasing to look at.

6. Sprint Backlog Items and Status

Backlog Item	Status
Identify Strengths and Weaknesses	Completed
Explanation of correctness	Completed
Leaderboard	Completed
Navigation Bar	Completed
Styling	Completed
Instructor View of Summary Report	Completed
Change Definition Questions	Completed

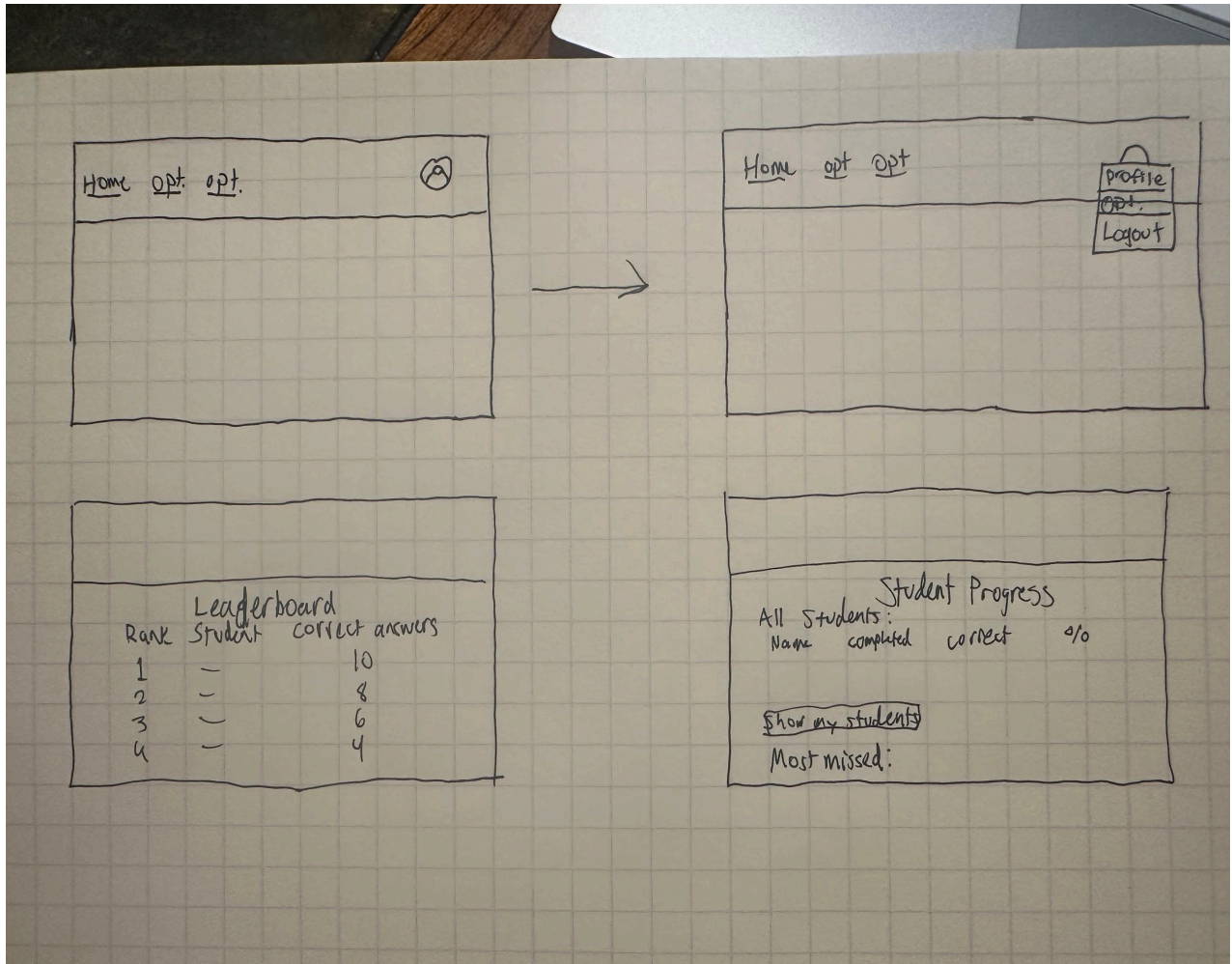
7. Burndown Chart

Sprint 3 Burndown



- **Notes:** In this sprint developers had a busy first week and had difficulties with finishing their tasks early. There was also a miscommunication about due dates that confused some developers and caused them to push off work. Developers worked hard to finish their tasks before the deadline.

