



Maharashtra State Board of Technical Education, Mumbai
Teaching and Examination Scheme for Post S.S.C. Diploma Courses

Program Name : Diploma in Information Technology

Program Code : IF

With Effect From Academic Year: 2017 - 18

Duration of Program : 6 Semesters

Duration : 16 Weeks

Semester : Fourth

Scheme : I

S. N.	Course Title	Course Abbreviation	Course Code	Teaching Scheme			Credit (L+T+P)	Examination Scheme														Grand Total
				L	T	P		Theory						Practical								
								Exam Duration in Hrs.	ESE		PA		Total		ESE		PA		Total			
									Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks		
1	Java Programming	JPR	22412	3	-	4	7	3	70	28	30*	00	100	40	50#	20	50	20	100	40	200	
2	Software Engineering	SEN	22413	3	-	2	5	3	70	28	30*	00	100	40	25@	10	25	10	50	20	150	
3	Database Management	DMA	22416	3	-	4	7	3	70	28	30*	00	100	40	50#	20	50	20	100	40	200	
4	Computer Network	CNE	22417	3	-	2	5	3	70	28	30*	00	100	40	25@	10	25	10	50	20	150	
5	GUI Application Development using VB.Net	GAD	22034	2	-	4	6	--	--	--	--	--	--	--	50#	20	50~	20	100	40	100	
Total				14	-	16	30	--	280	--	120	--	400	--	200	--	200	--	400	--	800	

Student Contact Hours Per Week: **30 Hrs.**

Medium of Instruction: **English**

Theory and practical periods of 60 minutes each.

Total Marks : **800**

Abbreviations: ESE- End Semester Exam, PA- Progressive Assessment, L - Lectures, T - Tutorial, P - Practical

@ Internal Assessment, # External Assessment, *# On Line Examination, ^ Computer Based Assessment

* Under the theory PA, Out of 30 marks, 10 marks are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the cognitive domain LOs required for the attainment of the COs.

~ For the courses having ONLY Practical Examination, the PA marks Practical Part - with 60% weightage and Micro-Project Part with 40% weightage

- **If Candidate not securing minimum marks for passing in the "PA" part of practical of any course of any semester then the candidate shall be declared as "Detained" for that semester.**
- **In-Plant Training during Summer vacation for minimum Six Weeks at the end of Fourth Semester (Second Year).**



Program Name : Computer Engineering Program Group
Program Code : CO/CM/IF/CW
Semester : Fourth
Course Title : Java Programming
Course Code : 22412

1. RATIONALE

Java is platform independent, open-source object oriented programming language enriched with free and open source libraries. In current industrial scenario Java has the broad industry support and is prerequisite with many allied technologies like Advanced Java, Java Server Pages, and Android Application Development. Thus, current industrial trends necessitate acquiring Java knowledge for Computer Engineering and Information Technology graduates. This course develops necessary skills in students to apply object oriented programming techniques in Java so that students will be able to develop complete applications using core Java.

2. COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

- **Develop Applications using Java.**

3. COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above mentioned competency:

- Develop programs using Object Oriented methodology in Java.
- Apply concept of inheritance for code reusability.
- Develop programs using multithreading.
- Implement Exception Handling.
- Develop programs using graphics and applet.
- Develop programs for handling I/O and file streams.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
L	T	P		Theory						Practical						
				Paper Hrs.	ESE		PA		Total		ESE		PA		Total	
Max	Min	Max	Min		Max	Min	Max	Min	Max	Min	Max	Min	Max	Min		
3	-	4	7	3	70	28	30*	00	100	40	50#	20	50	20	100	40

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.

Legends: L-Lecture; T – Tutorial/Teacher Guided Theory Practice; P -Practical; C – Credit, ESE -End Semester Examination; PA - Progressive Assessment.

5. COURSE MAP(with sample COs, PrOs, UOs, ADOs and topics)



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
6.	Develop programs to demonstrate use of 'while', 'do-while'	I	02*
7.	Develop a program for implementation of implicit type casting in Java. Part-I	I	02
8.	Develop a program for implementation of implicit type casting in Java. Part-II	I	02
9.	Develop a program for implementation of explicit type conversion in Java.		02*
10.	a) Develop a program for implementation of Constructor. b) Develop a program for implementation of multiple constructors in a class.	II	02*
11.	Develop a program for implementation of different functions of String Class. Part-I	II	02
12.	Develop a program for implementation of different functions of String Class. Part-II	II	02
13.	Develop a program for implementation of Arrays in Java.	II	02*
14.	Develop a program for implementation of Vectors in Java.	II	02*
15.	Develop a program for implementation of Wrapper Class to convert primitive into object.	II	02*
16.	Develop a program for implementation of Wrapper Class to convert object into primitive.	II	02*
17.	Develop program which implements the concept of overriding.	II	02*
18.	Develop a program for implementation of Single and Multilevel inheritance.	III	02*
19.	Develop a program for implementation of multiple inheritances.	III	02*
20.	Develop a program to import different classes in package.	III	02*
21.	Develop a program for implementation of multithreading operation Part-I	IV	02*
22.	Develop a program for implementation of multithreading operation Part-II	IV	02
23.	Develop a program for implementation of try, catch block. Part-I	IV	02
24.	Develop a program for implementation of try, catch block. Part-II	IV	02
25.	Develop a program for implementation of try, catch and finally block.	IV	02*
26.	Develop programs for implementation of throw, throws clause. Part-I	IV	02*
27.	Develop programs for implementation of throw, throws clause. Part-II	IV	02*
28.	Develop minimum two basic Applets. Display output with applet viewer and browser.	V	02*
	a) Develop a program on basic applet. b) Develop a program using control loops in applets.	V	02
29.	Write a program to create animated shape using graphics and applets. You may use following shapes: a) Lines and Rectangles. b) Circles and Ellipses.	V	02*
	c) Arcs d) Polygons with fillPolygon method.		02
30.	Develop a program to draw following shapes, graphics and applets. a) Cone b) Cylinders	V	02*



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	c) Cube		
	d) Square inside a circle e) Circle inside a square		02
31.	Develop a program for implementation of I/O stream classes.	VI	02*
32.	Develop a program for implementation of file stream classes.	VI	02*
	Total		64

Note

- i. A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicious mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.
- ii. The 'Process' and 'Product' related skills associated with each PrO are to be assessed according to a suggested sample given below:

Sr. No.	Performance Indicators	Weightage in %
1	Representation of classes and objects.	10
2	Coding.	40
4	Testing and Debugging of the Program.	30
5	Correctness of ProgramOutput.	10
6	Submission of report in time.	10
	Total	100

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

- a. Follow safety practices.
- b. Practice good housekeeping.
- c. Demonstrate working as a leader/a team member.
- d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

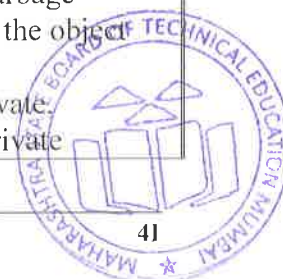


S. No.	Equipment Name with Broad Specifications	Pro S.No.
1	Computer with JDK1.8 or above	All
2	Any IDE for Java Programming such as Eclipse, Jcreator or any other product.	

8. UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I Basic Syntactical constructs in Java	1a. Write programs to create classes and objects for the given problem. 1b. Explain the characteristics of the given Java token. 1c. Explain the function of the given operator with example. 1d. Construct the expressions using implicit and explicit type conversions to solve the given problem. 1e. Develop the programs using relevant control structure to solve the given problem.	1.1 Java Features and the Java Programming Environment, Object Oriented, Compiled, Interpreted, Platform independent, Portable, Robust and Secure, Dynamic. 1.2 Defining a class, creating object, accessing class members 1.3 Java Tokens and Data types, Constants and Symbolic Constants, variables, dynamic initialization, data types, array and string, scope of variable, typecasting, and standard default values. 1.4 Operators and Expressions, Arithmetic Operators, Relational Operators, Logical Operators, Increment and Decrement, Conditional Operators, Bit wise Operators, Instance of Operators, Dot Operators, Operator precedence and associativity, Evaluation of Expressions, Type conversions in expressions, Mathematical Functions - min(),max(), sqrt(), pow(), exp(), round(), abs(). 1.5 Decision making and looping: If statement, if else statement, nested if else statement, if else if ladder, the switch statement, nested switch statement, The ?:operator, The while statement, the Do while statement, the 'for' statement, break, continue and return statement, nested loops, labeled loops, for-each version of the for loop.
Unit-II Derived Syntactical Constructs in Java	2a. Use Constructors for the given programming problem. 2b. Identify scope and lifetime of a variable in the given program code. 2c. Describe the given visibility control with example. 2d. Write the programs by implementing arrays to solve	2.1 Constructors and methods, types of constructors, nesting of methods, argument passing the 'this' keyword, command line arguments, varargs: variable-length arguments, garbage collection, finalize() method, the object class. 2.2 Visibility Control Public, Private, Protected, default, friendly private



Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	<p>the given problem.</p> <p>2e. Develop programs using vectors and wrapper classes for the given problem.</p>	<p>Protected access.</p> <p>2.3 Arrays and Strings: Types of arrays, creating an array, strings, string classes and string buffer, vectors, wrapper classes, enumerated types.</p>
Unit– III Inheritance , Interface and Package	<p>3a. Apply the identified type of inheritance for the given programming problem.</p> <p>3b. Differentiate between overloading and overriding for the given example.</p> <p>3c. Develop program using the specified interface.</p> <p>3d. Create user defined package for the given problem.</p> <p>3e. Add class and interface to the given package.</p>	<p>3.1 Inheritance: concept of inheritance Types of Inheritance</p> <p>3.2 Single Inheritance, multilevel Inheritance, Hierarchical Inheritance, method and constructor overloading and overriding, dynamic method dispatch, final variables, final methods, use of super, abstract methods and classes, static members.</p> <p>3.3 Interfaces: Define Interface, implementing interface ,accessing interface, variables and methods, extending interfaces, interface references, nested interfaces</p> <p>3.4 Package: Define package, type of package naming and creating packages, accessing package, import statement, static import, adding class and interfaces to a package.</p>
Unit– IV Exception Handling and Multithreading	<p>4a. Distinguish the errors and exceptions (if any) in the given example.</p> <p>4b. Develop program for handling the given exception.</p> <p>4c. Crate threads to run the given multiple processes in the given program.</p> <p>4d. Explain the function of the specified phase in thread life cycle using the given example.</p>	<p>4.1 Errors and Exception :Types of errors, exceptions, try and catch statement, nested try statement, throws and Finally statement, build-in exceptions, chained exceptions, creating own exception(throw clause), subclasses.</p> <p>4.2 Multithreaded Programming Creating a Thread: By extending to thread class and by implementing runnable Interface, Life cycle of thread: Thread Methods:wait(), sleep(), notify(), resume(), suspend(), stop().Thread exceptions, thread priority and methods, synchronization, inter-thread communication, deadlock.</p>
Unit –V Java Applets and Graphics Programmin g	<p>5a. Describe the given phase of applet life cycle using a typical example.</p> <p>5b. Develop programs using applet implementation for the given problem.</p> <p>5c. Develop program for implementing the given geometric shape.</p> <p>5d. Develop program for implementing the given font</p>	<p>5.1 Introduction to applets: Applet, Applet life cycle (skeleton), Applet tag, Adding Applet to HTML file, passing parameter to applet, embedding <applet> tags in java code, adding controls to applets.</p> <p>5.2 Graphics Programming: Graphics classes, lines, rectangles, ellipse, circle, arcs, polygons, color and fonts, setColor(), getColor(), setForeGround(), setBackground(), font class, variable defined by font class: name. pointSize.</p>

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	settings.	size, style, font methods: getFamily(), getFont(), getFontname (), getSize(), getStyle(), getAllFonts() and get available font family name() of the graphics environment class.
Unit –VI Managing Input /Output/ Files in Java	6a. Use I/O stream classes in a program to solve the given problem. 6b. Write programs for reading and writing character streams to and from the given files. 6c. Write programs for reading and writing bytes to and from the given files. 6d. Write program to demonstrate use of primitive Data types with the specified stream.	6.1 Introduction and Concept of Streams. 6.2 Stream Classes. 6.3 Byte Stream Classes: Input Stream Classes, Output Stream Classes. 6.4 Character Stream Classes, Using streams. 6.5 Using File Class: I/O Exceptions, Creation of Files, Reading/Writing characters, Reading/Writing Bytes, Handling Primitive Data types.

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Basic Syntactical constructs in Java	06	02	04	04	10
II	Derived Syntactical Constructs in Java	10	02	06	10	18
III	Inheritance, Interface and Package	10	02	04	06	12
IV	Exception Handling and Multithreading	08	02	04	06	12
V	Java Applets and Graphics Programming	08	02	04	04	10
VI	Managing Input/Output/Files in Java	06	02	02	04	08
Total		48	12	24	34	70

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)

Note: This specification table provides general guidelines to assist students for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:



- a. Prepare journals based on practical performed in laboratory.
- b. Follow coding standards.
- c. Develop variety of programs to improve the logical skills.
- d. Develop Application oriented real world programs.
- e. Prepare power point presentation or animation for understanding different Object Oriented Concepts.

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

- a. Massive open online courses (**MOOCs**) may be used to teach various topics/sub topics.
- b. '**L**' in item No. 4 does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
- c. About **15-20% of the topics/sub-topics** which is relatively simpler or descriptive in nature is to be given to the students for **self-directed learning** and assess the development of the COs through classroom presentations (see implementation guideline for details).
- d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- e. Use different Audio Visual media for Concept understanding.
- f. Guide student(s) in undertaking micro-projects.
- g. Demonstrate students thoroughly before they start doing the practice.
- h. Observe continuously and monitor the performance of students in Lab.

12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be **individually** undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should **not exceed three**.

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than **16 (sixteen) student engagement hours** during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

- a. Mini Banking System for handling deposits and withdrawal.
- b. Medical Store stock Management System.
- c. Library book issue Management System.
- d. Bus Reservation System.
- e. Attendance Management System.
- f. Develop a small animation using applet, graphics and multithreading.

GUIDELINES FOR DEVELOPING MICRO PROJECTS:

- i. Declare four to five classes and may include Interfaces if required.
- ii. Must use Most of the Object Oriented Concepts.



- iii. Must implement concepts of Inheritance and Exception Handling.
- iv. Must Create Own Package.
- v. May use the constructor overloading and overriding.
- vi. May Use Multithreading if required.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1.	Programming with JAVA	Balagurusamy E.	Mcgraw Hill Education (India) Private Limited, New Delhi, 5th Edition ISBN-13: 978-93-5134-320-2
2.	Java 8 Programming Black Book	DT Editorial Services	Dreamtech Press, New Delhi, ISBN:978-93-5119-758-4
3.	Java Complete Reference	Schildt Herbert	Mcgraw Hill Education, New Delhi ISBN:9789339212094
4.	Advanced Java Programming	Roy Uttam K	Oxford University Press, New Delhi ISBN :0-19-945550-3
5.	Jawa Programming	Dr. Rajendra Kawale	Devraj Publication

14. SUGGESTED SOFTWARE/LEARNING WEBSITES

- a. <https://docs.oracle.com/javase/8/docs/>
- b. http://www.nptelvideos.com/java/java_video_lectures_tutorials.php
- c. <http://nptel.ac.in/courses/106105084/25>
- d. <http://www.iitk.ac.in/esc101/08Jul/notes.html>



Program Name : Computer Engineering Program Group
Program Code : CO/CM/IF/CW
Semester : Fourth
Course Title : Software Engineering
Course Code : 22413

1. RATIONALE

Software Engineering is the foundation for professional processes to be followed involving principles, techniques, and practices for software development. The course provides a framework for software professionals for building quality assured software products. It enables students to blend the domain specific knowledge with the programming skills to create quality software products.

2. COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

- Use relevant software process model for developing software products.

3. COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above-mentioned competency:

- Select suitable Software Process model for software development.
- Prepare software requirement specifications.
- Use Software modeling to create data designs.
- Estimate size and cost of software product.
- Apply project management and quality assurance principles in software development.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
L	T	P		Theory						Practical						
				Paper Hrs.	ESE		PA		Total		ESE		PA		Total	
					Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
3	-	2	5	3	70	28	30*	00	100	40	25@	10	25	10	50	20

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.

Legends: L-Lecture; T – Tutorial/Teacher Guided Theory Practice; P - Practical; C – Credit, ESE - End Semester Examination; PA - Progressive Assessment

5. COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the course. in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.



