

# GenAI-powered Software Engineering

CSCI 7000-011  
Tue/Thu @2pm

Danny Dig





**Have you received the welcome email I sent yesterday?**

# Today's goals

What is GenAI for Software Engineering: practice and research examples

Discovery: learning about each other (Family Occupation Recreation Motivation), expectations

How can I be successful in CSCI 7000-011?

# Assistant augments our capacity

11:49   

 Workouts Sat, Sep 9 

 Outdoor Cycle  
Open Goal  
8:42 AM–9:34 AM  
Cottonwood Heights



## Workout Details [Show More](#)

Workout Time  
**0:51:15**

Distance  
**9.53MI**

Active Calories  
**1,148CAL**

Total Calories  
**1,256CAL**

Elevation Gain  
**2,599FT**

Avg. Speed  
**11.1MPH**

Avg. Heart Rate  
**142BPM**



85 Nm, 500 Wh

# 9 hardest things programmers do

Source: Elizabeth Churchill – UX Director @ Google

- Writing tests
- Writing documentation
- Implementing functionality you disagree with...
- Working with someone else's code
- Dealing with other people...
- Estimating time to complete a task
- Explaining what I do (or don't do)
- Naming things

# GenAI increases programmer productivity

Not displacing programmers, but making programmers more productive



## Amazon CodeWhisperer

Your AI-powered productivity tool for the IDE and command line

### A giant leap forward in developer productivity

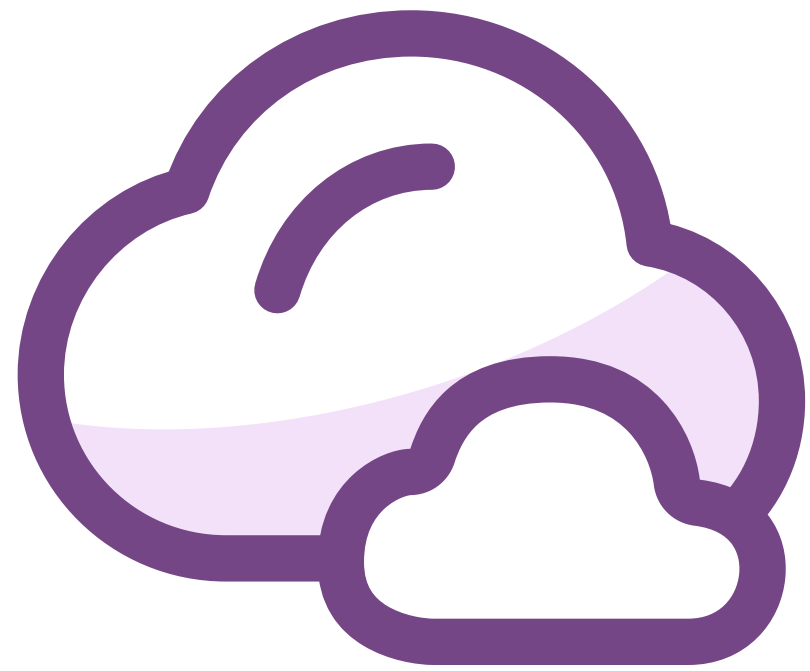
**57%** faster

**27%** more likely to succeed

Amazon ran a productivity challenge during the preview, and participants who used CodeWhisperer were 27% more likely to complete tasks successfully and did so an average of 57% faster than those who did not use CodeWhisperer.



**Have you used any AI-powered tools in your coding projects?**



**Which GenAI tools and services did you use for programming?**

# AI explains Runtime Errors

Programmers spend more time reading code than writing code

```
1 package org.example;
2
3 import java.util.stream.Stream;
4
5 public class Main {
6     public static void main(String[] args) {
7         var list = Stream.iterate( seed: 1, i -> i + 1)
8             .toList();
9         System.out.println(list.size());
10    }
11 }
```

Run Main

Exception in thread "main" java.lang.OutOfMemoryError: Java heap space Explain with AI

