# Systems/Software Analysis and Projects        2023-2024 Catal

[ARCHIVED CATALOG]

## SDEV 265 - Systems/Software Analysis and Projects

**PREREQUISITES:** SDEV 200 - Software Development using Jav OR SDEV 210 - Software Development using Visual Basic in the .NET Framework OR SDEV 220 - Software Development Using Python OR SDEV 230 - Software Development using C++ OR SDEV 240 - Software Development Using C# OR CSCI 102 - Computer Science II OR CSCI 201 - Computer Science II.
PROGRAM: Software Development
**CREDIT HOURS MIN:** 3
LECTURE HOURS MIN: 2
LAB HOURS MIN: 2
DATE OF LAST REVISION: Fall, 2017

Students will explore and learn about the fundamental parts of the systems development process, from analysis through design, implementation, and maintenance. Students will then develop a software product for an actual or simulated client who intends to use the product in a business environment. Students will form project teams to work through the full software development life cycle from understanding the business requirements to delivering a functioning product. Students will explore software development issues, select the appropriate development life cycle, estimate effort and development times, understand risk management, and write functional software. Students will make a series of presentations and reports of the project progress to the client and instructor. Students will learn methodologies pertinent to the assessment, design, and implementation of information systems.

MAJOR COURSE LEARNING OBJECTIVES: Upon successful completion of this course the student will be expected to:

1. Define the Systems Development Life Cycle (SDLC) and discuss the role of the systems analyst in the SDLC.
2. Identify key methodologies used in systems and software design, including waterfall, agile, etc.
3. Compare and contrast different methods for conversion from an existing system to a new system.
4. Conduct a feasibility study identifying the major properties of system planning as they relate to cost analysis.
5. Utilize standard tools such as DFDs, structure charts, UML, flowcharts, and data dictionaries in the design and development of systems and software.
6. Identify and contrast different project management tools utilized in systems and software design.
7. Identify the criteria used in the evaluation and selection of computer equipment and software.
8. Explain the significance of the various forms of documentation and documents as they pertain to the design of systems and software.
9. Perform a software risk evaluation, identifying and mitigating risks.
10. Define the human and political considerations in steering a project.
11. Demonstrate teamwork and communication associated with the design and development of systems and software solutions.
12. Implement the different stages of team formation, taking into account leadership styles, decision-making approaches, and conflict resolution techniques.
13. Work through the design phases of a software development project through requirements analysis and specification, software and UI design, developing documentation, and planning implementation, testing, and maintenance.
14. Implement a software development project, from start to finish, through requirements analysis and specification, through design, implementation, testing and documentation.

COURSE CONTENT: Topical areas of study include -

- Feasibility study

- Software life cycle
- Documentation
- User interface design
- Teamwork and communication
- Software development models
- Software and project metrics
- Estimating and tools
- Systems Analysis
- Configuration management
- Risk analysis
- Testing techniques and issues
- Project management and tracking
- Software requirements specification
- Software design description
- Software testing plan
- IEEE standards
- Systems Development Life Cycle (SDLC)

[Course Addendum - Syllabus (Click to expand)](#)