### Autonomous Program Structure of
Final Year B. Tech. (Information Technology)
Academic Year: 2019-2020 Onwards

#### Final Year B. Tech. (IT)  Semester – 7

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Teaching Scheme Hours /Week</th>
<th>Examination Scheme</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lecture</td>
<td>Tutorial</td>
<td>Practical</td>
<td>In Semester</td>
</tr>
<tr>
<td>IT 4101</td>
<td>Software Architecture &amp; Design Patterns</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>IT 4102</td>
<td>Cloud Computing</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>HS 4101</td>
<td>Green Computing*</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>OE 4101</td>
<td>Open Elective –I</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>IT 4103</td>
<td>Software Architecture &amp; Design Patterns Laboratory</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>IT 4104</td>
<td>Project Phase-I</td>
<td>0</td>
<td>2</td>
<td>14</td>
<td>100</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>12</td>
<td>2</td>
<td>16</td>
<td>300</td>
</tr>
<tr>
<td><strong>Grand Total</strong></td>
<td></td>
<td>30</td>
<td></td>
<td></td>
<td>600</td>
</tr>
</tbody>
</table>

*Advanced Entrepreneurship Development**

**Prerequisite: Basic Course ED

OE 4101 Open Elective-I
1. Software Testing and Quality assurance
IT 4101 Software Architecture and Design Patterns

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Object Oriented Paradigms, Software Engineering

Course Objectives:
Familiarize students with
1. Concepts of software architecture
2. Different types of software architectural styles
3. Concepts and applications of design patterns.
4. Different types of design patterns

Course Outcomes:
Students should be able to
1. Analyze and suggest architecture design for an application
2. Apply design patterns to software design
3. Evaluate and select appropriate design pattern for a situation
4. Compare the performance of the software on inclusion of various design patterns.

Unit – I: Software Architecture (07)
Overview of software Architecture, What drives software architecture, Quality attributes, Architecture design, Architecture documentation

Unit – II: Architectural Patterns (07)
Client server multitier architectural pattern, Even driven architectural pattern, Service Oriented Architectures, Component based architecture

Unit – III: Role of design patterns in architecture design (07)
Introduction to architecture design, introduction to design patterns, Types of design patterns Abstract factory, builder, factory method, singleton design patterns

Unit – IV: Creational Design Patterns (07)
Abstract factory, builder, factory method, singleton design patterns with case study
Unit – V:  Structural Design Patterns

Adapter, bridge, composite, facade, decorator, chain of responsibility with case study

Unit – VI:  Behavioral Design Patterns

State, Observer, Strategy, template method with case study,

Text Books:

2. Elizabeth Freeman, Kathy Seirra, Head first design patterns O’Reilly Media ISBN 0596007124

Reference Books:

IT 4102 Cloud Computing

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Operating Systems and Computer Networks

Course Objectives:
Familiarize students with
1. Distributed Systems and its ecosystem.
2. Basics of virtualization and its importance.
3. In-depth analysis of cloud computing capabilities.
4. Overview of cloud programming and services.

Course Outcomes:
Students should be able to
1. Recognize need of cloud based solutions.
2. Justify the importance of distributed systems.
3. Determine effective techniques to program cloud systems.
4. Evaluate current challenges and trade-offs in cloud computing.

Unit – I Introduction to Distributed Systems (07)

Unit – II Computer Clusters for Scalable Parallel Computing (07)

Unit – III Virtual Machines and Virtualization of Clusters and Data Centers (07)
Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation

Unit – IV Cloud Platform Architecture over Virtualized Data Centers (07)
**Unit – V Cloud Programming and Software Environments**
Features of Cloud and Grid Platforms, Parallel and Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments,

**Unit – VI Grids, P2P, and the Future Internet**

**Text Books**

**Reference Books**
HS 4101 Green Computing

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Basic Sciences

Course Objectives:
Familiarize students with
1. Knowledge of green computing practices to minimize negative impacts on the environment.
3. Green Computing and how it can help improve environmental sustainability.

Course Outcomes:
Students should be able to
1. Relate to the socio cultural aspects of green computing.
2. Create awareness about green computing and promote green agenda in their working environments leading to green movement.
3. Apply green computing skills such as energy efficiency, IT assets disposal, carbon footprint estimation, reporting and development of green products.
4. Justify green initiatives while developing applications and services in enterprises.

