



UNIVERSITY
of
TECHNOLOGY,
MAURITIUS

School of Business Informatics and Software Engineering

MSc Software Engineering

PROGRAMME DOCUMENT

VERSION 2.1

MSE v2.1

November 2014

University of Technology, Mauritius

La Tour Koenig, Pointe aux Sables, Mauritius

Tel: (230) 2075250 Fax: (230) 234 1747 Email: site@utm.intnet.mu

website: www.utm.ac.mu

MSc Software Engineering

A. Programme Information

The MSc in Software Engineering is primarily intended for graduates of computing subjects who want an in-depth focused in the software engineering discipline. This programme is a basis for further study, research and acquiring skills and knowledge in areas like design patterns, intelligent agent technologies and other specialist fields. The course comprises of a six months industrial project where the student will apply acquired knowledge and skills to solve real-life problem.

This course also helps the students to be conversant with all the software challenges that exist and shows them the right path to delivering code to the highest levels of quality, reliability and maintainability at the lowest cost.

For the award of the Masters Degree, all modules taken must be passed overall with passes in the examinations, coursework and other forms of assessment.

B. Programme Aims

The MSc Software Engineering programme prepares students to act project leaders. Furthermore, the students will be ready to join any public or private institution whether national or international where software development and maintenance are the main key activities.

This programme is also an important step towards research.

C. Programme Objectives

At the end of the programme, the students should be able to :

- Apply skills and knowledge acquired to solve real problems.
- Work in any software development institution.
- Carry out research in the software engineering area.

PART I - Regulations

D. General Entry Requirements

As per UTM Admission Regulations with reference to Masters Programmes.

E. Programme Entry Requirements

Applicants should have a computing background with programming and system design as strengths.

Admission decisions will be made by the School of Business Informatics and Software Engineering on a case by case basis. The primary criteria for acceptance of an applicant are: his or her anticipated successful completion of the programme.

The assessment of an applicant's likelihood of success can be carried out in a variety of ways e.g. entrance examination, interviews, etc. At the time of application, the student is expected to demonstrate his or her background in the field.

F. Programme Mode and Duration

Full Time: 1 years
Part Time: 1½ years

G. Teaching and Learning Strategies

Lectures and tutorials form part of the curriculum. Student should be encouraged to do self learning wherever applicable and interact with professionals in the industry via seminars, workshops and existing industry project.

H. Student Support and Guidance

Guidance will be provided on all aspects of the programmes through any means of communication with Programme Director and Programme Coordinator.

I. Attendance Requirements

As per UTM's Regulations and Policy.

J. Credit System

1 module = 3 or 4 credits
Industrial Project= 12 credits

K. Student Progress and Assessment

For the award of the Masters Degree, all modules must be passed overall with passes in the examinations, coursework and other forms of assessment.

All modules will carry 100 marks and will be assessed as follows (unless otherwise specified):

Written examinations up to 3-hours' duration and continuous assessment carrying up to 40% of total marks. A module can also be conducted as a project, that is, 100% coursework.

Continuous assessment can be based on seminars, practical labs and/or assignments or class tests.

Grading

| Grade | Marks x(%) |
|-------|------------------|
| A | $x \geq 70$ |
| B | $60 \leq x < 70$ |
| C | $50 \leq x < 60$ |
| D | $40 \leq x < 50$ |
| F | $x < 40$ |
| A-D | Pass |
| F | Fail |

L. Evaluation of Performance

Minimum Credits Required for Award of:

| | |
|---------------------------|----|
| Master's Degree: | 42 |
| Postgraduate Diploma: | 30 |
| Postgraduate Certificate: | 18 |

Breakdown as follows:

(i) MSc Software Engineering

8 Core modules: 30 credits

+Master Dissertation (Industrial Project): 12 credits

Candidates who obtain a mark of 70% and above will be awarded an MSc with Distinction.

