Shri Vile Parle Kelavani Mandal’s

Narsee Monjee College of Commerce and Economics

(Autonomous)

TYBSC IT SYLLABUS

Semester V & VI

Table of Contents

Course Structure (Semester V)

	T.Y BSc IT (To be implemented for the academic year 2022-2023)			
Semester V (Total Credits 18)				
Sr. No.	**Module Code**	**Module Name**		**Credits**
1	**NMUBSCIT504**	Advanced Web Programming	Skill Enhancement Courses (SEC)	3
2	**NMUBSCITP504**	Advanced Web Programming Practical		1
3	**NMUBSCIT505**	Enterprise Java		3
4	**NMUBSCITP505**	Enterprise Java Practical		1
5	**NMUBSCIT501**	Applied Artificial Intelligence	Core Course (CC)	3
6	**NMUBSCITP501**	Applied Artificial Intelligence Practical		1
7	**NMUBSCIT502**	Internet of Things		2
8	**NMUBSCITP502**	Internet of Things Practical		1
9	**NMUBSCIT503**	Cloud Computing		2
10	**NMUBSCITP503**	Cloud Computing Practical		1
	Total credits			**18**

SEMESTER V

Applied Artificial Intelligence

Programme : B. Sc IT (Information Technology)	**Semester V**
Course : Applied Artificial Intelligence	**Code : NMUBSCIT501**

Suggested Lectures per week	**03**
Practical Session per week (per Batch)	**01**

Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
45	**15 X 3 = 45**	**Nil**	**03 + 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**

Internal Component (Theory Break up)	
Class Test Duration 20 Mins	**Assignments**
10 Marks	**15 Marks**

Internal Component (Practical Break up)	
Examination (Duration 1 ½ Hrs)	**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**
30 Marks	**20 Marks**

Learning Objectives

1. To gain knowledge of Artificial Intelligence
2. To learn Evolutionary Computation
3. To learn Machine Learning, Artificial Neural Network, Natural Language Processing

Learning Outcomes :

1. Knowledge of Artificial Intelligence
2. Knowledge of GA and Evolutionary Computation
3. Knowledge of Machine learning, Artificial Neural Network, Natural Language processing

Pedagogy :

PPts,Videos,Case Studies , Small project(Group wise)

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	**What is AI?** The History of Artificial Intelligence, **Expert System and Applications:** Phases in Building Expert System Architecture **Searching Techniques** Searching for Solutions, Uninformed Search and Informed (Heuristic) Search		9

II	**Probability Theory:** joint probability, conditional probability, Bayes's theorem **Introduction to Fuzzy System** Fuzzy Sets and Membership, Fuzzy Logic,Fuzzy Systems		9
III	**Introduction to Evolutionary Computation:** Elementsof Genetic Algorithms, genetic programming concepts, evolutionaryprogramming, swarm intelligence Genetic Algorithms and Traditional Search Methods,Some Applications of Genetic Algorithms		9
IV	**Introduction to Machine Learning Paradigms:** Machine Learning systems, supervised and un-supervisedlearning **Introduction to ANN** Units in neural networks, Network structures, Single layer feed-forward neural networks (perceptrons), Applications of ANN		9
V	**Intelligent Agents:** Agents vs software programs, classification of agents, working of an agent, single agentand multiagent systems. **Natural Language Processing** What is natural processing processing.elements of natural language processing ,Applications Of natural Language Processing		9

<u>Books and References:</u>

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Artificial Intelligence- A Modern Approach	Stuart Russel, Peter Norvig	Pearson Education		
2.	Artificial Intelligence	Saroj Kaushik	Cengage	1 st	2019
3.	An Introduction to genetic algorithms	Melanie Mitchell			

4.	Fuzzy Logic with Engineering Applications	Timothy J. Ross			
5.	Artificial Intelligence	Elaine Rich, Kevin Knight			

List of Practical

1. Implementation of any uninformed search methods.
2. Simulation of TIC-TAC-TOE using MiniMax algorithm
3. Implementation of any informed search methods.
4. Implementation of a simple Neural Network using python’s inbuilt libraries.
5. Implementation of a simple Neural Network without using pythons inbuilt libraries.
6. Implementation of a simple Genetic Algorithm
7. Demonstration of fuzzy set operations.
8.Demonstration of fuzzy system.
9.Implementation of simple expert system in prolog10.

Creating a simple bot using aiml package.

Internet Of Things

Programme : B. Sc IT (Information Technology)						**Semester : V**	
Course : Internet Of Things						**Code : NMUBSCIT502**	
Suggested Lectures per week				**02**			
Practical Session per week (per Batch)				**01**			
Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Internal**	**External**
30	**15 X 2 = 30**	**Nil**	**02**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**

Internal Component (Theory Break up)	
Class Test Duration 30 Mins	**Assignments**
15 Marks	**10 Marks**

Internal Component (Practical Break up)	
Examination (Duration 1 ½ Hrs)	**Mini Project/Case study/ Field Visit (report to be submitted and certified prior to practical examination)**
30 Marks	20 Marks

Learning Objectives :

1. To teach students about internet of things
2. Identify the areas where the applications are used
3. Make students understand how the concept can be used in day to day life

Learning Outcomes :

1. Understand general concepts of Internet of Things (IoT)
2. Recognize various devices ,sensors and applications
3. Apply design concept to IoT solutions
4. Analyze various M2M and IoT architectures
5. Create IoT solutions using sensors, actuators and Devices

Pedagogy :

PPTs,Videos, Blackboard,ready made projects demo.

