



# UNIVERSITY OF MYSORE DIRECTORATE OF OUTREACH AND ONLINE PROGRAMS

MASTER OF COMPUTER APPLICATIONS

SEMESTER – III

## **Hard Core Courses**

1. Advanced Software Engineering
2. Python Programming
3. Machine Learning

## **Softcore courses**

1. Digital Image Processing
2. Internet of Things

## **Advanced Software Engineering (3:1:0)**

### **Course Outcome:**

- Identify unique features of various software application domains and classify software applications.
- Choose and apply appropriate lifecycle model of software development.
- Identify user needs and formulate software specifications, analyze requirements by applying various modeling techniques, Translate the requirements model into the design model.
- Understand the importance of User-interface design principles in software development, the concepts of clean room software development.
- Classify CASE tools and their applicability in software development.
- Understand the principles of agile development and distinguish agile process model from other process models.
- Learn the emerging trends in software development.

## **Course Content:**

### **Unit-I**

**Introduction:** The role of software engineering in system design, software products, emergence of software engineering, notable changes in software development practices, the changing nature of software, the software engineering challenges, software processes, desired characteristics of software process, the software life cycle, software development process models, comparison of process models.

**Software Requirement Specification:** Requirement analysis, need for SRS, characteristics of SRS, organization of SRS document, techniques for representing complex logic, functional specification with Use Cases, formal system development techniques.

### **Unit-II**

**Software Design:** Introduction, cohesion and coupling, software design approaches, design principles, module level concepts, Function-oriented software design, Object-oriented software design concepts: Overview, UML, object-oriented design methodology, OOD metrics and goodness criteria, user interface design concepts.

**Coding and Testing:** Coding standards, guidelines, code walkthroughs, code inspections, software documentation, unit testing, black box testing, white box testing. Debugging, approaches and guidelines, program analysis tools, integration testing, system testing, general issues associated with testing.

### **Unit-III**

**Software Project Management:** Main objectives of SPM, responsibility of software project managers, project planning, structure of software project management document, project size estimation metrics, project estimation techniques, project scheduling and staffing, work break down structure, Gantt charts, PERT charts, organization and team structures, attributes of a good software engineer, risk management and configuration management, software maintenance process models, estimation of maintenance costs, CASE and its scope, CASE support in software life cycle, characteristics of CASE tools, architecture of CASE environment.

**Software Quality Management:** Software quality factors, quality metrics, software quality management system, software reliability metrics, software reliability specification, reliability growth modeling.

## **Unit-IV**

**Emerging Technologies:** Agile software development concepts, Security concepts, security risk management, design for security, system survivability. Service-oriented software engineering- services as reusable components, service engineering, software development with services. Aspect- oriented software development- The separation of concerns, aspects, join points and point cuts, software engineering with aspects.

## **References Books:**

1. Software Engineering, Ian Sommerville, 8<sup>th</sup> Edition, Pearson Education Ltd.,
2. Software Engineering – A practitioners approach, Roger. S. Pressman, Tata-McGraw Hill 6th Edition.
3. Fundamentals of software engineering, Rajib Mall, Phi learning Pvt. Ltd, 3<sup>rd</sup> edition.
4. Pankaj Jalote – An Integrated Approach to Software Engineering, Third Edition.
5. Ghezzi, Jazayeri, Mandrioli – Fundamentals of Software Engineering, PHI.

## **Python Programming (2:0:2)**

### **Course Outcome**

- Acquire programming skills in core Python. Acquire Object Oriented Skills in Python.
- Develop the skill of designing Graphical user Interfaces in Python.
- Develop an ability to write database applications in Python.

### **Course Content**

#### **Unit-I**

Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

## **Unit-II**

Types, Operators and Expressions: Types – Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while break, continue, pass.

Data Structures Lists: Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences, Comprehensions.

## **Unit-III**

Functions: Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function- Global and Local Variables. Modules: Creating modules, import statement, from. Import statement, name spacing, Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages

## **Unit-IV**

Object-Oriented Programming OOP in Python: Classes, ‘self-variable’, Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding, Error, and Exceptions: Difference between an error and Exception, Handling Exception, try except for block, Raising Exceptions, User Defined Exceptions.

Brief Tour of the Standard Library: Operating System Interface – String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics Testing: Why testing is required ?, Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

## **Reference Books**

1. Charles Dierbach, "Introduction to Computer Science Using Python", 1st Edition, Wiley India Pvt Ltd.
2. Mark Lutz, "Programming Python", 4th Edition, O'Reilly Media, 2011.
3. Wesley J Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education India, 2015.

4. Roberto Tamassia, Michael H Goldwasser, Michael T Goodrich, “Data Structures and Algorithms in Python”, 1<sup>st</sup> Edition, Wiley India Pvt Ltd, 2016.
5. ReemaThareja, “Python Programming using problem solving approach”, Oxford university press, 2017.
6. Python Programming: A Modern Approach, VamsiKurama, Pearson
7. Learning Python, Mark Lutz, Orielly
8. Introduction to Python, Kenneth A. Lambert, Cengage.

