Source Book
of
Certificate Course in Software Testing

Table of Contents

1. **Course Objective:** The objective of this course is to provide essential knowledge of programming and expertise in Testing. Students who will complete this course can work in software testing.

2. **Eligibility Criteria:** Any Engineering /Science graduate with mathematics up to 10+2 level.

3. **Prerequisite:** Sound knowledge of Computing Fundamentals and Fundamentals of Programming.

4. **Teaching Schema:**

Sl. No.	Module Name	Hours
1	Fundamentals of Computer & OS Concepts	20
2	C Programming	30
3	Software Development Life Cycle	10
4	OOP with C++ with DS	40
5	Database Technology	20
6	Software Testing – Manual	50
7	Software Testing – Automation	50
8	Management Development Program	60
9	Project	40
	Total	**320**

5. **Suggested Schedule**

Week	Teaching Sessions & Academic Activity
1	Fundamentals of Computer & OS Concepts (20/20 Hours) **and** C Programming (10/30 Hours)
2	C Programming (20/30 Hours) **and** Software Development Life Cycle (10/10 Hours)
3	OOPs with C++ and DS (30/40 Hours)
4	OOPs with C++ and DS (10/40 Hours) **and** Database Technology (20/20 Hours)
5	Software Testing – Manual (30/50 Hours)
6	Software Testing – Manual (20/50 Hours) **and** Software Testing – Automation (10/50 Hours)
7	Software Testing – Automation (30/50 Hours)
8	Software Testing – Automation (10/50 Hours) **and** Management Development Program (20/60 Hours)
9	Management Development Program (30/60 Hours)
10	Management Development Program (10/60 Hours) **and** Project (20/40 Hours)
11	Project (20/40 Hours) **and** Exam Break 2 days)
12	1st Day – Exam, Two Days – Project Evaluation, 5th Day – Re-exam

Note: Course Delivery is 6 hours per day, 5 days per week (30 hours per week)

6. Session wise Breakup

Note: Each single session is of two hours duration for all subjects mentioned below.

Fundamentals of Computer & OS Concepts

Theory 10 hours + Lab 10 hours

Session 1:

- Computer Fundamental
- Types of computer
- Primary and Secondary storage
- Input-output devices
- Disks, tapes and CD-ROMs.
- VDUs, printers and other output devices
- Introduction to windows operating systems
- The desktop, The window, application window, document window, Dialog Window
- The Icons, Explore Your Computer, The Start Button and Taskbar.
- My Computer, Windows Explorer, Starting and Closing Programs,
- Installing Operating System
- Performing a New Installation for Windows
- Installing a Software other than OS
- Setting up a printer
- Uninstalling software

Session 2:

- Overview of Operating systems
- Types of Operating System
- What is process and thread
- Process management

Session 3:

- Process Scheduling
- CPU Scheduling
- Preemptive vs Non-Preemptive

Session 4:

- Deadlocks
- Memory Management
- Memory and I/O Maps

Session 5

- File Management
- I/O Management
- IPC

C Programming

Theory 16 hours + Lab 14 hours

Session 1:

- Introduction to Programming Language
- History of C
- Importance of C
- Basic structure of C program
- Programming style
- Types, Operators and Expressions
- Variable Names
- Data Types and Sizes
- Constants
- Declarations
- Executing a `C` program

Session 2:

- Arithmetic Operators
- Relational and Logical Operators
- Type Conversions
- Increment and Decrement Operators
- Bit wise Operators
- Assignment Operators and Expressions
- Conditional Expressions
- Precedence and Order of Evaluation

Session 3:

- Statements and Blocks
- If-Else
- Else-If
- Switch
- Loops - While and For
- Loops - Do-While
- Break and Continue
- Array
- One–dimensional and two–dimensional arrays
- Multi dimensional arrays
- Dynamic arrays
- Character Array and string
- Declaring and initializing String variables.
- Writing string to screen
- Comparison of strings
- String handling function

Session 4:

- Pointers
- Arrays using Pointers
- Function pointers
- Pointer to structure
- Functions, Call–by–value, Call–by–reference
- Recursion, Factorial Function
- Passing Arrays to Function, Function call and Run–time Stack

Session 5:

- The C Preprocessor
- File Inclusion

- File, File Access modes, File Input /Output
- Macro Substitution
- Conditional Inclusion

Session 6:

- User–defined data types
- Basics of Structures
  - Structures and Functions
  - Arrays of Structures
  - Array with in structure
  - Self-referential Structures
  - Bit field

Session 7:

- Union
- Enumeration
- Typedef

Session 8:

- Variable length list,
- Command line arguments
- Low–level interaction in C

Lab Assignments:

1. Write program, which finds the number of digits before the first even digit. If number is 1372465 then answer is 3 since the first even digit is 2.
2. Write program to print following pictures.

```
  i.  AAAAAA
 ii.  AAABBBCCCDDD
iii.  ABBCCCDDDD
 iv.  AAAABBBCCD
  v.  ACEGIKM
 vi.  ABCDEF
      ABCDE
      ABCD
      ABC
      AB
      A
vii.  A
      AB
      ABC
      ABCD
      ABCDE
      ABCDEF
viii. A
      BC
      CDE
      DEFG
      EFGHI
      FGHIJK
 ix.  A
      BC
```

DEF
GHIJ
KLMNO
PQRSTU

x. ABCDEF
BCDEF
CDEF
DEF
EF
F

xi. Aabcde
ABabcd
ABCabc
ABCDab
ABCDEa
ABCDEF

3. Write program, which reads 2 numbers and prints the sum of square of the first and cube of the second. Exa- input 5 3 output 5*5 + 3*3*3 = 52
4. Write program, which reads a, b, c, p, q and r. Let ax+by+c=0 and px+qy+r=0 be equation of lines. Print their point of intersection.
5. Write program, which reads three numbers. The program outputs the middle of these. Example- input 5 2 4 output 4, input 5 6 2 output 5.
6. Write program, which will print all numbers, which are even but not a multiple of either 3 or 5. e. g. 2 4 8 14 16 22 ….
7. Write program, which will print all numbers, which are either a multiple of 3 or 5 but not both.
8. $1 + 2x + 3x^2 + 4x^3 + \ldots\ldots + (n+1)x^n$
9. $1 - x + x^2 - x^3 + \ldots\ldots + (-1)^n x^n$
10. Write program to print the last digit, which is multiple of 3. e.g. input 23617 output 6.
11. Write program to print the second last even digit. e.g. input 23863 output 8(do not use if condition).
12. Write program, which reads a number the finds special product. If number is 2314 then output is 4 + 1*4 + 3*1*4 + 2*3*1*4 = 44.
13. Write program, which reads a number and finds the location of last even digit. If the number is 18263 then the output is 4 since the last even digit 6 occurs at 4th place.
14. Write program, which reads a number and finds how many times the first digit occurs. If the number is 34533253 then the answer is 4 since the first digit (3) occurs 4 times. [Hint: declare long int x; and Read number using scanf(“%ld”,&x);]

