COURSE OUTLINE: CSD124 - SYS. ANALYSIS/DESIGN

Prepared: D. Kachur
Approved: Martha Irwin, Dean, Business and Information Technology

Course Code: Title	CSD124: SYSTEMS ANALYSIS AND DESIGN
Program Number: Name	2095: COMPUTER PROGRAMMING
Department:	COMPUTER STUDIES
Academic Year:	2024-2025
Course Description:	This course provides learners with a structured, methodical approach to designing software systems, hardware networks and web-based cloud solutions. Analysis of existing industry projects will be conducted, identifying their successes and failures. As part of their project work, the learner will incorporate industry best practices while avoiding pitfalls. A number of diagramming methods and types will be used in the design and planning stages, including but not limited to uml object and class modeling, entity relationship modeling, sequence diagrams, business and process flow diagrams.
Total Credits:	3
Hours/Week:	3
Total Hours:	42
Prerequisites:	There are no pre-requisites for this course.
Corequisites:	There are no co-requisites for this course.
Substitutes:	CSD202
Vocational Learning Outcomes (VLO's) addressed in this course: **Please refer to program web page for a complete listing of program outcomes where applicable.**	**2095 - COMPUTER PROGRAMMING** VLO 1 — Identify, analyze, develop, implement, verify and document the requirements for a computing environment. VLO 5 — Communicate and collaborate with team members and stakeholders to ensure effective working relationships. VLO 7 — Apply project management principles and tools when working on projects within a computing environment. VLO 8 — Adhere to ethical, legal, and regulatory requirements and/or principles in the development and management of computing solutions and systems. VLO 9 — Support the analysis and definition of software system specifications based on functional and non-functional requirements. VLO 11 — Apply one or more programming paradigms such as, object-oriented, structured or functional programming, and design principles, as well as documented requirements, to the software development process.
Essential Employability Skills (EES) addressed in this course:	EES 1 — Communicate clearly, concisely and correctly in the written, spoken, and visual form that fulfills the purpose and meets the needs of the audience. EES 2 — Respond to written, spoken, or visual messages in a manner that ensures effective communication.

	EES 4 Apply a systematic approach to solve problems.
	EES 5 Use a variety of thinking skills to anticipate and solve problems.
	EES 6 Locate, select, organize, and document information using appropriate technology and information systems.
	EES 7 Analyze, evaluate, and apply relevant information from a variety of sources.
	EES 9 Interact with others in groups or teams that contribute to effective working relationships and the achievement of goals.
	EES 11 Take responsibility for ones own actions, decisions, and consequences.
Course Evaluation:	Passing Grade: 50%, D A minimum program GPA of 2.0 or higher where program specific standards exist is required for graduation.
Other Course Evaluation & Assessment Requirements:	A+ = 90-100% A = 80-89% B = 70-79% C = 60-69% D = 50-59% F < 50% Students are expected to be present to write all tests in class, unless otherwise specified. If a student is unable to write a test due to illness or a legitimate emergency, that student must contact the professor prior to class and provide reasoning. Should the student fail to contact the professor, the student shall receive a grade of zero on the test. If a student is not present 10 minutes after the test begins, the student will be considered absent and will not be given the privilege of writing the test. Students exhibiting academic dishonesty during a test will receive an automatic zero. Please refer to the College Academic Dishonesty Policy for further information. In order to qualify to write a missed test, the student shall have: a.) attended at least 75% of the classes to-date. b.) provide the professor an acceptable explanation for his/her absence. c.) be granted permission by the professor. NOTE: The missed test that has met the above criteria will be an end-of-semester test. Labs / assignments are due on the due date indicated by the professor. Notice by the professor will be written on the labs / assignments and verbally announced in advance, during class. Labs and assignments that are deemed late will have a 10% reduction per academic day to a maximum of 5 academic days at 50% (excluding weekends and holidays). Example: 1 day late - 10% reduction, 2 days late, 20%, up to 50%. After 5 academic days, no late assignments and labs will be accepted. If you are going to miss a lab / assignment deadline due to circumstances beyond your control and seek an extension of time beyond the due date, you must contact your professor in advance of the deadline with a legitimate reason that is acceptable. It is the responsibility of the student who has missed a class to contact the professor immediately to obtain the lab / assignment. Students are responsible for doing their own work. Labs / assignments that are handed in and are deemed identical or near identical in content may constitute academic dishonesty and result in a zero grade.

Students are expected to be present to write in-classroom quizzes. There are no make-up options for missed in-class quizzes.

Students have the right to learn in an environment that is distraction-free, therefore, everyone is expected to arrive on-time in class. Should lectures become distracted due to students walking in late, the professor may deny entry until the 1st break period, which can be up to 50 minutes after class starts or until that component of the lecture is complete.