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	Introduction to IoT: Sensing, Actuation, Networking basics, Communication Protocols, Sensor Networks, Machine-to-Machine Communications, IoT Definition, Characteristics. IoT Functional Blocks, Physical design of		6

	IoT, Logical design of IoT, Communication models & APIs		
II	IoT Platform Design Methodology - Introduction, IoT Design Methodology, Case study on IoT system for weather monitoring, motivation for using python.		6
III	M2M to IoT-The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics. Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT		6
IV	M2M vs IoT An Architectural Overview–Building architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. Reference Architecture and Reference Model of IoT		6
V	IoT physical Devices and Endpoints- What is an IoT device, Exemplary Devices: Raspberry Pi, About the board, Raspberry Pi interfaces Domain specific applications of IoT: Home automation, Industry applications, Surveillance applications, Other IoT application		6

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	"From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence",	Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle	Academic Press, 2014	1st Edition, .	2014
2	"Internet of Things (A Hands-onApproach)", 1st Edition, VPT, 2014	Vijay Madisetti and Arshdeep Bahga,			

List of Practical

1. Starting Raspbian OS Familiarising with Raspberry Pi Components and interface, Connecting to ethernet, Monitor,USB.
2. Displaying different LED patterns with Raspberry Pi.
3. Controlling Raspberry Pi with WhatsApp.
4. Fingerprint Sensor interfacing with Raspberry Pi/Arduino Uno
5. Raspberry Pi GPS Module Interfacing
6. Demonstrate Arduino and its pins
7. Perform Experiment using Arduino Uno to measure the distance of any object using Ultrasonic Sensor.
8. Perform Experiment using Arduino Uno to implement the working of Stepper Motor.
9. Creating a webpage and display the values available through Arduino
10. Perform Experiment using Arduino Uno to measure temperature and Humidity
11. Perform Experiment using Arduino Uno to interface 4 digit 7 segment display

Cloud Computing

Programme : B. Sc IT (Information Technology)					**Semester V**		
Course : Cloud Computing					**Code : NMUBSCIT503**		
Suggested Lectures per week				**02**			
Practical Session per week (per Batch)				**01**			
Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
30	**15 X 2 = 30**	**Nil**	**02 + 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**
Internal Component (Theory Break up)							
Class Test Duration 20 Mins				**Assignments**			
10 Marks				**15 Marks**			
Internal Component (Practical Break up)							
Examination (Duration 1 ½ Hrs)				**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**			
30 Marks				**20 Marks**			
Learning Objectives							
1. To learn basics of Cloud Computing 2. To learn Virtualization and Cloud Computing Architecture 3. To know Concepts of Cost metric and pricing models, GAE, AWS, and Azure							
Learning Outcomes :							
1. Knowledge of Cloud Computing 2. Knowledge of Cloud Computing Achitecture 3. Knowledge of GAE, Azure, AWS							
Pedagogy :							
PPts,Videos,Case Studies , Small project(Group wise)							

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	**Introduction to Cloud Computing:** Introduction, Characteristics and benefits of Cloud Computing, **Principles of Parallel and Distributed Computing:** Parallel v/s distributed computing, Elementsof Parallel Computing, Elements of distributed computing.		6

II	**Virtualization:** Introduction, Characteristics of virtualized environments, Taxonomy of virtualization techniques, Pros and cons of virtualization **Cloud Computing Architecture:** Introduction, Fundamental concepts and models, Cloud Delivery models, Cloud Deployment models.		6
III	**Fundamental Cloud Security:** Basics, Threat agents, Cloud security threats, additional considerations. **Cost Metrics and Pricing Models**: Business Cost Metrics, Cloud Usage Cost Metrics,.		6
IV	**Cost Metrics and Pricing Models**:Cost Management Considerations, **Public Cloud Platforms: GAE, AWS, and Azure**		6
V	**Amazon Web Services (AWS) Essentials:** Architecting on AWS, Building complex solutions with Amazon Virtual Private Cloud(Amazon VPC), AmazonRedshift a		6

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Mastering Cloud Computing Foundations and Applications Programming	Rajkumar Buyya, Christian , S. Thamarai Selvi	Elsevier		
2.	. Cloud Computing Concepts, Technology & Architecture	Thomas Erl, Zaigham Mahmood, and Ricardo Puttini	Prentice Hall		
3.	Distributed and Cloud Computing, From Parallel Processing tothe Internet of Things	Kai Hwang, Jack Dongarra, Geoffrey Fox	MK Publishers		

4.	AWS Training, http://aws.amazon.com/training .				

List of Practical

1. Implementation of virtualization using hyperV .
2. Exploring VmwareEsxi
3. Exploring Xen server with Citrix Xen application .
4. Deploying PAAS using Google App Engine.
5. Deploying PAAS using Azure.
6. Creating and launching Amazon EC2 instances
7. Monitoring of Amazon EC2 instances.
8. Simulation of cloud Environment using CloudSim.
9. Simulation of cloud Environment using CloudAnalyst
10. Creating private cloud using any tool
11. Implement Client Server communication model using TCP
12. Implement Client Server communication model using UDP.
13. Implement multicast Socket
14. Implement RPC

Advanced Web Programming

Programme : B. Sc IT (Information Technology)					**Semester V**		
Course : Advanced Web Programming					**Code : NMUBSCIT504**		
Suggested Lectures per week				**03**			
Practical Session per week (per Batch)				**01**			
Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
45	**15 X 3 = 45**	**Nil**	**03 + 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**
Internal Component (Theory Break up)							
Class Test Duration 20 Mins				**Assignments**			
10 Marks				**15 Marks**			
Internal Component (Practical Break up)							
Examination (Duration 1 ½ Hrs)				**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**			
30 Marks				**20 Marks**			

Learning Objectives

1. Gain a thorough understanding of the philosophy and architecture of Web applications using ASP.NET Core MVC;
2. Gain a practical understanding of.NET Core;
3. Acquire a working knowledge of Web application development using ASP.NET Core MVC 6 and Visual Studio
4. Persist data with XML Serialization and ADO.NET with SQL Server
5. Create HTTP services using ASP.NET Core Web API;
6. Deploy ASP.NET Core MVC applications to the Windows Azure cloud