Software Development Life Cycle

Theory 10 hours

Session 1:

- Introduction to Software and Software Engineering

- Software Process
- SDLC and Process Models
- Agile Development model
- Case study on Agile

Session 2:

- Requirement Engineering
- Requirement analysis
- Use case approach
    - Use cases & usage scenarios
    - Identifying use cases
    - Use cases & functional requirements
    - Benefits of Use cases

Session 3:

- Design concepts
- Component Definitions
- Class-based Component Design
- Class-based Component Design Principles
- Component-Level Design Guidelines
- Components

Session 4:

- Software implementation and maintenance
- Project Planning
- Project Management Definition
- Distinctive characteristics of Software
- Scope, Organizing, Planning,
  Scheduling, Graphical Schedule representations
- Activity Organization, Milestones, Deliverables

Session 5:

- Task Dependencies,
- Staffing, Communication
- Project Risk identification, analysis, planning monitoring
- Monitoring, Reviews,
- Control, Reporting
- Project Closure

CASE STUDY- I

Communities as well as non- profit and charitable organizations promoting reuse as an alternative to discarding unwanted items. It also covers market place functionality. User can buy and sell various products. Companies can also post details of the stuff they want to get rid of in bulk. This is a secure site where SSL has been used. Site traffic analyzer is also available to the site admin. It also has a blog associated with it where the admin can post case studies of the user experience.

The admin has the ability to create admin roles of various levels, view monthly transactions, create communities, hauling company details, customize email template and specify content, also give the feature in which to the admin where any corporate can contact the admin and they can provide details of the various goods that they want to get rid of in bulk.

For analysis of project please include followings documents:

1. SRS (software requirement specification)

2. SDLC (software development life cycle)
3. Database & technologies required
4. Project Mangement
5. Quality Management
6. Cost evaluation
7. Time requirement for the completion
8. Training ,Maintance & Documentation

Design a system which will meet all the above needs of the organization. The system should be flexible and should provide growth in the organizational activities.

CASE STUDY II

Objective

The objective of the project is to develop a software to assist ABC foundation in making decisions regarding home loans .The product will allow the client to add modify and delete information regarding the foundation investment, operating expenses and individual information. The product will perform the required calculations in the areas and produce the listing investment.

Assumptions & constraints

1. The deadline must be decided and met
2. validation and fulfillment of all documents for processing
3. the product must be reliable
4. The architecture must be open so that additional functionality may be added later
5. The product must be user friendly.

For analysis of project please include followings documents:

1. SRS (software requirement specification)
2. SDLC (software development life cycle)
3. Database & technologies required
4. Project Management
5. Quality Management
6. Cost evaluation
7. Time requirement for the completion
8. Training , Maintance & Documentation

Design a system which will meet all the above needs of the organization. The system should be flexible and should provide growth in the organizational activities

Objected Oriented Programming Concepts Using C++ & Data Structures

Theory 20 hours + Lab 20 hours

Session 1:

- Difference between C and C++
- Introduction to C++: Identifier, Keywords, Constants,
- Operators: Arithmetic, relational, logical, conditional and assignment. Sizeof operator, Operator precedence and associativity.
- Type conversion, Variable declaration, expressions, statements, manipulators
- Input and output statements, stream I/O, Conditional and Iterative statements, breaking control statements.
- OOP Concepts
- Class and Objects
- Executing sample C++ programs

Session 2:

- Static Member
- Inline Function
- Access specifiers
- String & Streams
- Constructors and Destructors
- Properties
- Types of constructors & destructors
- Inheritance
- Types of inheritance

Session 3:

- Friend functions
- Friend Classes
- Polymorphism
- Types of polymorphism
- Overloading functions
- Overloading Operators
- Copy Constructors
- Run Time Polymorphism
- Virtual Functions

Session 4:

- Exception Handling
- Namespaces
- File Handling
- Classes for file stream operations,
- opening and closing a file
- detecting end of file, file modes
- file pointers and their manipulations
- Sequential input and output operations
- Random access
- File operations error handling
- Command line argument

Session 5:

- Class and Function Templates
- subclass templates
- passing template classes to template parameters

Session 6:

- Dynamic memory allocation
- Allocating a block of memory: MALLOC
- Allocating a Multiple blocks of memory: CALLOC
- Introduction of REALLOC, Free

Session 7:

- Introduction to algorithm
- Analysis of algorithm
- Space complexity of algorithm
- Time complexity of algorithm
- Lists (Stacks)
- Lists (queues)

Session 8:

- Lists (Singly, Doubly and Circular)

Session 9:

- Searching (Sequential & Binary)
- Analysis of sorting & searching algorithms

Session 10:

- Sorting (Selection, Insertion, Bubble sort)
- Sorting (Merge, Heap & Quick)

Assignments:

1. Write a program which accept two numbers and print their sum.
2. Write a program which accept principle, rate and time from user and print the simple interest.
3. Write a program to check whether the given number is even or odd (using ? : ternary operator)
4. Write a Student class and use it in your program. Store the data of ten students and display the sorted data according to their roll numbers, date of births, and total marks.
5. Write the definition for a class called **time** that has hours and minutes as integer. The class has the following member functions:
   void settime(int, int) to set the specified value in object void showtime() to display time object time sum(time) to sum two time object and return time.
   a. Write the definitions for each of the above member functions.
   b. Write main function to create three time objects. Set the value in two objects and call sum() to calculate sum and assign it in third object. Display all time objects.
6. Write a program using basic concept of objects and classes to check whether given number is prime or not.
7. Using virtual and pure virtual functions implement hierarchy of computer printers.
8. Design a hierarchy of computer printers. Use multiple inheritances in your hierarchy. Also use friend functions and classes in your program.
9. Write Date and Time classes that allows you to add, subtract, read and print simple dates in dd/mm/yyyy and time in hh:mm:ss forms. Use function overloading in your program.
10. Write programs to overload =, ==, +, ++, --, <<, >> and [ ] operators
11. Write a template class for sorting method. Using this class, write a test program for sorting using different data types. Also implement exception handling.
12. Create a text file using any simple editor. Write a program that will reverse each line in the input file and store it in another file.
13. Create a C++ class named Grade that has the same functionality as the old struct Grade and associated functions in the original C file. When you are finished, your C++ file should only have one function definition outside of the class: main().

