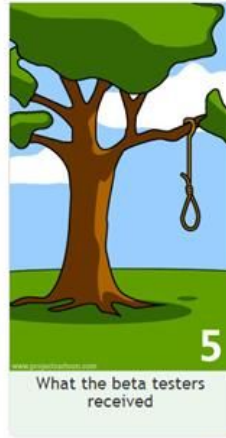


# COSC 499: Capstone Software Engineering Project



# Dr. Bowen Hui's Background

2020-

Position:

- Associate Professor of Teaching, Computer Science, UBCO

# Dr. Bowen Hui's Background

2020-

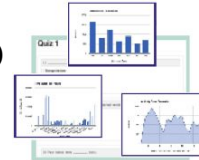
Position:

- Associate Professor of Teaching, Computer Science, UBCO

2012-

Teaching:

- Learning Analytics, Human-Computer Interaction, Intro Programming, Mobile Educational Game Development, Capstone



# Dr. Bowen Hui's Background

2020-

## Position:

- Associate Professor of Teaching, Computer Science, UBCO

2012-

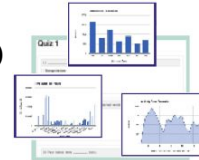
## Teaching:

- Learning Analytics, Human-Computer Interaction, Intro Programming, Mobile Educational Game Development, Capstone

2006-

## Research areas:

- Team formation/analytics, Novice programming (gamification)



# Dr. Bowen Hui's Background

2020-

Position:

- Associate Professor of Teaching, Computer Science, UBCO

2012-

Teaching:

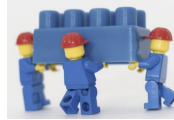
- Learning Analytics, Human-Computer Interaction, Intro Programming, Mobile Educational Game Development, Capstone



2006-

Research areas:

- Team formation/analytics, Novice programming (gamification)



1996-

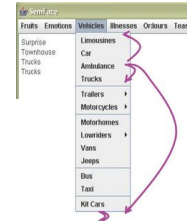
Past research:

- Edutainment design, computational thinking for kids, intelligent user interfaces, computational linguistics, second language acquisition



(23) Rules

- a. all:  $r \rightarrow \emptyset / \_ \sigma$
- b. early:  $r \rightarrow w / \sigma[ \_$
- inter:  $r \rightarrow w / \#[ \_$   
 $\quad \quad \quad 1 / \sigma[ \_$
- later:  $r \rightarrow 1 / \sigma[ \_$



# Dr. Bowen Hui's Background

Industry experience as  
System Analyst and Project Manager

2020-

Position:

- Associate Professor of Teaching, Computer Science, UBCO

2012-

Teaching:

- Learning Analytics, Human-Computer Interaction, Intro Programming, Mobile Educational Game Development, Capstone

2006-

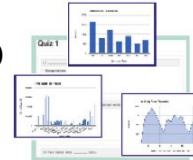
Research areas:

- Team formation/analytics, Novice programming (gamification)

1996-

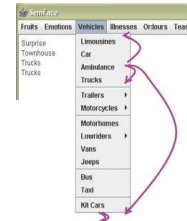
Past research:

- Edutainment design, computational thinking for kids, intelligent user interfaces, computational linguistics, second language acquisition



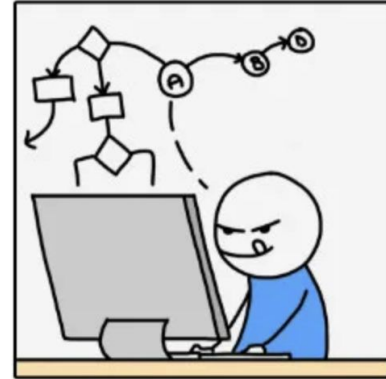
(23) Rules

- all:  $r \rightarrow \emptyset / \_ \_ ]\sigma$
- early:  $r \rightarrow w / \sigma[ \_ \_$
- inter:  $r \rightarrow w / \#[ \_ \_$   
 $\quad \quad \quad 1 / \sigma[ \_ \_$
- later:  $r \rightarrow 1 / \sigma[ \_ \_$



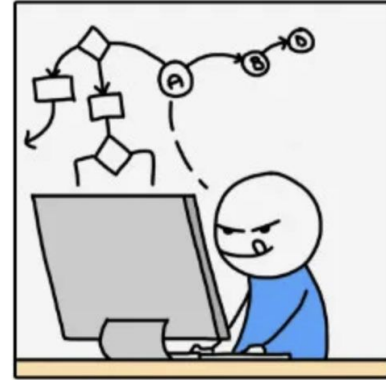
# From School to Industry

- A typical degree in COSC
  - Year 1: individual work, toy exercises, code snippets in template project
  - Year 2: logic, data structure, algorithms, inner-workings to "real" programs
  - Years 3+4: special topics in development and research, small projects
  - This curriculum structure hasn't changed in decades!!!
- Which skills do you think are most sought after by industry today?



