AI ASSISTED CODING LAB 5.2

Lab 5: Ethical Foundations – Responsible AI Coding Practices

ENROLLMENT NO: 2403A52006

NAME: MOHAMMED MUSTAFA

COURSE: CSM(AI/ML)

BATCH: 01

DATE: 09-09-2025

Task Description 1: (Privacy and Data Security):   
• Use an AI tool (e.g., Copilot, Gemini, Cursor) to generate a login system. Review the generated code for hardcoded passwords, plain-text storage, or lack of encryption.

Used Prompt:

using python, generate a login system for the user. take inputs username and password from the user. make sure from the input that the password contains uppercase letters, lowercase letters, numbers and symbols are optional. the min length of the password must be 6 characters, and max length must be 10 characters.

A screenshot of a computer program

AI-generated content may be incorrect.

A black rectangular object with a white line

AI-generated content may be incorrect.

Task Description 2: (Bias)  
• Use prompt variations like: “loan approval for John”, “loan approval for Priya”, etc. Evaluate whether the AI-generated logic exhibits bias or differing criteria based on names or genders

Used Prompt:

using python develop bank loan system. ask user to input these: check for the annual income for the user(use a condition), check for the credit score of the user(use a condition), and check for the age(must be >18) of the user(theses three are mandatory conditions). ask the user to input name, account number(exact 10 digits) and amount for the loan(these are just the inputs not mandatory conditions)

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Mitigation Techniques: In coding, **mitigation techniques** are practices used to reduce risks, bugs, and security vulnerabilities in software. One of the most important techniques is **input validation**, where all user inputs are checked and sanitized before being processed. It is about following **secure coding practices** that protect applications from vulnerabilities, reduce risks, and ensure data safety.

Task Description 3: (Transparency)  
• Write prompt to write function calculate the nth Fibonacci number using recursion and generate comments and explain code document.

Used Prompt: generate a python code to take a number as input from the user and find the Fibonacci series of that number. Also use comment lines wherever necessary to explain the code.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Explanation:

=> The program has a function that tells how to find a Fibonacci number — if you give it a number like 5, it calculates the Fibonacci series up to 5 numbers

=> It then asks you to type a number. If the input is not valid, it will terminate the program.

=> If you give a proper number, it prints the Fibonacci sequence from 0 to the number you entered.

Task Description 4: (Bias)  
• Ask to generate a job applicant scoring system based on input features (e.g., education, experience, gender, age). Analyse the scoring logic for bias or unfair weightings.

Used Prompt:

using python create a job applicant scoring system based on some input taken from the user such as qualifications, experience, age, soft skills(communication, engaging with teammates,leadership). check for all these conditions and based on the inputs by the user make a scoring system up to 10 . if it is >7 then he/she gets hired.

A screenshot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Task Description 5: (Inclusiveness)

def greet\_user(name,gender):

if gender.lower() == "male";

title = "Mr."

else :

title = "Mrs."

return (f"hello, {title} {name} ! Welcome").

Regenerate code that includes gender-neutral also

Used Prompt :

def greet\_user(name,gender):

if gender.lower() == "male";

title = "Mr."

else :

title = "Mrs."

return (f"hello, {title} {name} ! Welcome").

Now in the given code add a gender neutral category.

A screenshot of a computer program

AI-generated content may be incorrect.