15. Write a menu driven program in c, which consist followings:
    - To read and display the transpose of MxN matrix
    - To sum of two MxN matrix
    - To product of two MxN matrix
16. Write a C Program to construct a stack of integers and to perform the following Operations on it.

- push
- pop
- display

The program should print appropriate messages for stack overflow, stack underflow and stack empty.

17. Write a single C program to perform each of the following:
    - Create a queue of N elements
    - Reverse the queue so that last element becomes first & so on
18. Using pointers write your own functions for the following;
    - String comparison
    - String concatenate
    - String copy
    - String length.

    Note: Do not include <string.h> in your program
19. Write a program to cyclically permute a string one character at a time.
    E.g.: If space is the input the output should produce
    - space
    - paces
    - acesp
    - cespa
    - espac
20. Define a structure called cricket that will describe the following information
    - Player name
    - Team name
    - Batting average

    Using cricket, declare an array player with 10 elements and write a program to read the information about all the 10 players and print a team wise list containing names of players with their batting average.
21. Write a preprocessor directive to accomplish each of the following:
    - Define symbolic constant YES to have the value 1.
    - Define symbolic constant NO to have the value 0.
    - Include the header file common.h. The header is found in the same directory as the file being compiled.
    - Renumber the remaining line in the file beginning with line number 3000.
    - If symbolic constant TRUE is defined, undefined it and redefine it as1. Do not use #ifdef preprocessor directive.
    - If symbolic constant TRUE is defined, undefined it and redefine it as 1. Use #ifdef preprocessor directive.
22. Write a program to maintain a singly linked list having the following functions:
    - Creation of the list
    - Displaying the list.
    - Delete a node from given location.
23. Write a program to maintain a doubly linked list having the following functions:
    - Creation of the list.
    - Displaying the list by traversing from both ends.
    - Counting the number of nodes in the list.
24. Given an array of sorted list of integer numbers, write a function to search for a particular item using the method of binary search. And also show how this function may be used in a program. Use pointers and pointer arithmetic

Database Technology

Theory 10 hours + Lab 10 hours

Session 1:

- Introduction to DBMS – What is DBMS, Its need
- Areas where DBMS are used
- Types of DBMS: Introduction to Hierarchical Model, Network and Relational Models
- Data models (conceptual physical and logical)
- Data Integrity & integrity rules

Session 2:

- Introduction to Oracle
- SQL* Plus
- DDL Commands
- DML & DCL Commands
- Inbuilt Functions
- Grouping Things Together (Group By, Having Clause)
- Advance Subqueries (Correlated Sub query, Outer Joins)

Session 3:

- Set Operators (UNION, UNION ALL, INTERSECT, MINUS)
- Types of Views
- Creating Complex View
- Using Temporary Tables

Session 4:

- Introduction to PL/SQL
- PL/SQL Programming style
- PL/SQL Program blocks
- Cursors

Session 5:

- Procedures & Functions
- Triggers

Lab Assignments:

1. Write a program that computes the perimeter and the area of a rectangle. Define your own values for the length and width. (Assuming that L and W are the length and width of the rectangle, Perimeter = 2*(L+W) and Area = L*W. Display the output on the screen using dbms_output.put_line.
2. Write a program that declares an integer variable called num, assigns a value to it, and computes and inserts into the tempp table the value of the variable itself, its square, and its cube.
3. Convert a temperature in Fahrenheit (F) to its equivalent in Celsius (C) and vice versa. The required formulae are:-
   C= (F-32)*5/9
   F= 9/5*C + 32
   Display the output on the screen using dbms_output.put_line. Data has to be input by the user.

4. Convert a number of inches into yards, feet, and inches. For example, 124 inches equals 3 yards, 1 foot, and 4 inches. Display the output on the screen using dbms_output.put_line. Data has to be input by the user.
5. Write a program that enables a user to input an integer. The program should then state whether the integer is evenly divisible by 5. (Use decode instead of IF statement where required). Display the output on the screen using dbms_output.put_line. Data has to be input by the user.
6. Your block should read in two real numbers and tell whether the product of the two numbers is equal to or greater than 100. Display the output on the screen using dbms_output.put_line. (Use decode instead of IF statement where required). Data has to be input by the user.
7. In a PL*SQL block, create a datatype by the name of addr_type. It should contain the following components:-
   - name varchar2 (20)
   - street varchar2 (30)
   - city varchar2 (20)
   - state varchar2 (15)
8. Your block should accept the names and addresses of 4 employees in 4 different variables of datatype addr_type. Output the names and addresses of the 4 employees on the screen in the form of Labels as shown below:-

```
************************************************************
* Name:-      Jack              **    Name:-      Scott        *
* Street:-    M.G. Road         **    Street:-    Bhosale Marg *
* City:- Mumbai           **    City:-      Chennai      *
* State:-     Maharashtra       **    State:-     Tamil Nadu   *
************************************************************
************************************************************
* Name:-      King              **    Name:-      Adams        *
* Street:-    Lane No:-2        **    Street:-    P. M. Road   *
* City:- Nagpur                 **    City:-      Bangalore    *
* State:-     Maharashtra       **    State:-     Karnataka    *
************************************************************
```

9. Input a number and determine whether it is within a given range (for example, between 1 and 10). The low and high values of the range may be input by the user rather than be fixed by the program. Display the output on the screen using dbms_output.put_line.
10. Input three positive integers representing the sides of a triangle, and determine whether they form a valid triangle. Hint: In a triangle, the sum of any two sides must always be greater than the third side. Display the output on the screen using dbms_output.put_line.
11. Check if a given a year is a leap year. The condition is:-
    year should be (divisible by 4 and not divisible by 100) or (divisible by 4 and divisible by 400.) Display the output on the screen using dbms_output.put_line. The year should be input by the user.
12. Write a program that examines all the numbers from 1 to 999, displaying all those for which the sum of the cubes of the digits equal the number itself. Display the output on the screen using dbms_output.put_line.
13. Write a PL*SQL block that reads in a minimum and maximum value for a radius, along with an increment factor, and generates a series of radii by repeatedly adding the increment to the minimum until the maximum is reached. For each

value of the radius, compute and display the circumference, area, and volume of the sphere. (Be sure to include both the maximum and the minimum values.). Validate each of the input values to be sure they are positive. If the minimum is typed in place of the maximum, swap the values within the program, and continue execution. Display the results on the screen using dbms_output.put_line.