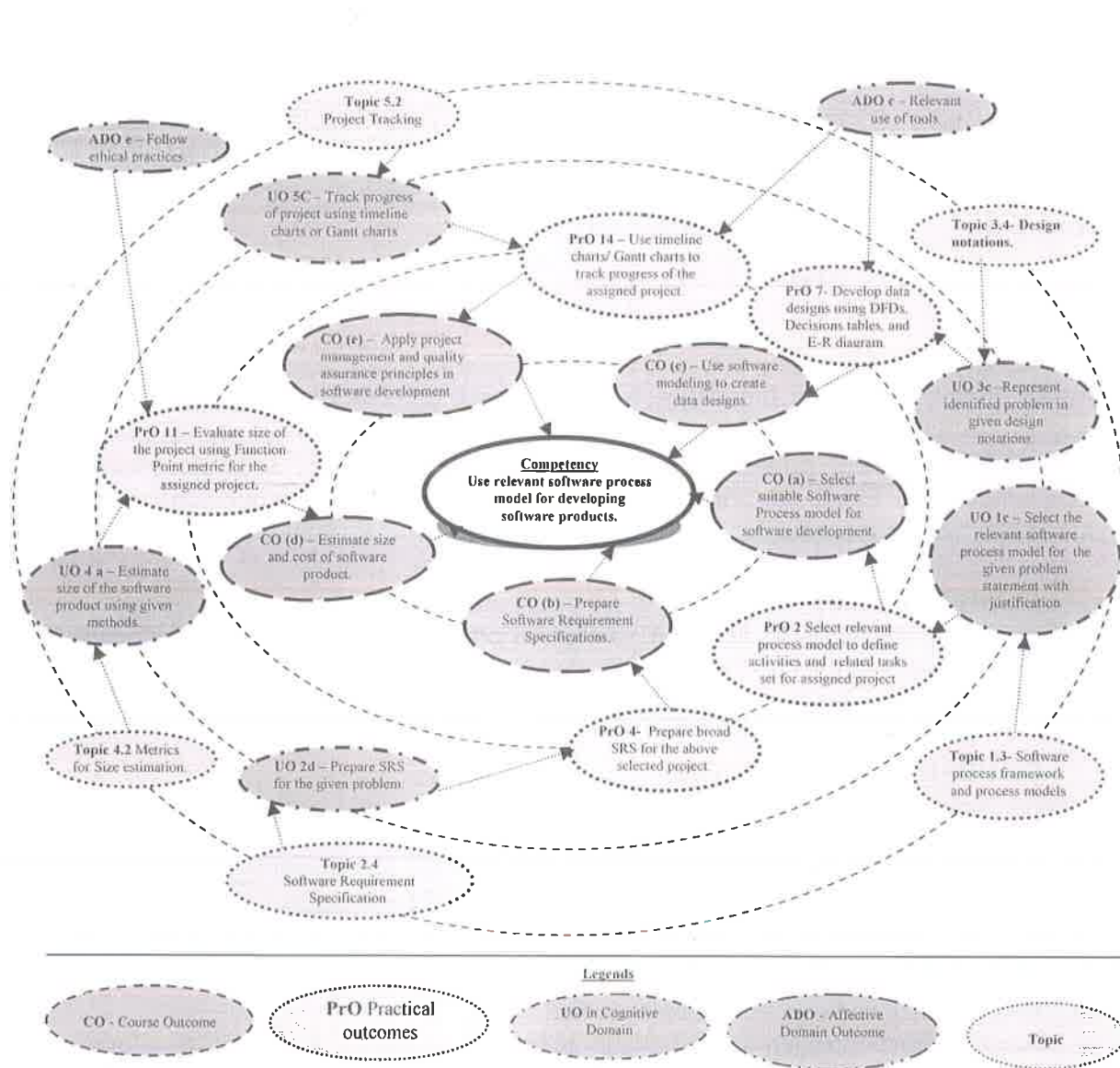


Figure 1 - Course Map

6. SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1	Write problem statement to define the project title with bounded scope of the project.	I	02*
2	Select relevant process model to define activities and related tasks set for assigned project.	I	02*
3	Gather application specific requirements for assimilate into RE (Requirements engineering) model.	II	02*
4	Prepare broad SRS (software requirement software) for the above selected project.	II	02*
5	Prepare use-cases and draw use – case diagram using Software Modeling Tool.	II	02*



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
6	Develop the activity diagram to represent flow from one activity to another for software development.	II	02
7	Develop data designs using DFDs (data flow diagram), Decision tables and E-R (entity-relationship) diagram.	III	02*
8	Draw class diagram, Sequence diagram, Collaboration diagram, State Transition Diagram for the assigned project.	III	02
9	Write test cases to validate requirements of assigned project from SRS document.	III	02*
10	Identify risks involved in the project and prepare RMMM (RMMM-Risk Management, Mitigation and Monitoring) plan.	IV	02
11	Evaluate size of the project using Function point metric for the assigned project.	IV	02*
12	Estimate cost of the project using COCOMO (Constructive Cost Model) / COCOMO II approach for the assigned project.	IV	02*
13	Use CPM (Critical Path Method) / PERT (Programme Evaluation and Review Technique) for scheduling the assigned project.	V	02*
14	Use Timeline charts/ Gantt charts to track progress of the assigned project.	V	02
15	Prepare SQA plan that facilitates various attributes of quality of process.	V	02*
16	Prepare SQA plan that facilitates various attributes of quality of product.	V	02*
Total			32

Note

- i. To carry out above listed practical /tasks, relevant software tool may be chosen (preferably open-source based).
- ii. A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicious mix of minimum 12 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.
- iii. The 'Process' and 'Product' related skills associated with each PrO is to be assessed according to a suggested sample given below:

S.No.	Performance Indicators	Weightage in %
1	Problem selection and its feasibility study	20
2	Logical thinking to decompose problem into modules	30
3	Ability to Estimate size and cost of a software	30
4	Presentation and technical documentation skills	10
5	Submission of reports within time.	10
Total		100

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

- a. Follow safety practices.
- b. Practice good housekeeping.
- c. Demonstrate working as a leader/a team member.



d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	Pro. S. No.
1	Hardware: Personal computer. (i3-i5 preferable), RAM minimum 2 GB	For all Experiments
2	Operating system: Windows 7/Windows 8/Windows 10/LINUX or any other .	
3	Software tools: Any UML tool	

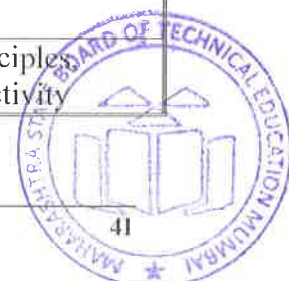
8. UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I Software Development Process	1a. Suggest the attributes that match with standards for the given software application. 1b. Recommend the relevant software solution for the given problem with justification. 1c. Select the relevant software process model for the given problem statement with justification. 1d. Suggest the relevant activities in Agile Development Process in the given situation with justification	1.1 Software, Software Engineering as layered approach and its characteristics, Types of software. 1.2 Software development framework. 1.3 Software Process Framework, Process models: Perspective Process Models, Specialized Process Models. 1.4 Agile Software development: Agile Process and its importance, Extreme Programming, Adaptive Software Development, Scrum, Dynamic Systems Development Method (DSDM), Crystal. 1.5 Selection criteria for software process model.
Unit– II Software Requirement Engineering	2a. Apply the principles of software engineering for the given problem. 2b. Choose the relevant	2.1 Software Engineering Practices and its importance, Core principles. 2.2 Communication Practices, Planning Practices, Modelling practices.



Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	<p>'requirement engineering' steps in the given problem.</p> <p>2c. Represent the 'requirement engineering' model in the given problem.</p> <p>2d. Prepare SRS for the given problem.</p>	<p>construction practices, software deployment (Statement and meaning of each principle for each practice).</p> <p>2.3 Requirement Engineering: Requirement Gathering and Analysis, Types of requirements (Functional, Product, organizational, External Requirements), Eliciting Requirements, Developing Use-cases, Building requirement models, Requirement Negotiation, Validation.</p> <p>2.4 Software Requirement Specification: Need of SRS, Format, and its Characteristics.</p>
Unit- III Software Modelling and Design	<p>3a. Identify the elements of analysis model for the given software requirements.</p> <p>3b. Apply the specified design feature for software requirements modeling.</p> <p>3c. Represent the specified problem in the given design notation.</p> <p>3d. Explain the given characteristics of software testing.</p> <p>3e. Prepare test cases for the given module.</p>	<p>3.1 Translating Requirement model into design model: Data Modelling.</p> <p>3.2 Analysis Modelling: Elements of Analysis model.</p> <p>3.3 Design modelling: Fundamental Design Concepts (Abstraction, Information hiding, Structure, Modularity, Concurrency, Verification, Aesthetics).</p> <p>3.4 Design notations: Data Flow Diagram (DFD), Structured Flowcharts, Decision Tables.</p> <p>3.5 Testing – Meaning and purpose, testing methods - Black-box and White-box, Level of testing – Unit testing.</p> <p>3.6 Test Documentation – Test Case Template, test plan, Introduction to defect report, test summary report.</p>
Unit-IV Software Project Estimation	<p>4a. Estimate the size of the software product using the given method.</p> <p>4b. Estimate the cost of the software product using the given empirical method.</p> <p>4c. Evaluate the size of the given software using CoCoMo model.</p> <p>4d. Apply the RMMM strategy in Identified risks for the given software development problem.</p>	<p>4.1 The management spectrum – 4P's</p> <p>4.2 Metrics for Size Estimation: Line of Code(LoC), Function Points(FP).</p> <p>4.3 Project Cost Estimation Approaches: Overview of Heuristic, Analytical, and Empirical Estimation.</p> <p>4.4 COCOMO (Constructive Cost Model), COCOMO II.</p> <p>4.5 Risk Management: Risk Identification, Risk Assessment, Risk Containment, RMMM strategy.</p>
Unit –V Software	<p>5a. Use the given scheduling technique for the</p>	<p>5.1 Project Scheduling: Basic principles, Work breakdown structure, Activity</p>



Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
quality assurance and Security	identified project. 5b. Draw the activity network for the given task. 5c. Prepare the timeline chart/ Gantt chart to track progress of the given project. 5d. Describe the given Software Quality Assurance (SQA) activity. 5e. Describe features of the given software quality evaluation standard.	network and critical path Method, Scheduling techniques (CPM, PERT). 5.2 Project Tracking: Timeline charts, Earned Value Analysis, Gantt Charts 5.3 Software Quality Management vs. Software Quality Assurance. Phases of Software Quality Assurance: Planning, Activities, audit, and review 5.4 Quality Evaluation standards: Six Sigma, ISO for software, CMMI: Levels, Process areas. 5.5 Software Security, Introduction to DevOps, Secure software engineering

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Software development process	08	04	04	04	12
II	Software Requirement Engineering	10	02	04	08	14
III	Software Modelling and Design	10	-	04	10	14
IV	Software Project Estimation	10	04	04	08	16
V	Software Project Management and quality assurance	10	04	04	06	14
Total		48	14	20	36	70

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)

Note: This specification table provides general guidelines to assist student for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

This specification table also provides a general guideline for teachers to frame internal end semester practical theory exam paper which students have to undertake.

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

- Prepare journals based on practical performed in laboratory.
- Give seminar on relevant topics.
- Study and analyze college website from perspective of software application.
- Study and analyze any available application software from perspective of software engineering.



11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

- a. Massive open online courses (**MOOCs**) may be used to teach various topics/sub topics.
- b. '**L**' in item No. 4 does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
- c. About **15-20% of the topics/sub-topics** which is relatively simpler or descriptive in nature is to be given to the students for **self-directed learning** and assess the development of the COs through classroom presentations (see implementation guideline for details).
- d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- e. Guide student(s) in undertaking micro-projects.
- f. Demonstrate students thoroughly before they start doing the practice.
- g. Encourage students to refer different websites to have deeper understanding of the subject.
- h. Observe continuously and monitor the performance of students.

12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be **individually** undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should **not exceed three**.

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than **16 (sixteen) student engagement hours** during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

- a. Study and analyze given software and write the characteristics and functions of the same.
- b. Case study of application specific software product for requirement engineering
 - i. Identify the problem statement
 - ii. Perform feasibility analysis
 - iii. Identify application specific requirements by following RE steps
 - iv. Prepare SRS
- c. Choose any problem statement and use data models to represent the solution
 - i. Search and utilize different UML tools to represent models
- d. Choose a problem, create activity network and use different project scheduling and tracking tools for the same.

