Software Evolution (Maintenance)	Code & No:	CS 434
	Credits:	3 (3,0,1)
	Pre-requisite:	CS 360
	Co-requisite:	None
	Level:	9 or 10

Course Description:

This course introduces the concept of software as an evolving and complex entity. Deliver knowledge about technical and business issues connected to legacy systems. Topics include:

- Relationships between evolving entities
- Models of software evolution
- Working with Legacy Systems
- Program Comprehension
- High level Reverse Engineering
- System and Process Re-Engineering
- Program Migration (technical and business)
- Refactoring
- Impact Analysis
- Introduction to Data Reverse Engineering

Course Aims:

1) Introducing basic concepts of maintenance
2) How the concept of system evolution fits into maintenance
3) Present different technical and managerial problems of maintenance
4) Addresses the formal types of maintenance
5) Discusses standard maintenance processes.

Student Outcomes (SOs):

☐(a) An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline

☐(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

☒(c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

☐(d) An ability to function effectively on teams to accomplish a common goal

☐(e) An understanding of professional, ethical, legal, security and social issues and responsibilities

☐(f) An ability to communicate effectively with a range of audiences

☐(g) An ability to analyze the local and global impact of computing on individuals, organizations, and society

☐(h) Recognition of the need for and an ability to engage in continuing professional development

☒(i) An ability to use current techniques, skills, and tools necessary for computing practice.

☒(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices. [CS]

☐(k) An ability to apply design and development principles in the construction of software systems of varying complexity. [CS]

☐(j) An ability to use and apply current technical concepts and practices in the core information technologies of human computer interaction, information management, programming, networking, and web systems and technologies. [IT]

☐(k) An ability to identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems. [IT]

☐(l) An ability to effectively integrate IT-based solutions into the user environment. [IT]

☐(m) An understanding of best practices and standards and their application. [IT]

☐(n) An ability to assist in the creation of an effective project plan. [IT]

Course Learning Outcomes (CLOs):

1. Learn the maintenance process
2. Understand system evolution
3. Understand configuration management
4. Apply re-engineering and refactoring
5. Apply concepts of reuse

SOs and CLOs Mapping:

CLO/SO	a	b	c	d	e	f	g	h	i	j	k	l	m	n
CLO1			√											
CLO2			√											
CLO3										√				
CLO4									√					

CLO5									√					

No.	Topics	Weeks	Teaching hours
1	Evolution and Maintenance, Models of software evolution	2	6
2	Taxonomy of Software Maintenance and Evolution	2	6
3	Evolution and Maintenance Models	2	6
4	Re-Engineering, Legacy information systems	2	6
5	Impact Analysis, Refactoring	2	6
6	Program Comprehension, Reuse and Domain Engineering	4	12
Total		**14**	**42**

Textbook:

- Software Evolution and Maintenance: A Practitioner’s approach, Priyadarshi Tripathy, Kshirasagar Naik,John Wiley & Sons, 2014

Essential references:

- Software Evolution, Tom Mens& Serge Demeyer. Springer, 2008
- Experiences in software evolution and reuse : twelve real world projects, by Hallsteinsen, S. and Paci, M, 1997, Berlin ; New York: Springer.
- IEEE Standard for Software Maintenance, IEEE Std 1219-1998,
- Software engineering - Software life cycle processes – Maintenance. ISO/IEC FDIS 14764:2005(E),
- Advances in software maintenance management: technologies and solutions Hershey, PA: Idea Group Pub.