14. A table consists of the following fields:-

Invoice Number	Varchar2	4
Invoice Date	Date	
Customer Code	Number	1
Product Code	Number	1
Quantity Sold	Number	3

There are ten customers with codes 0 to 9 and five products with codes 0 to 4. The rates of products are Rs. 15, 35, 42, 51 and 60 respectively. Write a program to find the total purchase in Rs. of each customer and total sale of each product using this table and insert these values in two other tables.

15. Write a PL*SQL block to accept a character string from the user. The user should enter a number spelt out. With the help of PL*SQL arrays, write a program for Word to number conversion up to 99 crores. The program should cater to Rs. and paise also.

For example, if the user enters:-

Rs. Twelve crores, Thirty Four lakhs, Fifty One thousand, Two hundred and Fifty and Seventy five paise only

The output of your program should be: - 123451250.75

If the user enters:-

Rs. Nine thousand, Seven hundred and Twenty Eight only

The output of your program should be: - 9728

16. Create the following 3 tables and insert sample data as shown:-

Ord_mst

Ord_no	**Cust_cd**	**Status**
1	C1	P

Ord_dtl

Ord_no	**Prod_cd**	**Qty**
1	P1	100
1	P2	200

Prod_mst

Prod_cd	**Prod_name**	**Qty_in_stock**	**Booked_qty**
P1	Floppies	10000	1000
P2	Printers	5000	600
P3	Modems	3000	200

- Write a Before Insert trigger on Ord_dtl. Anytime a row is inserted in Ord_dtl, the Booked_qty in Prod_mst should be increased accordingly.
- Write a Before Delete trigger on Ord_dtl. Anytime a row is deleted from Ord_dtl, the Booked_qty in Prod_mst should be decreased accordingly.

17. Write a stored procedure by the name of Comp_intr to calculate the amount of interest on a bank account that compounds interest yearly. The formula is:-

$I = p\ (1+ r)^{y} – p$

where:-

I is the total interest earned.

p is the principal.

r is the rate of interest as a decimal less than 1, and
y is the number of years the money is earning interest.

Your stored procedure should accept the values of *p*, *r* and *y* as parameters and insert the Interest and Total amount into tempp table.

18. Create a stored function by the name of Age_calc. Your stored function should accept the date of birth of a person as a parameter. The stored function should calculate the age of the person in years, months and days e.g. 35 years, 3 months, 17 days. The stored function should return the age in years directly (with the help of Return statement). The months and days are to be returned indirectly in the form of OUT parameters. Write a PL*SQL block to accept the date of birth of an employee from the user, call the stored function, and display the age of the employee on the screen. Display the above results on the screen using dbms_output.put_line.
19. Write a SELECT statement to display the experience of all the employees (Sysdate – Hiredate). Your output should be as follows:-

    5 years 7 months 11 days
    9 years 3 months 16 days etc

    Don't assume that there are 365 days in a year or that there are 30 days in a month. Your solution should even take care of leap years.
20. Create a view of the Salespeople table called Commissions. This view will include only the snum and comm fields. Through this view, someone could enter or change commissions, but only to values between .10 and .20.
21. Some SQL implementations have a built-in constant representing the current date, sometimes called "CURDATE" or "SYSDATE". The word CURDATE can therefore be used in a SQL statement, and be replaced by the current date when the value is accessed by commands such as Select or Insert. We will use a view of the Orders table called Entryorders to insert rows into the Orders table. Create the Orders table, so that CURDATE is automatically inserted for odate if no value is given. Then create the Entryorders view so that no values can be given.
22. Write a command that will enable a user to pull orders grouped by date out of the Orders table quickly.
23. Write a command that puts the following values, in their given order, into the salespeople table: city – San Jose, name – Blanco, comm – NULL, cnum – 1100.
24. Write a command that removes all orders from customer Clemens from the Orders table.

Software Testing – Manual

Theory 24 hours + Lab 26 hours

Session 1:

- Software Testing Fundamentals
- Term and modification of testing
- Fundamental Test process
- Types of testing
- Life cycle of testing
- Fundamentals of Quality

Session 2:

- Test Management
- Testing Concepts

- Testing Principles
- Testing Approaches

Session 3:

- Black Box Testing
- White Box Testing
- Static Testing
- Dynamic Testing
- Testing Process

Session 4:

- Test Planning
- Administrative Plan
- Risk management
- Test Focus
- Test Objective

Session 5:

- Test Strategy
- The Build Strategy
- Problem management
- Problem Control
- Test case Design
- “V” Model

Session 6:

- Levels of testing
- Unit Testing
- Integration Testing
- Bottom Up
- Top Down

Session 7:

- Big bang
- Sandwich
- System testing
- GUI
- Usability

Session 8:

- Configuration
- Compatibility
- Availability
- Reliability
- Installation

Session 9:

- System Integration Testing
- User Acceptance Testing
- Alpha Testing
- Beta testing

Session 10:

- Testing Types
- Functional testing

Session 11:

- Structure Testing
- Specialized Testing

Session 12:

- Planning Your TEST efforts

Assignments:

1. Implement functional testing in employee management system.
2. Create a scenario to check usability testing
3. Analyze the Employee management system and decide what all types of testing are required to test user interfaces.
4. Perform black box testing and white box testing for railway reservation system.
5. Define parameters for user acceptance testing of online examination system.
6. Analyze an e-library system and describe top down and bottom up approaches.
7. Analyze any billing system and differentiate between unit testing and integration testing.
8. Consider Hotel Reservation System and describe structured testing.
9. Analyze levels of testing in online examination system.

Software Testing – Automation

Theory 20 hours + Lab 30 hours

Session 1

- Test Planning
- (Administrative Plan, Risk management, Test Focus, Test Objectives, Test Strategy, The Build Strategy, Problem management and Control)
- Test case Design
- Writing Test cases,
- Test procedures,

Session 2

- Test data,
- Test Reports,
- Defect Management – Using Test Director,
- Problem management,
- Change Management,
- Measuring the Success/Test Metrics and defect Life cycle,

Session 3

- Automated Testing and Tools,
- Automated Vs Manual Testing,
- Benefits of Automation,
- Types of Test case Management Tools,
- Automation Tools,
- Defect Management Tools,

Session 4

- **QTP:**
- Introduction
- Frameworks
- Test Automation
- Environment Setup

- Record and Play Back
- Object Repository
- Actions
- Data Tables
- Checkpoints
- Synchronization
- Smart Identification

Session 5

- Debugging
- Error Handling
- Recovery Scenarios
- Environment Variables
- Library Files
- Test Results
- Running Tests,
- Tracking Defects,

Session 6

- GUI Objects
- Virtual Objects
- Accessing Databases
- working with XML
- Descriptive Programming,
- Automation Object Model
- Recording Tests,
- Synchronizing Tests,