Learning Outcomes :

1. Complete understanding of ASP.Net Core Technology
2. Development Ability – MVC and Core
3. Ability to create Microservice and deploy it

Pedagogy :

PPts,Videos,Case Studies , Small project(Group wise)

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	Introduction to .NET Framework , Introduction to ASP.NET Core 5. Understanding C# Programming C# Programming language. The C# Language: C# Language Basics, Variables and Data Types, Variable Operations, Object-Based Manipulation, Conditional Logic, Loops, Methods.		9
II	Building ASP.NET Websites with HTML, CSS and C# code Using Web controls, Validation Controls, User Controls, Databases, Getting External data, Ajax ToolKit		9
III	Fundamentals of web application development in Cross-Platform Setup - Leveraging the .NET framework, Combining cross-platform and single-platform code.		9
IV	Understanding Razor View Engine. Learning basics of Razor pages . HTML Helpers . Rendering data from view model.		9
V	Understanding MVC Pattern. Creating an MVC Application. Building To Do Application with MVC		9

Books and References:

1. C# 2015 Anne Bohem and Joel Murach Murach Third
2. Web Programming ASP.NET Core Hans-Petter Halvorsen 2021
3. ASP.NET_Core_5_for_Beginners__Kickstart_your_ASP.NET_web_development_journey_(2020) (Andreas Helland, Vinceck Maverick)

List of Practical

1. Building C# Console Applications

2. Building a website using ASP.NET

3. Creating MVC Application

Enterprise Java

Programme : B. Sc IT (Information Technology)					**Semester V**		
Course : Enterprise Java					**Code : NMUBSCIT505**		
Suggested Lectures per week				**03**			
Practical Session per week (per Batch)				**01**			
Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
45	**15 X 3 = 45**	**Nil**	**03 + 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**
Internal Component (Theory Break up)							
Class Test Duration 20 Mins				**Assignments**			
10 Marks				**15 Marks**			
Internal Component (Practical Break up)							
Examination (Duration 1 ½ Hrs)				**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**			
30 Marks				**20 Marks**			
Learning Objectives							
1. To gain knowledge of Java Development 2. To learn IDE 3. To learn Enterprise Applications							
Learning Outcomes							
1. Knowledge of Java Development 2. Knowledge of Java IDE 3. Introductory Enterprise Applications							
Pedagogy							
Presentations, Programming Simulators							

I	**Understanding Java EE:** What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Glassfish server **Java EE Architecture, Server and Containers:** Types of System Architecture, Java EE Server, Java EE Containers. **Introduction to Java Servlets:** The Need for Dynamic Content, Java Servlet Technology, Why Servlets? What can Servlets do? **Servlet API and Lifecycle:** Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle, A Simple Welcome Servlet **Working With Servlets:** Getting Started, Using Annotations Instead of Deployment Descriptor. **Working with Databases:** What Is JDBC? JDBC Architecture, Accessing Database, The Servlet GUI and Database Example.		9
II	**Request Dispatcher:** Resquestdispatcher Interface, Methods of Requestdispatcher, Requestdispatcher Application. **COOKIES:** Kinds Of Cookies, Where Cookies Are Used? Creating Cookies Using Servlet, Dynamically Changing The Colors Of A Page **SESSION:** What Are Sessions? Lifecycle Of Http Session, Session Tracking With Servlet API, A Servlet Session Example **Working With Files:** Uploading Files, Creating an Upload File Application, Downloading Files, Creating a Download File Application. **Working With Non-Blocking I/O:** Creating a Non-Blocking Read Application, Creating The Web Application, Creating Java Class, Creating Servlets, Retrieving The File, Creating index.jsp		9

	Introduction To Java Server Pages: Why use Java Server Pages? Disadvantages Of JSP, JSP v\s Servlets, Life Cycle of a JSP Page, How does a JSP function? How does JSP execute? About Java Server Pages **Getting Started With Java Server Pages:** Comments, JSP Document, JSP Elements, JSP GUI Example. **Action Elements:** Including other Files, Forwarding JSP Page to Another Page, Passing Parameters for other Actions, Loading a Javabean. **Implicit Objects, Scope And El Expressions:** Implicit Objects, Character Quoting Conventions, Unified Expression Language [Unified El], Expression Language. **Java Server Pages Standard Tag Libraries:** What is wrong in using JSP Scriptlet Tags? How JSTL Fixes JSP Scriptlet's Shortcomings? Disadvantages Of JSTL, Tag Libraries.		9
IV	**Introduction To Enterprise Javabeans:** Enterprise Bean Architecture, Benefits of Enterprise Bean, Types of Enterprise Bean, Accessing Enterprise Beans, Enterprise Bean Application, Packaging EnterpriseBeans **Getting started with Enterprise Javabeans:** Creating a Web Application, Creating an Enterprise Bean, Creating a Web Client [Servlet], Creating a JSP File, Build the Web Application, Running the Web Application. **Working With Session Beans:** When to use Session Beans? Types of Session Beans, Remote and Local Interfaces, Accessing Interfaces, Lifecycle of Enterprise Beans, Packaging Enterprise Beans, Example of Stateful Session Bean, Example of Stateless Session Bean, Example of Singleton Session Beans. **Working with Message Driven Beans:** Lifecycle ofa Message Driven Bean, Uses of Message Driven Beans, The Message Driven Beans Example. **INTERCEPTORS:** Request And Interceptor, Defining An Interceptor, AroundInvoke Method, Applying Interceptor, Adding An Interceptor To An Enterprise Bean, Build and Run the Web Application		9

