# SE/COM S 3190 - Construction Of User Interfaces
# Assignment 3

**Total Points:** 100
**Published On:** Apr 8, 2025, 12:00 AM CST
**Due Date:** Apr 15, 2025, 11:59 PM CST

# Contents

Prepared by: ***Pranava Sai Maganti***

This is a team assignment. Please review the rubric carefully, as points will be awarded for effective communication and collaboration. Ensure tasks are evenly distributed between both members. Plan accordingly and complete the project the right way.

If you need help, try office hours—the schedule is on **_Canvas_**. Lots of help is also available through the Piazza discussions. Please start the assignment as soon as possible to get your questions answered right away.

# 1 Overview

This document outlines the requirements, expectations, and grading rubric for Assignment 03 in COM S/SE 3190. In this assignment, you will develop a simplified online learning platform, inspired by platforms like Udemy and Coursera. The client is a small online education provider that offers a variety of courses. The platform should allow users to browse available courses, add them to their cart, and confirm their purchase. This project will be developed using React and styled with Tailwind CSS, with a focus on creating a fully functional, single-page application.

The assignment is worth 5% of the final grade and is a team assignment. Students are expected to design and implement the following key views:

1. **Browse Courses View:**

   - Displays a list of available courses, each with a image, title, instructor, description, price, and an **Add to Cart** button.

2. **My Cart View:**

   - A mandatory view where users can view the courses added to their cart, remove items, and proceed to checkout.
   - The cart will display the total cost of selected courses, and users will be able to continue with the purchase or return to browsing courses.

3. **Confirmation View:**

   - Displays the order summary, including the courses purchased and the total cost.
   - Users can return to the Browse Courses view after completing their order.

The assignment is due on April 15th, 2025, at 11:59 PM. Late submissions will be accepted until April 18th, 11:59 PM. Students are expected to work in teams on this assignment, and are encouraged to seek clarification and feedback during scheduled office hours or via email.

The grading will be based on the successful implementation of all required views and features, including proper navigation, state management, and adherence to the specified design guidelines using Tailwind CSS.

---

# 2 Implementation Details

This section outlines the steps and considerations for implementing the Browse Courses, My Cart, Payment Gateway and Order Summary views for the online learning platform. Each view needs to function seamlessly within a single-page React application, styled with Tailwind CSS. The implementation should ensure that users can smoothly interact with the platform, from browsing courses to completing a purchase.

## 2.1 Browse Courses View

The Browse Courses view is where users can explore and browse through available courses to enroll in. The view must provide a clear, organized layout of courses, each with essential details such as a title, description, and price.

- **Course Listings**: Each course should be displayed as a card, showing the course image, title, instructor, a brief description, and the price. Include a button to allow users to enroll in the course, which will add the course to their cart.

- **Tailwind CSS Styling**: Use Tailwind's grid and flexbox utilities to create a clean, responsive design that adapts well to different screen sizes. The layout should be visually appealing, easy to navigate, and user-friendly.

## 2.2 My Cart View

The My Cart view is where users can manage the items they have added to their cart. This view should display a list of the courses in the cart, allowing users to modify quantities, remove courses, and proceed to checkout.

- **Cart Items**: Display each course in the cart with its title, price, and instructor. Allow users to remove it entirely from the cart.

- **Total Price**: The cart should show the total cost of the selected courses, including any applicable taxes or fees.

- **Proceed to Checkout**: Include a button that lets users proceed to the Confirmation View, where they can review their order. Also, provide an option to return to the Browse Courses View if they wish to add more courses.

- **Tailwind CSS Styling**: Ensure the cart is easy to read and navigate, using clear borders, buttons, and spacing. Tailwind's utilities for layout, typography, and color can help make the cart visually distinct and simple to interact with.

## 2.3 Payment Gateway

The Payment Gateway View is the final step in the checkout process, where users can confirm their order and provide payment information. This view calculates the total of the courses in the cart, and collects the necessary payment details.

- **Order Review**: Display the list of courses that the user has added to the cart, showing the course name, price, and the total price for each course. This section should include the total cost for all items, including taxes and fees.

---

- **User Information**: This view must collect user information for fields such as name, email, and shipping address (if applicable) to complete the payment process.

- **Payment Fields**: Include fields to collect payment information, such as:

  - Credit card number
  - Expiry date
  - CVC

- **Submit Payment**: Provide a **Submit Payment** button that simulates the payment process and redirects the user to the Order Summary View after the payment is successfully processed.

- **Tailwind CSS Styling**: Ensure the form is clean and responsive, using Tailwind's utilities for form input, spacing, and layout. Group payment information and order details logically, and make sure the form is accessible on both mobile and desktop screens.

## 2.4 Order Summary View

The Order Summary View displays a final confirmation of the user's order after they have submitted their payment details. This view shows the full order details, including courses purchased, user information, and a payment confirmation.

- **Order Details**: List the courses the user is purchasing, showing the course name, price, quantity, and total for each item. Display the total cost of the order, including taxes, shipping, and any other fees.

- **User Information**: Show the user's information (e.g., name, email, shipping address) that was provided during the checkout process.