8. Design Diagrams



9. Documentation of Changes

- **Changes to Design:**
 - Changed a few navigation bar page names to better represent page content
 - User icon in top right corner is not just to access profile, also accesses logout function
- **Changes to Cucumber Tests:**
 - Explanation of Correctness feature file was modified to account for existing steps that could be utilized and to account for user login

- Leaderboard feature file was modified to divide user feedback scenario into different outcomes for unranked, ranked, and not a student
- Navigation Bar feature file was modified to account for new pages added and also divides scenarios into instructor, student, and admin views
- Strengths and Weaknesses feature file was modified by adding a list of submissions for each scenario in order to properly test the functionality; performance trends were deleted because they do not match with our current design and “report card” terminology was deleted as well

10. Evaluations of Code and Test Quality

Smells

Control Parameter

This was found in our User implementation. We have a long list of parameters, hindering readability, so to fix this we should reduce the number of parameters in our User implementation.

Data Clump

This was found in our OAuth implementation. It means that we have multiple variables used together in multiple places in a program, making code harder to understand, maintain, and change. To fix this we should combine variables into a single object.

Duplicate Code

This was found in numerous different files. To fix this, we should make a function for the duplicate code and call the function in the multiple areas that the duplicate code was found in.

Duplicate Method Call

This was found in numerous different files. To fix this we would need to store the value called by the function and then use this to manipulate data instead of calling the function each time.

Feature Envy

This was found in numerous different files. This occurs when a class uses another class's methods excessively. To fix this, we must extract the method from the other class and move it to the class that uses it excessively.

High Complexity

This was found in numerous different files. This occurs when a function is too convoluted and can be fixed by breaking it up into multiple functions.

Instance Variable Assumption

This was found in numerous different files because an instance variable wasn't properly initialized. This can be fixed by making sure to initialize all variables.

Irresponsible Module

This was found in numerous different files. This occurs when a module is responsible for too many things, and can be fixed by breaking it down into smaller modules.

Nil Check

This was found in numerous different files. It indicates poor handling of nil values through the use of too many `nil?` checks, so code should be adjusted to reduce this and use guard clauses early on.

Too Many Instance Variables

This was found in numerous different files. If a class has too many instance variables, it has too many responsibilities and should be broken up or the amount of instance variables should be reduced.

Too Many Statements

This occurs when a method performs too many tasks and can be fixed by breaking down methods into smaller, focused methods.

Uncommunicative Variable Name

This was found in numerous different files. To fix this, we must adjust the variable names so they are more descriptive.

Utility Function

This was found in numerous different files. Utility functions are too generic and can lead to duplication or a lack of context, so they should be encapsulated within appropriate classes/modules with clearer context and responsibility.

Very High Complexity

This was found in numerous different files. This occurs when a function is extremely convoluted and can be fixed by breaking it up into multiple functions to reduce complexity.

Style Offenses

- Style issues were addressed with rubocop, there were no issues that could not be fixed either by “rubocop -a” or reading the offenses and manually fixing them.

Coverage

- Cucumber
 - 41 scenarios (11 undefined, 30 passed)
 - 254 steps (1 skipped, 14 undefined, 239 passed)
 - Line Coverage: 96.53% (362 / 375)
- Rspec
 - 136 examples, 0 failures
 - Line Coverage: 99.47% (373 / 375)
- **Details:** With our efforts to write effective tests we were able to get above 90% test coverage for both Cucumber and RSpec testing.