V	**Persistence, Object/Relational Mapping And JPA:** What is Persistence? Persistence in Java, Current Persistence Standards in Java, Why another Persistence Standards? Object/Relational Mapping, **Introduction to Java Persistence API:** The Java Persistence API, JPA, ORM, Database and the Application, Architecture of JPA, How JPA Works? JPA Specifications. **Writing JPA Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database And Tables in Mysql, Creating a Web Application, Adding the Required Library Files, Creating a Javabean Class, Creating Persistence Unit [Persistence.Xml], Creating JSPS, The JPA Application Structure, Running The JPA Application. **Introduction to Hibernate:** What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works? **Writing Hibernate Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, Creating a Web Application, Adding The Required Library Files, Creating a Javabean Class, Creating Hibernate Configuration File, Adding a Mapping Class, Creating JSPS, Running The Hibernate Application.		9

Books and References:

Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Java EE 7 For Beginners	Sharanam Shah, Vaishali Shah	SPD	First	2017
2.	Java EE 8 Cookbook: Build reliable applicationswith the most robust and mature technology for enterprise development	Elder Moraes	Packt	First	2018
3.	Advanced Java Programming	Uttam Kumar Roy	Oxford Press		2015

List of Practical

Implement the following simple servlet applications

1. Create a simple calculator application using servlet

2. Create a servlet for a login page. If the username and password are correct then it says message “Hello <username>” else a message “login failed”

3. Create a registration servlet in Java using JDBC. Accept the details such as Username, Password, Email, and Country from the user using HTML Form and store the registration details in the database.

Implement the following Servlet applications with Cookies and Sessions.

4. Using Request Dispatcher Interface create a Servlet which will validate the password entered by the user, if the user has entered "Servlet" as password, then he will be forwarded to Welcome Servlet else the user will stay on the index.html page and an error message will be displayed.

5. Create a servlet that uses Cookies to store the number of times a user has visited servlet.

6. Create a servlet demonstrating the use of session creation and destruction. Also check whether the user has visited this page first time or has visited earlier also using sessions.

Implement the Servlet IO and File applications

7. Create a Servlet application to upload and download a file.

8. Develop Simple Servlet Question Answer Application using Database.

9. Create simple Servlet application to demonstrate Non-Blocking Read Operation.

10. Implement the following JSP applications.

11. Develop a simple JSP application to display values obtained from the use of intrinsic objects of various types.

12. Develop a simple JSP application to pass values from one page to another with validations. (Name-txt, age-txt, hobbies-checkbox, email-txt, gender-radio button).

13. Create a registration and login JSP application to register and authenticate the user based on username and password using JDBC.

Implement the following JSP JSTL and EL Applications.

14. Create an html page with fields, eno, name, age, desg, salary. Now on submit this data to a JSP page which will update the employee table of database with matching eno.

15. Create a JSP page to demonstrate the use of Expression language.

16. Create a JSP application to demonstrate the use of JSTL.

17. Implement the following EJB Applications.

18. Create a Currency Converter application using EJB.

19. Develop a Simple Room Reservation System Application Using EJB

20. Develop simple shopping cart application using EJB [Stateful Session Bean].

Implement the following EJB applications with different types of Beans.

21. Develop simple EJB application to demonstrate Servlet Hit count using Singleton Session Beans.

22. Develop simple visitor Statistics application using Message Driven Bean [Stateless Session Bean].

23. Develop simple Marks Entry Application to demonstrate accessing Database using EJB.

Implement the following JPA applications

24. Develop a simple Inventory Application Using JPA.

25. Develop a Guestbook Application Using JPA.

26. Create simple JPA application to store and retrieve Book details.

Implement the following JPA applications with ORM and Hibernate.

27. Develop a JPA Application to demonstrate use of ORM associations.

28. Develop a Hibernate application to store Feedback of Website Visitor in MySQL Database.

29. Develop a Hibernate application to store and retrieve employee details in MySQL Database.

Implement the following Hibernate applications.

30. Develop an application to demonstrate Hibernate One- To -One Mapping Using Annotation.

31. Develop Hibernate application to enter and retrieve course details with ORM Mapping.

32. Develop a five page web application site using any two or three Java EE Technologies.

Course Structure (Semester VI)

	T.Y BSc IT (To be implemented for the academic year 2022-2023)			
Semester VI (Total Credits 18)				
Sr. No.	**Module Code**	**Module Name**		**Credits**
1	**NMUBSCIT605**	Advanced Mobile Programming	Skill Enhancement Courses (SEC)	2
2	**NMUBSCITP605**	Advanced Mobile Programming Practical		1
3	**NMUBSCIT604**	Ethical Hacking		2
4	**NMUBSCITP604**	Ethical Hacking Practical		1
5	**NMUBSCIT601**	Software Project Management	Core Course (CC)	3
6	**NMUBSCITP601**	Project		1
7	**NMUBSCIT602**	Software Testing & Quality Assurance		3
8	**NMUBSCITP602**	Software Testing & Quality Assurance Practical		1
9	**NMUBSCIT603**	Data Science		3
10	**NMUBSCITP603**	Data Science Practical		1
	Total credits			**18**

SEMESTER VI

Software Project Management

Programme : B. Sc IT (Information Technology)	Semester VI
Course : Software Project Management	Code : NMUBSCIT601
Suggested Lectures per week	03
Practical Session per week (per Batch)	01

Teaching Scheme				Evaluation Scheme			
Lecture	Practical	Tutorial	Credits	Theory		Practical	
				Internal	External	Component 1	Component 2
45	15 X 3= 45	Nil	03+ 01	25 Marks	75 Marks	20 Marks	30 Marks

Internal Component (Theory Break up)	
Class Test Duration 20 Mins	Assignments
10 Marks	15 Marks

Internal Component (Practical Break up)	
Examination (Duration 1 ½ Hrs)	Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)
30 Marks	20 Marks

Learning Objectives

1. To understand the concepts of Project Management object oriented programming
2. Identify the various approaches to project development
3. Developing students to develop project.