- at java.base/java.lang.Integer.valueOf(Integer.java:1081)
- at org.example.Main.lambda\$main\$0(Main.java:7)
- at org.example.Main\$\$Lambda\$14/0x00000008000c0940.apply(Unknown Source)
- at java.base/java.util.stream.Stream\$1.tryAdvance(Stream.java:1464)
- at java.base/java.util.Spliterator.forEachRemaining(Spliterator.java:332) <3 internal lines>
- at java.base/java.util.stream.AbstractPipeline.evaluateToArrayNode(AbstractPipeline.java:260) <3 internal lines>
- at org.example.Main.main(Main.java:8)

Process finished with exit code 1

# AI suggests names

Programmers don't spend enough time giving intention revealing names to code entities, though it is crucial for code readability

What's in a name?

Romeo and Juliet, William Shakespeare

```
public class TheAlgorithm {  
    private static final int COLINEAR = 0;  
    private static final int CLOCKWISE = 1;  
    private static final int COUNTERCLOCKWISE = 2;
```



```
@ public static int t(D p, D q, D r) {
```

```
    int val = (q.y - p.y) * (r.x - q.x) -  
              (q.x - p.x) * (r.y - q.y);
```

```
    if (val == 0) return COLINEAR;
```

```
    return (val > 0) ? CLOCKWISE : COUNTERCLOCKWISE;
```

```
}
```

```
@ public static List<D> run(D[] dts) {
```

```
    if (dts.length < 3) return Collections.emptyList();
```

# Multi-file changes

Programmers fail to propagate changes consistently

```

public class Message {
    String text;
    String sender;

    public Message(String text, String sender) {
        this.text = text;
        this.sender = sender;
    }

    public String getText() {
        return text;
    }




    public void setText(String text) {
        this.text = text;
    }

    public String getSender() {
        return sender;
    }

    public void setSender(String sender) {
        this.sender = sender;
    }
}

```

Java

Apply   

```

        messageService.deleteMessage(id);
        return new ResponseEntity<>(HttpStatus.NO_CONTENT,
    }
}

```

Collapse ^


## Explanation:


1. **Dependencies:** Added Spring Data JPA and MySQL Connector dependencies.
2. **Database Configuration:** Set up the database connection properties.
3. **Entity Class:** Modified the Message class by:
  - Adding `@Entity` and `@Table` annotations
  - Adding an ID field with appropriate annotations
  - Adding a default constructor (required by JPA)
4. **Repository:** Created a MessageRepository interface that extends JpaRepository for CRUD operations.
5. **Service Layer:** Added a service class to handle business logic.
6. **Controller:** Created a REST controller to expose API endpoints.

Make sure you have MySQL installed and running on your system. The configuration will create a database named "messagedb" if it doesn't exist.

