COURSE CODE	COURSE TITLE	L	T	P	C
UCS1617	MINI PROJECT	0	0	3	1.5

OBJECTIVES

- To capture the requirements specification for an intended software system
- To draw the UML diagrams for the given specification
- To map the design properly to code
- To test the software system thoroughly for all scenarios
- To improve the design by applying appropriate design patterns.

SUGGESTIVE LIST OF EXPERIMENTS

Draw standard UML diagrams using an UML modeling tool for a given case study and map design to code and implement a 3 layered architecture. Test the developed code and identify whether the SRS is satisfied.This could be done in an incremental way and the next increment should undergo the same process of design, coding and testing to satisfy the SRS. Any one of the suggested domains could be chosen for the mini project.The student could also choose his own interested domain base on the approval from the concerned faculty.

1. Develop a problem statement.
2. Document theSoftwareRequirements Specification (SRS) for theabove problem statement.
3. Identify use cases and develop the Use Case model.
4. Identify theconceptual classesanddevelop a Domain Modeland derivea Class Diagram from the already designed Domain Model.
5. Usingtheidentified scenarios, findtheinteraction between objectsand represent themusingUML Sequence and Collaboration Diagrams
6. Draw relevant State Chart and Activity Diagrams for the same problem.
7. Implement the problem with the detailed design structure (1-6) and satisfy the SRS.
8. Test the developed software system.Identify the areas in which refinement is needed.
9. Refine the design of the modules using appropriate design patterns and implement the same.
10. Test the software again after refinement to check, whether it satisfies theSRS, and carry over the refinement if needed.

SUGGESTED DOMAINS FOR MINI PROJECT

1. Passport automation system.
2. Book bank
3. Exam Registration
4. Stock maintenance system.
5. Online course reservation system
6. E-ticketing
7. Software personnel management system
8. Credit card processing
9. e-book management system
10. Recruitment system
11. Foreign trading system
12. Conference Management System
13. BPO Management System
14. Library Management System
15. Student Information System

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the requirements for a given problem specification. (K2)
- Designand draw UML diagrams. (K3)
- Map design to code and implement usingUML diagrams. (K4)
- Designand generate test cases. (K3)
- DesignUML diagrams by applying appropriate design patterns (K3).

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

• Standalone Desktops - 38 Nos

Software:

• ArgoUML that supports UML 1.4 and higher

• Selenium, JUnit or Apache JMeter