Session 7

- **QC:**
- Introduction
- Environment Setup
- Server Side Configurations
- Common Functionalities
- Management Requirements
- Test Plan
- Test Lab
- QC Integration

Session 8

- Defects
- Dashboard Analysis
- Project Customization
- Creating Data,
- Reading Text
- Creating Batch Tests
- Maintaining Your Test Scripts

Session 9 & 10

- Test Cases

Assignments:

1. Try to design Test case for login page.

2. Use case based test case design exercise.

   You are required to study the application description and the associated use case description given below to

   1. Identify the use paths
   2. Prepare test cases to execute all use case paths
   3. Prepare additional test cases for data validation
   4. Document test suite and test case descriptions

   Book point Sales System – Book Availability Inquiry and Supply of Books for Approval to purchase: Book point receives inquiries from libraries, companies and individuals regarding availability of books. Libraries and companies generally inquire about books on various subjects of various publishers and / or various books published by an author. Sometimes they inquire about a specific title also.

   In the current manual system the manager/clerk searches book catalogs of various publishers on the subjects. He prepares a list of books on the subject(s). He verifies the stock register and the book-racks to check availability of the book in the books stores. He verifies which books can be obtained from publishers or their agents. He intimates the list of books available to the customer.

   He prepares a proforma invoice, if the customer is interested in obtaining the books, on approval basis for purchase. The invoice contains information on books that can be arranged from within the store, and those need to be procured from publishers or their agents against firm order. As per the proforma invoice books are collected and packed. Books being supplied on approval basis are recorded in the stock register. The proforma invoice and the packaged books are sent to the library/company on approval basis for selection and decision on purchase of books.

   The Book Point bookstore intends to automate the book inquiry operation by means of a web based application. As part of the system analysis the book enquiry operation has been documented as per the following use case.

   Use case Name: Book Purchase Inquiry

   Description: Provide necessary information related to books on various subjects. Generate list of books available on given title, subject, authorize and publisher. Generate and print proforma invoice for supply of books for approval to purchase, if required.

   Actors: Manger, Clerk

   Initiation: This use case is executed when the book availability and purchase inquiry from to customers (libraries, companies and individuals) is received by the manager/clerk of the bookstore.

   Precondition: Stock register and publisher catalog should be up to date.

3. Assignment
   1. Study test case design document developed in assignment 1 for testing MS Word processor and do the following.
      1. Execute the tests and record results
      2. Get your test results evaluated by peer and obtain test evaluation report from him
      3. Prepare test evaluation report for your peer
   2. Study the test case design document of the client project assigned to you. Also study the test result documents (defect reports, test case forms) and the test evaluation forms. Interact with the testing team members to understand how test evaluation activity is carried out.

4. Assignment

1. Study the features of a capture/playback testing tools (such as Win Runnert, X Runner) and prepare a short technical note on its features.
2. Study the automated testing mechanisms being used in the real life client project assigned to you. Interact with the project ream members to understand how tests are automated. Study few test scripts for automating testing.

5. Assignment
   1. Study the defect reports assigned to you. Find ways by which clarity in the defect report can be improved. Verify whether the defect can be actually reproduced (if access is provided to the software system).

   2. Study defect log and / or defect-tracking tool being used by the testing project assigned to you. Interact with the testing project team members to understand how defects are logged and tracked to closure.
6. Assignment
   1. Study the test metrics data of the testing project assigned to collect the following information.
      a. Number of tests planned, conducted and blocked (build wise)
      b. Testing effort (build wise)
      c. Defects – found, open, closed, rejected (build wise),
      d. Defect distribution - category (root cause) wise, severity wise, functionality wise
      e. Days to correct defects (time elapsed between reporting to closure of the defect)
      f. Test productivity information – test case design, testing (for each build)

      Analyze test information and develop charts for the following
      a. Open defects Vs build (or time)
      b. Closed defects Vs build (or time)
      c. Cumulative open defects Vs time
      d. Cumulative closed defects Vs time
      e. Defect detection rate (defects found /testing effort) Vs time (weekly)
      f. Defect resolutions rate (defects closed per week) Vs time
      g. Cumulative defects found over time Vs time
      h. Defect distributions root cause wise
      i. Defects found functionality wise
      j. Testing productivity build wise

Management Development Program

Theory 30 hours + Practice 30 hours

Session 1:

- Introduction to communication,
- Barriers to communication, Kind of communication,
- Confidence building Non-verbal Communication

Session 2:

- Fluency and vocabulary
- Synonyms
- Antonyms

- Grammar, Noun Pronoun,
- Verb, Adjective, Preposition, Conjunction

Session 3:

- Words of Idioms & phrases
- Sentence Construction
  Pronunciation,

Session 4:

Greeting,
Conversation practice,
Polite Conversation,

Session 5:

Resume Writing,
Covering letter,
Email,

Session 6:

Presentation Skill,
What is group discussion?
Interview skills, Mock interview

Session 7:

- Analogy, Series Completion (Number, Alphabet, Letter Series)
- Coding-Decoding for Number
- Alphabet and Letter
- Blood Relations

Session 8:

- Puzzle Test: Classification Type questions
- Compression Type questions
- Sequential order questions
- Section based on given conditions
- Questions involving family members

Session 9:

- Alphabet test
- Order of words
- Letter words problems
- Rule detection
- Alphabetical quibble
- Word formation
- Number
- Ranking
- Time Sequence Test
- Mathematical operations
- Logical sequence of words

Session 10:

- Arithmetic reasoning
- Logical reasoning
- Statement-Arguments
- Statement-Assumptions
- Statement-courses of Action
- Statement-Conclusions
- Deriving conclusion from passages

Session 11:

- General Aptitude
- Addition
- Multiplication
- Divisibility
- Squaring
- Cube
- HCF and LCM
- Fraction

Session 12:

- Number system
- Permutation & combination
- Probability
- Ratio & Preparation

Session 13:

- Partnership
- Percentage
- Average
- Problem on Ages
- Profit and loss

Session 14:

- Simple Interest
- Compound Interest
- Time and work
- Work and Wages

Session 15:

- Trains
- Streams Pronoun
- Alligation
- Clock
- Pipes and cisterns

Lab Practice:

- Faculty needs to conduct GD, presentation for speaking, conducting mock interviews etc.
- Faculty needs to conduct tests, Surprise tests, assignments etc.