Ask AI Assistant, use @mentions (#mentions) or /commands

Codebase Off

+  Claude 3.7 Sonnet v

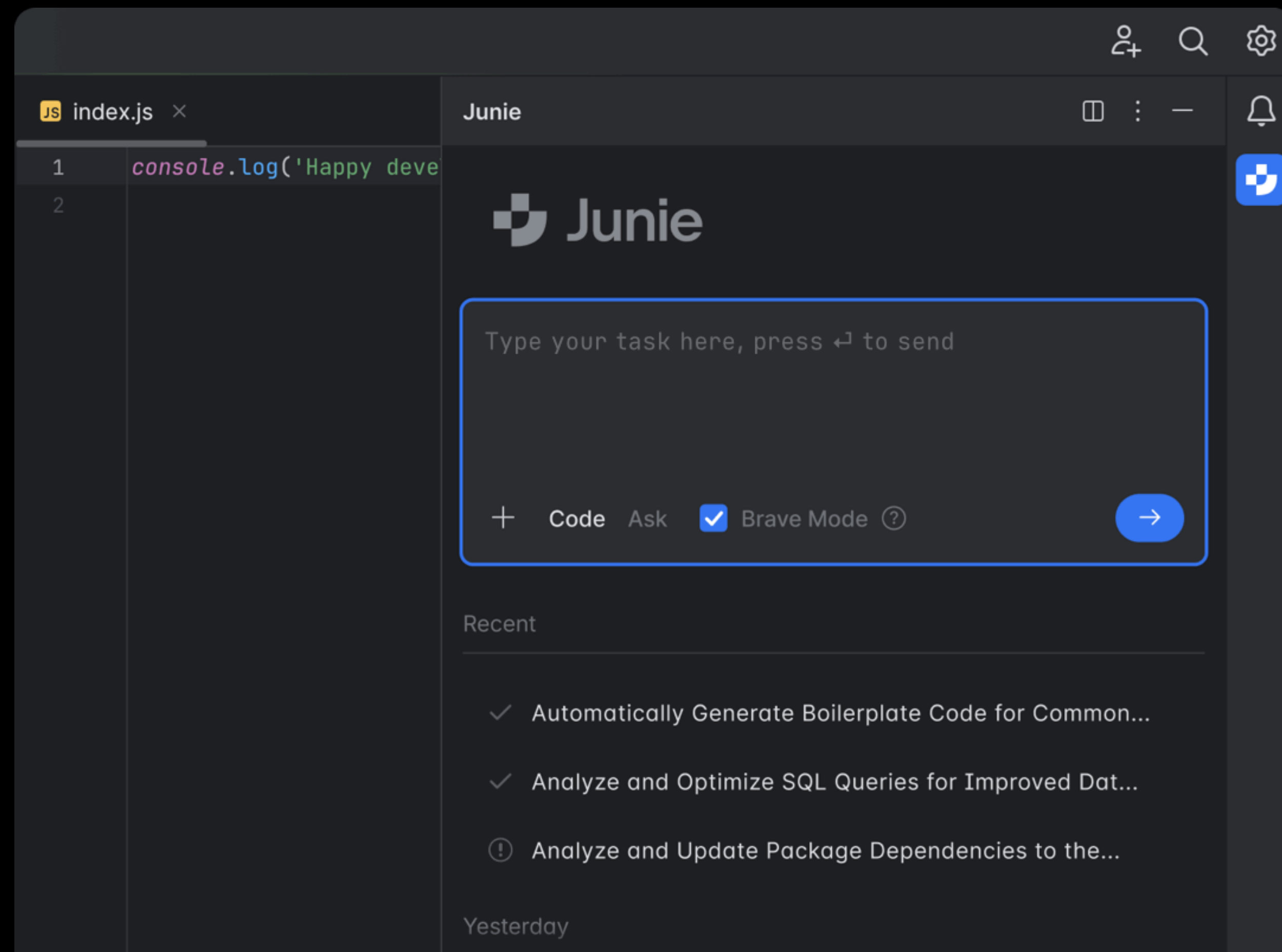
Chat 

Share feedback ↗



# Smart coding agent in the IDE

Lighten programmer's load





# Code and ASK mode

Lighten programmer's load

What\_neural\_network\_model\_works

Junie

Web App for Automatic Kettlebell Jerk Rep Tracking

What neural network model works best for pose detection in low light conditions? Make sure that the options are compatible to be used in our project

Response to Your Question

# Neural Network Models for Pose Detection in Low Light Conditions with TensorFlow.js

Based on the analysis of the project and current research in pose detection, here are the best neural network models for pose detection in low light conditions that are...

Open in Editor

Write follow up here

+

Code

Ask

☒ Brave Mode ?