## **Machine Learning (3:0:1)**

### **Course Outcome**

- Gain knowledge about basic concepts of Machine Learning.
- Identify machine learning techniques suitable for a given problem.
- Solve the problems using various machine learning techniques
- Design and implement machine learning solutions to classification, regression, and clustering problems; and be able to evaluate and interpret the results of the algorithms.

### **Course Content**

#### **Unit-I**

Introduction, Machine learning definition, importance of machine learning, machine learning framework, types of machine learning, relation to other fields, examples of machine learning applications, designing a learning system, issues in machine learning.

#### **Unit-II**

Introduction to Supervised Learning, Decision tree based classifier, Bayesian theory based classifier, Neural network based classifier, Nearest neighbour classifier, Support vector classifier, performance evaluation.

#### **Unit-III**

Introduction to Unsupervised Learning, Clustering methods, Criteria functions for clustering, Similarity measures, Component analysis, Low dimensional analysis and multidimensional scaling.

## **Unit-IV**

Additional topics, Reinforcement learning, Genetic algorithms, Analytical learning, Ensemble of classifiers, Design and analysis of machine learning experiments.

### **Reference Books**

1. Machine Learning: a Probabilistic Perspective by Kevin Patrick Murphy, MIT Press, March 2014.
2. Introduction to Machine Learning by Alex Smola and S.V.N. Vishwanathan, Cambridge University Press.
3. Understanding Machine Learning: From Theory to Algorithms by Shai Shalev-Shwartz and Shai Ben-David  
Published 2014 by Cambridge University Press.
4. Published 2014 by Cambridge University Press.

### **Softcore Courses**

#### **Digital Image Processing (2:0:2)**

##### **Course Outcome:**

- Develop and implement algorithms for digital image processing.
- Apply image processing algorithms for practical object recognition applications.

##### **Course Content**

###### **Unit-I**

Introduction to digital image processing, Stages, Application areas, components, electromagnetic spectrum, image sensing and acquisition, image sampling and quantization, relationships between pixels, Enhancement in spatial domain: Intensity transformation functions.

## **Unit–II**

Spatial filtering, Frequency domain enhancement: Discrete Fourier transform (DFT) properties of the 2D discrete Fourier transform, filtering in the frequency domain, Introduction to Color image processing.

## **Unit–III**

Segmentation – Intensity based – point, line and edge. Region based – Boundaries, region growing, Thresholding, splitting and merging, segmentation by morphological watersheds, the use of motion in segmentation.

## **Unit–IV**

Morphological operations: Preliminaries, opening and closing, the hit-or-miss transformation, some basic morphological algorithms, gray-scale images. Image representation.

Some applications: Document image processing, Biometrics, robot vision, medical applications.

## **Reference Books**

1. R. C. Gonzalez, R. E. Woods, Digital Image Processing, 3-rd ed. Prentice Hall, Pearson publication.
2. Anil K Jain, Digital Image Processing, PHI Publication
3. Milan Sonka, Image Processing, Analysis, and Machine Vision, 3rd Edition, CL Engineering(2013)

## **Internet of Things (2:0:2)**

### **Course Outcome**

- Understand the key technologies in internet of things, wireless sensor network architecture and its framework along with WSN applications,
- Understand the resource management and business models for the internet of things.



## **Course Content**

### **Unit-I**

Introduction to IoT: Definition and Characteristics, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies.

M2M and IoT: Introduction to M2M, Difference between IoT and M2M, SDN and NFV for IoT.

IoT Protocols: IEEE 802.15.4, BACNet Protocol, Modbus, KNX, Zigbee Architecture, 6LoWPAN, RPL

### **Unit-II**

Developing Internet of Things: IoT Platforms Design Methodology, Python packages of Interest for IoT, IoT Physical Devices and Endpoints.

IoT and Cloud: IoT Physical Servers and Cloud Offerings, IoT Tools: Chef, Puppet.

### **Unit-III**

Data Analytics for IoT: Big Data Platforms for the IoT, Hadoop Map Reduce for Batch Data Analysis, Apache Oozie Workflows for IoT Data Analysis, In-Memory Analytics using Apache Spark, Apache Storm for Real Time Data Analysis, Sustainability Data and Analytics in Cloud based M2M Systems, Fog Computing: A Platform for IoT and Analytics

### **Unit-IV**

Domain Specific IoTs: Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health and Lifestyle, Virtual Reality Internet Advertising, Intelligent Transportation Systems, Health Information System: Genomics Driven Wellness Tracking and Management System (Go-WELL).

## **Reference Books**

1. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-on Approach, 2015, 1st Edition, Universities Press.
2. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things – Key applications and Protocols, 2012, Wiley Publication.

3. Honbo Zhou, *The Internet of Things in the Cloud: A Middleware Perspective*, 2012, CRC Press.
4. Dieter Uckelmann; Mark Harrison; Florian Michahelles *Architecting the Internet of Things*,