13. SUGGESTED LEARNING RESOURCES



S. No.	Title of Book	Author	Publication
1	Software Engineering: A practitioner's approach	Pressman, Roger S.	McGraw Hill Higher Education, New Delhi, (Seventh Edition) ISBN 978-0-07-337597-7
2	Software Engineering Concepts	Fairly, Richard	McGraw Hill Education New Delhi – 2001, ISBN-13: 9780074631218
3	Software Engineering: Principles and practices	Jain, Deepak	Oxford University Press, New Delhi ISBN 9780195694840

14. SUGGESTED SOFTWARE/LEARNING WEBSITES

- a. <http://www.rspa.com/spi/>
- b. www.tutorialspoint.com/software_engineering/
- c. www.versionone.com/agile-101/
- d. www.sei.cmu.edu
- e. www.nptel.ac.in/courses/
- f. <https://techbeacon.com/secure-devops>



Program Name : Diploma in Information Technology
Program Code : IF
Semester : Fourth
Course Title : Database Management
Course Code : 22416

1. RATIONALE

Managing data as a resource is an important function of information management. Organizations need to maintain huge amount of data in effective manner. In software industry the relational database management system is being used predominantly to manage the data stored in the database. This subject enables students to create, store, modify data and extract information from a database using SQL. After learning this subject student will be able to use RDBMS as a backend for developing database applications.

2. COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

- Apply Database Management concepts using SQL.

3. COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above mentioned competency:

- Create Database using SQL commands
- Manage Database using SQL commands.
- Implement Advanced SQL concepts on Database.
- Write PL/SQL code for database application.
- Apply security and safety on Database.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
L	T	P		Theory						Practical						
				Paper Hrs.	ESE		PA		Total		ESE		PA		Total	
			Max		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	
3	-	4	7	3	70	28	30*	00	100	40	50#	20	50	20	100	40

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.

Legends: L-Lecture; T – Tutorial/Teacher Guided Theory Practice; P - Practical; C – Credit, ESE - End Semester Examination; PA - Progressive Assessment.

5. COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the



course, in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.

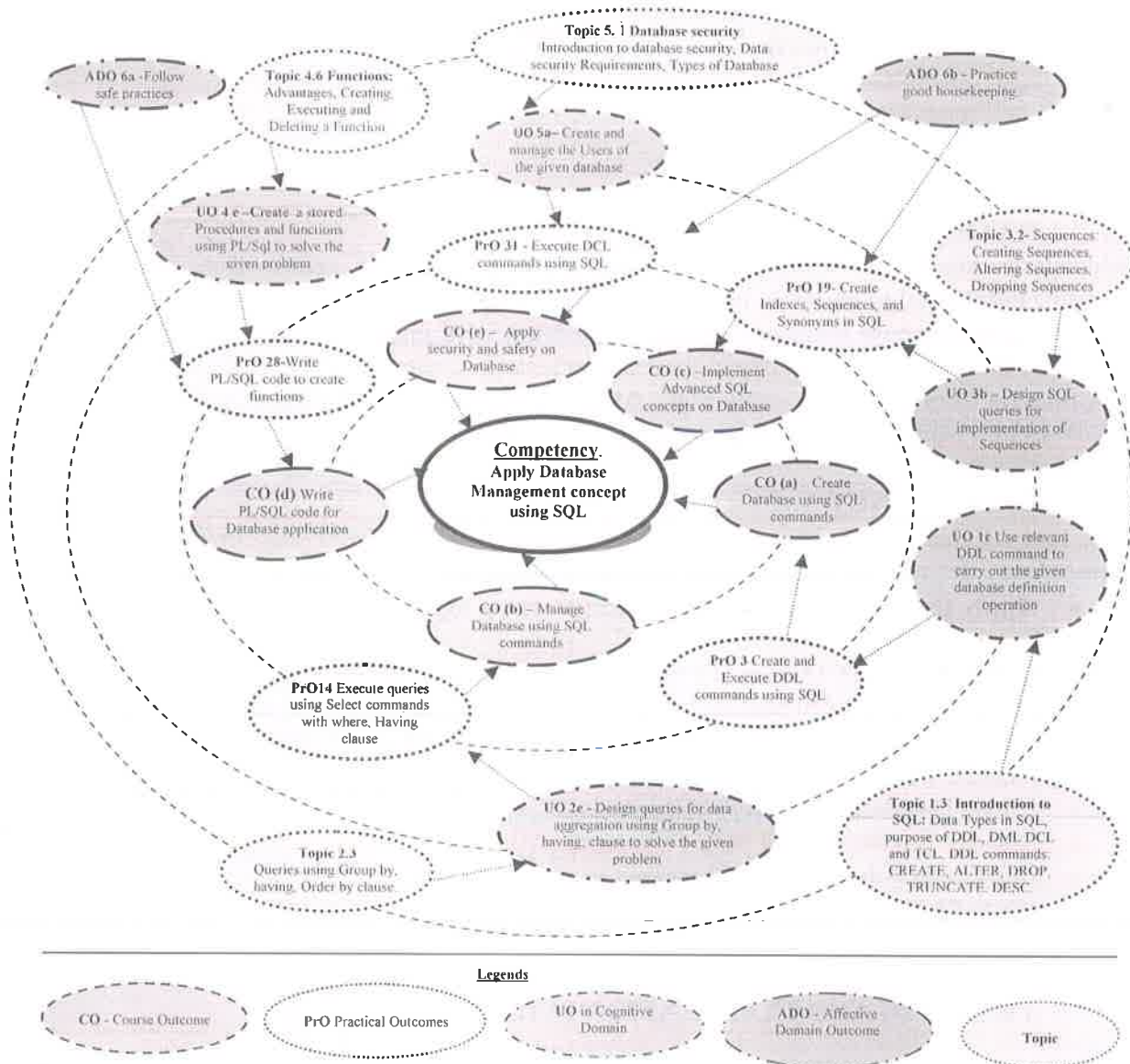


Figure 1 - Course Map

6. SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1.	Install and configure Database product (such as MySQL, ORACLE or any other relational database product)	I	02*
2.	Write Queries using Relational Algebraic operations.	I	02*
3.	Create and Execute DDL commands using SQL Part-I.	I	02*
4.	Create and Execute DDL commands using SQL Part-II.	I	02*
5.	Apply following Integrity constraints on table: Primary key, Foreign key, Unique key constraint	I	02*



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
6.	Apply following Integrity constraints on table: Null, Not Null and Check constraint.	I	02
7.	Create and Execute DML commands using SQL Part-I.	II	02*
8.	Create and Execute DML commands using SQL Part-II.		02
9.	Write Queries using following operators: Arithmetic Operators, Comparison Operators, Logical Operators.	II	02*
10.	Write Queries using following operators: Set Operators, Range Searching operators-Between, Pattern matching operators-Like.	II	02*
11.	Write Queries using String, Arithmetic Functions.	II	02*
12.	Write Queries using Date and time Functions.	II	02*
13.	Write Queries using Aggregate Functions- Min, Max, Sum, Avg, Count.	II	02*
14.	Execute Queries using the Select command with Where, Having clauses.	II	02*
15.	Execute Queries using the Select command with Group by and order by clauses.	II	02*
16.	Execute queries using Inner Joins.	II	02*
17.	Execute queries using Outer Joins.	II	02*
18.	Implement Views to perform following operations: a. Create views. b. Insert, modify and delete records through views. c. Delete the views.	III	02*
19.	Create Indexes, Sequences, and Synonyms in SQL.	III	02*
20.	Execute Views, Indexes, Sequences and synonyms in SQL.	III	02*
21.	Write a PL/SQL programs using if then else, for.	IV	02*
22.	Write a PL/SQL programs using while and nested loop.	IV	02
23.	Write a PL/SQL code to implement implicit cursors.	IV	02*
24.	Write a PL/SQL code to implement explicit cursors.	IV	02
25.	Write PL/SQL Programs based on Exceptions handling. (Predefined exceptions).	IV	02*
26.	Write PL/SQL Programs based on Exceptions handling. (user-defined exceptions)	IV	02*
27.	Write PL/SQL code to create Procedures.	IV	02*
28.	Write PL/SQL code to create functions.	IV	02*
29.	Write PL/SQL code to create triggers on given database Part-I.	IV	02*
30.	Write PL/SQL code to create triggers on given database Part-II.	IV	02
31.	Executing DCL commands using SQL a. Create users b. Grant privileges to users c. Revoke privileges from users.	V	04*
Total			64

Note

i. A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicious mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student



reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.

ii. The 'Process' and 'Product' related skills associated with each PrO are to be assessed according to a suggested sample given below:

S. No.	Performance Indicators	Weightage in %
1	Coding of SQL queries and PL/SQL programming	60
2	Database Integrity	10
3	Quality of result displayed by SQL queries/ PL/SQL programming.	10
4	Answer to sample questions	10
5	Submit report in time	10
Total		100

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

- a. Follow safety practices.
- b. Practice good housekeeping.
- c. Demonstrate working as a leader/a team member.
- d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	PrO. S. No.
1.1	Computer system (Any computer system with basic configuration)	All
1.2	Any RDBMS software (MySQL/SQL server or any other)	All

8. UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I Creating	1a. Describe the given RDBMS term with example.	1.1 Relational Database Design. Concept of Relational Database



