School of Computer Science

P.G.D.C.A. (Post Graduate Diploma in Computer Applications) Programme

Subject Code & Name: PGDCA206-2 - Practical based on PGDCA204-2 and PGDCA205-2

Teaching And Evaluation Scheme:

Teaching Scheme				Examination Scheme			
Credits				Component Weightage			
				CCE		SEE	
L	T	P	Total	TH	PWE	TH	PWE
-	-	4	4	-	50%	-	50%

Programme Name	P.G.D.C.A.
Semester	2
Course Code	PGDCA206-2
Course Title	Practical based on PGDCA204-2 and PGDCA205-2
Course Content Type (Th. / Pr.)	Pr.
Course Credit	4
Sessions + Lab Per Week	4
Total Teaching/ Lab Hours	120
*** 2 Lab = 1 session**	

Learning Objectives
Upon successful completion of this course, students will be able to: • Understand the fundamental concepts, history, evolution, and applications of Artificial Intelligence.

- Differentiate among Artificial Intelligence, Machine Learning, Deep Learning, and Generative AI technologies.
- Understand the architecture, capabilities, and limitations of Large Language Models (LLMs) and Generative AI systems.
- Apply prompt engineering techniques and frameworks to interact effectively with AI systems.
- Analyse AI-generated outputs and optimize prompts for improved accuracy and performance.
- Evaluate ethical, legal, security, and societal implications of AI technologies.
- Design AI-assisted workflows, intelligent applications, and domain-specific prompt solutions.
- Develop practical skills in using Generative AI tools for research, software development, data analysis, and content creation.
- Understand the concepts of mobile applications, Android architecture, development tools, and Flutter development environment setup
- Apply Dart programming constructs, object-oriented programming principles, collections, exception handling, and asynchronous programming techniques to solve computational problems
- Develop mobile user interfaces and application functionality using Flutter widgets, layouts, forms, navigation, state management, and API integration.
- Design and implement interactive mobile applications using Flutter animations, Firebase services, authentication, cloud database operations, and deployment techniques.

Learning Outcomes

After successful completion of the course, students will be able to:

- Explain the concepts, evolution, classifications, technologies, and applications of Artificial Intelligence and its role in modern computing systems.
- Utilize Generative AI tools and Large Language Models to perform knowledge discovery, content generation, research assistance, and problem-solving tasks.
- Construct effective prompts using standard prompt engineering techniques and frameworks for obtaining desired outputs from AI systems.
- Analyse AI-generated responses and evaluate prompt effectiveness by identifying limitations, biases, hallucinations, and optimization opportunities.
- Assess ethical, legal, privacy, security, and governance challenges associated with the deployment and use of Artificial Intelligence technologies.
- Design and develop AI-assisted workflows, domain-specific prompt libraries, and innovative solutions using advanced prompt engineering and Generative AI technologies.

- Explain different types of mobile applications, Android architecture, Android software stack, and Flutter development tools
- Develop Dart programs using variables, control statements, functions, collections, OOP concepts, exception handling, and asynchronous programming constructs.
- Create responsive Flutter applications incorporating widgets, user interactions, routing, state management, assets, and RESTful API data consumption.
- Build Flutter applications with animations, Firebase authentication, CRUD operations, and APK generation for real-world mobile platforms.

CLO–PLO Mapping Matrix:

CLOs/PLOs	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CLO1	3	2	2	3	3	3
CLO2	2	3	2	3	3	3
CLO3	2	2	2	3	3	3
CLO4	3	3	3	2	2	2
CLO5	3	3	3	3	2	2
CLO6	3	3	3	3	3	3

L:: Lecture, **T::** Tutorial, **P::** Practical

CCE:: Continuous & Comprehensive Evaluation

(CCE theory includes Mid Semester Examination, Assignment, MCQ Quizzes, Seminar, Reflective Notes, Class Participation, Case Analysis and Presentation, Slip Tests (Announced/ Surpirsed), Attendance etc. or any combination of these)

PWE: Practical Work Examination

(PWE includes Laboratory Practical Work, Project Work, Viva simulation exercise work etc.)

