University of Asia Pacific (UAP)

Department of Computer Science and Engineering (CSE)

BSc in CSE Program

Course Outline – Object Oriented Programming II Lab: Visual and Web Programming Lab

Part A – Introduction

1. **Course No. / Course Code:** CSE 302
2. **Course Title:** Object Oriented Programming II: Visual and Web Programming Lab
3. **Course Type:** Core course
4. **Level/Term and Section:** 5th Semester (3nd Year/1st Semester)
5. **Academic Session:** Spring 25
6. **Course Instructor:** Durjoy Mistry, Atia Rahman Orthi
7. **Prerequisite (If any):** None
8. **Credit Value:** 1.5
9. **Contact Hours:** 3.0
10. **Total Marks:** 100
11. **Course Objectives and Course Summary:**

This lab course focuses on the practical application of Django and modern web development techniques to build real-world, data-driven applications. Students will integrate backend and frontend components, apply object-oriented design principles, and utilize industry-standard tools to deliver functional, user-friendly solutions.

Course Objectives:

- Understand and apply object-oriented design principles in the context of Django-based web applications.

- Design and implement dynamic, database-driven web applications using Django's MVT architecture.
- Integrate frontend technologies (HTML, CSS, JavaScript) with Django templates for interactive user experiences.
- Utilize modern development tools, frameworks, and version control systems to streamline the development process.
- Collaborate effectively in teams, demonstrating professional ethics, clear communication, and problem-solving skills.

12. Course Learning Outcomes: at the end of the Course, the Student will be able to

CLO1	Demonstrate foundational knowledge of basic programming and web development, with OOP for the backend and HTML/CSS for the frontend.
CLO2	Identify complex problems in web application development by analyzing user requirements and system constraints.
CLO3	Design a web application using MVT/MVC framework, incorporating models, views, templates, forms, and database integration.
CLO4	Implement and deploy a secure, responsive, and functional Django-based web application addressing a real-world scenario.
CLO5	Collaborate effectively in diverse teams to manage project timelines, roles, and deliverables using version control (e.g. Git, GitHub etc.)
CLO6	Prepare technical documentation and deliver effective oral presentations justifying design decisions and technical outcomes.

13. Mapping / Alignment of CLOs with Program Learning Outcomes (PLO) (Optional):

CLO No	Corresponding PLOs	Bloom's Taxonomy Domain/Level	Delivery Methods and Activities	Assessment Tools	K	P	A
CLO1	PLO-a	1 / Apply	Demonstration, tutorials, guided labs	Quizzes, lab tests	K3	P1	A2
CLO2	PLO-b	1 / Analyze	Requirement analysis sessions, group discussions	Project Proposal, Requirement analysis, Viva	K5	P2	A2

CLO3	PLO-c	1 / Create	Group project design phase, weekly mentoring	Design documentation, demo	K4, k5	P3	A2
CLO4	PLO-e	2 / Precision	Coding sprints, demo labs	Final project demo, code inspection	K6	P7	A4
CLO5	PLO-i	Affective (3) / Valuing	Version control demo, team mentoring, peer reviews	Peer evaluation, Viva voce, GitHub repo, commit history, version control report	K6	P1	A5
CLO6	PLO-j	Knowledge (1) / Analyze	Report writing sessions, presentation rehearsals	Project report, final presentation, Viva voce	K7	P3	A2

Part B – Content of the Course

14. Course Content:

15. Alignment of topics of the courses with CLOs:

SL. No	Topics / Content	Course Learning Outcome (CLO)
1	**OOP / Language Basic**	**CLO1**
2	**Identify CEP, Requirement analysis**	**CLO2**
3	**Designing project architecture**	**CLO3**
4.	**Development of the project**	**CLO3, CLO4**
5.	**Effective Collaboration**	**CLO5**
6.	**Technical document presentation**	**CLO 6**

16. Class Schedule/Lesson Plan/Weekly plan:

Week	Topics	Course Outcome (CLO)	Delivery Methods and Activities	Reading Materials