Unit – I: Introduction to Green Computing (07)
Environmental Impacts of IT, Need of green computing, Green IT Standards, Enterprise Green IT Strategy, Hardware: Reuse, Recycle and Dispose, present scenario in industry, health issues relevance, Software: Energy-Saving Software Techniques

Unit – II: Software Development and Green Data Centers (07)
Sustainable Software, Software Sustainability Attributes, Software Sustainability Metrics, Sustainable Software Methodology, Data Centres and Associated Energy Challenges, Data Centre IT Infrastructure, Data Centre Facility Infrastructure: Implications for Energy Efficiency, IT Infrastructure Management, Green Data Centre Metrics

Unit – III: Green Data Storage and Networks (07)

Unit – IV: Enterprise Green IT Strategy (07)
Approaching Green IT Strategies, Business Drivers of Green IT Strategy, Business Dimensions for Green IT Transformation, Multilevel Sustainable Information,
Sustainability Hierarchy Models, Product Level Information, Individual Level Information, Functional Level Information, Organizational Level Information, Regional/City Level Information

**Unit – V: Green Computing Services and Roles**

Factors Driving the Development of Sustainable IT, Sustainable IT Services (SITS), Sustainable IT Roadmap, Organizational and Enterprise Greening, Information Systems in Greening Enterprises, Greening the Enterprise

**Unit – VI: Regulating Green Computing**

The Regulatory Environment and IT Manufacturers, Nonregulatory Government Initiatives, Industry Associations and Standards Bodies, Green Building Standards, Green Data Centres, Social Movements

**Text Books:**

**Reference Books:**
OE 4101 Software Testing and Quality Assurance

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Any programming language

Course Objectives:
Familiarize students with
1. Application of testing strategies in projects.
2. Test management strategies and tools for testing
3. Various quality assurance models

Course Outcomes:
Students should be able to
1. Analyze the project scenario and to select proper testing technique
2. Apply testing techniques to deliver a project
3. Choose quality assurance models for the project
4. Choose one of the latest testing tools suitable for the project

Unit – I: Software testing fundamentals (07)

Testing as a Process, Software testing principles, The tester’s role in a software development organization, Origins of defects, Defect classes, Testing fundamentals, the defect repository and test design, Defect examples, Developer /Tester support for developing a defect repository. Process model to represent Different phases, Lifecycle models

Unit – II: Levels of testing (07)

Need for levels of testing, Unit testing, Integration testing, System Testing - Usability and Accessibility Testing, Configuration Testing, Compatibility Testing, Stress testing, Regression testing, Alpha, Beta and Acceptance testing.

Unit – III: Testing techniques (07)

Using White Box Approach to Test design - Static Testing, Structural Testing, Unit Functional Testing, Challenges in White box testing, Using Black Box Approaches to Test Case Design, Random Testing, Requirements based testing, Decision tables, State-based testing, Cause-effect graphing, Error guessing, Compatibility testing.

Unit – IV: Fundamentals of software quality assurance (07)

SQA basics, Components of the Software Quality Assurance System, software quality in
business context, planning for software quality assurance, product quality and process quality, software process models, 7 quality control Tools and Modern Tools.

**Unit – V: Quality assurance models**


**Unit – VI: Software test automation and current industry trends**


**Text Books:**


**Reference Books:**

IT 4103 Software Architecture and Design Pattern Laboratory

Teaching Scheme:
Practical: 2 hours/week
Tutorial: NIL

Examination Scheme:
Oral: 50 marks
Credits: 1

Prerequisites: Web Engineering Technology, Programming skill development laboratory

Course Objectives:
Familiarize students with
1. One client side programming Technology
2. One server side programming Technology
3. Developing a multiuser application

Course Outcomes:
Students should be able to
1. Apply appropriate technology to design the client side of the application
2. Apply appropriate technology to design the server side part of the application
3. Design the persistent layer classes their connection to database
4. Deploy and run the complete application

List of Assignments
1. Identify a system having three or four user expectations. Prepare its use case model
2. For the same system, prepare its analysis class model.
   Implement it using java language
3. For the same system, refine the analysis model and prepare the design class model.
   Implement it using java. Include appropriate applicable design patterns while designing the system.
4. Add view classes to your model and run the code handling appropriate events.
5. Design persistent layer classes and connect the business logic to database.
6. Deploy the application on server and ensure that it runs for various clients.
   Comment on the Quality attributes addressed in the system.

Text Books
2. Deitel, Deitel and Nieto, Internet and World wide web how to program

Reference Books
IT 4104-PROJECT PHASE – I

Teaching Scheme:
Tutorial: 2 hrs/week
Practical: 14 hrs/week

Examination Scheme:
In semester: 100 marks
Oral: 50 marks
Credits: 9

Course Objectives:
Familiarize students with:

1. The practical implementation of theoretical knowledge gained till date.
2. Implementation of their ideas/real time industrial problem/current application of Computer Science or Information Technology.

Course Outcomes:
At the end of this course the student should be able to:

1. Formulate a statement for the problem in Computer Science or Information Technology domain.
2. Prepare prototype for the identified problem.
4. Work in team using ethical practices.

Following activities are expected to be completed in Project Phase-I:

1. Identification of Problem
2. Feasibility study
3. Formulation of Problem Statement
4. Abstract writing
5. Literature Survey
6. Project planning and maintaining log
7. High level System Design
9. Study of technology/platform
10. Technical Report writing
11. Start project based online course.

All this should be done with frequent meetings with internal and external guide.
The log has to be maintained.
Every project group has to give 2 Reviews in Semester-I
In Review-I, Point 1 to 4 should be completed. Demonstration and discussion with reviewers will be done.
In Review-II, Point 5 to 11 should be completed. Demonstration and discussion with reviewers will be done.