# From School to Industry

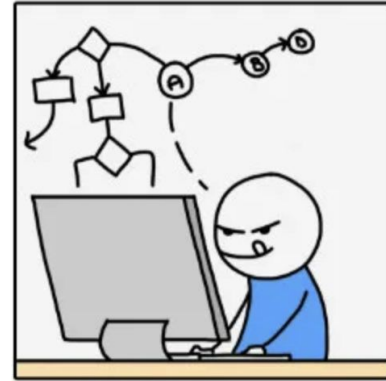
- A typical degree in COSC
  - Year 1: individual work, toy exercises, code snippets in template project
  - Year 2: logic, data structure, algorithms, inner-workings to "real" programs
  - Years 3+4: special topics in development and research, small projects
  - This curriculum structure hasn't changed in decades!!!
- Industry expectations (in 2023)
  - Top 5 from Medium: cloud computing, data structures/algorithms, Github, containers, vim/IDEs
  - Top 5 from Indeed: programming languages, database, data structures/algorithms, source control, testing procedures
  - Top 5 from LinkedIn: programming, software architecture, testing, object-oriented design, project team experience





# From School to Industry

- A typical degree in COSC
  - Year 1: individual work, toy exercises, code snippets in template project
  - Year 2: logic, data structure, algorithms, inner-workings to "real" programs
  - Years 3+4: special topics in development and research, small projects
  - This curriculum structure hasn't changed in decades!!!
- Industry expectations (in 2023)
  - Top 5 from Medium: cloud computing, data structures/algorithms, Github, containers, vim/IDEs
  - Top 5 from Indeed: programming languages, database, data structures/algorithms, source control, testing procedures
  - Top 5 from LinkedIn: programming, software architecture, testing, object-oriented design, project team experience
- How to bridge the gap?



# Aspirations of the Capstone



✦✦ Pool together knowledge from Years 1, 2, 3+ and apply them to a year-long project

# Aspirations of the Capstone



- ◆◆ Pool together knowledge from Years 1, 2, 3+ and apply them to a year-long project
- ◆◆ Work on project from start to end (of a working prototype), experience different aspects of a software lifecycle development phase
  - Know that there are roles other than "programmer" in a software team

# Aspirations of the Capstone




- ◆◆ Pool together knowledge from Years 1, 2, 3+ and apply them to a year-long project
- ◆◆ Work on project from start to end (of a working prototype), experience different aspects of a software lifecycle development phase
  - Know that there are roles other than "programmer" in a software team
- ◆◆ Exercise industry-relevant practices and learn industry tools and trends
  - Technology is fast-changing and every sector uses different tools
  - Pick a direction and learn new things

# Aspirations of the Capstone





- ◆◆ Pool together knowledge from Years 1, 2, 3+ and apply them to a year-long project
- ◆◆ Work on project from start to end (of a working prototype), experience different aspects of a software lifecycle development phase
  - Know that there are roles other than "programmer" in a software team
- ◆◆ Exercise industry-relevant practices and learn industry tools and trends
  - Technology is fast-changing and every sector uses different tools
  - Pick a direction and learn new things
- ◆◆ Work effectively in a team, embrace differences, recognize each other's strengths, learn from each other, and produce more than any single individual
  - Learn to work with people who are not you
  - Accept differences in opinions and not take them personally

# Lessons Learned from Past Capstones


- 
- Past successes:
- Students hired by external client company or receive positive reference letters from clients
  - Students learn new technical skills
  - Students push through deadlines to meet client needs or to enhance portfolio
  - Students' self-discovery, through reflections/conflicts, sometimes through several months


# Lessons Learned from Past Capstones

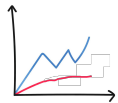
- 
- Past successes:
- Students hired by external client company or receive positive reference letters from clients
  - Students learn new technical skills
  - Students push through deadlines to meet client needs or to enhance portfolio
  - Students' self-discovery, through reflections/conflicts, sometimes through several months

- 
- Things we should avoid/change:
- Some client actions or wishes conflict with our course timeline
  - Reports are designed for grading, but have no direct, positive impact on students or the project itself
  - Teams are balanced based on skills in order to maximize project success
  - Performance is measured based on the quality of the deliverables, but largely ignores the quality of teamwork

# Lessons Learned from Past Capstones

-  Past successes:
- Students hired by external client company or receive positive reference letters from clients
  - Students learn new technical skills
  - Students push through deadlines to meet client needs or to enhance portfolio
  - Students' self-discovery, through reflections/conflicts, sometimes through several months

-  Things we should avoid/change:
- Some client actions or wishes conflict with our course timeline
  - Reports are designed for grading, but have no direct, positive impact on students or the project itself
  - Teams are balanced based on skills in order to maximize project success
  - Performance is measured based on the quality of the deliverables, but largely ignores the quality of teamwork



Enrollment continues to grow, while resources/support remain uncertain



# New Approach to Capstone



- Project option
  - Themes are provided, each team comes up with their own solution (within reason)
- Client involvement
  - Serves as external validation only, adopts a hackathon model of "judges"

# New Approach to Capstone



- Project option
  - Themes are provided, each team comes up with their own solution (within reason)
- Client involvement
  - Serves as external validation only, adopts a hackathon model of "judges"
- Minimal reporting
  - Long, written reports are eliminated from the course deliverables

# New Approach to Capstone



- Project option
  - Themes are provided, each team comes up with their own solution (within reason)
- Client involvement
  - Serves as external validation only, adopts a hackathon model of "judges"
- Minimal reporting
  - Long, written reports are eliminated from the course deliverables
- Team formation
  - Based on your preferences, after team matching activity
- Team coaching, reflection, self-management
  - Weekly updates that tell us about the project progress and team dynamics
  - Teams should reflect on the data reported to us

# Course Logistics

- In-person classes Tuesdays and Thursdays:
  - Lectures
  - Team reviews with teaching staff
- Weekly team checkpoints
  - Gives the teaching staff a "temperature" of the team
- Additional deliverables
  - Short reports, mini presentations (to class), video demos, and/or project portfolio descriptions
  
- Review Canvas course and syllabus
  - See: evaluation criteria
  - See: tentative schedule

# Course Re-Design and Team Formation Software

- Go to Canvas course for Capstone, select "Quizzes"
  - Select "Consent to Study"
  - In the description, click on Qualtrics link
  - Consent to give us access to your survey data after course
  - Complete for participation marks (regardless of your consent decision)

