

# Course Syllabus

 Edit

## Overview

An introduction to software development on the web. The course covers the development of programs that operate on the web and survey of technological alternatives, with emphasis on modern web development technologies. Concepts, including the Internet and the web, static client content, dynamic client content, dynamically served content, n-tiered architectures, web development processes, and security on the web, are discussed. Assignments involve increasingly more complex web-based programs, and the term ends with an open-ended project for students to showcase their mastery over the course content.

## Learning outcomes

At the end of the course, you will:

- Explain the structure of the web and how HTTP works
- Create web pages with HTML, CSS, and JavaScript
- Build interactive frontend applications with React and React Router.
- Build backend applications using Express.js, and design and implement RESTful APIs.
- Work with MongoDB to store, retrieve, and manage data, and implement authentication using JWT.
- Identify and prevent common web security issues
- Follow best practices in web development to deploy full-stack web applications

## Requirements

- Programming experience & Python (CSC108)
- Advanced programming & OOP (CSC207 & CSC148)
- Basic shell & system programming (CSC209)
- **Corequisite** : Database systems (CSC343). Note that this is not strictly enforced, but highly recommended.

This course is suitable for anyone interested in learning web programming and may be seeking a relevant job in the future. It assumes no prior knowledge or experience in web development. So everything will be discussed from the very basics.

## Course Information

Instructor:

Name: **Kuei (Jack) Sun**

Office: BA 4231 (not for office hours)

Office Hours: Mondays and Wednesdays 11:45am to 12:45pm, room BA4290

*Meetings may be scheduled outside of regular office hours by request.*

## Communications:

Course Email: [csc309-2025-01@cs.toronto.edu](mailto:csc309-2025-01@cs.toronto.edu) (<mailto:csc309-2025-01@cs.toronto.edu>)

Please use email for personal issues and Piazza for all other course-related questions. We will try to respond to email by the end of the next day. However, due to volume, it may take longer, especially on weekends. (We are often not able to answer email more than once on the weekend.)

## Lectures

*All lectures will be held in-person at the locations indicated.*

<p><b>Afternoon Section</b></p> <p>LEC0101 and LEC2001</p>	<p>Tuesdays 3pm - 5pm</p> <p>Room: <a href="#">MC 254</a></p> <p><a href="https://map.utoronto.ca/?id=1809#!m/494474?share">https://map.utoronto.ca/?id=1809#!m/494474?share</a></p>
--	--

## Lecture Format

In the lectures, we will cover core material of the course. During the lectures, you may be asked to participate through in-class activities and exercises. Lectures will be recorded and the recordings will be posted as soon as possible for asynchronous viewing. However, students are expected to attend the lectures in person and complete the exercises during, or shortly after, the lectures.

*This course, including your participation, will be recorded on video and will be available to students in the course for viewing remotely and after each session.*

*Course videos and materials belong to your instructors, the University, and/or other sources depending on the specific facts of each situation, and are protected by copyright. In this course, you are permitted to download session videos and materials for your own academic use, but you should not copy, share, or use them for any other purpose without the explicit permission of the instructors.*

*For questions about recording and use of videos in which you appear please contact your instructors.*

## Tutorials:

Update: in-person tutorial sessions have been canceled. However, the following CS lab rooms are still reserved for CSC309 students on Wednesdays between 3pm and 4pm: **BA3175**, **BA3185**, **BA3195**, and **BA2220**.

The purpose of the tutorials is to provide hands-on practice of related tools and frameworks and reinforce lecture materials by guiding students through examples and exercises of key web programming concepts. If you need help with the tutorial content, please attend TA office hours and/or the instructor's office hours.

## Textbook(s):

There are no textbooks for this course. You can Google related keywords to find reference manuals on each major topic. Midterm and final exam material will solely be sourced from the lecture notes and tutorial content.

## Website and Discussion Board:

You will be able to find all course materials from this Quercus site. The Piazza discussion board is required reading. Please use Piazza to ask general questions, and remember to search to check if someone else has already answered the question. The instructor and/or TAs will be monitoring it daily.

# Marking Scheme

## Overview

ITEM	WEIGHT
Exercises (10/12)	1% (0.1% each)
Tutorials (10/11)	20% (2% each)
Assignments (2/2) A1: 10% A2: 14%	24%
Term Project (1/1)	18%
Midterm Test	12%
Final Exam	25%

## Detailed Description

### Lecture Exercises (1%):

There will be in-class exercises associated with the lectures during most classes. These will typically take the form of quizzes on Quercus and will be graded on best effort. In other words, you will get full marks for an exercise if we can tell that you have made a serious attempt at it. You will not get credit for the exercise if you do nothing and just submit. We will take the best  $n-2$  out of  $n$  scores. We expect that  $n$  will be 12 so that each completed lecture exercise is worth 0.1%. We strongly encourage you to do these exercises in class to get a clearer understanding of the material. We welcome questions about these activities during the synchronous lectures. All lecture exercises for the week will be due on Sunday at the end of the day (11:59 p.m.).