Learning Outcomes :

1. Project Management Skills
2. Project Development Life Cycle
3. Cost Analysis of Project

Pedagogy :
Presentations, Programming Simulators, Search Engines , Videos,Online Tutorials

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	**Introduction to Software Project Management:** Introduction, Why is Software Project Management Important? What is a Project? Software Projects versus Other Types of Project, Contract Management and		09

	Technical Project Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Some Ways of Categorizing Software Projects, Project Charter, Stakeholders, Setting Objectives, The Business Case, Project Success and Failure, What is Management? Management Control Project Management Life Cycle, Traditional versus Modern Project Management Practices. **Project Evaluation and Programme Management :** Introduction, Business Case, Project Portfolio Management, Evaluation of Individual Projects, Cost– benefit Evaluation Techniques, Risk Evaluation, Programme Management, Managing the Allocation of Resources within Programmes, Strategic Programme Management,Creating a Programme, Aids to ProgrammeManagement, Some Reservations about Programme Management, Benefits Management, An Overview of Project Planning.		
II	**Selection of an Appropriate Project Approach:** Introduction, Build or Buy? Choosing Methodologies and Technologies, Software Processes and Process Models, Choice of Process Models, Structure versus Speed of Delivery, The Waterfall Model, The Spiral Model, Software Prototyping, Other Ways of Categorizing Prototypes, Incremental Delivery, Atern/Dynamic Systems Development Method, Rapid Application Development, Agile Methods, Extreme Programming (XP), Scrum, Lean Software Development, Managing Iterative Processes, Selecting the Most Appropriate Process Model. **Software Effort Estimation:** Introduction, Where are the Estimates Done? Problems with Over- and Under- Estimates, The Basis for Software Estimating, Software Effort Estimation Techniques, Bottomup Estimating, The Top-down Approach and Parametric Models, Expert Judgement, Estimating by Analogy, Albrecht Function Point Analysis, Function Points Mark II, COSMIC Full Function Points, COCOMO II: A Parametric Productivity Model, Cost Estimation, Staffing Pattern, Effect of Schedule compression, Capers Jones Estimating Rules of Thumb.		09
III	**Activity Planning** : Introduction, Objectives of Activity Planning, When to Plan, Project Schedules, Projects and Activities, Sequencing and Scheduling Activities, Network Planning Models, Formulating a Network Model, Adding the Time Dimension, The Forward Pass, Backward Pass,		09

	Identifying the Critical Path, Activity Float, Shortening the Project Duration, Identifying Critical Activities, Activity- on-Arrow Networks. **Risk Management**: Introduction, Risk, Categories of Risk, Risk Management Approaches, A Framework for Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the Schedule, Boehm‟s Top 10 Risks and Counter Measures, Applying the PERT Technique, Monte Carlo Simulation, Critical Chain Concepts.		
IV	**Resource Allocation:** Introduction, Nature of Resources, Identifying Resource Requirements, Scheduling Resources, Creating Critical Paths, Counting the Cost, Being Specific, Publishing the Resource Schedule, Cost Schedules, Scheduling Sequence. **Monitoring and Control:** Introduction, Creating the Framework, Collecting the Data, Review, Visualizing Progress, Cost Monitoring,Earned Value Analysis, Prioritizing Monitoring, Getting the Project Back toTarget, Change Control, Software Configuration Management (SCM). **Managing Contracts**: Introduction, Types of Contract, Stages in Contract Placement, Typical Terms of a Contract, Contract Management, Acceptance. .		09
V	**Managing People in Software Environments**: Introduction, Understanding Behaviour, Organizational Behaviour: A Background, Selecting the Right Person for the Job, Instruction in the Best Methods, Motivation, The Oldham–Hackman Job Characteristics Model, Stress, Stress Management, Health and Safety, Some Ethical and Professional Concerns **Working in Teams:** Introduction, Becoming a Team, Decision Making, Organization and Team Structures, Coordination Dependencies, Dispersed and Virtual Teams, Communication Genres, Communication Plans, Leadership. **Project Closeout**: Introduction, Reasons for ProjectClosure, Project Closure Process, Performing a Financial Closure, Project Closeout Report.		09

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Project Management	Bob Hughes, Mike Cotterell, Rajib Mall	TMH	6th	2018
2.	Project Management and Tools & Technologies – An overview	Shailesh Mehta	SPD	1st	2017

Practical

Project Development (Live/Dummy)

Software Testing & Quality Assurance

Programme : B. Sc IT (Information Technology)	**Semester : VI**
Course Name: Software Testing & Quality Assurance	**Code: NMUBSCIT602**

Suggested Lectures per week	**03**
Practical Session per week (per Batch)	**01**

Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Internal**	**External**
45	**15 X 3 = 45**	**Nil**	**03 +1=4**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**

Internal Component (Theory Break up)	
Class Test Duration 30 Mins	**Assignments**
15 Marks	**10 Marks**

Internal Component (Practical Break up)	
Examination (Duration 1 ½ Hrs)	**Mini Project/Case study/ Field Visit (report to be submitted and certified prior to practical examination)**
30 Marks	20 Marks

Learning Objectives :

1. Give students understanding of quality management
2. Fundamentals of testing and different types of testing
3. Need for verification and validation

Learning Outcomes :

1. Knowledge of Software Testing
2. Student should be well versed with the concept of Total Quality Management
3. Student can perform testing at different levels during project development

Pedagogy :

PPts,Videos, Website, Test cases

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	**Introduction to Quality:** Historical Perspective of Quality, What is Quality? (Is it a fact or perception?), Definitions of Quality, Core Components of Quality, Quality View, Financial Aspect of Quality, Customers, Suppliers and Processes, Total Quality Management		9