- **Payment Confirmation**: Display a message confirming that the payment was successful. Optionally, include a reference number or transaction ID for the user's records.

- **Return to Browse Courses**: Include a button that allows users to return to the Browse Courses View, clearing the cart and resetting the state for a new session.

- **Tailwind CSS Styling**: Use Tailwind's utilities for layout, spacing, and typography to create a clear, easy-to-read summary. Make sure that all order details are well-organized, and the confirmation message is prominent and clear for the user.

## 2.5 Extra Credit (Optional)

**Total Points:** 5 points

- Implement **onChange search functionality** in the Browse Courses View.

- As the user types in the search input box, the course list should dynamically update in real time to reflect matching results by the course title.

---

- This feature should improve user experience by providing immediate feedback without requiring the user to press Enter or click a search button.

- Ensure the filtering logic is efficient and the UI remains responsive, even with a large number of courses.

- The implementation should be clearly demonstrated in the video walkthrough to earn credit.

# 3  Task Division

To ensure equal contribution from both team members, the tasks for Assignment 03 will be divided as follows. Each member will be responsible for different views, but both will collaborate on shared tasks like styling, state management, and documentation.

## 3.1  Member-wise Task Division

### Member 1: Focus on Browse Courses and Order Summary View

- Browse Courses View:

  1. Implement the display of courses, showing each course's title, description, price, and an "Add to Cart" button.
  2. Implement search functionality to allow users to search for courses by name.
  3. Implement filters for courses by difficulty level (Beginner, Intermediate, Expert) and subject.
  4. Manage the state of the courses and handle filter or search interactions to dynamically update the course list.

- Order Summary View:

  1. Display a summary of the user's order, including courses purchased, their prices, quantities, and total cost.
  2. Show user details (name, email) provided during checkout.
  3. Display payment confirmation and optionally include a reference or transaction ID.
  4. Include a button to return to the Browse Courses View, resetting the cart for a new session.

### Member 2: Focus on My Cart View and Payment Gateway View

- My Cart View:

  1. Implement the cart functionality, allowing users to see the courses they've added, with options to modify quantities and remove items.
  2. Display the total cost of the courses, including taxes and any additional fees.
  3. Include a button to proceed to the Payment Gateway View and another button to return to the Browse Courses View.

---

- Payment Gateway View:

  1. Display the total cost of the selected courses, along with payment fields for the user to enter credit card information (e.g., credit card number, expiry date, CVC).
  2. Include a **Submit Payment** button that simulates the payment process and redirects users to the Order Summary View after a successful transaction.
  3. Validate the form to ensure all required payment fields are filled out.

## 3.2 Collaborative Tasks

- Both members will collaborate on the overall styling using TailwindCSS, ensuring the platform is visually consistent and responsive across all views.

- Both members will work together on state management to ensure smooth transitions between views (e.g., cart updates, order details passed to the Order Summary).

- Both members will give voice for a collaborative 3-minute explanatory video, each explaining a portion of the platform's functionality (e.g., Browse Courses, Cart Management, Payment, and Order Confirmation) which they worked on.

## 3.3 Design Expectations

The design of your application will be evaluated on the following criteria:

- **Responsiveness:** The user interface must adapt gracefully to various screen sizes, including desktops, tablets, and mobile devices.

- **User Experience:** Layout and navigation should be intuitive, user-friendly, and logically structured to ensure smooth interaction.

- **Visual Appeal and Creativity:** The project should demonstrate thoughtful design choices and creativity. Submissions with minimal or rushed UI/UX effort will receive point deductions.

**TA Setup:** TAs will initialize GitLab repositories for each team with a standard folder structure. Teams are expected to build upon this structure in their submissions.

## 3.4 Required Deliverables

- **Software Submission (Due: April 15)**
  Complete your web application. Ensure it is fully functional, meets all requirements, and is pushed to GitLab repositories.

- **Demo Video (Due: April 15)**
  A recorded walkthrough of your completed application, showcasing features and user flow. Upload to the `Documents/` folder.

---

# 4 Grading Rubric

| Item | Item Description | Points | Max Points |
|---|---|---|---|
| Task Division | Equal division of tasks between team members | 20 | 20 |
| Browse Courses View | Implementation of the course listing and Add to Cart functionality | 20 | 20 |
| My Cart View | Functionality for managing cart items and proceeding to checkout | 20 | 20 |
| Payment Gateway | Implementation of payment form, order review, and state management for payment | 20 | 20 |
| Order Summary View | Displaying order details, user info, payment confirmation, and smooth transition to Browse Courses view | 15 | 15 |
| Overall Design | Use of TailwindCSS for a clean, responsive, and user-friendly design across all views | 5 | 5 |
| Extra Credit | Optional real-time search feature in Browse View | 5 | 0 |
| **Total** | | **105** | **100** |

# 5 Submission Details & Deadline

**Due Date:** Apr 15, 2025, 11:59 PM CST
**Format:** Submit the GitLab repo link on Canvas
**Late Penalty:** -5 points per day (accepted through Apr 18, 2025)

# 6 Additional Notes

## 6.1 Expectations

This is a team-based project. Equal contribution is expected and should be as per the Task Division section. Contact TAs or instructor if you have any questions.