- Exercises for the first 3 weeks are all due on Sunday, January 26 at 11:59 p.m., to allow waitlisted students to complete them.

### Tutorial Exercises (20%):

For each tutorial, you will be asked to turn in a small piece of work or to work on an activity during the tutorial. These exercises will be auto-graded and should be completed individually. Where possible, you will be allowed to resubmit and we will take your final score. Many of these will be directly related to the assignments. We will take the best 10 out of 11 scores (2% each).

Tutorial exercises are due on Sunday at the end of the day (11:59 p.m.), with the following exception:

- Tutorials for the first 3 weeks are all due on Sunday, January 26 at 11:59 p.m., to allow waitlisted students to complete them.
- There will be no tutorial on week 7, during midterm week.

### Programming Assignments (24%):

Over the term, you will complete 2 assignments that consist of problems that challenge your understanding of the concepts and will be auto-graded. **All assignments must be completed individually.** All assignments must be submitted by checking your work into your MarkUs repository. Assignment 1 is worth 10% and assignment 2 is worth 14%

### Term Project (18%):


The project simulates a real-world website that you are likely to develop in the future as a freelance web developer. Even though the scope is considerably small, it is designed to give a sense about how creating a real website and being a full-stack developer would look like. For the project, you can make groups of up to **3 members**. You could also do it alone, but it is not recommended as the workload might be excessive for one person.

The project will be based on the RESTful APIs that you developed during assignment 2. After the project deadline, your final deliverable will be graded through an interview with a TA during which they work with

your website to check if the requirements are implemented correctly (interview booking link TBA). In normal circumstances, every member will receive identical grade. However, if you run into situations where some members are not contributing fairly to the project, you should contact the instructor via course email about the issue.

### Midterm Test (12%):

There will be one 50-minute test, which should be completed during your regular lecture time slot. Midterm coverage will be the course content taught prior to reading week. It will be done completely **online** and **open book**. The exact detail of the midterm format will be released one week before the start of the reading week. You can do the midterm at home if you wish. However, I will be available in room **BA 1170** if you prefer to write in person. In this case, you will need to bring a laptop or tablet with Internet connectivity.

The midterm date will be **Wednesday February 26th**, during the regular tutorial hour (3pm to 4pm). If you cannot write the Midterm test due to extraordinary circumstances beyond your control, please submit [this form](https://forms.office.com/r/ZrWsRfiwGS)  (<https://forms.office.com/r/ZrWsRfiwGS>) with the supporting documentation to request a special consideration as soon as possible. Special consideration requests will be evaluated on a case-by-case basis.


### Final Exam (25%):

The Final Exam is scheduled by Arts & Sciences and the exact date and time will be released later. It will cover all course material, including topics that were tested on the midterm test and questions about the tutorials and assignments. The final exam format will be **online** and **open book**. A minimum grade of **40% on the final exam is required to pass this course**.

## Minimum Standards for Submitted Work:

For your tutorial work or assignment to be graded, it must meet the minimum standards of a professional computer scientist. **All** required files must be submitted. If the submission contains a web page, it should work on either Firefox 128.5.2 ESR or Google Chrome 131.0.6778.204/205. If the submission needs to run on Node.js, it should work on v22.12.0 LTS. Last-minute difficulties with git can easily be avoided by ensuring all files are added to the repository well before the deadline, and that you know how to commit and push them. **Submissions that are missing files or do not run correctly will receive a grade of 0.**

### Late Work:

All assignments must be submitted electronically by **11:59:59 p.m. sharp** on the due date. There are no grace tokens or automatic extensions of any kind. Late submissions will not be accepted unless under exceptional circumstances. If you find yourself in a serious medical or emergency situation, you will have the opportunity to submit [this form](https://forms.office.com/r/ZrWsRfiwGS)  (<https://forms.office.com/r/ZrWsRfiwGS>) to explain your

circumstances and your specific special consideration request. Special consideration requests will be evaluated on a case-by-case basis.

Please note that a submission made at 12:00:00 a.m. after the due date will be considered late. You should ensure that your work is not submitted at the very last second. Since you will be using version control, it is very easy to commit regularly to avoid running into the deadline.

## Religious Holidays:

If a religious holiday will keep you from completing any assigned work, please let us know as soon as possible (but no later than two weeks before the due date), and we will work out a mutually agreeable accommodation.