	(TQM), Quality Principles of Total Quality Management, Quality Management Through Statistical Process Control, Quality Management Through Cultural Changes, Continual (Continuous) Improvement Cycle, Quality in Different Areas, Benchmarking and Metrics, Problem Solving Techniques, Problem Solving Software Tools. **Software Quality:** Introduction, Constraints of Software Product Quality Assessment, Customer is aKing, Quality and Productivity Relationship, Requirements of a Product, Organisation Culture, Characteristics of Software, Software Development Process, Types of Products, Schemes of Criticality Definitions, Problematic Areas of Software Development Life Cycle, Software Quality Management, Why Software Has Defects? Processes Related toSoftware Quality, Quality Management System Structure, Pillars of Quality Management System, Important Aspects of Quality Management.		
II	**Fundamentals of testing:** Introduction, Necessity of testing, What is testing? Fundamental test process, The psychology of testing, Historical Perspective of Testing, Definitions of Testing, Approaches to Testing, Testing During Development Life Cycle, Requirement Traceability Matrix, Essentials of Software Testing, Workbench, Important Features of Testing Process, Misconceptions About Testing, Principles of Software Testing, Salient Features of Good Testing, Test Policy, Test Strategy or Test Approach, Test Planning, Testing Process and Number of Defects Found in Testing, Test Team Efficiency, Mutation Testing, Challenges in Testing, Test Team Approach, Process Problems Faced by Testing, Cost Aspect of Testing, Establishing Testing Policy, Methods, Structured Approach to Testing, Categories of Defect, Defect, Error, or Mistake in Software, Developing Test Strategy, Developing Testing Methodologies (Test Plan), Testing Process, Attitude Towards Testing (Common People Issues), Test Methodologies/Approaches, People Challenges in Software Testing, Raising Management Awareness for Testing, Skills Required by Tester, Testing throughout the software life cycle, Software development models, Test levels, Test types, the targets of testing, Maintenance testing		9

III	**Unit Testing: Boundary Value Testing:** Normal Boundary Value Testing, Robust Boundary Value Testing, Worst-Case Boundary Value Testing, Special Value Testing, Examples, Random Testing, Guidelines for Boundary Value Testing, **Equivalence Class Testing:** Equivalence Classes, Traditional Equivalence Class Testing, Improved Equivalence Class Testing, Edge Testing, Guidelines and Observations. **Decision Table–Based Testing:** Decision Tables, Decision Table Techniques, Cause-and-Effect Graphing,Guidelines and Observations, **Path Testing:** Program Graphs, DD-Paths, Test Coverage Metrics, Basis Path Testing, Guidelines andObservations, **Data Flow Testing:** Define/Use Testing, Slice-Based Testing, Program Slicing Tools.		9
IV	**Software Verification and Validation:** Introduction, Verification, Verification Workbench, Methods of Verification, Types of reviews on the basis od StagePhase, Entities involved in verification, Reviews in testing lifecycle, Coverage in Verification, Concerns of Verification, Validation, Validation Workbench, Levels of Validation, Coverage in Validation, Acceptance Testing, Management of Verification and Validation, Software development verification and validation activities. **V-test Model:** Introduction, V-model for software, Testing during Proposal stage, Testing during requirement stage, Testing during test planning phase, Testing during design phase, Testing during coding, VV Model, Critical Roles and Responsibilities.		9
V	**Levels of Testing:** Introduction, Proposal Testing, Requirement Testing, Design Testing, Code Review, Unit Testing, Module Testing, Integration Testing, Big-Bang Testing, Sandwich Testing, Critical Path First, Sub System Testing, System Testing, Testing Stages. **Special Tests:** Introduction, GUI testing, Compatibility Testing, Security Testing, Performance Testing, Volume Testing, Stress Testing, Recovery Testing, Installation Testing, Requirement Testing, Regression Testing, Error Handling Testing, Manual Support Testing, Intersystem Testing, Control Testing, Smoke Testing,Adhoc Testing, Parallel Testing, Execution Testing, Operations Testing, Compliance Testing, Usability Testing, Decision Table Testing, Documentation		9

	Testing, Training testing, Rapid Testing, Control flow graph, Generating tests on the basis of Combinatorial Designs, State Graph, Risk Associated with New Technologies, Process maturity level of Technology, Testing Adequacy of Control in New technology usage,Object Oriented Application Testing, Testing of Internal Controls,COTS Testing, Client Server Testing, Web Application Testing, Mobile Application Testing, eBusiness eCommerce Testing, Agile Development Testing, Data Warehousing Testing.		

Books and References:

Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Software Testing and Continuous Quality Improvement	William E .Lewis	CRC Press	Third	2016
2.	Software Testing: Principles, Techniques and Tools	M. G. Limaye	TMH		2017
3.	Foundations of Software Testing	Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black	Cengage Learning	3rd	
4.	Software Testing: A Craftsman‟s Approach	Paul C. Jorgenson	CRC Press	4th	2017

Practical

1. You have got the brilliant idea of setting up a company that sells testing services to software houses. Make a strategic plan for your company, taking into account the following issues:

- What is the testing process that will be followed in the company?
- What is the focus of the testing services?
- What kind of people are you going to hire as staff for the company?
- How are you going to validate that a testing project carried out in the company has been beneficial to the customer?
- What kind of automated tools will the company use?

2. Prepare a small project and submit SRS, design, coding and test plan.

3. The program reads an arbitrary number of temperatures (as integer numbers) within the range -60°C … +60°C and prints their mean value. Design test cases for testing the program with the black-box strategy.