11. Customer Meeting

- **Meeting Date:** 2/28
- **Meeting Time:** 2:45
- **Meeting Place:** Zoom
- **Summary:** Ran through styling options and got feedback, client expressed that the navigation bar is her most important concern at this time.

- **Meeting Date:** 3/16
- **Meeting Time:** 2:15
- **Meeting Place:** Zoom
- **Summary:** Showed sprint 3 MVP and all features. She was very happy with styling and our features so far. Client suggested making changes to the leaderboard to sort by accuracy and require students to answer a minimum number of questions before appearing on the leaderboard

12. BDD & TDD

Cucumber steps:

Feature: Admin promotion

As an administrator,
so that I can manage accounts with professor privileges,
I want to be able to promote tame accounts to professor status

Scenario: Admin views list of accounts

Given I am logged in as an administrator
And I click "View Accounts"
Then I should be on the accounts page
And I should see a list of all accounts and their roles

Scenario: Admin promotes an account to professor

Given I am on the view accounts page
And there is an account with the student role
When I select instructor in the role dropdown
And I click Update Role in that row
Then the selected account will have the instructor role

Feature: Explanation of Correctness

Background:

Given a predefined question exists
And I am logged in with a valid tamu email
And I visit the practice problems page
And I select topic "Velocity"
And I select question type "Free Response"
And I press "Submit"
Then I should be on the problem generation page

Scenario: Understanding the correct answer

And I input the correct solution
When I click "Submit"

Then I should see the explanation for this answer

Scenario: Answering the question incorrectly

And I input an incorrect solution

When I click "Submit"

Then I should see the explanation for this answer

Scenario: Trying Another Problem

And I input the correct solution

When I click "Submit"

When I click "Try Another Problem"

Then I should be on the problem generation page

Feature: Definition Questions

Background:

Given I am logged in as a student

Scenario: Selecting the term based on its definition

When I am on the question page

And I have a definition question

Then I should be able to select a term to match a definition

Feature: Instructor creates a custom template

As an instructor

I want to create reusable templates

So that I can quickly build assignments or modules

Background:

Given I am logged in as an instructor

Scenario: Successfully create a custom template

Given I am on the instructor home page

When I click on "Custom Template"

And I select "Velocity" from "Select Topic"

And I select "Definition" from "Select Type"

And I fill in "Question Template Text" with "My Custom Template"

And I fill in "Equation" with "F / m"

And I fill in "Variables (comma separated)" with "F, m"

And I fill in "Answer Format" with "F / m"
And I press on the button: "Create Question Template"
Then I should see the string "Question template created successfully!"

Feature: Instructor Summary Report

As an instructor

I want to view summary data about my students and common trouble areas
So that I can focus my teaching efforts

Background:

Given I am logged in as a person instructor

Scenario: View summary report of all students

When I navigate to the instructor summary report page
Then I should see star "All Students"
And I should see star "Student Progress Summary"
And I should see star "Topic with the Most Missed Questions"

Feature: Student Leaderboard

Scenario: Students view the leaderboard and correct rank

Given "student1" has completed 10 problems with 8 correct answers
Given "student2" has completed 15 problems with 12 correct answers
Given "student3" has completed 7 problems with 5 correct answers
Given I am logged in with a valid tamu email
Given a predefined question exists
And I visit the practice problems page
And I select topic "Velocity"
And I select question type "Free Response"
And I press "Submit"
And I input the correct solution
When I click "Submit"
When I visit the leaderboard page
Then the system displays "student2" then "student1" then "student3" on the leaderboard
Then the system informs the student "Your current rank is 4"

Scenario: Student with no practice sees leaderboard info prompting to practice

Given I am logged in with a valid tamu email
When I visit the leaderboard page

Then the system informs the student "You are not currently on the leaderboard. Practice some problems to join the leaderboard."

Scenario: Non-student user does not see a rank display

Given I have an instructor account

When I visit the leaderboard page

Then the system does not have any message about rank or system

Feature: Login as Admin

As an admin,

so that I can access the admin homepage,

I want to log in using my @Tamu Google account.