## Emergencies:

In the event of an illness or other catastrophe that affects your ability to do your academic work, consult the course instructors right away. Normally, you will be asked for documentation in support of your specific circumstances. This documentation may take the following forms:

- Absence declaration via [ACORN \(https://www.acorn.utoronto.ca/\)](https://www.acorn.utoronto.ca/)
  - *The University has updated its policies on the use of the Absence Declaration, which can now only be used in case of (a) a health condition or personal injury, (b) a personal or family emergency, or (c) bereavement. Students may submit **one absence declaration per academic term**, to declare an absence for a **maximum period of seven consecutive calendar days**. The seven-day declaration period can be retroactive for up to six days in the past, or up to six days in the future, but it must cover the period in which the missed academic obligation occurred.*
- [U of T Verification of Illness or Injury Form \(VOI\)](http://www.illnessverification.utoronto.ca/index.php)  
(<http://www.illnessverification.utoronto.ca/index.php>)
  - *The VOI indicates the impact and severity of the illness, while protecting your privacy about the details of the nature of the illness. If you cannot submit a VOI due to limits on terms of use, you can submit a different form (like a letter from a doctor), as long as it is an original document, and it contains the same information as the VOI (including dates, academic impact, practitioner's signature, phone and registration number). To download a copy of the VOI, please see <http://www.illnessverification.utoronto.ca> (<http://www.illnessverification.utoronto.ca/index.php>).*
- College Registrar's letter
- Letter of Academic Accommodation from Accessibility Services

For more information on documentation of absences for Arts and Science students, including limitations on the use of the Absence Declaration tool, refer to the [A&S Student Absences \(http://www.artsci.utoronto.ca/absence\)](http://www.artsci.utoronto.ca/absence) page. It is always easier to make alternate arrangements before a due date, so please inform us as soon as you know that you will need accommodation. If you get a

concussion, break your hand, or suffer some other acute injury, you should register with Accessibility Services as soon as possible.

## Remark Requests:

If you believe there was an error in the grading of your tutorial work or assignment, you may submit a remark request through MarkUs. Your request must clearly state one of the following:

1. **Marking error:** Clearly and concisely describe the specific error you believe occurred in the grading.
2. **Minor submission issue:** Identify a minor problem in your submission that can be resolved with no more than one line of code, e.g., so that your code would compile again. If your request is accepted under this category, a 20% penalty will be applied.

Remark requests must be submitted within **one week** of the marks being released. Please note that remarking may result in an increased grade, no change, or a *decreased grade*.

## Academic Integrity:

All of the work you submit must be done by you and your work must not be submitted by someone else. Plagiarism is academic fraud and is taken very seriously. The department uses software that compares programs for evidence of similar code. Please refer to the UofT [Academic Integrity website \(https://www.academicintegrity.utoronto.ca/\)](https://www.academicintegrity.utoronto.ca/) and read the [Code of Behaviour on Academic Matters \(https://governingcouncil.utoronto.ca/secretariat/policies/code-behaviour-academic-matters-july-1-2019\)](https://governingcouncil.utoronto.ca/secretariat/policies/code-behaviour-academic-matters-july-1-2019). Here are a few guidelines to help you avoid plagiarism:

Honesty and fairness are fundamental to the University of Toronto's mission. Plagiarism is a form of academic fraud and is treated very seriously. Please refer to the University of Toronto [Academic Integrity website \(https://www.academicintegrity.utoronto.ca/\)](https://www.academicintegrity.utoronto.ca/) and read the [Code of Behaviour on Academic Matters \(https://governingcouncil.utoronto.ca/secretariat/policies/code-behaviour-academic-matters-july-1-2019\)](https://governingcouncil.utoronto.ca/secretariat/policies/code-behaviour-academic-matters-july-1-2019). Here are a few guidelines to help you avoid plagiarism.

## Assignments

Assignments are individual tasks, meaning you are expected to complete them independently. Seeking advice from other students or copying someone else's code—whether from a peer or open source repositories—is strictly prohibited. You may refer to online resources, tutorials, and Q&A websites to assist with your learning and problem-solving process. However, the code you submit must be entirely your own work. The use of AI-generated code is also prohibited. Any violation of this policy will result in disciplinary action to the fullest extent allowed under academic regulations.

## Project

For the project, you are permitted to use open-source packages and code available on the internet. However, sharing any part of your code with other teams, or using code shared by other teams, is strictly forbidden. If you use online code, it must be properly referenced, including a citation of the webpage

from which it was sourced. Violations of this rule will be treated as serious breaches of academic integrity.

## Tests

While all tests in this course are open-book, you are expected to consult only official materials, such as online manuals, lecture notes, and other authorized resources during the test. Copying code or answers from external online forums, such as Stack Overflow, Reddit, or AI tools like ChatGPT, is strictly prohibited. Any student found submitting identical code or answers, either partially or entirely, will be charged with an academic offense.

## Accessibility Needs:

The University of Toronto is committed to accessibility. If you require accommodations for a disability or have any accessibility concerns about the course, the classroom, or course materials, please contact **Accessibility Services** (<https://studentlife.utoronto.ca/departments/accessibility-services/>) as soon as possible via email ([accessibility.services@utoronto.ca](mailto:accessibility.services@utoronto.ca) (<mailto:accessibility.services@utoronto.ca>)) or phone ([416-978-8060](tel:416-978-8060) (<tel:416-978-8060>)).