Course Outcomes and Learning Objectives:

Course Outcome 1	Learning Objectives for Course Outcome 1
1. Discuss the elements of Systems Analysis and Design	1.1 Review existing industry IT projects for their successes and failures 1.2 Identify key components that were incorporated in successful projects 1.3 Prepare documentation that includes common issues found in failed systems 1.4 Diagram examples of software engineering, networking and cloud-based systems 1.5 Using an organization chart, diagram and explain the roles of employees in the IT Department and their relation to systems design 1.6 Explain and contrast B2C and B2B 1.7 Contrast waterfall vs agile styles of the systems analysis and design process

Course Outcome 2	Learning Objectives for Course Outcome 2
2. Analyze a business case	2.1 Explain the concept of a business case and how a business case affects an IT project 2.2 Describe the strategic planning process and why it is important to the IT team 2.3 Explain the purpose of a mission statement 2.4 Conduct a SWOT analysis and describe the four factors involved 2.5 Explain how the SDLC serves as a framework for systems development 2.6 List reasons for systems projects and factors that affect such projects 2.7 Describe systems requests and the role of the systems review committee 2.8 Define operational, technical, economic, and schedule feasibility 2.9 Describe the steps and the end product of a preliminary investigation

Course Outcome 3	Learning Objectives for Course Outcome 3
3. Describe the various tools and techniques that relate to Managing the Systems Project	3.1 Explain project planning, scheduling, monitoring, and reporting 3.2 Draw a project triangle that shows the relationship among project cost, scope, and time 3.3 Describe work breakdown structures, task patterns, and critical path analysis

SAULT COLLEGE | 443 NORTHERN AVENUE | SAULT STE. MARIE, ON P6B 4J3, CANADA | 705-759-2554

3.4 Explain techniques for estimating task completion times and costs 3.5 Describe various scheduling tools, including Gantt charts and PERT/CPM charts 3.6 Analyze task dependencies, durations, start dates, and end dates 3.7 Describe project management software and how it can assist you 3.8 Discuss the importance of managing project risks 3.9 Explain why projects sometimes fail	

Course Outcome 4	Learning Objectives for Course Outcome 4
4. Describe the Systems Analysis Phase and Requirements Modeling	4.1 Describe systems analysis phase activities 4.2 Explain joint application development (JAD), rapid application development (RAD), and agile methods 4.3 Use a functional decomposition diagram (FDD) to model business functions and processes 4.4 Describe the Unified Modeling Language (UML) and examples of UML diagrams 4.5 List and describe system requirements, including outputs, inputs, processes, performance, and controls 4.6 Explain the concept of scalability 4.7 Use fact-finding techniques, including interviews, documentation review, observation, questionnaires, sampling, and research 4.8 Define total cost of ownership (TCO) 4.9 Develop effective documentation methods to use during systems development

Course Outcome 5	Learning Objectives for Course Outcome 5
5. Describe the Data and Process Modeling Phase	5.1 Describe data and process modeling concepts and tools, including data flow diagrams, a data dictionary, and process descriptions 5.2 Describe the symbols used in data flow diagrams and explain the rules for their use 5.3 Draw data flow diagrams in a sequence, from general to specific 5.4 Explain how to level and balance a set of data flow diagrams 5.5 Describe how a data dictionary is used and what it contains 5.6 Use process description tools, including structured English, decision tables, and decision trees 5.7 Describe the relationship between logical and physical models

Course Outcome 6	Learning Objectives for Course Outcome 6
6. Explore fundamental concepts of Object Modeling	6.1 Explain how object-oriented analysis can be used to describe an information system 6.2 Define object modeling terms and concepts, including objects, attributes, methods, messages, classes, and instances 6.3 Explain relationships among objects and the concept of inheritance

	6.4 Draw an object relationship diagram 6.5 Describe Unified Modeling Language (UML) tools and techniques, including use cases, use case diagrams, class diagrams, sequence diagrams, state transition diagrams, and activity diagrams 6.6 Explain the advantages of using CASE tools in developing the object model 6.7 Explain how to organize an object model
Course Outcome 7	**Learning Objectives for Course Outcome 7**
7. Explore Software Development Strategies	7.1 Describe the concept of Software as a Service 7.2 Define cloud computing 7.3 Explain software acquisition alternatives, including traditional and Web-based software development strategies 7.4 Describe software outsourcing options, including offshore outsourcing and the role of service providers 7.5 Explain advantages and disadvantages of in-house software development 7.6 Discuss cost-benefit analysis and financial analysis tools 7.7 Describe a request for proposal (RFP) and a request for quotation (RFQ) 7.8 Describe the system requirements document 7.9 Explain the transition from systems analysis to systems design
Course Outcome 8	**Learning Objectives for Course Outcome 8**
8. Design an effective user interface	8.1 Explain the concept of user interface design and human computer interaction, including basic principles of user-centered design 8.2 Explain how experienced interface designers perform their tasks 8.3 Describe rules for successful interface design 8.4 Discuss input and output technology issues 8.5 Design effective source documents and forms 8.6 Explain printed output guidelines 8.7 Describe output and input controls and security 8.8 Explain modular design and prototyping techniques

Evaluation Process and Grading System:

Evaluation Type	Evaluation Weight
Lab Assignments	40%
Test #1	30%
Test #2	30%

Date: June 16, 2024

Addendum: Please refer to the course outline addendum on the Learning Management System for further information.