# **Creating Modern Web and Mobile Applications**

# CSCI1320, Spring 2020, Prof. Reiss

# **Course Overview**

CSCI1320, 'Creating Modern Web and Mobile Applications', is a spring semester course within the Brown CS department. The course takes a holistic look at the process of developing web and mobile applications and aims to bring students to a point of mastery of several of the most used web and recent technologies and development practices.

The course has three intertwined parts. The first involves learning the fundamentals of web and mobile applications. In this part, we will look at the organization and mechanisms used by these applications. There are a number of different technologies used today for developing such applications. We do not expect students to be an expert in all of these, but we do expect that students should be aware of them, what they are good for, and how they work so they can reasonably determine what should be used for a new application and can talk intelligently about web applications. There are also a number of factors to consider in building web and mobile applications including human factors, universal access and security covered in this part of the course. This part of the course consists mainly of lectures, labs, and the final exam.

The second part of the course involves learning a specific set of technologies for building web and mobile applications. Here we will concentrate on HTML, CSS, JavaScript, and VUE for web front ends; NativeScript for mobile front ends; and Node.JS for back ends. We will also cover databases, both SQL and No-SQL, and various forms of client-server interaction. These will be covered in five programming or design assignments where the students get to apply the technologies.

The third part of the course involves a semester long group final project in which the students will be working in teams with external companies, non-profits, faculty, and other organizations to create or augment a real modern web or mobile interface. It is also possible for students to propose and direct their own project if it is well defined at the start of the semester.

The course has two tracks, one intended for CS concentrators, and one intended for non-Concentrators with previous design experience. The concentrator track is for students with prior programming experience (CSCI0150/0160 or CSCI0170/0180 at a minimum, CSCI0320 or CSCI0330 recommended). Concentrators will learn how to build and maintain all parts of a modern web or mobile application. The specific technology assignments and labs for concentrators will involve programming, generally in JavaScript. The design track does not assume any programming background, but does assume a strong design background. Designers will learn how to apply their skills to web and mobile applications and learn how to interact with programmers to create good-looking and effective applications. The

technology assignments and labs for designers will involve a minimum amount of programming and a significant amount of design.

The course can serve as a Computer Science **capstone**. If a student is taking it as a capstone, they must be in their last two semesters, and must either propose and supervise a team project or act as the team leader on their group project.

Students interested in **entrepreneurship** in Computer Science will find that the course provides several opportunities. Students can propose a project that could eventually be the basis of a start-up, and have a team of work on it. Project development will include ample opportunities to get experience in selling the underlying ideas including elevator talks, a poster presentation, and in-class presentations. The prototype system that you produce as the project can be used as a prototype for a potential start-up.

## **Course Mechanics**

The class will meet Monday, Wednesday and Friday from 10:00 to 10:50 in the auditorium at 85 Waterman Street. Students are encouraged to attend each class. Information is posted on the course web site (<u>http://www.cs.brown.edu/courses/csci1320</u>). Students are expected to keep up to date with the announcements both in class and on the web site.

There is no textbook per se for the course. The course web site provides a set of reference links that can be used to get information about specific technologies. Homeworks, generally prerequisites for the various labs, will include web-based tutorials providing additional information.

Laptops will be required (with some exceptions) for in-class labs. (If you do not have a laptop, do not worry – all labs are collaborative and you can work with someone who does.) Outside of this, the use of laptops and similar devices in class is strongly discouraged. Several studies have shown that taking notes via laptop is not nearly as effective as taking them by hand. Using laptops for anything other than taking notes is disruptive not only to you but also to all the students around you. (Using your phone has similar problems.) To be fair to the rest of the class, we ask that you not use laptops or phones in class unless the class is a lab.

The course will involve a team project with various milestones during the semester. It will involve five programming/design assignments. There will be in-class labs on specific technologies. The labs will be preceded by a homework assignment that should introduce students to the material in a tutorial manner. There will be optional Sunday evening sessions to go over these pre-labs. (If you cannot attend, you should be able to ask the TAs for equivalent help during TA hours.) The labs themselves will let students demonstrate the associated technology in a creative way. There will also be short homeworks that will serve to frame the subsequent lectures. Students are expected to do these homeworks in advance of the lecture and to be able to discuss them in class.