Scenario: Login as admin with valid tamu email

Given I am on the welcome page

And I have an admin account

And I click "Login with Google"

Then I will be on the Admin Homepage

Scenario: Login as admin with invalid credentials

Given I am on the welcome page

When I have an invalid admin account

And I click "Login with Google"

Then I should be on the welcome page

And I should see a message "Please login with an @tamu email"

Feature: Login as Instructor

As an instructor,

so that I can access the instructor homepage,

I want to log in using my @Tamu Google account.

Scenario: Login as instructor with valid tamu email

Given I am on the welcome page

And I have an instructor account

And I click "Login with Google"

Then I will be on the Instructor Homepage

Scenario: Login as instructor with invalid credentials

Given I am on the welcome page

When I have an invalid instructor account
And I click "Login with Google"
Then I should be on the welcome page
And I should see the message "Please login with an @tamu email"

Feature: Login as Student

As a student,
so that I can access the student homepage,
I want to log in using my @Tamu Google account.

Scenario: Login with valid @tamu Google account

Given I am on the welcome page
When I have a valid @tamu Google account
And I click "Login with Google"
Then I will be on the Student homepage

Scenario: Login with non @tamu Google account

Given I am on the welcome page
When I have a non-tamu Google account
And I click "Login with Google"
Then I should be on the welcome page
And I should see the message "Please login with an @tamu email"

Scenario: Try to access the student homepage while not being logged in

Given I am not logged in
When I try to go to the student homepage
Then I should be on the welcome page
And I should see the message "You must be logged in to access this section."

Feature: Logout

As a user,
so that I can finish using the platform,
I want to log out of my account.

Scenario: Logout from student homepage

Given I am logged in as a student
And I am on the student homepage
When I click "Logout"
Then I will be on the welcome page

And I should see the message "You are logged out."

Scenario: Logout from student homepage

Given I am logged in as an instructor

And I am on the instructor homepage

When I click "Logout"

Then I will be on the welcome page

And I should see the message "You are logged out."

Feature: Navigation Bar

Scenario: Display navigation bar

Given I am logged in with a valid tamu email

When I am on any page of the application

Then I should see a navigation bar at the top of the screen

Scenario: Navigation bar contains links for student view

Given I am logged in as a student

And I am on any page of the application

Then the student navigation bar should have links to "Home", "Profile", "Logout", "Problems", "Practice Tests", "Leaderboard", and "Progress"

Scenario: Navigation bar contains links for instructor view

Given I am logged in as an instructor

And I am on any page of the application

Then the instructor navigation bar should have links to "Home", "Profile", "Logout", "Custom Template", and "Student Progress Summary"

Scenario: Navigation bar contains links for admin view

Given I am logged in as an admin

And I am on any page of the application

Then the admin navigation bar should have links to "Home", "Profile", "Logout", and "View Accounts"

Scenario: Clicking on a navigation link routes to the correct page

Given I am logged in with a valid tamu email

When I am on any page of the application

When I click on the "Home" link in the navigation bar

Then I should be redirected to the "Home" page

Scenario: Active page is highlighted in the navigation bar

Given I am logged in with a valid tamu email

When I am on the "Home" page

Then the "Home" link on the navigation bar should be bold and underlined

Feature: Practice Test Functionality

As a user

So that I can take a ENGR 216 practice test

I want to access a system that allows me to attempt and submit a practice test with feedback

Background:

Given a predefined question exists

Given another predefined question exists

Given I am logged in with a valid tamu email

Given I visit the practice tests page

Scenario: Successfully generate and complete a practice test

Given I select the topics "Velocity" and "Accuracy and precision of measurements, error propagation"

And I select the question types "Definition" and "Free Response"

And I submit the form

Then I should be redirected to the practice test generation page

Given I should see multiple randomly selected problems

And I should see multiple input fields to submit my answers

When I answer all the questions

And I submit my practice exam

Then I should see my score

And I should receive feedback on my test answers

Scenario: Attempt to generate a practice test without any topics or question types selected

Given I don't select any topics

And I don't select any question types

When I submit the form

Then I should see the message "No questions available for the selected criteria."

