

# **IIT Mandi**

## **Proposal for a New Course**

<b>Course number</b>	:	CS592
<b>Course name</b>	:	Reactive Design Patterns
<b>Credits</b>	:	2-0-0-2
<b>Prerequisites</b>	:	Instructor's permission
<b>Intended for</b>	:	UG and PG
<b>Distribution</b>	:	Elective for BTech CSE, MS, PhD

### **1. Preamble**

Software Design Patterns describe recurring problems in software design to meet the functional requirements of a software systems and templates of solutions which can be adapted/adopted by the designer.

Non-functional requirements are generally addressed as add-ons and not factored at the design time. These non-functional requirements become much more complex to address after system design. Industries need system which are scalable, able to react to load variations and failures, maintainable, extensible etc. These aspects of system become more complex in internet scale distributed systems. Reactive design patterns address these issues at the design stage and not after it.

### **2. Course objectives**

The main objective the course is to enable students to learn basic concepts associated with development of internet scale distributed applications and appreciate the issues related with non-functional requirements and their possible solutions . They will learn about:

- Limitations of Synchronous Communication & Difficulties due to concurrency
- Asynchronous Communication for distributed applications & scalability
- Enabling encapsulation and isolation with explicit asynchronous message passing
- Location transparency
- Horizontal scalability
- Principled failure handling by hierarchical modules
- To manage sharing of state of mutable objects

### **3. Course modules**

- Review of Multicore processor, cache & memory hierarchy and cache coherency [2 lecture hours]
- Review of Process & thread [1 lecture hour]
- Thread Pool & their scheduling & User Mode scheduling [3 lecture hours]
- Event and message communication & their context & handling [1 lecture hour]
- Asynchronous Communication, Promise & Future [2 lecture hour]
- Method execution as an independent thread & Green Threads [2 lecture hour]
- Message Flow Patterns: Request-Response & Ask [4 lecture hours]
- Flow Control Patterns: Pull, Managed Queue, Drop [3 lecture hours]
- Replication Patterns: Active-Reactive, Active-Active [4 lecture hours]
- Resource Management Patterns: Resource Pool & Complex Command Pattern [3 lecture hours]
- Mutability & Synchronization primitives [2 lecture hours]
- Software Transactional Memory [ 4 lecture hours]