Relational Database	<p>1b. Design SQL queries to create the specified Relational database</p> <p>1c. Use relevant DDL command to carry out the given database definition operation.</p> <p>1d. Categorise the given SQL command in DDL, DML, DCL.</p> <p>1e. Apply the given data constraint for data integration.</p> <p>1f. Describe the procedure to create the given database using SQL commands.</p>	<p>Design, different types of RDBMS software, different relational algebra operator.</p> <p>1.2 RDBMS Terminology –Relation, Attribute, Domain, Tuple, cardinality, Degree.</p> <p>1.3 Introduction to SQL: Data Types in SQL, purpose of DDL, DML DCL and TCL. DDL commands: CREATE, ALTER, DROP, TRUNCATE, DESC, RENAME, Use of data constraints</p>
Unit– II Interactive SQL for data extraction	<p>2a Design relevant query for extracting the data based on the given criteria.</p> <p>2b Use ‘Like’ clause in SQL query with the given matching Pattern.</p> <p>2c Use the given built-in function in SQL query.</p> <p>2d Design queries using the given aggregate function.</p> <p>2e Design queries for data aggregation using Group by, having clause to solve the given problem.</p> <p>2f Describe the procedure to manage the given database using SQL commands.</p>	<p>2.1 DML commands: INSERT, UPDATE, DELETE, SELECT SQL operators: arithmetic operators, comparison operators, Logical operators, Set operators, Range Searching operators- Between, Pattern matching operators- Like</p> <p>2.2 In built Functions: String, Arithmetic, ate and time, Aggregate Functions and Miscellaneous Functions. Conversion Functions, Special Date formats using To char () function.</p> <p>2.3 Queries using Group by, having, Order by clause.</p> <p>2.4 Joins, Types of Joins, Sub queries.</p> <p>2.5 TCL: Commit, Save point, Rollback, set transaction</p>
Unit III- Advance features of SQL	<p>3a. Design SQL queries for implementing the given View/ sub query.</p> <p>3b. Design SQL queries for implementation of the given Sequences.</p> <p>3c. Create Indexes using SQL Query to solve the given problem.</p> <p>3d. Describe the procedure to create Synonym using SQL Query to solve the given problem.</p>	<p>3.1 Views: Concept of View, The Create View Command, Updating Views, Views and Joins, Views and Sub queries, Dropping Views.</p> <p>3.2 Sequences: Creating Sequences, Altering Sequences, Dropping Sequences.</p> <p>3.3 Indexes: Index Types, Creating of an Index: Simple Unique, and Composite Index, Dropping Indexes</p> <p>3.4 Synonyms: Creating Synonyms, Dropping Synonyms.</p>



Unit-IV PL/SQL Programming	<p>4a. Describe the use of the given component of PL/SQL.</p> <p>4b. Use the relevant control structure in PL/SQL Code to solve the given conditional/iterative problem.</p> <p>4c. Write the PL/SQL Code to handle the given exception.</p> <p>4d. Write the PL/SQL codes to create cursor for retrieving multiple records for the given problem.</p> <p>4e. Create a stored Procedures and Functions using the PL/SQL to solve the given problem.</p> <p>4f. Write PL/SQL code for the given database application .</p>	<p>4.1 Introduction of PL/SQL, Advantages of PL/SQL, The PL/SQL Block Structure, PL/SQL execution environment, PL/SQL data Types, Variables, Constants.</p> <p>4.2 Control Structure: Conditional Control, Iterative Control, Sequential Control.</p> <p>4.3 Exception handling: Predefined Exception, User defined Exception.</p> <p>4.4 Cursors: Implicit and Explicit Cursors, Declaring, Opening and Closing a Cursor, Fetching a Record from Cursor, Cursor for loops, Parameterized Cursors.</p> <p>4.5 Procedures: Advantages, Creating, Executing and Deleting a Stored Procedure.</p> <p>4.6 Functions: Advantages, Creating, Executing and Deleting a Function.</p> <p>4.7 Database Triggers: Use of Database Triggers, apply database Triggers, Types of Triggers, Syntax for Creating Trigger, Deleting Trigger.</p>
Unit –V Database security and Transaction Processing	<p>5a. Describe the procedure to Create the users of the given database.</p> <p>5b. Describe the procedure to Provide security to database by assigning/revoking the given privilege to the user.</p> <p>5c. Explain the given ACID property of transactions in database.</p> <p>5d. Describe the procedure to Carry out backup of the given Database by following the standard process.</p>	<p>5.1 Database security: Introduction to database security, Data security Requirements, Types of Database Users, Creating, altering and Deleting Users.</p> <p>5.2 Protecting the data within database- Database Privileges: Systems privileges and object Privileges, Granting and Revoking Privileges: Grant and Revoke command.</p> <p>5.3 Transaction and Concurrency Control: Concept, Properties and States of Transaction, Concurrency issues, need for transactions, Necessary properties of transactions (ACID properties), Transaction states, Lock-Based Concurrency Control</p> <p>5.4 Database Backup -Types of Failures, Causes of failures, Database Backup</p>

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.



9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Creating Relational Database	4	02	02	04	08
II	Interactive SQL for data extraction	12	02	04	12	18
III	Advance features of SQL	12	02	04	10	16
IV	PL/SQL Programming	12	02	04	10	16
V	Database security and Transaction Processing	08	04	04	04	12
Total		48	12	18	40	70

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)

Note: This specification table provides general guidelines to assist student for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related **co-curricular** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

- Prepare journals based on practical performed in laboratory.
- Give seminar on relevant topic.
- Undertake micro-projects.

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various learning outcomes in this course:

- Massive open online courses (**MOOCs**) may be used to teach various topics/sub topics.
- '**L**' in item No. 4 does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
- About **15-20% of the topics/sub-topics** which is relatively simpler or descriptive in nature is to be given to the students for **self-directed learning** and assess the development of the COs through classroom presentations (see implementation guideline for details).
- With respect to item No.10, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- Guide student(s) in undertaking micro-projects.
- Demonstrate students thoroughly before they start doing the practice.
- Encourage students to refer different websites to have deeper understanding of the subject.
- Observe continuously and monitor the performance of students in Lab.
- Demonstrate students thoroughly before they start doing the practice.
- Encourage students to refer different websites to have deeper understanding of the subject.



12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be **individually** undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should **not exceed three**.

The micro-project could be industry application based, internet-based, workshop-based, laboratory based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than **16 (sixteen) student engagement hours** during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

- For Hospital Management: Patient data base/Doctor database/Billing (any one database)
- College Admission: Student personal Information System/Merit list database(any one data base)
- Medical Purchase: Database of medicine inventory records.
- Library Management: book issue /book stock database.
- Any other micro-projects suggested by subject faculty on similar line.

13. SUGGESTED LEARNING RESOURCES :

S. No.	Title of Book	Author	Publication
1	Introduction to Database Management Systems	ISRD Group	McGraw Hill Education, 2005, New Delhi, ISBN-13:9780070591196
2	Database System Concepts	Korth, Henery Abraham, Silberschatz Sudarshan ,S	McGraw Hill Education, New Delhi, 2013, ISBN-13: 978-9332901384
3	Complete Reference: Mysql	Vaswani Vikram	McGraw Hill Education, New Delhi, ISBN-13: 9780070586840
4	SQL, PL/SQL The Programming Language of ORACLE	Bayross, Ivan	BPB Publications, New Delhi ^{3rd} Edition. ISBN-13: 978-8176569644
5	Database Management	Leon Alexis & Leon Mathews	Vikas Publishing, New Delhi ISBN-13:9788182092228

14. SUGGESTED SOFTWARE/LEARNING WEBSITES

- www.tutorialpoint.com (Important website)
- wielyIndia.com or DreamtechPress.com
- <http://plindia.com/gupta/chapter/chapter1.pdf>
- www.williamstannings.com



Program Name : Diploma in Information Technology
Program Code : IF
Semester : Fourth
Course Title : Computer Networks
Course Code : 22417

1. RATIONALE

To utilize the potential of Information and communication technology fully, all workplaces contain computers, and they are invariably interconnected. Apart from at home and in the office, computer networks are present in all domains of work. This demands increased need of skilled man power and well trained work force to create and maintain computer networks. This course introduce basic concept of networks, network classification, network topologies, network devices, Network communication models, concept of TCP/IP protocols, IP addressing to help the students to setup computer network and apply current Computer Network technology to maintain it.

2. COMPETENCY

The aim of this course is to help the student to attain the following *industry identified* competency through various teaching learning experiences:

- Setup different types of computer networks.

3. COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry oriented COs associated with the above mentioned competency:

- Use basic concepts of networking for setting-up computer networks.
- Setup a computer network for specific requirements.
- Configure basic network services.
- Configure the different TCP/IP services.
- Implement subnetting for improved network address management.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme				Credit (L+T+P)	Examination Scheme											
L	T	P	Theory						Practical							
			Paper Hrs.		ESE		PA		Total		ESE		PA		Total	
Max	Min	Max		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min		
3	-	2	5	3	70	28	30*	00	100	40	25@	10	25	10	50	20

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.