4. When getting a person's weight and height as input, the program prints the person's body weightindex. The weight is given in kilograms (as a real number, for instance: 82,0) and the height in meters (as a real number, for instance: 1,86). The body weight index equals weight divided by height squared: weight / (height ´height). Design test cases for testing the program with the black-box strategy.

5. Let us study the following program:

```
x=0; read(y);
while (y > 100) { x=x+y; read(y); }
if (y < 200) print(x) else print(y);
```

a) Construct a control-flow graph for the program.

b) Design test cases for reaching complete branch coverage over the program. Use as few testcases as possible.

6. Design test cases for the following program with the "simple loop" strategy:

```
x=0; read(y);
while ((y > 100) && (x < 10)) { x=x+1; read(y); }
print(y);
```

a) Construct a data-flow graph for the program with respect to variable x.

b) Which execution paths have to be traversed during testing, in order to reach completealldefinitions coverage with respect to variable x?

c) Minimize the number of paths and tests.

d) Which execution paths have to be traversed during testing, in order to reach complete alluses coverage with respect to variable x? Minimize the number of paths and tests.

e) Design test cases for reaching the (minimal) complete all-uses coverage with respect to variable x.

7. MANUAL TESTING for the project

a) Walkthrough

b) Whitebox Testing

c) Blackbox Testing

d) Unit Testing

e) Integration Testing

8. Functional Testing

a) Boundary value Testing
b) Equivalence class testing
c) Decision Table based testing
d) Cause-effect graphs
9. Regression Testing using automated testing for website.
10. AUTOMATED TESTING for websites
a) Load Testing
b) Performance Testing

Data Science

Programme : B. Sc IT (Information Technology)					**Semester VI**		
Course : Data Science					**Code : NMUBSCIT603**		
Suggested Lectures per week				**03**			
Practical Session per week (per Batch)				**01**			
Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
45	**15 X 3 = 45**	**Nil**	**03+ 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**
Internal Component (Theory Break up)							
Class Test Duration 20 Mins				**Assignments**			
10 Marks				**15 Marks**			
Internal Component (Practical Break up)							
Examination (Duration 1 ½ Hrs)				**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**			
30 Marks				**20 Marks**			

Learning Objectives

1. To understand data science models
2. To begin to construct DS models of their own.
3. To be able to compare performance
4. To be able to implement DS algorithms into high-level programming language.
5. To formulate simple algorithms to solve problems, and can code them in a high-level language appropriate for data science work (e.g., Python, SQL, R, Java).

Learning Outcomes :

1. Learners will develop relevant programming abilities.
2. Learners will demonstrate proficiency with statistical analysis of data.
3. Learners will develop the ability to build and assess data-based models.
4. Learners will execute statistical analyses with professional statistical software.
5. Learners will demonstrate skill in data management.

Pedagogy :
Presentations, Programming Simulators,Search Engines ,Videos,Online Tutorials

Module	Module Content		Duration of Module
Unit I	**Introduction to Data Science:** What is Data? Different kinds of data, Exploratory Data Analysis (EDA) ,Data Visualization, Different types of data sources, **Data Management**: Data Collection, Data cleaning/extraction, Data analysis & Modeling		**09**
Unit II	**Data Curation:** Query languages and Operations to specify and transform data, Structured/schema based systems as users and acquirers of data Semi-structured systems as users and acquirers of data, Unstructuredsystems in the acquisition and structuring of data, Security and ethicalconsiderations .		**09**
Unit III	**Data transformations**: Dimension reduction, Feature extraction, Smoothing and aggregating **Statistical Modelling and Machine Learning:** Regularization,Ridge and LASSO regressions		**09**
Unit IV	**Supervised Learning**: Classification: classification trees, Logistic regression, sseparating hyperplanes, k-NN		**09**
Unit V	**Unsupervised Learning**: Principal Components Analysis (PCA), k-means clustering, Hierarchical clustering		**09**

Textbook(s):

1) Doing Data Science, Rachel Schutt and Cathy O'Neil, O'Reilly,2013
2) Mastering Machine Learning with R, Cory Lesmeister, PACKT Publication,2015

Additional Reference(s):

1) Hands-On Programming with R, Garrett Grolemund,1st Edition, 2014
2) An Introduction to Statistical Learning, James, G., Witten, D., Hastie, T.,Tibshirani, R.,Springer,2015

Practicals:- (To be conducted using R/Python)

1. Implementation of Data collection, Data curation and management(with any NoSQL DBMS)
2. Implementation of Principal Component Analysis
3. Implementation of k-means Clustering
4. Implementation of Simple Linear Regression
5. Implementation of Logistics Regression
6. Implementation of Hypothesis testing
7. Implementation of KNN
8. Implementation of Decision Tree

Ethical Hacking

Programme : B. Sc IT (Information Technology)						**Semester VI**	
Course : Ethical Hacking						**Code : NMUBSCIT604**	
Suggested Lectures per week				**02**			
Practical Session per week (per Batch)				**01**			
Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
30	**15 X 2 = 30**	**Nil**	**02+ 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**

Internal Component (Theory Break up)	
Class Test Duration 20 Mins	**Assignments**
10 Marks	**15 Marks**
Internal Component (Practical Break up)	
Examination (Duration 1 ½ Hrs)	**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**
30 Marks	**20 Marks**

Learning Objectives

1. To make the learner understand the significance of Ethical hacking through practice.
2. To make the learner aware of different types of Malware and attacks
3. To give the learner a brief understanding of Social Engineering
4. To provide brief knowledge of hacking webservers and web application

Learning Outcomes :

1. Learner will develop an understanding of Ethical hacking and how it can be used to protect different attacks on the internet
2. The learner can Plan a vulnerability assessment and penetration test for a network.
3. The learner can report on strengths and vulnerability of the penetration testperformed

Pedagogy :
Presentations, Programming Simulators, Search Engines ,Videos, Online Tutorials