(ii) Postgraduate Diploma Software Engineering

8 Core modules: 30 credits

(iii) Postgraduate Certificate Software Engineering

5 Core modules: 18 credits

M. Award Classification

| Overall weighted mark x (%) | Classification |
|-----------------------------|----------------------|
| $x \geq 70$ | MSc with Distinction |
| $60 \leq x < 70$ | MSc with Merit |
| $40 \leq x < 60$ | MSc |

N. Programme Organisation and Management

Programme Director and Co-ordinator: Mr. J. Narsoo

Contact Details : Tel: 2075250 Fax: 234-1767 Email: jnarsoo@utm.intnet.mu

PART II - Programme Structure

O. MSc Software Engineering – Full Time (Version 2.1)

| <i>Semester 1</i> | | | | <i>Semester 2</i> | | | |
|-------------------|-------------------------------|---------|---------|-------------------|--|---------|---------|
| Code | Module | L + T/P | Credits | Code | Module | L + T/P | Credits |
| OSS5101 | Unix Programming | 2 + 2 | 4 | SEM5102 | Applied Software Project Management | 2 + 2 | 4 |
| SDT5101 | Design Patterns | 2 + 2 | 4 | MCT5103 | Mobile Systems Architecture and Applications | 2 + 2 | 4 |
| SEM5201 | Software Quality Assurance | 2 + 1 | 3 | DTB5101 | Database Systems Integration | 2 + 2 | 4 |
| WAT5101 | Service Oriented Architecture | 2 + 2 | 4 | ISM5104 | Global IT Management | 2 + 1 | 3 |
| PRJ5204 | Industrial Project | - | - | PRJ5204 | Industrial Project | - | 12 |

P. MSc Software Engineering – Part Time (Version 2.1)

| <i>Semester 1</i> | | | | <i>Semester 2</i> | | | |
|-------------------|-------------------------------|---------|---------|-------------------|--|---------|---------|
| Code | Module | L + T/P | Credits | Code | Module | L + T/P | Credits |
| OSS5101 | Unix Programming | 2 + 2 | 4 | SEM5201 | Software Quality Assurance | 2 + 1 | 3 |
| SDT5101 | Design Patterns | 2 + 2 | 4 | SEM5102 | Applied Software Project Management | 2 + 2 | 4 |
| WAT5101 | Service Oriented Architecture | 2 + 2 | 4 | MCT5103 | Mobile Systems Architecture and Applications | 2 + 2 | 4 |
| | | | | PROJ5101 | Industrial Project | - | - |

| <i>Semester 3</i> | | | |
|-------------------|------------------------------|---------|---------|
| Code | Module | L + T/P | Credits |
| DTB5101 | Database Systems Integration | 2 + 2 | 4 |
| ISM5104 | Global IT Management | 2 + 1 | 3 |
| PROJ5101 | Industrial Project | | 12 |

Q. MODULE OUTLINE

OSS5101: UNIX PROGRAMMING

A Tour of UNIX and Linux. Versions of UNIX., File I/O, File Descriptors and Open File Descriptions, Symbols for File Permission Bits, open and creat System Calls, umask System Call, unlink System Call, Creating Temporary Files, Disk Special Files and File Systems, Hard and Symbolic Links, Pathnames, Accessing and Displaying File Metadata, Directories, Changing an I-Node, Reading from a Terminal, Sessions and Process Groups (Jobs), Setting Terminal Attributes, Terminal-Identification System Calls, Processes and Threads: exec System Calls, Implementing a Shell, fork System Call, exit System Calls and Process Termination, User and Group Ids, Setting User and Group Ids, Process Ids, chroot System Call, Priority, Interprocess Communication: Pipes, dup, Unidirectional Pipes and Bidirectional Pipes, FIFOs, or Named Pipes, Simple Messaging Interface (SMI), System V IPC (Interprocess Communication), System V Message Queues, Semaphores, File Locking, Networking and Sockets: Socket Basics, Socket Addresses, Socket Options, Simple Socket Interface (SSI), Socket Implementation of SMI, Connectionless Sockets, Signals and Timers: Signal Basics, Deprecated Signal System Calls, Realtime Signals Extension (RTS). Global Jumps, Clocks and Timers, Shell programming