Week 1	Python syntax, data types, control structures, functions, loops, list	1	Lecture, multimedia, demonstration	Slides
Week 2	Fundamentals of Web Development (Front End): HTML, CSS, Portfolio Development	3	Lecture, multimedia, demonstration	Slides
Week 3	Version Control with Git, Basic Requirement Analysis, Class Diagram, CEP, Project Idea formulation	5,3	Lecture, multimedia, demonstration	Slides
Week 4	Django environment setup, MVT structure, URL routing, views, first Django application	4	Lecture, multimedia, demonstration	Slides
Week 5	Formal Project Proposal Presentation	6,2	—	—
Week 6	Django Administration: Superuser creation, model registration in admin interface, CRUD operations with class-based objects and methods	4	Lecture, multimedia, demonstration	Slides
Week 7	Django Forms: View-Based CRUD Operations	4	Lecture, multimedia, demonstration	Slides
Week 8	Bootstrap for clean, responsive UI design, cloning and collaboration on project repositories, potential problem discussion, OOP concepts in Django (inheritance, access modifiers), user authentication (Login/Signup)	4	Lecture, multimedia, demonstration	Slides
Week 9	Midterm Lab Examination: Individual Application Development, View CRUD	1,3,4	—	—
Week 10	Media file handling in Django applications	4	Lecture, multimedia, demonstration	Slides
Week 11	Managing relationships: multiple models and Django applications (Foreign Keys)	4	Lecture, multimedia, demonstration	Slides
Week 12	Search and filters functionality in Django	4	Lecture, multimedia, demonstration	Slides
Week 13	Advanced features in Django: context processors, APIs (Optional)	4	Lecture, multimedia, demonstration	Slides
Week 14	Final Project Submission and Demonstration	6	Presentation skills workshop, Q&A	—

17. Teaching-Learning Strategies: Demonstration, Hands-On Lab work, Group Project

18. Assessment Techniques of each topic of the course:

SL. No	**Topics / Content**	**Assessment Techniques**
1	**Language Basic**	Assignment, Viva
2	**OOP Basics**	Lab Test, Project

3	**Full stack web development**	Continuous Web Project Monitoring
4.	**Technical documentation and presentation**	Assignment and presentation.

Part C – Assessment and Evaluation

19. Assessment Strategy

- **Assignment** – Submit all tasks within the deadline; demonstrate accuracy, completeness, and proper formatting.
- **Viva** – Answer coding questions clearly; explain logic, syntax, and problem-solving approach.
- **Lab Test** – Complete the given problem within the allotted time; ensure correctness, efficiency, and functionality.
- **Project** – Identify and design a CEP problem; develop an effective solution collaboratively, showing creativity, functionality, and code quality.
- **Documentation** – Prepare a structured technical report detailing the problem, solution approach, system architecture, and implementation details.
- **Presentation** – Deliver a clear, engaging, and well-organized presentation; demonstrate the solution and respond confidently to queries.

Weighting COs with Assessment methods:

Assessment Type	% Weight	CLO1	CLO2	CLO3	CLO4	CLO5	CLO6
Project Design and Implementation(Continious)	35.00%			10	15	10	
Project Report	15.00%		5	5			5
Project Presentation	10.00%						10
Lab Performance	20.00%	10	10				
Lab Test	20.00%	5		5	10		
Total	100.00%	15	15	15	25	10	20

20. Evaluation Policy

Grades will be calculated as per the university grading structure and individual students will be evaluated based on the following criteria with respective weights.

1.	Lab Performance	20%
2.	Lab test	20%
3.	Project	60%

UAP Grading Policy

Numeric Grade	Letter Grade	Grade Point
80% and above	A+	4.00
75% to less than 80%	A	3.75
70% to less than 75%	A-	3.50
65% to less than 70%	B+	3.25
60% to less than 65%	B	3.00
55% to less than 60%	B-	2.75
50% to less than 55%	C+	2.50
45% to less than 50%	C	2.25
40% to less than 45%	D	2.00
Less than 40%	F	0.00

Part D – Learning Resources

21. Text Book

1. ***Django for Beginners*** – Will Vincent (best for starting and building real projects).

2. ***Django for Professionals*** – Will Vincent (production-ready apps with Docker, PostgreSQL, testing).

3. ***Two Scoops of Django*** – Daniel & Audrey Roy Greenfeld (industry best practices).