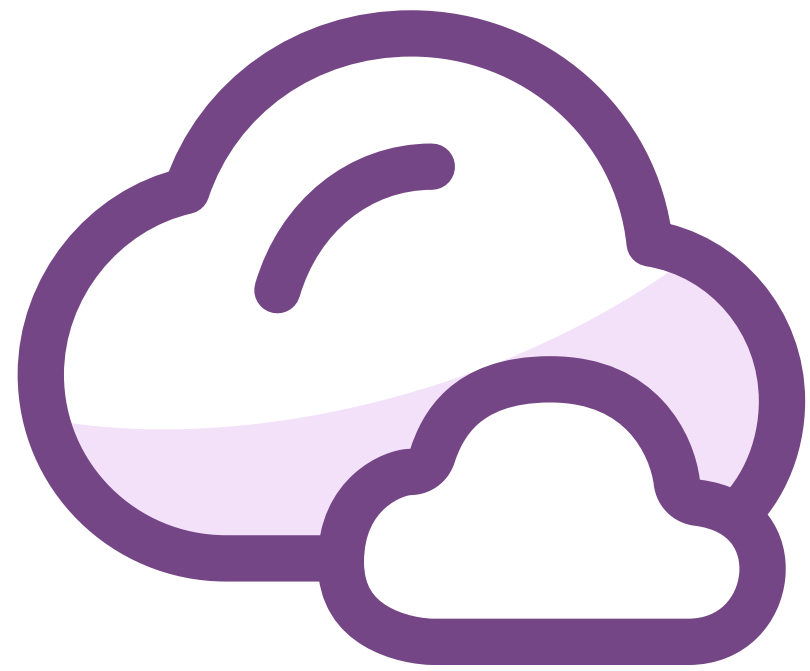
→

## Neural Network Models for Pose Detection in Low Light Conditions

Based on the analysis of the project and current research in pose detection, here are the best neural network models for pose detection in low light conditions that are...

### Best Options for Low Light Conditions

1. **MoveNet Thunder** - Already integrated into TensorFlow.js
  - More accurate than LightningPose
  - Better performance in challenging conditions
  - Slower but more robust for low light conditions
  - Fully compatible with the current project setup
2. **BlazePose** - Available in TensorFlow.js
  - Specifically designed for more accurate pose detection
  - Higher accuracy in challenging conditions
  - Supports 33 keypoints (vs. 17 for MoveNet)
  - Compatible with TensorFlow.js



**Which GenAI features would you find most useful (i.e., excite you the most) in your software development?**

# Examples: GenAI-powered SE Research

# Next Generation Refactoring: LLM Insights and IDE for ExtractMethod



Dorin  
Pomian



Boulder



Abhiram  
Bellur



Boulder



Malinda  
Dilhara



Boulder



Zarina  
Kurbatova



Andrey  
Sokolov



Egor  
Bogomolov



Timofey  
Bryksin



Danny  
Dig



Boulder

# Executive Summary:

IDE + LLM + Human >> Sum of the individual parts

Extract Method Refactoring: Splitting up large method (for modularity, reuse, ...)

- No suggestions from IntelliJ, though Execution is fully automated
- Previous research tools generate suggestions that do not align with developers

- LLM's create wow effects! but also have a high hallucination rate – 59.2%
- Tame Hallucinations, non-determinism
- Evaluating quality
  - By replicating thousands of OSS refactorings, EM-Assist outperforms previous SOTA (Recall is 53% vs 39% other tools)
  - User study – 18 JetBrains devs, 94% suggestions positively rated

EM-Assist aligns with how expert developers perform extract-method.

*"These suggestions made me look at this code with new eyes, and I will refactor it."*

# Software Testing Research @JetBrains



**Objective:** Develop new approaches to support developers and testers in their software testing tasks

## Projects



LLM-based &  
Search-based  
Unit Test Generation



Language-agnostic  
LLM-based Test Generation



Testing AI features

Metrics:

Compilation rate  
Line/branch coverage

Automatically  
Reproducing  
Reported Crashes



1) TestSpark paper: <https://arxiv.org/pdf/2401.06580>

2) Test Wars: A Comparative Study of SBST, Symbolic Execution, and LLM-Based Approaches to Unit Test Generation [ICST'25]

**Getting to know you**

# Quiz: About YOU

Your name

- **FORM** (family, occupation, recreation, motivation)
- **Grad Program** (e.g., CS PhD, MS, MSCP, etc.), year of study, who is your grad advisor
- Your **background** (e.g., industry experience, other CS background – such as strong SE, ML/AI, Systems, etc.)
- What is the ONE Thing that you **expect** to take out of CSCI 7000-011?

What are your plans **post graduation**?