SDT5101: DESIGN PATTERNS

Review of software development processes, Object Oriented modelling Using UML, Introduction to Design Patterns, Examples of design patterns: Singleton, Façade, Factory, Adapter, Bridge, Composite, Decorator, Flyweight, Proxy, Builder, Prototype, Interpreter, Template, Chain of responsibility, Command, Iterator, Mediator, Memento, Observer, State, Strategy, Visitor

SEM5201: SOFTWARE QUALITY ASSURANCE

Differentiating between Quality Assurance, Quality Control and Testing, importance of quality, the cost of quality, two views of quality, Quality attributes, Quality audits, Management commitment to quality, Organisational structure, Placement within the organization, Quality coordination, Setting quality standards, Software Quality Assurance and CMMI: Process and product quality assurance, quality assurance goals, Processes for each goal, Risk identification and management, Software Quality Management and ISO 9001: Management responsibility, Internal quality audits, Control quality records, Document and data control, Training, Corrective and Preventive actions, Quality Tools: Tools for ideas and information, numbers, presentation, Software Quality Assurance Plan (SQAP): Relationship of SQAP to SQA program, SQAP components, Purpose, Reference documents, Standards, practices, and conventions, Reviews and audits, Problem reporting, Tools, techniques, and methodologies, Code control, Media control, Supplier control, Records collection, maintenance, and retention, Implementation of SQAP: Management acceptance, Acceptance by development personnel, Planning for SQAP implementation, Resources, Scheduling, Risk Assessment, Training, Distribution of SQAP, Execution of SQAP

WAT5101: SERVICE ORIENTED ARCHITECTURE

Service-Oriented Architecture in the enterprise: response to change and transformation, The evolution of the service idea, Managing upside and downside risk, SOA and Business Process Management: Service-oriented government and business, Line-of-business services and service domains, Service-level abstractions, Reusable business vs. technical services, Fundamentals of service-oriented architecture: Architecture focus, the concepts of the service contract, Enabling infrastructure, The Enterprise Architecture perspectives, The Zachman Framework and SOA, Process and use-case views, Data and information views, Logical view, Contract view, Implementation view, Service-oriented prerequisites: Security, Service management, Process and composition, Metadata management, Key components for implementing SOA: SOA Meta model – Understanding the role of the service contract, Policy-based services, Service-oriented processes, Atomic services and composite services, Building a service and managing the business side: Granularity, Context/domain dependency.

DBT5101: DATABASE SYSTEMS INTEGRATION

Distributed Database Management Systems, Heterogeneous Databases, Distributed Transaction processing, Data Mining, Data Warehousing, Online Analytical Processing (OLAP), Optimisation, Recover and Backup, Database Connectivity technologies, Database Middleware, World Wide Web Database Architecture, distribution and manipulation, Object-oriented Databases

ISM5104: GLOBAL IT MANAGEMENT

Global information technology environment, Framework for building national information structure, International business opportunities on the information superhighway, Problem of data quality in a developing economy, How can information technology enable developing countries to integrate into the global economy? Information technology planning and architectures for networked global organizations, Information systems resource structure and management in multinational organizations, Issues in Global Outsourcing, Global software outsourcing, Global management support systems, Strategic application of information technology in global business, Developing and implementing global information systems

SEM5102: APPLIED SOFTWARE PROJECT MANAGEMENT

This module can be conducted as a project-based module.

Project planning and management, Project Management Plan, Product development life cycles, Software life cycles, Development life cycles, Software maintenance, Life cycles incorporating software reuse, How to pick the appropriate life cycle?, Software development processes, Process concepts, Process maturity (CMM), Process Interaction, Supporting or interfacing processes, Marketing, Activity planning and scheduling, Cost and resource estimating, Resource management, Determining project status, Contracting and outsourcing, Monitoring and Control, Organisational strategies, Managing teams, Environmental and social factors, Standards. Case Studies

SEM5103: MOBILE SYSTEMS ARCHITECTURE AND APPLICATIONS

Mobile Architecture, Mobile Platforms and environment, Mobile Software Development Tools, Mobile application design, development, testing and implementation and deployment, User Interface Design, Memory Management and Security, Data Transfer, Disconnected Architecture, Middleware, Bluetooth Technologies and applications.