And I should be redirected to the practice test form page

Feature: Selecting topics and question types

Scenario: User selects topics and question types in the problem form

Given I am logged in with a valid tamu email
And I visit the practice problems page
When I select the topics "Velocity" and "Acceleration"
And I select the question types "Definition" and "Free Response"
And I submit the form
Then I should be redirected to the problem generation page
And I should see "Velocity" and "Acceleration"
And I should see "Definition" and "Free Response"

Feature: Practice Problem Generation

As a student,
so that I can practice problems relevant to the selected concept,
I want to be shown a randomly generated example problem after selecting a category.

Background:

Given a predefined question exists

Scenario: Unique questions within a set of problems

Given I am logged in with a valid tamu email
And I visit the practice problems page
And I select topic "Velocity"
And I select question type "Free Response"
And I press "Submit"
Then I should be on the problem generation page
And I should see a randomly selected problem
And I should see an input field to submit my answer

Scenario: Displaying problem with randomized values

Given I am logged in with a valid tamu email
And I visit the practice problems page
And I select topic "Velocity"
And I select question type "Free Response"
And I press "Submit"
Then I should be on the problem generation page
Given a problem with variables "u", "a", and "t"
When the problem is displayed
Then the values for "u", "a", and "t" should be randomly generated
And the question text should include these values

Feature: Question Templates

As a teacher,

I want to be able to add a new practice question template,

So that I can continuously update the questions that are given to students as the curriculum changes.

Background:

Given I am logged in as an instructor

Scenario: Teacher wants to add a new practice question template

When I click the add a new practice question button

Then I am prompted to enter the template and submit

Feature: Student selects an instructor

As a student user

I want to view my profile

And optionally assign an instructor to my account

So that my instructor can see my progress

Background:

Given I am logged in as a student

Scenario: View my profile

When I visit my profile page

Then I should see b "Howdy" in the page content

And I should see b "No instructor selected"

Scenario: Successfully save a valid instructor

Given an instructor user exists

When I visit my profile page

And I select that instructor from the dropdown

And I press star "Save Instructor"

Then I should see b "Instructor saved successfully!"

And I should see the instructor's name in my profile

Scenario: Fail to save an instructor when none selected

Given an instructor user exists

When I visit my profile page

And I do NOT select any instructor from the dropdown

And I press star "Save Instructor"
Then I should see b "Failed to save instructor."
And I should see b "No instructor selected"

Feature: Report Card Highlights Strengths and Weaknesses

Scenario: Display strengths and weaknesses on the Progress page

Given I am logged in
And the following submissions exist:
Topic	Question	Correct?
Velocity	What is gravity?	false
Velocity	What is gravity?	false
Velocity	What is gravity?	false
Velocity	What is gravity?	true
Acceleration	What is gravity?	true
Acceleration	What is the formula for acceleration	true
Acceleration	What is the formula for acceleration	true
Acceleration	What is the formula for acceleration	true
When I am on the "Progress" page
Then my strongest topics with an accuracy of 85 or higher should be highlighted as strengths
And my weakest topics with an accuracy of 60 or lower should be highlighted as weaknesses

Scenario: No topics above 80% accuracy

Given I am logged in
And the following submissions exist:
Topic	Question	Correct?
Velocity	What is gravity?	false
Velocity	What is gravity?	false
Velocity	What is gravity?	false
Velocity	What is gravity?	true
Acceleration	What is gravity?	false
Acceleration	What is the formula for acceleration	false
Acceleration	What is the formula for acceleration	true
Acceleration	What is the formula for acceleration	true
When I am on the "Progress" page
Then I will not have topics highlighted as strengths

Scenario: No topics below 60% accuracy

Given I am logged in

And the following submissions exist:

Topic	Question	Correct?
Velocity	What is gravity?	true
Velocity	What is gravity?	true
Velocity	What is gravity?	true
Velocity	What is gravity?	true
Acceleration	What is gravity?	true
Acceleration	What is the formula for acceleration	true
Acceleration	What is the formula for acceleration	true
Acceleration	What is the formula for acceleration	true

