



Course Syllabus

Course Code	Course Title	ECTS Credits
COMP-201	Systems Analysis and Design	6
Prerequisites	Department	Semester
Sophomore Standing	Computer Science	Fall/Spring
Type of Course	Field	Language of Instruction
Required for Computer Science Elective for Data Science	Computer Science	English
Level of Course	Lecturer(s)	Year of Study
1 st Cycle	Dr Vasso Stylianou	2 nd
Mode of Delivery	Work Placement	Corequisites
Face-to-face	N/A	None

Course Objectives:

1. Identify organisational problems that can be addressed by information systems.
2. Evaluate project feasibility from economic, technical, operational, legal, security, and environmental viewpoints.
3. Apply structured, object-oriented, and Agile methodologies (SDLC, SSADM, Scrum, Kanban).
4. Model processes and architecture with DFDs, BPMN 2.0, UML, C4 Level 1–2, and Domain-Driven Design (bounded contexts).
5. Elicit and document requirements through a mix of stakeholder interviews, questionnaires, observation, document analysis, facilitated workshops, user-story mapping, Behaviour-Driven Development (BDD) acceptance criteria, and requirements-traceability matrices.
6. Produce conceptual data models (relational, NoSQL, graph) and reason about storage choices.
7. Incorporate security, privacy (GDPR), and ethical considerations through threat modelling and impact assessments.
8. Apply fundamental project-management techniques—scope definition, Work Breakdown Structure (WBS), Scrum release planning, Gantt scheduling, PERT critical-path analysis, burndown charts, and risk registers—using basic version-control and issue-tracking tools.
9. Plan and monitor projects with basic version-control and issue-tracking tools; explain the purpose of Continuous Integration/Delivery (CI/CD).
10. Prepare user-training plans, conversion strategies, evaluation metrics, and reflect on environmental impact via green-software guidelines.

Learning Outcomes:

On completion of the course, students are expected to be able to

1. Analyse an organisational setting and articulate the problems or opportunities that justify an information-system initiative.
2. Conduct and document a feasibility study that integrates economic, technical, operational, legal, security, and environmental criteria, culminating in a defensible business case.
3. Compare structured, object-oriented, and Agile life-cycle models (SDLC, SSADM, Scrum, Kanban) and justify the selection of a methodology for a specific project context.
4. Model business processes, data flows, system structure, and high-level architecture by producing coherent sets of DFDs, BPMN 2.0 diagrams, UML artefacts, C4 Level 1–2 views, and bounded-context maps.
5. Elicit, document, and validate system requirements by applying interviews, questionnaires, observation, document analysis, facilitated workshops, user-story mapping, BDD acceptance criteria, and a requirements-traceability matrix.
6. Create conceptual data models for relational, NoSQL, and graph-based storage options and defend the chosen data-management approach in writing.
7. Perform threat modelling and GDPR-aligned privacy impact analysis, and explain the ethical considerations associated with the proposed system.
8. Create and maintain core project-management artefacts (project charter, WBS, Scrum release plan, Gantt chart, PERT network diagram with critical-path analysis, burndown chart, and risk register) and use them to monitor progress and communicate status to stakeholders.
9. Plan and monitor project work using basic version-control and issue-tracking tools, and accurately explain the goals and benefits of Continuous Integration and Continuous Delivery (CI/CD).
10. Develop a user-training programme, a data-conversion or cut-over strategy, and post-deployment evaluation metrics, while outlining actions that support environmentally sustainable software operation.

Course Content:

1. Foundations & Opportunity Identification: Information systems in organisations; systems thinking. Identifying IT-enabled opportunities; drafting a project charter.
2. Project Initiation & Feasibility Analysis: Feasibility dimensions: economic, technical, operational, legal, environmental, security. Project-planning artefacts: scope statement, Work Breakdown Structure (WBS), PERT network & critical-path analysis, Gantt chart creation and interpretation, initial risk register.
3. Development Life-Cycles & Agile Overview: Plan-driven models: SDLC, SSADM. Agile values and frameworks (Scrum, Kanban); backlog and sprint planning.
4. Requirements Elicitation: Stakeholder analysis. Interviews, questionnaires, observation, document analysis. Facilitated workshops, story-mapping. User stories, BDD acceptance criteria; requirements-traceability matrix.
5. Process & Data-Flow Modelling: Data-flow diagrams (DFDs). BPMN 2.0 processes; data dictionaries.
6. Object & Domain Modelling: UML use-case, class, and sequence diagrams. Domain-Driven Design: bounded-context discovery, ubiquitous language.
7. Architecture Modelling: C4 Level 1–2 context and container views. Conceptual cloud patterns; microservice vs. monolith trade-offs.

8. Conceptual Data Design & Basic Security Controls: Relational, NoSQL, and graph data models. Core security/privacy attributes; introduction to GDPR constraints.
9. Security Engineering, Risk & Testing Concepts. STRIDE threat modelling; OWASP Top-10. Test planning and levels (unit, integration, system); CI/CD purpose and vocabulary.
10. Human-Computer Interaction & UX: Usability heuristics; accessibility guidelines. Rapid prototyping with Figma/Balsamiq; UX evaluation techniques.
11. Implementation Strategies & Course Wrap-Up: Packaged vs. custom solutions; outsourcing; conversion and training plans. Success metrics, post-implementation evaluation, green-software considerations; comprehensive review.

Learning Activities and Teaching Methods:

- Interactive lectures with short preparatory videos (flipped segments)
- Hands-on workshops (story-mapping, BPMN, DDD, threat modelling, traceability)
- CASE and modelling-tool demonstrations (draw.io, PlantUML)
- Sprint-based group project with in-class studio sessions for mentoring
- Guest talks from industry practitioners (Agile practice, UX, cloud concepts)

Assessment Methods:

Group Project, Final Exam, Continuous Assessment (quizzes, peer evaluation, learning journal)

Required Textbooks / Readings:

Title	Author(s)	Publisher	Year	ISBN
Systems Analysis and Design, 8 th ed.	Dennis A, Wixom, B., & Roth, R.	Wiley	2021	9781119803782
Systems Analysis and Design, 11 th ed.	Kendall, K. & Kendall, J.	Pearson	2023	9780137947850

Recommended Textbooks / Readings:

Title	Author(s)	Publisher	Year	ISBN
Systems Analysis and Design in a Changing World, 8 th ed.	Satzinger, J., Jackson, R., & Burd, S.	Cengage	2020	
Domain-Driven Design Distilled	Evans, E. & Vernon, V.	Addison-Wesley	2016	
Accelerate	Forsgren, N., Humble, J., & Kim, G.	IT Revolution	2018	

Designing Data-Intensive Applications	Kleppmann, M.	O'Reilly	2017	
---------------------------------------	---------------	----------	------	--