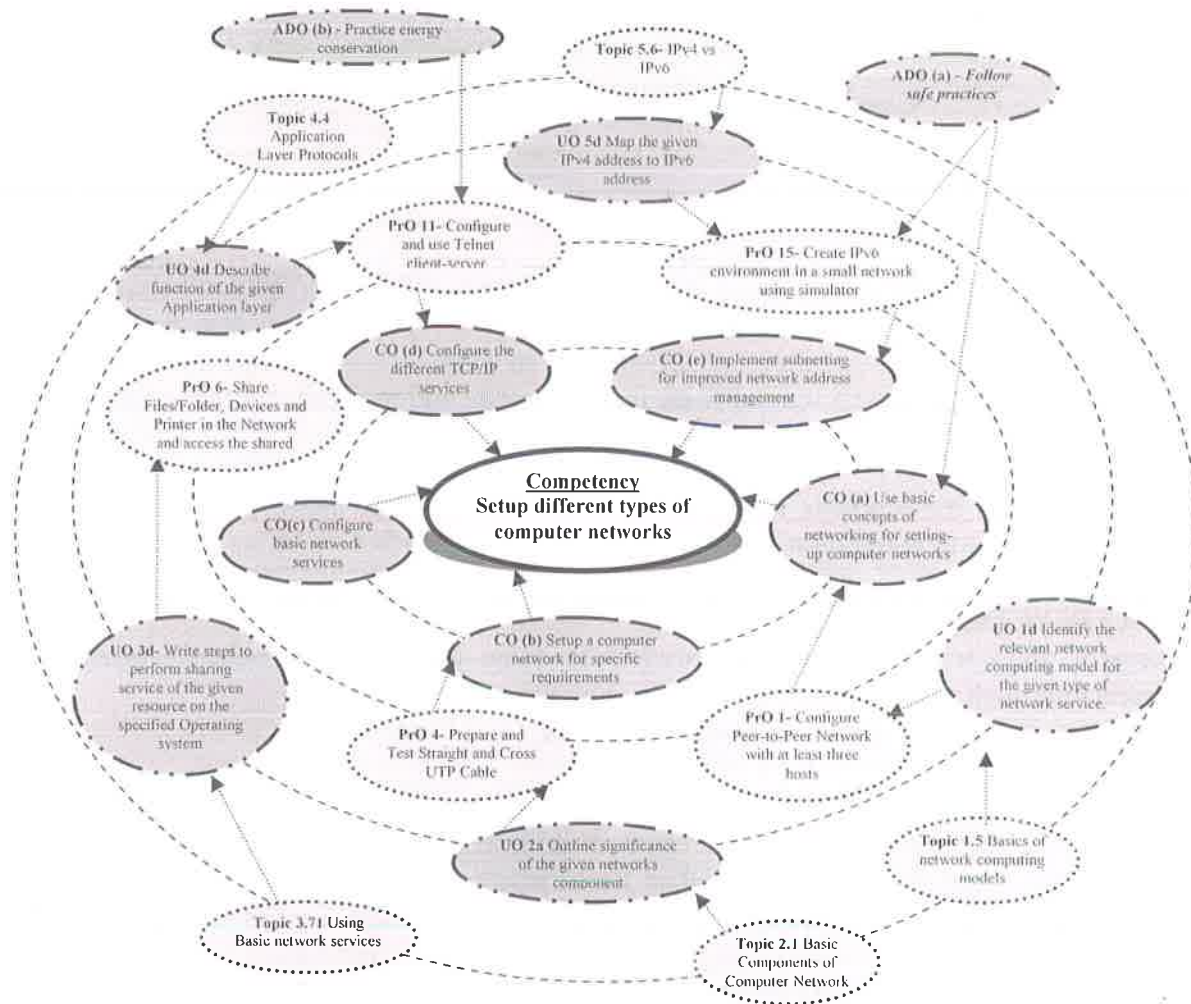
Legends: L-Lecture; T- Tutorial/Teacher Guided Theory Practice; P - Practical; C – Credit, ESE - End Semester Examination; PA - Progressive Assessment.

5. COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the



course, in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.



Legends



Figure 1 - Course Map

6. SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1.	Configure Peer-to-Peer Network with at least three hosts.	I	02*
2.	Create a small physical network using computers, Network connecting device and cables.	II	02*
3.	Draw the network layout with its topology for network set-up of your Laboratory .	II	02
4.	Prepare and Test Straight and Cross UTP Cable.	II	02*
5.	Install and configure Network Interface Card and identify its MAC Address.	II	02*



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
6.	Share Files/Folder, Devices and Printer in the Network and access the shared resource from the other node.	III	02*
7.	Run the following TCP/IP Commands with Options and record their output: Ping, ipconfig, Tracert, Netstat, Wireshark, ARP	III	02*
8.	Use Wireshark packet sniffer software and capture TCP, UDP, IP, ARP, ICMP, Telnet, FTP packets.	IV	02
9.	Setup FTP Client/ Server and Transfer the file using FTP.	IV	02*
10.	Install TCP/IP Protocols and configure Advanced features of TCP/IP Protocols like IP address, subnetmask, gateway, primary and secondary DNS.	IV	02*
11.	Configure and use Telnet client-server.	IV	02
12.	Configure and work with Remote desktop application available with Operating System.	IV	02
13.	Configure DHCP server.	IV	02*
14.	Create two subnets and implement it with calculated subnet masking.	IV	02*
15.	Create IPv6 environment in a small network using simulator (preferably open source based) Part-I	V	02*
16.	Create IPv6 environment in a small network using simulator (preferably open source based) Part-II	V	02*
Total			32

Note

- i. A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicious mix of minimum 12 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.
- ii. The 'Process' and 'Product' related skills associated with each PrO are to be assessed according to a suggested sample given below:

S. No.	Performance Indicators	Weightage in %
1	Completion of given task.	25
2	Correctness of the given task.	50
3	Answer to sample questions.	15
4	Submit report in time.	10
Total		100

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

- a. Follow safety practices.
- b. Practice good housekeeping.
- c. Demonstrate working as a leader/a team member.
- d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs



according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	PrO. No.
1	Networked computers.	All
2	UTP Cable CAT 6, Crimping Tool, Punch Tool, Network Cable Tester, RJ45 Connectors and switch with minimum 8 ports.	02
3	Network Interface Card	05
4	Wireshark packet sniffer software.	10

8. UNDERPINNING THEORY COMPONENTS

The following topics/subtopics should be taught and assessed in order to develop UOs in cognitive domain for achieving the COs to attain the identified competency:

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I Fundamentals of Computer Network	1a. Outline benefits of the given computer network application. 1b. Classify Computer Network based on the given criteria. 1c. Describe the steps to share the given network resource on the specified Operating System . 1d. Select the relevant network computing model for the given type of network service with justification. 1e. Elaborate the given feature of NOS.	1.1 Needs, uses of Computer Network, Applications of Computer Network, Advantages/Benefits of Computer Network: Sharing of Information, Sharing Resources, Centralized management of resources, Backing up of data 1.2 Classification of Networks: Geographical Classification, Classification Based on Transmission Technology, Classification Based on Network Relationships 1.3 Basics of network computing models: per-to peer, client sever, distributed 1.4 Network Operating System (NOS): its types, features.
Unit-II Network Components and Topologies	2a. Outline the significance of the given networks component. 2b. Explain the function of the given network connecting device. 2c. Explain merits/demerits of the specified type of network topology. 2d. Write the cable/connector/ connecting device requirement to	2.1 Basic Components of Computer Network : cables, Host, Communication Subnet, NIC. 2.2 Network Devices and their role : Repeaters. Hub. Bridge. Switches. Router, and Gateway. 2.3 Wireless Infrastructure components : Access point. Clients.



Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	<p>implement the given network topology.</p> <p>2e. Describe the procedure to set up the computer network for the given requirement</p>	<p>2.4 Network Topologies: Concept, Significance, Bus, Star, Ring, Tree, Mesh, Hybrid.</p>
Unit– III Reference Models for Computer Networks.	<p>3a. Describe the significance of the given design issue for layering in Protocol.</p> <p>3b. Explain major functions of the given layer of OSI Reference Model.</p> <p>3c. Explain major functions of the given layer of TCP/IP protocol suit.</p> <p>3d. Write steps to perform sharing service of the given resource on the specified Operating system.</p> <p>3e. Describe the procedure to configure the given type of computer network.</p>	<p>3.1 Protocol Hierarchies- Layered Approach</p> <p>3.2 Interfaces, Services, Protocols and Packets</p> <p>3.3 Design issues for layering.</p> <p>3.4 OSI reference Model: Layers and their functions.</p> <p>3.5 TCP/IP Protocol: Layers and their functions</p> <p>3.6 OSI Model Vs. TCP/IP</p> <p>3.7 Using Basic network services: File sharing, Print sharing, Device sharing</p>
Unit– IV TCP/IP Protocol Suite	<p>4a. Explain the salient features of the given Host-to-Network Layer Protocol.</p> <p>4b. Explain the working of the given internet layer protocol.</p> <p>4c. Identify the use of TCP or UDP at transport layer, based on functioning of the given application layer protocol.</p> <p>4d. Describe function of the given Application layer protocol.</p> <p>4e. Describe the procedure to configure the given type of TCP/IP services.</p>	<p>4.1 Host-to-Network Layer Protocols: SLIP, PPP</p> <p>4.2 Internet Layer Protocols: IP, ARP, RARP, ICMP.</p> <p>4.3 Transport Layer Protocols: TCP, UDP.</p> <p>4.4 Application Layer Protocols: FTP, HTTP, SMTP, TELNET, DNS, BOOTP, DHCP</p>
Unit –V IP Addressing	<p>5a. Calculate address range of hosts on each subnet for the given IP address.</p> <p>5b. Calculate subnet mask for the given subnet.</p> <p>5c. Illustrate the method of multicast operation using given IP address.</p> <p>5d. Map the given IPv4 address to IPv6 address</p>	<p>5.1 Addressing: Physical Address, Logical Address, Prot Address,</p> <p>5.2 IP Address: Concept, Notation, Address Space</p> <p>5.3 IPv4 Addressing: Classes, Subnet Mask, Subnetting, Supernetting.</p> <p>5.4 Unicast, Multicast, Broadcast Address, Multicast operation .</p> <p>5.5 IPv6 address, basic structure</p> <p>5.6 IPv4 vs IPv6</p>

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.



