# Introduction to the Course



Building Modern Web Applications - VSP2022

Karthik Pattabiraman Kumseok Jung

#### What is this course about?

- 1. What is this course about?
- 2. Logistics
- 3. Policies
- 4. Grading
- 5. Other



#### What is this course about?

- Core principles behind building modern web applications
- Abstractions and design principles
- Application of core web technologies such as HTML, CSS, JavaScript, Node.js to the above



#### What is it NOT about?

## Learning of specific technologies

- These will get outdated by the time you finish
- Fast changing field, so new technologies continuously appear and disappear.
- Can learn any technology if you understand the principles and concepts behind web development

## Frameworks or libraries (e.g., jQuery)

- These are built on the principles and concepts
- Too many to cover in a reasonable time



#### **Bottom line**

You will understand the principles behind web application development



- Not simply copy-paste code from websites to string together a web application
- You will understand why technologies are the way they are, rather than accept it as a statement of fact, and perhaps change them if needed
- It enables you to design novel techniques and technologies in the web application space
- If you put in the effort, this course will be really fun! :-)

# **Logistics**

1. What is this course about?



- 3. Policies
- 4. Grading
- 5. Other



#### **Instructor: Karthik Pattabiraman**

## Karthik Pattabiraman (karthikp@ece.ubc.ca)

- Professor at UBC (joined 2010)
  - PhD from University of Illinois Urbana-Champaign
  - Detour via Microsoft Research in 2009

#### Research

- Web applications' reliability and security
- Error resilient applications
- Internet of Things (IoT) security
- Machine Learning Dependability



# **TA: Kumseok Jung**

## Kumseok Jung (<u>kumseok@ece.ubc.ca</u>)

- PhD Student at UBC
  - o BSc & MASc from University of British Columbia



- Internet of Things (IoT)
- Cloud/Edge Computing
- Software Engineering
- Distributed Systems



## **Logistics - Lectures**

Lectures delivered by the instructor in class



- Will consist of a mix of teaching (lecturing) sessions mixed with in-class activities
  - Please bring your laptops fully charged with you to class.
  - o Contact us if you do not have a laptop.
  - You will work in teams of 2
  - Participation in activities is important
- Lecture notes will be distributed ahead of time
  - No course textbook required; However, you should keep your own notes

## **Logistics - Software**

Any OS: Windows, Mac OSX, or Linux



- Your favorite web browser + built-in web dev tools
  - Firefox
  - Chrome
  - Edge
- The text editor/IDE of your choice :-)
  - VScode highly recommended
  - Sublime
  - Atom
  - Notepad++

## **Logistics - Interactions**

## **Github**



github.com/ubc-vsp22/classroom



- Main hub to find all the links
- Lecture slides, class activities, assignments

Do not distribute

Assignment submissions

#### **Slack**



Ask and answer questions - bonus points for participation

No Email (unless it's private)

## **Logistics - Resources**

- There's no textbook for the course
  - Lectures will cover all the material
  - Augment with online resources as needed
  - Attendance expected at all lectures
- Assignments will test you on material not necessarily covered in the lectures
  - You're free to use publicly available online resources on the web, as long as you cite them



### **Policies**

- 1. What is this course about?
- 2. Logistics
- 3. Policies
- 4. Grading
- 5. Other



#### **Policies**

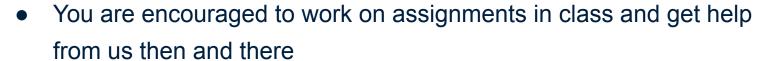
- You are responsible for all material you hand in
  - Review UBC's policies for academic dishonesty



- Plagiarism of any kind will NOT be tolerated
  - Automatically result in you getting an F
  - Lack of knowledge of policies is not a valid excuse
- No collaboration allowed on assignments (except with your partners more later)

#### **Policies**

- All material in the exam will be from the lectures covered in class.
  - Will NOT test you on material NOT in the lecture notes!
  - Missing a lecture means that you may miss out
  - o Encouraged to ask questions in class and online



- Office hours will not typically be held outside class
- o If really needed, we can schedule special sessions with you



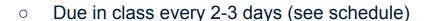
# **Grading**

- 1. What is this course about?
- 2. Logistics
- 3. Policies
- 4. Grading
- 5. Other



## **Grading**

- Assignments (60%)
  - 4 assignments counting for 15% each



- Done in teams of 2 (form teams by today)
- Encouraged to work during class on laptops
- Use Github to commit code (do this periodically)
- No late assignments (no exceptions)



## **Grading**

- Final Exam (40%)
  - o To be held on Aug 9th



- Must be done individually (NO collaboration)
- Closed notes and Closed book part consisting of multiple choice questions (15%)
- Open notes and Open book part consisting of 5 programming problems (25%) –
   please bring your laptop for this

Exam will be auto-graded with manual checking. No partial points.

## **Assignments - Git**

Open source distributed version control system



- We will be using Git for version control and GitHub for hosting
- Each group will receive a private GitHub repository

## **Assignments - Git**

Assignment submissions will take place through GitHub



- Create an assignment branch (i.e., assignment-1, assignment-2, assignment-3, assignment-4) by the due date (we will give more details on this)
  - No other means to submit an assignment will be accepted!
- No late commits will be accepted (unless with instructor permission).
  - Please push your latest changes to the appropriate branch before 11:59:59 PM on the due date!

## **Class Participation**

- To learn and benefit from this class, all of you need to participate
  - Asking and answering questions in class and on Slack
  - Participating in in-class exercises
  - Does NOT mean simply showing up in class

0

We will award bonus points for class participation



## Other

- 1. What is this course about?
- 2. Logistics
- 3. Policies
- 4. Grading
- 5. Other



## Other thoughts

- Hope you have fun and learn too



- If you're struggling, let us know early so we can help to the extent possible – or it may be too late
- Feel free to give us feedback and suggestions for improvement etc. –
   these will NOT impact your grade in any way



## **TODO for Today**

• Find a partner to do the assignments with (teams of 2) to get a Github account from Kumseok.



- Let us know about your team by end of the first class. Use the following google form to submit your group information:
  - https://forms.gle/3VHAhKsUkrQAg9sR7
- We will then assign a GitHub repository for your team, and both members will be added as collaborators. Make sure you can work with it from your laptops

## **TODO for Today - Git Demo**

- 1. Clone repository
- 2. Commiting changes
- 3. Pushing/pulling changes from repository
- 4. Branching

#### <u>Useful Git Commands</u>

git clone git pull origin master git push origin master

#### **Creating Branches**

git branch assignment-X
git checkout
assignment-X
git push -u origin
assignment-X

git checkout master

git branch

git branch -r



# **TODO for Today**

- Node.js Setup
- Git setup



## **Extra Resources on JavaScript**

## If you want to go beyond the VSP course:



- 1. "Eloquent JavaScript: A Modern Introduction to Programming" by Marijn Haverbeke
- 2. "JavaScript: The Good Parts" by Douglas Crockford (where JavaScript quiz is from)
- 3. "Programming JavaScript Applications: Robust Web Architecture with Node, HTML5, and Moderns JS Libraries" by Eric Elliott
- "Effective JavaScript: 68 Specific Ways to Harness the Power of JavaScript" David Herman
- 5. "JavaScript: The Definitive Guide" by David Flanagan
- 6. "You Don't Know JS" by Kyle Simpson

## Not required for this VSP course!