When I am on the "Progress" page

Then I will not have topics highlighted as weaknesses

Feature: Student View Personal Progress

As a student

So that I can see what areas I need to work on

I want to be able to view the questions I got right and wrong

Background:

Given I am logged in

Scenario: Student wants to view question progress

Given the following submissions exist:

Topic	Question	Correct?
Velocity	What is gravity?	false
Velocity	What is gravity?	false
Velocity	What is gravity?	false
Velocity	What is gravity?	true
Acceleration	What is gravity?	true
Acceleration	What is the formula for acceleration	false
Acceleration	What is the formula for acceleration	false
Acceleration	What is the formula for acceleration	true

When I navigate to the progress page

Then I should see my overall performance

And I should see my performance by topic

Feature: Submit Solution Attempt

As a student,

so that I can test my understanding,

I want to submit my attempt at solving a problem as a computed numeric value.

Background:

- Given a predefined question exists
- And I am logged in with a valid tamu email
- And I visit the practice problems page
- And I select topic "Velocity"
- And I select question type "Free Response"
- And I press "Submit"
- Then I should be on the problem generation page

Scenario: Student submits correct answer

- And I input the correct solution
- When I click button "Submit"
- Then I should see "Correct!"
- And the user's counters should show a correct submission
- Then I print the user counters

Scenario: Student submits incorrect answer

- And I input an incorrect solution
- When I click button "Submit"
- Then I should see "Incorrect!"
- And the user's counters should show an incorrect submission
- Then I print the user counters

Rspec Tests:

AdminController

- GET #index
 - assigns the correct instance variables and renders the index template

AdminRolesController

- GET #index
 - assigns @users and @roles
- PATCH #update_role
 - with valid parameters
 - updates the user role and redirects to admin roles path
 - with a non-existent user
 - raises an ActiveRecord::RecordNotFound error

ApplicationController

```
#current_user
  when session[:user_id] is present and valid
    returns the user associated with the session user_id
  when session[:user_id] is present but invalid
    returns nil when no user exists with the session user_id
  when session[:user_id] is not present
    returns nil

#logged_in?
  when current_user is present
    returns true
  when current_user is nil
    returns false

#require_login
  when user is logged in
    allows the action to proceed
  when user is not logged in
    redirects to the welcome page with an alert
```

InstructorHomeController

```
GET #index
  when the user is an instructor
    returns a successful response
    renders the index template
    assigns the current_user as @instructor
  when the user is not signed in
    redirects to the sign-in page

GET #custom_template
  when the user is an instructor
    returns a successful response
    renders the custom_template template
  when the user is not an instructor
    redirects to the root path
    sets an alert message

GET #summary
  when the user is an instructor
    assigns the correct students
    assigns the most missed topic
```

- when the user is not an instructor
redirects to the root path

POST #create_template

- when the user is an instructor
creates a new question template
- when the user is not an instructor
does not create a new question template
- redirects to the root path
- sets an alert message

PracticeTestsController

POST #create

- when submitting topics and types
saves the selected topics and question types in session
- handles empty selections

GET #practice_test_generation

- when generating practice tests
fetches the correct topics and question types based on session
- retrieves a valid question set
- stores questions in session
- when no questions match the criteria
redirects to the practice test form

POST #submit_practice_test

- when submitting a correct answer
redirects to the results page
- records a correct submission
- stores test results in session
- when submitting an incorrect answer
redirects to the results page
- records an incorrect submission
- stores test results in session
- when submitting a text-based answer (case-insensitive)
accepts correct answers with different letter cases
- rejects incorrect answers