9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Fundamentals of Computer Network	08	04	06	04	14
II	Network Components and Topologies	12	06	04	06	16
III	Reference Models for Computer Networks.	10	04	08	04	16
IV	TCP/IP Protocol Suite	08	02	06	04	12
V	IP Addressing	10	02	04	06	12
Total		48	18	28	24	70

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)

Note: This specification table provides general guidelines to assist students for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

- Draw OSI Reference model viz a viz TCP/IP protocol stack on chart.
- Prepare chart showing wire color coding for making UTP cable and straight cable.

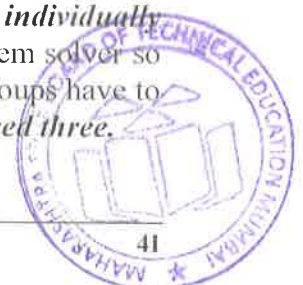
11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

- Massive open online courses (*MOOCs*) may be used to teach various topics/sub topics.
- 'L' in item No. 4 does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
- About **15-20% of the topics/sub-topics** which is relatively simpler or descriptive in nature is to be given to the students for *self-directed learning* and assess the development of the COs through classroom presentations (see implementation guideline for details).
- With respect to item No.10, teachers need to ensure to create opportunities and provisions for *co-curricular activities*.
- Use animations to explain various theorems in circuit analysis.
- Guide student(s) in undertaking micro-projects.

12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be *individually* undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should **not exceed three**.



The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than **16 (sixteen) student engagement hours** during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

- a. **Survey on Guided Transmission media:** Prepare a report on recent and widely used Guided media in industries depending on Cost, speed, efficiency, reliability.
- b. **Survey on Unguided Transmission Media:** Prepare a report on recent and widely used Unguided media in industries depending on Cost, speed, efficiency, reliability.
- c. **Present design of computer network for the institute spread across a campus with several buildings:** Elaborate/justify the Network for Inter building connectivity and within building. Present layout plan highlighting type of network selected, devices required, addressing used/servers used.
- d. **Survey on Structured cabling with CAT(5/5e/6/6a..):** Prepare a report on structured cabling system. Elaborate its importance, components, devices (passive and/active) required with their broad specifications for structured cabling with UTP cables.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1	Computer Networks	Tanenbaum , Andrew S.	PHI Learning, New Delhi, ISBN-13: 978-0-13-212695-3
2	TCP/IP PROTOCOL SUITE	Forouzan, Behrouz A.	McGraw Hill, New Delhi, 2006, ISBN 978-0-07-337604-2
3	Data communications and networking.	Forouzan, Behrouz A.	McGraw Hill, New Delhi, 2006, ISBN : 9780-07-296775-3
4	Data Communication and Networks	Godbole, Achyut	McGraw Hill, New Delhi, 2006, ISBN : 0070472971
5	Computer Networking	T.M.Bansod	
6	Computer Network Top down approach	Korus	Pearson.

14. SOFTWARE/LEARNING WEBSITES

- a. <http://www.nptelvideos.in/2012/11/computer-networks.html>
- b. <http://nptel.ac.in/courses/106105081/1>
- c. <http://freevideolectures.com/Course/2276/Computer-Networks>
- d. <https://www.youtube.com/playlist?list=PL5419FC33483A2563>
- e. <http://nptel.ac.in/downloads/106105080/>
- f. http://nptel.ac.in/courses/Webcourse-contents/IIT%20Kharagpur/Computer%20networks/ New_index1.html
- g. <http://www.studytonight.com/computer-networks/>
- h. <http://homepages.herts.ac.uk/~comqrgd/docs/network-notes/network-notes.pdf>
- i. <http://studentstudyhub.com/wp-content/uploads/2013/12/CN-Full-Notes-Download-1st-Part.pdf>





Program Name : Computer Engineering Program Group
Program Code : CO/CM/IF/CW
Semester : Fourth
Course Title : GUI Application Development using VB.Net
Course Code : 22034

1. RATIONALE

VB.NET is the programming language based on Object Oriented Concepts which is prominently used to develop GUI based Applications. Graphical User Interface (GUI) based application includes various user friendly controls to accept or display data. This course will give the students an in-depth understanding of the concepts used in VB .NET and necessary skills to use programming techniques to develop .NET based applications and deploy the same.

2. COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

- **Develop GUI based application using VB.net.**

3. COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above mentioned competency:

- Use Visual Studio IDE to design application.
- Develop GUI Application using Form Controls and its events.
- Apply Object Oriented concepts in GUI Application.
- Use Data access controls to store data in Database and retrieve it.
- Use Data Binding in GUI Application.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
L	T	P		Theory						Practical						
				Paper Hrs.	ESE		PA		Total		ESE		PA		Total	
Max	Min	Max	Min		Max	Min	Max	Min	Max	Min	Max	Min	Max	Min		
2	-	4	6	--	--	--	--	--	--	--	50#	20	50~	20	100	40

'#':No practical Examination, (~²): For the *practical only courses*, the PA has two components under practical marks i.e. the assessment of practicals (seen in section 6) has a weightage of 60% (i.e.30 marks) and micro-project assessment (seen in section 12) has a weightage of 40% (i.e.20 marks). This is designed to facilitate attainment of COs holistically, as there is no theory ESE.

Legends: L-Lecture; T – Tutorial/Teacher Guided Theory Practice; P - Practical; C – Credit, ESE - End Semester Examination; PA - Progressive Assessment, '#': No Theory Examination

5. COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)



This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the course. in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.

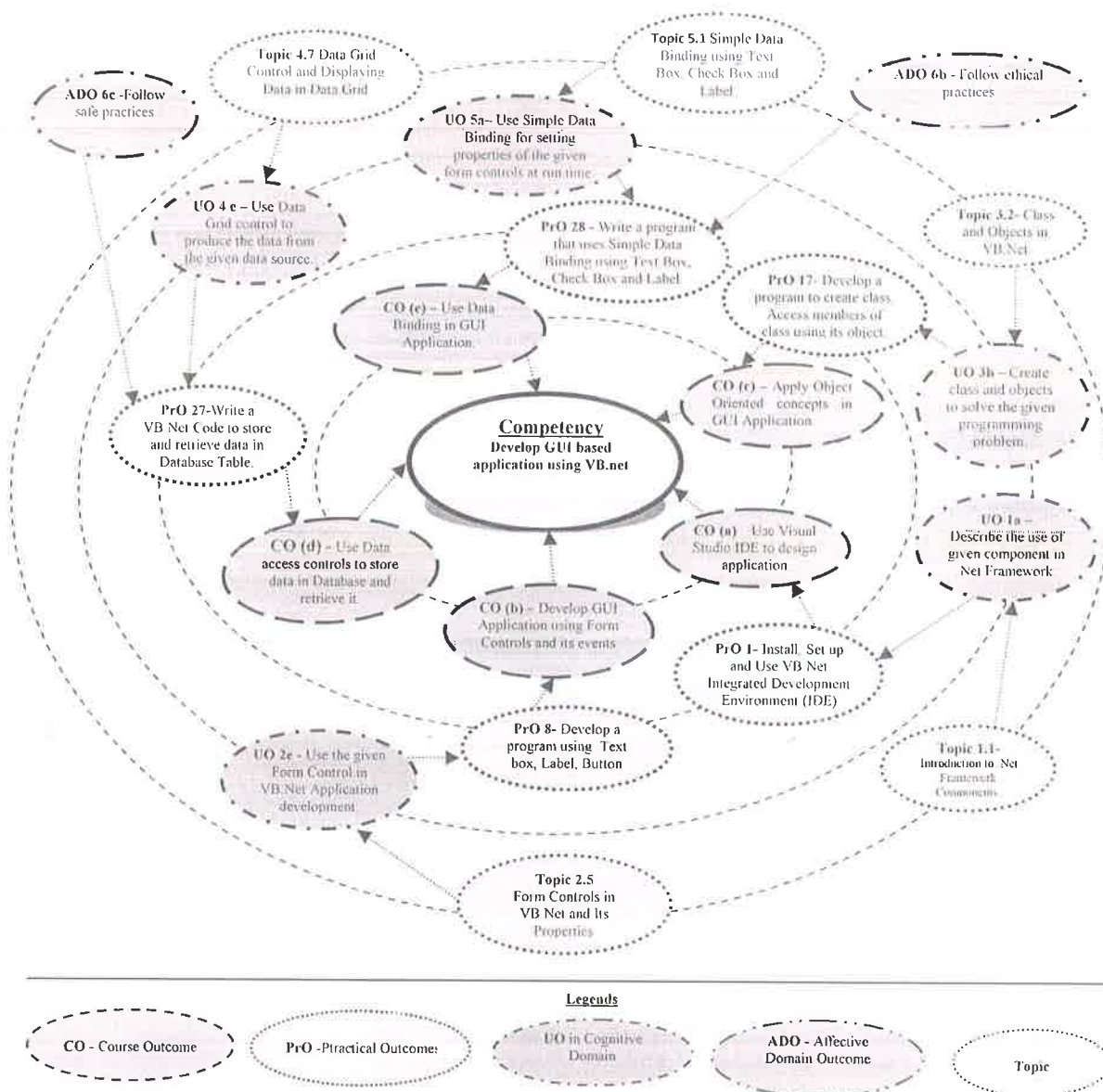


Figure 1 - Course Map

6. SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1.	Install, Set up and Use VB.Net IDE (Integrated Development Environment).	I	
2.	Use Existing Namespaces and Create user defined Namespace in VB.Net.	I	