# **What are your expectations from CSCI 7000-011: GenAI for SE?**

- A. Do a cool GenAI Project that helps programmers/society**
- B. Learn about exciting GenAI Applications for SE**
- c. Exposure to research on GenAI for SE**
- D. Learn about the Challenges in building GenAI software**
- E. Learn how GenAI software differs from classical software**
- F. How to leverage the strengths of each tool**
- G. How academia changes to adapt to AI**
- H. Best practices and ethical use of these tools**
- I. How to make scalable GenAI systems**



**What are your expectations from CSCI 7000-011: GenAI for SE**

# Course Administration

---

**Check webpage:**

**[https://danny.cs.colorado.edu/courses/csci7000-011\\_F25/](https://danny.cs.colorado.edu/courses/csci7000-011_F25/)**

**Work items due before next Tue (08/26):**

- Familiarize with class webpage**
- sign up on Piazza (all communications through Piazza, no email after this week)**
- Read and write critiques for two research papers (see template on webpage)**

**Check prerequisites: computing background (either practical experience or undergrad-level knowledge of SE, Systems, ML), please check with me after the class**

# CS 7000-011 is Different!!!

---

This class is **NOT**:

- an introduction to Software Engineering (take instead CSCI 5828)
- a tutorial on GenAI APIs/tools (see free resources on the internet)
- an asynchronous course (you need to participate live each class)

# CS 7000-011 is Different!!!

---

## **Research**-based course:

- at times it would feel it is not "organized"
- there are lots of choices, you need to select
- structure is fixed, but content is dynamic

**Complete a research or industrial-novel project of your choice (teams of 4 students)**

- follow the steps of open-ended/risky research (proposal, fit in literature, evaluate empirically)
- at the end of the term you would have produced a research paper that you can submit to conference
- **WHY:** equips you to lead **novel R&D**

# CS 7000-011 is Different!!!

---

Participate in **live** class discussion and activities.

Read 1-2 research papers for every class meeting (11 pages each, double column => total of 500+ research pages)

- later on, you choose papers that match your project

Paper Critiques: for each class meeting, for each research paper, submit before class (by 5pm previous day)

- WHY: equips you with **critical thinking**

Research presentation: you prepare and deliver for the selected research papers

- WHY: equips you to **communicate** your ideas

# Projects on GenAI for SE

---

**For new grad student, project gives ideas for dissertation**

**For experienced grad student, project advances your research**

**Technological shifts/opportunities for GenAI:**

**- any process & tool we use in software development lifecycle (requirements, design, coding, testing, evolution)**

**Academic-research:** generate new insights and knowledge

**Industrial-innovation:** application of established practices to produce a novel software powered by GenAI

**Research projects (not an app), teams of 4 people**

# Testimonials

---

***“This course was exceptionally difficult and carried a massive time requirement. I only took two courses this quarter and even then I'm not sure I was able to give either of the classes the amount they deserved because of the time requirement for this class.”*** [student from previous years]

***“... But that doesn't mean that I'm not satisfied, have students who previously took the course convey this”***

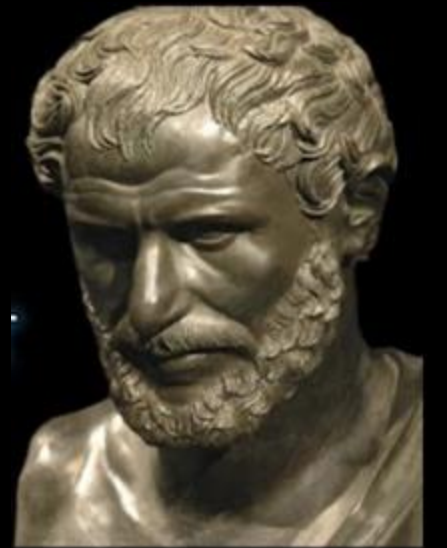
**Announcements for next class:**

- **Read the webpage, signup on Piazza**
- **Read two papers and submit Paper Critique in Canvas (by Monday at 5pm)**

**Office hours: right after the class**

# Executive Summary

Change is the only guaranteed constant



GenAI + Tools + Human

"...Game changer! Enter the AI assistant, the great equalizer." -Lenovo CTO

"AI **will not** displace programmers. But programmers who use AI **will** displace programmers who do not use AI." – Prof Danny Dig

**Together We Go Further**