Dr. Reiss will have office hours Monday and Thursday from 1 -3 pm. TA hours will announce during the first week of classes.

# **Project**

Much of the course will involve a team project proposed by either outside sponsors or students. Project teams should ideally be four people. The project schedule is tentatively:

DATE	PROJECT EVENT
1/25	Student Project Proposals due
1/27	Project proposals available
1/31	Project preferences due
2/5	Project teams announced
2/14	Initial client report
2/24	Project specifications due
3/2	Project elevator talks
3/6	User feedback report
3/9	Initial project design presentation to mentor TA
3/16	Project front end design due
3/16,18	Poster fair
4/10	Project implementation design due
4/20	Project prototype target date
4/24	Client feedback report on prototype
4/29	Project testing reports
5/4	Project final presentations
5/8	Final project hand-in

## Grading

Your grade will be determined from programming assignments, labs, class participation, homeworks, final project, and final exam:

Programming Assignments	35% (7% each)	
Homeworks, labs, participation	23%	
Group Project	32%	
Final Exam	10%	

A rough estimate of the required time would be 120 hours over the semester for the final project, 10 hours each for the five programming assignments, 20 hours for the labs and pre-labs, and 2 hours doing the pre-lecture homeworks.

Each student can use five late days throughout the whole semester. Late days can be used only for assignments, not for the final project, with at most three being used for any one assignment. Without late days, late assignments are penalized 15% per day. Late days will be applied optimally at the end of the semester on your behalf. All assignments must be turned in, even if they get zero points. Failure to do so will result in an NC for the course. The final project will not be accepted late. Absences or delays

due to planned travel (e.g. job interviews) are not excusable (plan your late days accordingly). Students who need more time due to illness or other emergencies should get a dean's note or a note from health services.

Students are also required to work on and hand-in in-class labs. Labs must be completed (what this means depends and the lab) up to two classes beyond the lab. After that, they receive no credit.

## **Collaboration**

In general, this course encourages collaborative efforts. The final projects are collaborative. The in-class labs are collaborative. Homeworks can be collaborative. Especially for their final projects, students are encouraged to use publically available libraries and materials as long as they are appropriately cited and the intellectual property rights are appropriate for the project.

However, **the programming/design assignments and final exam are not collaborative exercises**. They should be the student's own work. A TA must approve any potential use of outside packages for the programming or design assignments. Details of the policy can be found on the web site and the collaboration policy that must be signed by all students.

## **Diversity**

Brown University is committed to full inclusion of all students. Please inform Dr. Reiss early in the term if you have a disability or other conditions that might require accommodations or modification of any of these course procedures. You may speak with me after class or during office hours. For more information, please contact <u>Student and Employee Accessibility Services</u> at 401-863-9588 or SEAS@brown.edu. Students in need of short-term academic advice or support can contact one of the deans in the Dean of the College office.

## **Tentative Class Schedule**

Lecture: Course Introduction Lecture: The Browser Lecture: Accessibility Lab: HTML/CSS Lecture: JavaScript Lecture: Dynamic Web Pages Lecture: Dynamic Web Pages Lecture: Requirements and Specifications Lab: JavaScript/DOM Lecture: VUE Lab: JavaScript Frameworks --- HOLIDAY Lecture: The Web Server Lecture: Node.JS Lecture: Web App Architectures Lab: Note.JS

Lecture: SQL Databases Project: Elevator Talks Lecture: NoSQL Databases Lab: Databases Lecture: Backend Technologies Lecture: Mobile Applications Lecture: NativeScript Project: Poster fair Project: Poster fair Lab: Mobile ----- Spring Break Lab: Cloud-based deployment Lecture: HCI Design Lab: Design Lecture: Security I Lecture: Security II Lab: Security Challenge Lecture: Privacy Lab: Security Challenge Lecture: Testing I Lecture: Testing II Lab: Testing Lab: Testing Lab: Performance Lecture: Ethical Design Lecture: Actual web applications Project: Final Presentations Final Exam