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
3.	(a) Write a simple program to display a welcome message using <i>msgbox()</i> . (b) Develop programs to solve Arithmetic expressions.	II	02*
4.	Develop programs to demonstrate use of <i>IF</i> , <i>IF-else</i> Control structures in VB.net.	II	02*
5.	Develop programs to demonstrate use of <i>Case</i> Control structures in VB.net.	II	02*
6.	Develop programs to demonstrate use of <i>While</i> , <i>DO Loops</i> in VB.net.	II	02*
7.	Develop programs to demonstrate use of <i>For</i> , <i>For-each</i> Loops in VB.net.	II	02*
8.	Develop a program using Text box, Label, Button	II	02*
9.	Develop a program using Radio button, check box,	II	02*
10.	Develop a program using List box, Combo box.	II	02*
11.	Write a program using Picture Box, Panel.	II	02*
12.	Write a program using Tab Control, and Timer.	II	02*
13.	Write a program to perform validation using regular expression and error provider.	II	02*
14.	Write a program to perform validation using regular expression and error provider.	II	02*
15.	Write a program to demonstrate use of Sub-procedures and Parameterized Sub-Procedures.	III	02
16.	Write a program to demonstrate use of Simple function and parameterized Functions.	III	02*
17.	Develop a program to create class. Access members of class using its object.	III	02*
18.	Create constructor to initialize object of class. Use Destructor to de-allocate memory using <i>finalize</i> method.	III	02*
19.	Develop a program to inherit members of super class in sub class using simple inheritance.	III	02*
20.	Develop a program to demonstrate Overloading a method	III	02*
21.	Develop a program to demonstrate Overriding in inheritance	III	02*
22.	Develop a program to demonstrate Shadowing in inheritance		02
23.	Construct a program to handle runtime errors by using Exception handling.	III	02*
24.	Write a program to fetch data from table and display in Data Grid.	IV	02*
25.	Write a program to perform following operation using Data Adapter: Fill and Update data in Database.	IV	02*
26.	Write a program to perform following operation using Data Adapter: Fetch data from multiple tables in Dataset.	IV	02
27.	Write a VB.Net Code to store and retrieve data in Database Table.	IV	02*
28.	Write a program that uses Simple Data Binding using Text Box, Check Box and Label.	V	
29.	Write a program that uses Complex Data Binding using Combo	V	02



S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	Box.		
30.	Write a program that uses Complex Data Binding using List Box.	V	02
31.	Write a program to Navigate across existing data in table.	V	02
32.	Create Executable file of VB.Net Application and Deploy it to other computer.	V	02*
	Total		64

Note

- i. A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicious mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.
- ii. The 'Process' and 'Product' related skills associated with each PrO are to be assessed according to a suggested sample given below:

S.No.	Performance Indicators	Weightage in %
1	Use of relevant VB.Net tool to design given GUI application.	20
2	Write appropriate code to generate desired output in GUI Application.	30
3	Debug, test and execute the programs/modules.	30
4	Able to answer oral questions.	10
5	Submission of report in time.	10
	Total	100

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

- a. Follow safety practices.
- b. Practice good housekeeping.
- c. Demonstrate working as a leader/a team member.
- d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	PrO S. No.

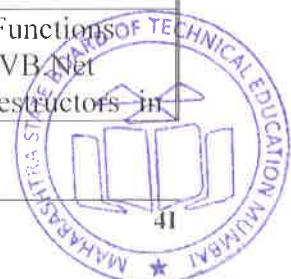


S. No.	Equipment Name with Broad Specifications	PrO S. No.
1	Personal computer, (preferably i3-i5 processor based), RAM minimum 2 GB, Hard disk 10 GB minimum available space.	For all Experiments
2	Operating system: Windows 7/8/10	
3	Microsoft Visual Studio 2012 or later.	

8. UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Major Learning Outcomes (in cognitive domain)	Topics and Sub-topics
Unit – I Overview of GUI Program ming	1a. Describe use of the given component in .Net Framework. 1b. Describe use of use the given element in VB.Net IDE. 1c. Apply the given System Namespace in VB.net Application. 1d. Create Event Handler to respond to the given event.	1.1 Introduction to .Net Framework Components. i. Common Language Runtime (CLR) ii. Microsoft Intermediate Language (MSIL) iii. Just-In-Time Compiler 1.2 Exploring VB.Net IDE 1.3 System Namespaces in VB.Net 1.4 Events and Event handling
Unit– II Decision control and Loop control	2a. Select relevant data type for the given problem with justification. 2b. Write expression using operators for the given VB.Net Application. 2c. Use relevant control structure to apply the given criteria for decision making and branching. 2d. Implement relevant type of loop to solve the given iterative problem. 2e. Use the given Form Control in VB.Net Application development.	2.1 Data Types in VB.Net 2.2 Operators in VB. Net ii. Arithmetic Operators iii. Logical Operators iv. Bit Shift Operators v. Relational Operators vi. Assignment Operators 2.3 Control Structures ii. IF Statement iii. IF – ELSE Statement iv. Select Case Statement 2.4 Loops in VB.Net i. For Loop ii. While Loop iii. Do Loop iv. For Each Loop 2.5 Form Controls in VB.Net and Its Properties - Button, Text box, Label, Radio button, Check Box, List Box, Combo Box, Picture Box, Panel, Tab Control, Timer.
Unit– III Object Oriented	3a. Write Sub-procedure/function to solve the given problem. 3b. Create class and objects to solve	3.1 Sub Procedures and Functions 3.2 Class and Objects in VB.Net 3.3 Constructors and Destructors in



Unit	Major Learning Outcomes (in cognitive domain)	Topics and Sub-topics
Programming in VB.Net	<p>the given programming problem.</p> <p>3c. Create constructor to initialize the given object.</p> <p>3d. Apply Inheritance to inherit members of the super class in the given problem.</p> <p>3e. Use Overloading/Overriding/Shadowing in the given situation.</p> <p>3f. Develop exception handling mechanism to handle the given exception.</p>	<p>VB.Net</p> <p>3.4 Inheritance in VB.Net, Simple Inheritance using <i>Override</i> Keyword</p> <p>3.5 Overloading, Overriding and Shadowing</p> <p>3.6 Exception Handling</p>
Unit-IV Data access in VB.NET	<p>4a. Use the relevant component of ADO.Net architecture based on the given situation.</p> <p>4b. Select relevant data provider to solve the given problem.</p> <p>4c. Use the given data provider to the given Access database.</p> <p>4d. Produce data using Data Adapter Control for communication between the given dataset and the data source.</p> <p>4e. Use Data Grid control to produce the data from the given data source.</p>	<p>4.1 Architecture of ADO.Net.</p> <p>4.2 Accessing Data with Server Explorer.</p> <p>4.3 Data Providers.</p> <p>4.4 Connections, Data Reader, Data Adapters and Datasets.</p> <p>4.5 Creating new Data Connection.</p> <p>4.6 Creating Dataset.</p> <p>4.7 Data Grid Control and Displaying Data in Data Grid.</p> <p>4.8 Data Access using Data Adapter.</p>
Unit –V Data Binding and Deployment	<p>5a. Use Simple Data binding for setting properties of the given form control at run time.</p> <p>5b. Use Complex Data binding for setting properties of the given form control at run time.</p> <p>5c. Implement VB.Net Application to navigate data in the given database.</p> <p>5d. Deploy the given VB.Net Application.</p>	<p>5.1 Simple Data Binding using Text Box, Check Box and Label.</p> <p>5.2 Complex Data Binding using Combo box and List box.</p> <p>5.3 Navigating Database.</p> <p>5.4 Deploying VB.Net Application.</p>

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER

–Not Applicable–

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:



- a. Prepare journals based on practical performed in laboratory.
- b. Library/E-Book survey regarding 'VB.Net' used in software industries.
- c. Undertake a survey of different GUI applications and compare with the following points.
 - i. Available Applications.
 - ii. Application Profile.

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

- a. Massive open online courses (*MOOCs*) may be used to teach various topics/sub topics.
- b. '*L*' in item No. 4 does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
- c. About *15-20% of the topics/sub-topics* which is relatively simpler or descriptive in nature is to be given to the students for *self-directed learning* and assess the development of the COs through classroom presentations (see implementation guideline for details).
- d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for *co-curricular activities*.
- e. Guide student(s) in undertaking micro-projects.
- f. Demonstrate students thoroughly before they start doing the practice.
- g. Observe continuously and monitor the performance of students in Lab.

12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be *individually* undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should *not exceed three*.

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than *16 (sixteen) student engagement hours* during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

- a. **Hotel Management applications**– Prepare
 - Hotel room booking system having variety of rooms,
 - Hotel billing system for various services used by the guest.
- b. **Store Management Application** - Prepare a menu driven application for inventory management in Store.
- c. **Students' Record System** -Prepare a menu driven application to Maintain Academic record of students from multiple streams.
- d. **Traffic signals control Design**- Design an application for traffic signal control using Timer Control.

13. SUGGESTED LEARNING RESOURCES



S. No.	Title of Book	Author	Publication
1	Visual Basic .NET The Complete Reference	Jeffrey R. Shapiro	McGraw-Hill, California, USA ISBN0-07-213381-3
2	Visual Basic .NET Programming Black Book	Holzner Steven	Dreantech Press, 2015, New Delhi, ISBN-13:978-81-7722-609-6.
3	Beginning Visual Basic 2012	Bryan Newsome	Wrox Press, USA, Edition: 2012; ISBN: 9781118311813,
4	GUI Application Development using VB.Net	Dr. Rajendra Kawale	Devraj Publication,

14. SUGGESTED SOFTWARE/LEARNING WEBSITES

- a. <http://www.vbtutor.net/index.php/visual-basic-2012-tutorial>
- b. <http://howtostartprogramming.com/vb-net>
- c. <https://www.tutorialspoint.com/vb.net>
- d. <http://vb.net-informations.com>
- e. <http://www.java2s.com/Tutorial/VB/CatalogVB.htm>
- f. <http://www.functionx.com/vbnet>
- g. <http://www.dfit.dfinalsolution.com/dotnet%20tutorial%20for%20beginners.pdf>

