MANGALORE UNIVERSITY

DEPARTMENT OF M.Sc. COMPUTER SCIENCE

MASTER OF COMPUTER APPLICATIONS (MCA) PROGRAMME

MCAS406:SOFTWARE ARCHITECTURE		
Hours/Week: 4 **Credits : 4**		**I.A. Marks: 30** **Exams. Marks: 70**

Course Outcomes:

CO1: Students will cite knowledge of various approaches to document a software system (Remembering)
CO2: Students will be able to describe functional and non-functional requirements (Understanding)
CO3: Students will be able to use proper architecture for software (Applying)
CO4: Students will be able to categorize different components used in the software system (Analyzing)
CO5: Students will be able to choose from different architectural styles (Evaluating)
CO6: Students will be able to improve quality of software by selecting proper architecture (Creating)

	UNIT-I	12 Hours

Introduction: The Architecture Business Cycle: Where do architectures come from? Software processes and the architecture business cycle; What makes a "good" architecture? What software architecture is and what it is not; Other points of view; Architectural patterns, Reference models and reference architectures; Importance of software architecture; Architectural structures and views. **Architectural Styles And Case Studies:** Architectural styles; Pipes and filters; Data abstraction and object-oriented organization; Event-based, implicit invocation; Layered systems; Repositories; Interpreters; Process control; Other familiar architectures; Heterogeneous Architectures. Case Studies: Keyword in Context; Instrumentation software; Mobile robotics.

	UNIT-II	12 Hours

Quality: Functionality and architecture; Architecture and quality attributes; System quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics.

	UNIT-III	12 Hours
Architectural Patterns: Introduction, Distributed Systems: Broker; Interactive Systems: MVC, Presentation-Abstraction-Control. Adaptable Systems: Microkernel; Reflection. **Some Design Patterns:** Structural decomposition: Whole – Part; Organization of work: Master – Slave; Access Control: Proxy.		
	UNIT-IV	**12 Hours**
Designing And Documenting Software Architecture: Architecture in the life cycle; designing the architecture; Forming the team structure; Creating a skeletal system. Uses of architectural documentation; Views; choosing the relevant views; Documenting a view; Documentation across views.		

REFERENCE BOOKS

1. Len Bass, Paul Clements, Rick Kazman**, Software Architecture in Practice,** Pearson Education, 2003, 2nd Edition.
2. Frank Buschmann, ReginеMeunier, Hans Rohnert, Peter Sommerlad, Michael Stal**, Pattern-Oriented Software Architecture, A System of Patterns -Volume 1,** John Wiley and Sons, 2006.
3. **Mary Shaw and David Garlan**, Software Architecture-Perspectives on an Emerging Discipline, Prentice-Hall of India, 2007.
4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, **Design Patterns- Elements of Reusable Object-Oriented Software**, Addison- Wesley, 2003.