GET #result

- renders the results page with correct variables

#generate_random_values

- generates random values for all variables

#format_template_text

formats the template text with given values

#evaluate_equation

evaluates the equation with given values

returns nil when equation has invalid syntax

ProblemsController

POST #create

when submitting topics and types

saves the selected topics and question types in session

handles empty selections

#generate_random_values

generates random values for all variables

#format_template_text

formats the template text with given values

#evaluate_equation

evaluates the equation with given values

returns nil when equation has invalid syntax

returns nil when values are empty

returns nil when equation is empty

handles missing variables

handles division by zero

GET #problem_generation

when generating problems with selected topics and types

fetches the correct topics and question types based on session

retrieves a valid question

stores question ID in session

when no questions are found

redirects back to the problem form page

POST #submit_answer

when submitting a correct answer

redirects to the problem_generation page with correct data

creates a submission and updates user data

when submitting an incorrect answer

redirects to the problem_generation page with correct data

creates a submission with correct being false

SessionsController

GET #logout

resets the session and redirects to welcome_path with a notice

GET #omniauth

- when UID and provider are present
- with a valid @tamu.edu email
- creates a new user with student role and redirects to student_home_path
- creates a new user with instructor role and redirects to instructor_home_path
- creates a new user with admin role and redirects to admin_path
- with an invalid email domain
- redirects to welcome_path with an alert
- when UID or provider is missing
- redirects to welcome_path with an alert
- when there is an error saving the user
- redirects to welcome_path with an alert

StudentHomeController

GET #index

- assigns @current_user to the current user
- assigns @profile_path to the user profile path
- assigns @logout_path to the logout path
- assigns @progress_path to the progress path
- responds successfully with an HTTP 200 status code
- renders the index template

UsersController

GET #show

- when the user exists
- assigns the requested user to @current_user
- returns a successful response
- when the user does not exist
- raises an ActiveRecord::RecordNotFound error

#full_name

- returns the concatenated full name
- handles nil first_name
- handles nil last_name
- handles both first_name and last_name being nil

POST #save_instructor

- when a valid instructor is selected
- updates the current_user with the selected instructor
- when an invalid instructor ID is given
- does not update the user and sets an error flash message

GET #progress

- when user has submissions
 - assigns user
 - assigns total_submissions
 - assigns correct_submissions
 - assigns correct accuracy
 - assigns topic names correctly
- when use has no submissions
 - assigns user
 - assigns total_submissions to 0
 - assigns correct_submissions to 0
 - assigns correct accuracy to 0
 - assigns topic names to be empty

WelcomeController

GET #index

- when the user is logged in
 - redirects to student_home_path with a notice
- when the user is not logged in
 - does not redirect and renders the index template

ApplicationHelper

#active_class

- returns 'active-link' when the current page matches the given path
- returns nil when the current page does not match the given path

Question

valid information

- is valid
- is invalid without a topic_id
- is invalid without a type_id
- is invalid without a template_text
- is invalid with a short template_text

table associations

- example at ./spec/models/question_spec.rb:35
- example at ./spec/models/question_spec.rb:36
- belongs to a topic
- belongs to a type

Submission

valid information

- is valid with a user, question, and correct value
- is invalid without a user
- is invalid without a question
- is invalid without a value for 'correct'

table associations

- example at ./spec/models/submission_spec.rb:32
- example at ./spec/models/submission_spec.rb:33

after_create update_user_stats

- increments total submissions
- increments correct submissions if correct
- does not increment correct submissions if correct

User

valid information

- is valid
- is invalid without a first name
- is invalid without a last name
- is invalid without an email
- is invalid with a duplicate email
- defaults role to student

table associations

- example at ./spec/models/user_spec.rb:39
- example at ./spec/models/user_spec.rb:40
- example at ./spec/models/user_spec.rb:41

#full_name

- returns full name correctly

#update_user_submissions

- increments total submissions
- increments correct submissions when correct
- does not increment correct submissions when incorrect

#total_accuracy

- correctly calculates the percentage of questions answered correctly
- correctly handles zero for the total_submissions

#submissions_by_topic

- when there are many submissions
- groups submissions by submitted topics and counts correct answers
- when there are less submissions

groups submissions by submitted topics and counts correct answers
when there are no submissions for a user
returns empty
does not use other users submissions