SEE:: Semester End Evaluation

Sample Practical Exercises based on the subject PGDCA204-2-Introduction to AI and Prompt Engineering

1. Generate a short story using Generative AI.
2. Generate a poem using AI.
3. Generate a business letter using AI.
4. Generate a resume using AI assistance.
5. Demonstrate tokenization using sample text.

6. Analyse token count of different prompts.
7. Create examples of AI hallucinations.
8. Write a simple prompt and analyse output.
9. Improve an ineffective prompt.
10. Create prompts with clear instructions.
11. Create prompts with contextual information.
12. Design prompts for educational content generation.
13. Design prompts for business report generation.
14. Design prompts for coding assistance.
15. Design prompts for data analysis.
16. Create prompts for translation tasks.
17. Create prompts for summarization tasks.
18. Perform Zero-Shot Prompting.
19. Perform One-Shot Prompting.
20. Perform Few-Shot Prompting.
21. Implement Chain-of-Thought prompting.
22. Create Role-Based prompts for teachers.
23. Create Role-Based prompts for software developers.
24. Create Role-Based prompts for researchers.
25. Implement Step-by-Step prompting.
26. Experiment with Tree-of-Thought prompting.
27. Design prompts using RTF framework.
28. Design prompts using CARE framework.
29. Design prompts using RISEN framework.
30. Design prompts using CRISPE framework.
31. Evaluate prompt quality using defined metrics.
32. Generate blog articles using AI.
33. Generate marketing content using AI.
34. Generate social media posts using AI.
35. Create presentation outlines using AI.
36. Use AI for code generation.
37. Use AI for debugging Python programs.
38. Generate SQL queries using AI.
39. Generate charts and visualization suggestions using AI.
40. Create lesson plans using AI.

Sample Practical Exercises based on the subject PGDCA205-2-Mobile Application Development using Flutter

1. Installation and environment setup of the Flutter and Dart SDKs within Android Studio.
2. Write a Dart program to demonstrate variables, constants, and different data types.
3. Write a Dart program to perform arithmetic, relational, logical, and assignment operations.
4. Write a Dart Program to implement control flow statements (if, if-else, switch-case, for, while, do-while).
5. Develop a Dart application to find factorial, Fibonacci series, and prime numbers.
6. Write a Dart program demonstrating user-defined functions with parameters and return values.
7. Write a Dart Program that utilizes anonymous functions and lambda expressions.
8. Write a Dart program to perform operations on List collections.
9. Develop a Dart application using Set collections to manage unique records.
10. Write a Dart program demonstrating Map collections for key-value data storage.
11. Develop a Dart application implementing classes, objects, and constructors.
12. Write a Dart program demonstrating inheritance and method overriding.
13. Develop a Dart application implementing abstraction and encapsulation concepts.
14. Write a Dart program demonstrating exception handling using try-catch-finally blocks.
15. Develop a Dart application using Future, async, and await for asynchronous operations.
16. Develop a Flutter application displaying Text, Image, Icon, and Button widgets.
17. Create a Flutter application demonstrating Stateless Widgets.
18. Create a Flutter application demonstrating Stateful Widgets with dynamic data updates.
19. Design a user interface using Row, Column, and Container widgets.
20. Develop a Flutter application using Stack widget for layered UI design.
21. Create a scrolling interface using ListView and GridView widgets.
22. Develop a Flutter application to accept user input using TextField, Checkbox, Radio Button, and Dropdown widgets.
23. Create a user registration form with validation using Flutter Forms.
24. Develop a multi-screen Flutter application using Navigation and Routing.
25. Create a Flutter application that displays local assets such as images and icons.
26. Develop a Flutter application using external packages from pub.dev.
27. Create a Flutter application demonstrating state management using setState().

28. Develop a Flutter application that consumes data from a REST API using the HTTP package.
29. Create a Flutter application to fetch and display JSON data in a ListView.
30. Create a Flutter application demonstrating local data persistence using SharedPreferences.
31. Develop a Flutter application demonstrating implicit animations.
32. Create a Flutter application using explicit animations with Animation Controller.
33. Configure Firebase and integrate it with a Flutter application.
34. Develop a Flutter application implementing Firebase Email and Password Authentication.
35. Develop a Flutter application performing CRUD operations using Firebase Firestore Database.
36. Generate an APK file and test the application on a physical Android device.