Administrative Issues



EECS3342 E: System Specification and Refinement
Fall 2026

CHEN-WEI WANG

Instructor

- How may you call me?

  “Jackie” (most preferred),

  “Professor Jackie”, “Professor”, “Professor Wang”, “Sir”, “Hey”, “Hi”, “Hello”
- When you need ***advice*** on the course, speak to me!
- There will be a bonus opportunity for you to fill out an informal, anonymous ***midterm course survey*** during the reading week.
- Throughout the semester, feel free to suggest ways for helping your learning.

If You Are Not Enrolled Yet

- Send me an email ASAP requesting access to the course eClass site, with your ***name***, ***student #***, ***Passport York ID***.
- Still keep up with lectures & study materials.
- Still complete labs & tests (***no extension***).

Writing E-Mails to Your Instructor

- Think of me as your ***colleague*** who is happy to help you learn.
  - ***formality*** is unnecessary
  - ***courtesy*** is expected
- This sounds ***very rude*** (and may be delayed, if not ignored):

```
On the link you sent us for our mark
my mark for lab1 did not appear on it
and i submitted lab1 during my lab session
```

- This sounds ***much nicer***:

```
Hello Jackie, the link you sent didn't work.
I did submit my lab1. Could you please look into this?
Thanks! Jim
```

- ***in-person*** communication may be the *most effective*

  Slow/No responses to your email inquiries ⇒
  Jackie is happy to help during office hours and/or appointments.

Classroom Conduct & Expectations

- ***Lectures*** are recorded.
- ***Attendance*** is encouraged for ***learning*** and ***engagement***.
- *If you decide to attend,* ***respect*** *the learning environment:*
  - **No side conversations** – even low voices distract.
  - **Phones away** – except for iClicker checks.
  - **Headphones away**
  - **Laptops** – for taking notes and following slides **only**.
    Working on assignments *(this or other courses) =* ***disrespectful****.*
- This announcement serves as a **general warning**.
  I ***may*** *ask a student to leave* ***without*** *additional notice.*
- *I address behaviour based on* ***my assessment*** *of disruption.*
  That decision is ***not open to debate*** *during class time.*
- Refusal to leave ⇒ Wasting time of those who want to learn
  ⇒ **Campus security** may be involved as a last resort.

Open Letters from Jackie

This course is taught with a particular emphasis on foundational ***reasoning***, individual ***mastery***, professional ***standards***, and rigorous reasoning about software systems.

You are expected to read two open letters **carefully and early**:

1. ***General Open Letter***
   Explains my broader approach to ***learning***, ***assessment***, ***grading***, and ***professional responsibility***.

   `https://www.eecs.yorku.ca/~wangcw/teaching/lectures/2026/F/Letter-from-Jackie-Fall-2026.pdf`

2. ***EECS3342 Companion Letter***
   Builds on the first letter and explains why ***formal specification***, ***refinement***, ***verification***, and ***assurance*** matter in this course — including the reasoning behind its assessment structure.

   `https://www.eecs.yorku.ca/~wangcw/teaching/lectures/2026/F/EECS3342/notes/EECS3342-F26-Q&A.pdf`

Course Information

- A single eClass site:
  - ***LE/EECS 3342 E - System Specification and Refinement (Fall 2026-2027)***
    - Announcements
    - Reference Textbook
    - Lab Exercises
    - Written Tests [instructions & submissions]
    - Midterm Course Evaluation
    - End-of-Term Survey on Lecture Digests
- Please check your emails regularly!

In-Person Lectures

- Slides are ***self-contained***, so I may **not** just read them off.
- I will focus on explaining core concepts with examples.
- Your ***engagement*** is the key: ask ***questions***!

Required Study Materials

- Lecture materials (recordings, iPad notes, slides, codes) will be posted for you to ***re-iterate concepts and examples***:
  `https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS3342_F26`
- The ***course syllabus*** is posted in the above lectures site.
- Though Jackie **attempts** to record each lecture entirely:
  - *Not meant to be a replacement for classes!*
  - The purpose of recording is that you can focus on reaching ***maximum comprehension***.
    - *Ask questions!*
    - Take (even ***incomplete***) notes: they help when re-visiting lectures.
    - Review points which you need to ***re-iterate*** from the recordings.
  - It'd be ***your call*** to use the posted ***lecture recordings***:
    - either as a way to **review** details not understood for the first time;
    - or as an **excuse** to skip classes!

Course Syllabus

Let's go over the ***course syllabus***.

Foundations Matter More in the Age of AI

- Fact: AI can ***automate*** many routine tasks — and sometimes produce impressive solutions to difficult ones.
- Also fact: AI can still:
  - ***misinterpret*** requirements or hidden assumptions;
  - ***overlook*** edge cases;
  - produce output that ***looks right*** but fails under closer examination.
- The key issue is **not** whether AI can solve hard problems.
- It is whether **you can judge, repair, adapt, and take responsibility for what it produces**.

AI does not remove professional judgment.
AI makes professional judgment more valuable.

Leveraging AI Demands Solid Foundations

- Foundations build:
  - ***abstraction*** and ***logic***;
  - ***testing*** and ***debugging***;
  - **independent** ***reasoning***.
- When AI gives you an answer, these foundations let you:
  - ***state*** requirements, assumptions, and constraints precisely;
  - ***test*** edge cases and look for failures;
  - ***modify, debug, and adapt*** when conditions change;
  - ***decide what to trust*** — and explain why.
- Without them, shortcuts can create "***cognitive debt***": short-term productivity can hide gaps that surface later in debugging, maintenance, extensions, or unfamiliar problems.

AI can accelerate learning.
But AI cannot replace understanding.

Becoming a Software Engineer

- How a real ***software engineer*** works:
  - ***Problems*** are explained via the expected methods' ***API*** (i.e., I/O types) and some ***use cases***, without visualization!
  - A set of ***tests*** must be *re-run automatically* upon changes.
    [***regression testing***]
  - Thinking *abstractly* without seeing changes on a physical device is an important skill to acquire before graduating.
- This course, like EECS2011/2101, trains you for ***programming interviews***.
  - In 2011/2101, you think via ***concrete*** data structures.
  - In 3342, you think via ***abstract*** machines (and math functions).

  e.g., Watch *interviews at Google*: Given problems described in English, solve them on a whiteboard.

Index (1)

Instructor

If You Are Not Enrolled Yet

Writing E-Mails to Your Instructor

Classroom Conduct & Expectations

Open Letters from Jackie

Course Information

In-Person Lectures

Required Study Materials

Course Syllabus

Foundations Matter More in the Age of AI

Leveraging AI Demands Solid Foundations

Index (2)

Becoming a Software Engineer