7. List of Reference Books

Name of the Module	Title of the Book	Author/Publication	Edition	ISBN
C Programming	The C programming Language 2nd Edition	Kernighan, Retchie / PHI Learning	2012 Printing	9788120305960
	Programming in ANSI C 6th Edition	E. Balaguruswamy / Tata Mc-Graw Hill Publishing	2012 Printing	9781259004612
	Let Us C 13th Edition	Yashavant Kanetkar / BPB Publication	2012 Printing	9788183331630
Software Development Life Cycle	Fundamentals of Software Engineering	Rajib Mall / PHI Learning	3rd Edition	9788120338197
	Software Engineering: A Practitioner's Approach	Roger S. Pressman / Tata McGraw – Hill Publication	7th Edition	9780071267823
	Software Engineering	Ian Sommerville / Pearson Publication	9th Edition	9788131762165
Objected Oriented Programming Concepts Using C++ and Data Structures	Thinking in C++ : Introduction to Standard C++ Vol - 1	Bruce Eckel / Pearson	2nd Edition	9788131706619
	Object-oriented Programming Using C++	Dehuri Satchidananda, Jagadev Alok Kumar, Rath Amiya Kumar / PHI Learning	1st Edition	9788120330856
	Object - Oriented Programming Using C++	Gopalan N. P., Sivaselvan B., Mala C / PHI Learning Private Limited	1st Edition	9788120339231
	Data Structures through C++ 1st Edition	ISRD Group/Tata McGraw - Hill Education	2011 Printing	9780071072779
	Fundamentals of Computer Algorithms 2nd Edition	Sartaj Sahni, Sanguthevar Rajasekaran, Ellis Horowitz / Universities Press	2008	9788173716126
Database Technology	Oracle Database 11g The Complete Reference, 1st Edition	Kevin Loney / Tata McGraw - Hill Education	2008 Printing	0070140790
	Mastering Database Technologies	Ivan Bayross / BPB Publication	2005 Edition	9788183331302
	Database Management Systems	Raghu Ramakrishnan, Johannes Gehrke / Tata McGraw – Hill	3rd Edition	9780071231510

Software Testing – Manual	Software Quality Assurance : From Theory to Implementation	Galin / Pearson	Year 2004	9788131723951
	Software Testing : Principles and Practices	Srinivasan Desikan, Gopalaswamy Ramesh / Pearson	1st Edition	9788177581218
	Software Engineering and Quality Assurance	K Chandrashekar Shet / BPB Publicatons	1st Edition	9788183330145
	Software Testing	Ron Patton / Pearson	2nd Edition	9788177580310
	Software Testing Concepts and Tools	Nageswara Rao Pusuluri / Dreamtech Press	1st Edition	9788177227123
Software Testing – Automation	Software Testing Tools : Covering WinRunner, Silk Test, LoadRunner, JMeter and TestDirector with case studies (With CD)	K.V.K.K. Prasad / Dreamtech Press	1st Edition	9788177225327
	Making Sense of Software Quality Assurance	Authored By Raghav S Nandyal / Tata McGraw - Hill Education	1st Edition	9780070633780
	Jumpstart to Software Quality Assurance	Vishnuvarthanan Moorthy / Createspace	2013 Edition	9781491203514
	Software Testing Concepts And Tools	Nageshwar Rao Pusuluri / Dreamtech Press	1st Edition	9788177227123
	Software Testing And Quality Assurance: Theory And Practice	Priyadarshi Tripathy, Kshirasagar Naik / Wiley India Pvt Ltd	2012 Edition	9788126525935
Management Development Program	High School English Grammar & Composition Revised Edition	Wren, Martin / S. Chand Publisher	2011 Edition	9788121900096
	Communication Skills Publication Year 2011	Sanjay Kumar, Pushp Lata / Oxford University Press	2011 Edition	9780198069324
	Professional Communication Skills	Praveen S R Bhatia / S.Chand Publishing	2011 Edition	9788121920926
	Quantitative Aptitude For Competitive Examinations	R. S. Aggarwal / S. Chand Publishing	17th Edition	9788121924986
	A Modern Approach	R. S. Aggarwal /	Year	9788121905510

	To Verbal & Non-Verbal Reasoning	S.Chand Publishing	2012 Edition	
	How to Prepare for GD and Interview (With CD) 3rd Edition	Hari Mohan Prasad, Rajnish Mohan/TMH	2010	0070706344

8. Evaluation Guidelines

8.1 Evaluation

Evaluation is a necessary and essential part of conducting the Certificate Course in Software Testing programme, as it provides important feedback and inputs to both the institute as well as the student. The institute gets an idea about the relative performance of each student, which also serves as feedback about the design and conduct of the programme. The student gets a clear picture of his academic standing, individually and in comparison to his fellow students.

In order to ensure timely and efficient evaluation and certification of all students, the following guidelines are being issued and should be followed religiously.

8.2 Evaluation Methodology

8.2.1 Each centre should have a Designated Responsible Member (DRM) for Evaluation.

8.2.2 The DRM Evaluation would be responsible for coordinating all activities relating to evaluation at the training centre and for communicating with CDAC ACTS, Pune.

8.2.3 Evaluation is a compulsory part of the process of obtaining Certificate Course in Software Testing. All students are required to pass in each subject of the course in order to be eligible to receive the C-DAC Certificate.

8.2.4 The faculty of every subject should outline the objectives of the evaluation to be conducted for that particular subject, so as to enable the student to prepare himself/ herself properly.

8.2.5 The performance of students is constantly evaluated through surprise quizzes, hourly examinations, assignments throughout the term, submission of term reports, presentations and final examinations at the end of the course.

8.2.6 Mode of exams will be in online / offline, but prior information will be given by C-DAC, ACTS about the mode of the exam and it will be final.

8.3 EVALUATION METHODS

8.3.1 Course End Evaluation

After completion of the all subjects, a written examination CEE (Course End Examination) will be held, which will test the knowledge of the students of each subject and it is a compulsory part of the evaluation. Conducting CEE involves performing duty with responsibility. A small mistake in the process may hamper the whole system. Everyone has to play their role in an effective manner. It is a joint effort work which has to be carried out in a combined way. Right from receiving question paper from ACTS, C-DAC to sending the OMR answer sheet (in case of offline exam) and the response file (in case of online exam) for evaluation dealt with lot of responsibility.

ACTS, C-DAC in its pursuit of excellence, believes in providing a congenial atmosphere to the students during all exams in order to get them to perform at their optimum level. However, there are certain norms which the students are expected to be aware of and observe both in letter and spirit. These norms are:

3.1.1 Impersonation may lead to permanent expulsion from the Institute.

3.1.2 Cell phones are strictly prohibited in the exam hall/room.

3.1.3 Valid ID card is mandatory for entry to the exam room / hall.

3.1.4 Punctuality is most important at all times. Students are expected to check their exam location and be seated at least 10 minutes prior to the exam time.

