|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE** | | | | | **DEPARTMENT OF COMPUTER SCIENCE ENGINEERING** | | | | |
| **ProgramName:**B. Tech | | | | **Assignment Type: Lab** | | | **AcademicYear:**2025-2026 | | |
| **CourseCoordinatorName** | | | | Venkataramana Veeramsetty | | | | | |
| **Instructor(s)Name** | | | | |  | | --- | | Dr. V. Venkataramana (Co-ordinator) | | Dr. T. Sampath Kumar | | Dr. Pramoda Patro | | Dr. Brij Kishor Tiwari | | Dr.J.Ravichander | | Dr. Mohammand Ali Shaik | | Dr. Anirodh Kumar | | Mr. S.Naresh Kumar | | Dr. RAJESH VELPULA | | Mr. Kundhan Kumar | | Ms. Ch.Rajitha | | Mr. M Prakash | | Mr. B.Raju | | Intern 1 (Dharma teja) | | Intern 2 (Sai Prasad) | | Intern 3 (Sowmya) | | NS\_2 ( Mounika) | | | | | | |
| **CourseCode** | | | 24CS002PC215 | **CourseTitle** | | AI Assisted Coding | | | |
| **Year/Sem** | | | II/I | **Regulation** | | R24 | | | |
| **Date and Day**  **of Assignment** | | | Week3 - Tuesday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:5.2**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | Lab 5: Ethical Foundations – Responsible AI Coding Practices  **Lab Objectives:**   * To explore the ethical risks associated with AI-generated code. * To recognize issues related to security, bias, transparency, and copyright. * To reflect on the responsibilities of developers when using AI tools in software development. * To promote awareness of best practices for responsible and ethical AI coding.   **Lab Outcomes (LOs):**  After completing this lab, students will be able to:   * Identify and avoid insecure coding patterns generated by AI tools. * Detect and analyze potential bias or discriminatory logic in AI-generated outputs. * Evaluate originality and licensing concerns in reused AI-generated code. * Understand the importance of explainability and transparency in AI-assisted programming. * Reflect on accountability and the human role in ethical AI coding practices..   **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.   **Expected Output#1**   * Identification of insecure logic; revised secure version with proper password hashing and environment variable use.   **Prompt1:**  Generate a simple login system in Python with username and password checking.  **Code**    **Prompt 2**  Generate a simple login system in python with 20 in-built usernames and passwords checking  **Code**    **Observations**   1. The program is a login system with multiple users and passwords stored in a dictionary. 2. If the login is successful, it stops immediately with a success message.   **Code Explanation**   1. A dictionary users stores usernames and passwords. 2. Program asks for username and password. 3. If username exists and password matches → Login sucessful. 4. If wrong → it reduces attempts and shows how many are left.   **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.   **Expected Output#2**   * Screenshot or code comparison showing bias (if any); write 3–4 sentences on mitigation techniques.   **Prompt 2**  Generate a simple Python function that decides loan approval for a person based on their name,credit score and income.  **Code**    **Observations**   1. Works correctly for basic loan approval. 2. Too simple for real banks . 3. Uses fixed numbers . 4. No check if input is wrong .   **Code Explanation**   1. The program asks for name**, credit score, and income**. 2. If **credit score ≥ 650** and **income ≥ 30,000** → loan is **Approved**. 3. Otherwise → loan is **Denied**. 4. It prints the result with the person’s name.   **Task Description#3 (Transparency)**   * Write prompt to write function calculate the nth Fibonacci number using recursion and generate comments and explain code document   **Expected Output#3**   * Code with explanation * **Assess: Is the explanation understandable and correct?**   **Prompt 3**  generate a Python function to calculate the nth Fibonacci number using recursion.  **Code**    **Observations**   1. correctly calculates Fibonacci numbers using recursion. 2. Has input validation (non-negative and integer check). 3. Recursion is **slow for large numbers** .   **Code Explanation**   1. A function fibonacci\_recursive(n) is defined. 2. The program asks the user for a number . 3. If the number is **negative**, it shows an error message. 4. If the number is valid, it prints the nth Fibonacci number. 5. If the input is not an integer (like letters), it shows "Invalid input. Please enter an integer."   **Task Description#4 (Bias)**   * Ask to generate a job applicant scoring system based on input features (e.g., education, experience, gender, age). Analyze the scoring logic for bias or unfair weightings.   **Expected Output#4**   * Python code * Analyze is there any bias with respect to gender or any   **Prompt**  Generate a Python function that scores job applicants based on features such as education, experience, gender, and age.  **Code**    **Observations**   1. Uses **education, experience, and age** to calculate score. 2. Gender is included in input but not affecting score → avoids **bias**. 3. Gender input is unnecessary if it’s not used at all.   **Code Explanation**   1. A function score\_applicant() is defined. 2. Education is scored from a dictionary:   · High School = 10, Bachelor = 20, Master = 30, PhD = 40.   1. Experience adds **5 points per year**, up to a maximum of 50. 2. Age gives extra points:   · If 25–45 → +15 points,If 20–25 or 45–55 → +5 points.   1. Gender is taken as input but **not used** in scoring. 2. Finally, the total score is returned and printed.   **Task Description#5 (Inclusiveness)**   * Code Snippet     **Expected Output#5**   * Regenerate code that includes **gender-neutral** also   **Prompt**  Make a Python program that greets a person by name. It should not assume only male or female. If gender is unknown or non-binary, use a neutral title.  **Code**    **Observations**   1. supports **male, female, and non-binary/unspecified**. 2. Neutral option "Mx." makes it more respectful. 3. Easy to use with simple inputs.   **Code Explanation**   1. The function greet\_person(name, gender) greets people by name. 2. If gender is **male**, it uses "Mr.". 3. If gender is **female**, it uses "Ms.". 4. If gender is **other/unknown**, it uses "Mx." (neutral title). 5. The program asks the user for:   · Their **name,**Their **gender** (Male, Female, Other, or leave blank).  6.Finally, it prints a greeting  **Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots**  **Evaluation Criteria:**   | **Criteria** | **Max Marks** | | --- | --- | | Transparency | 0.5 | | Bias | 1.0 | | Inclusiveness | 0.5 | | Data security and Privacy | 0.5 | | **Total** | **2.5 Marks** | | | | | | | Week3 - Wednesday |  |