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	**Introduction to information security** : Asset, Access Control, CIA, Authentication, Authorization, Risk, Threat, Vulnerability, Attack, Attack Surface, Malware, Security-Functionality-Ease of Use Triangle **Types of malware** :Worms, viruses, Trojans, Spyware, Rootkits		06
II	**Introduction**: Black Hat vs. Gray Hat vs. White Hat (Ethical) hacking, Why is Ethical hacking needed?, How is Ethical hacking different from security auditing and digital forensics?, Signing NDA, Compliance and Regulatory concerns, Black box vs. White box vs. Black box, Vulnerability assessment and Penetration Testing. **Phases:** Footprinting and Reconnaissance, Scanning Networks, Enumeration		06
III	Social Engineering:-Technical Non Technical Techniques,preventive measures, Denial of Service :- DOS and DDOS,Tools and Techniques used,Preventive measures, Hacking Mobile Platforms,		06
IV	Hacking Webservers and Hacking Web Applications, SQL Injection Attacks, Hacking Wireless Networks,		06
V	IDS, Firewalls and Honeypots, Cryptography, Penetration Testing		06

Textbook(s):

1) Certified Ethical Hacker Study Guide v9, Sean-Philip Oriyano, Sybex; Study Guide Edition,2016
2) CEH official Certified Ethical Hacking Review Guide, Wiley India Edition, 2007

Additional Reference(s):

1) Certified Ethical Hacker: Michael Gregg, Pearson Education,1st Edition, 2013
2) Certified Ethical Hacker: Matt Walker, TMH,2011
3)Ethical Hacking Review Guide Kimberly Graves Wiley Publishing
4) Ethical Hacking Ankit Fadia 2 nd Edition Macmillan India Ltd, 2006

Practicals:-

1. Using the tools for whois, traceroute, email tracking, google hacking.

2. Demonstrating BufferOverflow Attack.

3. Demonstrating Format String Attack

4. Use Cain and Abel for cracking Windows account password using Dictionary attack and to decode wireless network passwords

5. Use NMap scanner to perform port scanning of various forms – ACK, SYN, FIN, NULL, XMAS

6. Use Wireshark (Sniffer) to capture network traffic and analyze

7. Perform SQL injection attack

8. Study of Denial of Service attack tools

9. Study of Web server attack tools

10. Using cryptanalysis tool.

Advanced Mobile Programming

Programme : B. Sc IT (Information Technology)	**Semester V**
Course : Advanced Mobile Programming	**Code : NMUBSCIT605**
Suggested Lectures per week	**02**
Practical Session per week (per Batch)	**01**

Teaching Scheme				**Evaluation Scheme**			
Lecture	**Practical**	**Tutorial**	**Credits**	**Theory**		**Practical**	
				Internal	**External**	**Component 1**	**Component 2**
30	**15 X 2 = 30**	**Nil**	**02 + 01**	**25 Marks**	**75 Marks**	**20 Marks**	**30 Marks**

Internal Component (Theory Break up)	
Class Test Duration 20 Mins	**Assignments**
10 Marks	**15 Marks**

Internal Component (Practical Break up)	
Examination (Duration 1 ½ Hrs)	**Mini Project/Case study/ Field Visit (report to be submittedand certified prior to practical examination)**
30 Marks	**20 Marks**

Learning Objectives

1. Introduction to Basic Android Programming
2. Introduction to concepts like layouts, intents, activities, fragments, dialog
3. Introduction to database, services, broadcast receivers

Learning Outcomes :

1. Knowledge of Basic Android Programming
2. Knowledge of concepts like layouts, intents, activities, fragments, dialog
3. Knowledge of database, services, broadcast receivers

Pedagogy :

PPts,Videos,Case Studies , Small project(Group wise)

Module	Module Content	Module wise Pedagogy Used	Duration of Module
I	Your First Android Application – App Basics, Creating an Android Project, UI, Widgets, Toasts, different components like button, textview,etc. Android and Model-View- Controller - Creating a New Class, MVC, Updating the view layer, Adding Icon, Listeners		6

II	The Activity Lifecycle – Logging the Activity Lifecycle, Logcat, Rotation and the Activity Lifecycle Debugging Android Apps- Exceptions and Stack Traces Your Second Activity – Starting activity, Passing data between activities		6
III	UI Fragments and the Fragment Manager- Introducing Fragments, Hosting a UI Fragment, Fragment Life cycle Displaying Lists with RecyclerView - RecyclerView, Adapter, and ViewHolder Multimedia – Audio, Video, Image		6
IV	Creating User Interfaces with Layouts and Widgets – Different layouts, More on Layout Attributes Dialogs- Creating a DialogFragment, Passing Data Between Two Fragments Background Services		6
V	Toolbars – AppCompat, menus, SQLite Databases - Defining a Schema, Initial database, Reading Updating and writing to database Broadcast Receivers		6

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Android Programming: The Big Nerd Ranch Guide	Bill Phillips, Chris Stewart and Kristin Marsicano		3	2017

List of Practical

1 Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals: Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple "Hello World" program.

2 Programming Resources Android Resources: (Color, Theme, String, Drawable, Dimension, Image),

3 Programming Activities and fragments Activity Life Cycle, Activity methods, Multiple Activities,

Life Cycle of fragments and multiple fragments.
4 Programs related to different Layouts Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.
5 Programming UI elements AppBar, Fragments, UI Components6
Programming menus, dialog, dialog fragments
7 Programs on Intents, Events, Listeners and Adapters The Android Intent Class, Using Eventsand Event Listeners
8 Programs on Services, notification and broadcast receivers9
Database Programming with SQLite
10 Programming threads, handles and asynchronized programs11
Programming Media API and Telephone API
12 Programming Security and permissions
13 Programming Network Communications and Services (JSON)14
Calculator