3.1.5 In case of offline exam, as per ACTS, C-DAC policy all question papers are to be returned along with the answer script.

3.1.6 Students are required to bring their own stationary as no lending or borrowing is permitted during examination.

3.1.7 Programmable calculators or any other kind of electronic devices are strictly prohibited inside the exam area.

3.1.8 Indiscipline in the exam hall/ room will not be tolerated.

3.1.9 Possession of any written material related to the subject or communication with their fellow students, will result in disciplinary actions.

3.1.10 A student must score a minimum of 40 percent marks, in order to successfully clear the course.

3.1.11 It is recommended that the students should ensure 100% attendance for each course. 10% absences are permissible, only in case of illness, or emergencies. These have to be approved by the Centre Head. Approval is contingent upon the evidence provided.

3.1.12 There will be 150 questions to answer in 3 hours duration in CEE as per the following distribution mentioned in Table – 1.

Table – 1

Sl. No.	Module Name	Hours	No. of Questions
1	Fundamentals of Computer & OS Concepts	20	10
2	C Programming	30	15
3	Software Development Life Cycle	10	05
4	OOP with C++ with DS	40	20
5	Database Technology	20	10
6	Software Testing – Manual	50	35
7	Software Testing – Automation	50	35
8	Management Development Program	60	20
9	Project	40	-
	Total	**320**	**150**

8.3.2 GENERAL GUIDELINES FOR AWARD OF GRADES:

The marks of obtained in the CCEE shall be calculated to get total marks out of 100. The rounding off shall be done on the higher side. The grades shall be awarded on the basis of cut off in the absolute marks, as mentioned in Table – 2.

Table 2

Lower range of marks	Grade	Upper range of marks
91	≤ A+ <	100
81	≤ A <	90
71	≤ B+ <	80
61	≤ B <	70
51	≤ C+ <	60
41	≤ C <	50
0	≤ F <	40

8.3.3 Guidelines of CEE:

CEE will be conducted normally before the commencement of Project work of the course. The written examination should be of 180 minutes duration. It should consist of objective questions. A typical objective type exam paper should contain the following types of questions: –

- Multiple choice
- Yes or No
- True or False

Objective questions are useful in testing the recognition and recall abilities of students. They also help in keeping the exam short and easier to evaluate.

For the pure objective type question papers, there will be 150 objective type questions with 4 maximum answer options having only one correct option. The value of each objective type question is of one mark only. There will not be any negative marks for the wrong answers given by the students.

8.3.4 Guidelines for setting Question Papers:

While setting the question papers for theory Exam the following weightages should be assigned as per the difficulty level of the questions.

Levels	Requirements	Weightage
Level A – Easy	Requires elementary knowledge which may be obtained by attending all lectures and completion of mandatory lab assignments	25%
Level B – Intermediate	Requires thorough study of all course material, attendance at all lectures and completion of mandatory assignments	50%
Level C – Difficult	Requires study and lab work beyond the prescribed course material and mandatory assignments	25%

8.4 Guidelines for generating questions:

8.4.1.1 Question paper setter has to use sample paper format provided by C-DAC, ACTS Pune

8.4.1.2 Mention the subject name without fail.

8.4.1.3 Language of the question should be easy to understand.

8.4.1.4 The answers must have relevant objective type choices and “only one” correct answer.

8.4.1.5 The questions must be prepared by referring appropriate books, reference books, reference material, and course material having good information.

8.4.1.6 The question must be created by the domain expert afresh and should not be copied directly from any book, website, existing previous question papers etc.

8.4.1.7 The question should be unique and should have not been published anywhere.

8.4.1.8 Please mention the source of the question wherever possible, as it may help us in referring the same for detailing if required.

8.4.1.9 The caliber of the question should suffice the growing need of competition.

8.4.1.10 The question paper should have questions covering the entire syllabus.

8.4.1.11 The questions have to be typed in MS Word with “Arial” having letter size 12 point. Do not bold any letter, word or sentence in any part of the question paper.

8.4.1.12 It is essential to give password to the word document and send/tell the password separately.

8.4.1.13 It is essential that utmost care is taken at your end to maintain the secrecy of the soft copy at all time.

8.4.1.14 An expert team will review all questions. The questions will be filtered as per following:

- If the question is incomplete
- If the answer of the question is wrong
- If the question is not there in the syllabus
- If the question appears more than once
- If the question is too lengthy
- If the question is irrelevant
- If the options to the questions are irrelevant

8.4.1 Template for generation of Questions

Date:

Question generated by: Mr. /Ms.

Subject Name:

Q. No.
Question: <Text of the question>

Answer Choices
A:
B:
C:
D:

Difficulty Level: Easy / Intermediate / Difficult

Reference: (Name of books)
(If question taken from book) (Mention name of the book, author, ISBN)
Total Number of Questions Generated: ________________

8.4.2 Template for Answer Key:

Module name:			
Question No.	**Answer Keys**	**Question No.**	**Answer Keys**
1		76	
2		77	
3		78	
4		79	
5		80	
6		81	
7		82	
8		83	
9		84	
10		85	

11		86	
12		87	
13		88	
14		89	
15		90	
16		91	
17		92	
18		93	
19		94	
20		95	
21		96	
22		97	
23		98	
24		99	
25		100	
26		101	
27		102	
28		103	
29		104	
30		105	
31		106	
32		107	
33		108	
34		109	
35		110	
36		111	
37		112	
38		113	
39		114	
40		115	
41		116	
42		117	
43		118	
44		119	
45		120	
46		121	
47		122	
48		123	
49		124	
50		125	
51		126	
52		127	
53		128	
54		129	
55		130	
56		131	
57		132	
58		133	
59		134	

60		135	
61		136	
62		137	
63		138	
64		139	
65		140	
66		141	
67		142	
68		143	
69		144	
70		145	
71		146	
72		147	
73		148	
74		149	
75		150	

8.4.3 Evaluation of answer papers:

For Offline mode: Use of OMR sheets will be useful for processing the result of multiple choice exams. OMR is an effective way to collect data, process for the result and also it takes less time with greater accuracy in less effort. Centres need to follow the best way for scanning the OMR sheets, process the result and publish the result. Centres which are not using OMR can user OCR to conduct the exams and evaluate the students. Centre which are not using OMR or OCR can evaluate the students manually and process the result.

For Online mode: Course end exam will be through online s/w. Evaluation will be through that Exam s/w.

If a student requests for re-evaluation then the student has to pay `150 /- and it should be routed through training centre. The Re-evaluation fee should be paid to respective C-DAC training Centres, in case of Authorized Training Centres associated to C-DAC, Pune, payment to be made in favour of "C-DAC, ACTS" and payable at Pune. (This is applicable only for theory exam)

8.5 Moderation:

Grace marks would be awarded as per the methodology below:

8.5.1. Maximum of 4% of total term end theory exam marks can be awarded to a candidate.

Sr. No.	Name of the course	Total Marks	Maximum grace marks
1	Certificate Course in Software Testing	150	6

On completion of the moderation exercise the revised marks should be updated in the marks database.

8.6 Re-examinations:

The following conditions will be applicable for the course end re-exam:

6.1. Students who do not appear for an exam on the scheduled date will not have an automatic right to re-examination. Only those students who, in the opinion of the centre/course coordinator have a genuine reason for being absent may be allowed to appear for a re-exam.

6.2. Students who have failed an exam may be allowed to appear for a re-exam.

6.3. The re-exam should be conducted following the same process as the regular examination.

6.4. Students, who failed/remained absent in the Course End Examination conducted by C-DAC, shall be allowed to appear in the re-examination only once.

6.5. Students who remain absent or fail in the re-examination will not get any further chance for appearing for a third attempt or further. In such case the candidate can receive the Performance Statement and the certificate of participation without any grade.

6.6. On evaluation of their answer sheets 20% of the marks obtained by the students will be deducted (towards de-rating for re-examination) for arriving at the final score, i.e. in order to clear the module test the student has to score a minimum of 50% marks instead of 40%.

8.7 Project Module:

8.7.1. Project work should be start at the time of Software Engg. Module and database design should be complete at the time of Database Technology Module.

8.7.2. After that students should be ready with all mandatory documents with database design and then completion of all teaching modules they can do the project.

8.7.3. Performance in the Project module will be awarded in grade. The Project grade will be mentioned separately on the certificate & will have no effect on the overall grade obtained by a student.

8.7.4. Students may do industry-sponsored projects, but will be required to do the project work within the centre.

8.7.5. Evaluation of the Project module will take place as following:

8.7.5.1. Internal evaluation will be take place at mid of the module

8.7.5.2. External evaluation will take place at the end of the module

Based on both evaluations, final grade will be awarded & communicated to C-DAC ACTS, Pune

8.7.6. **Guidelines for Project Evaluation**

Evaluation of Project work needs to be carried out as per the following guidelines:

a. Literature study.

b. Submission of abstract for their colloquium/seminar/project work along with the references.

c. Submission of the detailed work report

d. Two presentations each for 15 minutes on the work done restricted to 15 – 20 slides followed by evaluation.

e. The evaluation for 100 marks will be splited up as follows:

i.	Literature survey	– 10
ii.	Contents of the project work	– 20
iii.	Contents Flow of Presentation	– 15
iv.	Communication and Presentation Skills	– 20
v.	Depth of Knowledge in the topic	– 15
vi.	Viva Voce	– 15
vii.	Attendance	– 5

f. Soft copy of the presentation should be submitted to C-DAC.

8.8 Ensuring Security of Evaluation data/records:

8.8.1. Ensure that all data relating to evaluation of students is stored in a secure place that cannot be accessed by unauthorized personnel.

8.8.2. All question papers must be prepared and stored in a separate area specifically designated for the purpose.

8.8.3. Whenever any external faculty sets a question paper, ensures that he should follows the guidelines given by C-DAC ACTS Pune.

8.8.4. Ensure that only one copy of any question paper is prepared in physical (printed) form for review and revision.

8.8.5. When the question paper is finalized, print out one master copy and gets it signed by the paper setter, Reviewer and DRM Evaluation.

8.8.6. Prepare required number of photocopies of the question paper and store them in a safe and secure location before the exam.

8.8.7. The data relating to evaluation of students, such as soft copies of question papers and answer keys, student marks database and performance statements etc. must be kept in a separate domain/directory which is accessible only to authorized personnel. Ensure that the data is regularly backed up.

8.8.8. The question papers for the theory as well as the laboratory examinations at all the centres will be set by CDAC, ACTS Pune. The centres according to guidelines provided by C-DAC, ACTS Pune, will conduct the evaluation of the laboratory and assignments locally.

Note: The Evaluation Guidelines, Rules and Regulations issued by C-DAC, ACTS – Pune from time to time shall be binding on all the centers and all the students. C-DAC, ACTS, Pune reserves the right to add, modifies or deletes any or entire contents of this document at any point of time without giving any notice. It’s the responsibility of the centre coordinator to inform such changes to the students in form of a formal notice with a duly signed copy to C-DAC, ACTS, Pune.

9. Requirements (S/W and H/W)

Computing Facilities for C-DAC Certificate courses in Software Testing	
A. Servers	
1. Unix / Linux / Server	
2. Windows 2003 / Windows 2008	
3. Application / Dummy Servers Configured for various modules	
Severs Configuration	
1. Processor (min 3.2 Ghz)	
2. RAM (min 8 GB)	
3 HDD (min 500 GB)	
4. Network Card	
5. AGP Card with 4/8 MB VRAM	
6. 2 Serial ports, 1 parallel port, 104 Keys Keyboard.	
7. DVD RW Drive	
B. Clients Machines Configuration	
1. Processor (Min 3.2 GHz)	
2. RAM (Min 4 GB)	
3. HDD IDE / EIDE (min 250 GB)	
4. AGP-64 bit Card with 8 MB / 4MB VRAM	
5. PCI Network Card 10/100 Base T, UTP Ethernet	
6. Multimedia Kit	
C. Network	
1. 10/100 Base T UTP Hub(s)	
2. UTP CAT-5 Cabling with RJ-45 connectors	
3. UTP Patch Cables	
D. Communication and Internet	
1 Internet Access	
2. ISDN Connectivity	
3. Modem 512 KBPS	
E. Printers	
1. Laser Printer	
F. Additional Lab Equipments	
1. Amplified Speakers, Headphones & Mikes	
2. Hi-Lumen OHP	
3. Video Projector (XGA / SVGA Compatible)	
4. TWAIN Compliant Color Scanner	
G. Module Specific Software Environments, Operating Systems and Hardware	
Fundamentals of Computer & OS Concepts	Open Suse 13.1 / Windows 7
C Programming	gcc 4.9.0 compiler
Software Development Life Cycle	MS Project 2007
OOP with C++ with DS	gcc 4.9.0 compiler
Database Technology	Oracle 11g
Software Testing – Automation	Quick Test Professional (QTP), Selenium
Software Testing – Manual	-