Darshan UNIVERSITY

योगः कर्मसु कौशलम्

Darshan Institute of Eng. and Tech.

Academic Year 2024-25 | Semester-V

Lab Manual

B.Tech.

2101CS503/2301CS502 - Software Engineering

(Theory: 3 | Practical: 2 | Credits: 4)

Prof. R. B. Gondaliya

Sr No	Lab No	Practical Description
1	1	**Introduction of version control using Git Lab**

2	1	**Project Definition and Analysis**
		• Allocation of project title.
		• Write a project abstract (Short description of project minimum 600 word).
		• Write a project detailed description (minimum 1500 word).
		• List out project requirement.
		  ○ Functional requirement
		  ○ Non-Functional requirement

3	2	**Prepare Use case, Activity and Swimlane diagram for given project**
4	3	**Prepare Sequence and State diagram for given project**
5	4	**Prepare Class diagram for given project**
6	5	**Prepare DFD level 0 & level 1 diagram for given project**
7	6	**Project Design** • Prepare screen list. • Design each screen. (for mobile App use MarvelApp and for website use Responsive HTML Template) • Describe each screen along with the fields used in screen.
8	7	**Database Design** • Prepare database schema. • Prepare ER diagram for given project • Manage following information along with schema: Design Date, Verified By, Verification Date, List of Changes, Approved By, Approved Date
9	8	**Prepare API Document (Use POSTMAN tool to save API details). Maintain following information for each API.** • Name of API • Input parameter • Output parameter • Description of API • Use Swagger API for creating dummy API • Refer https://www.getpostman.com for POSTMAN tools details.
10	9	**Test Case Writing** • Write a test case for the screen. • Demonstration of web testing tool Selenium (Automation testing)
11	10	**Compute Function Point value** **Exercise 1(A)**: A system has 10 external inputs, 20 external outputs, 25 different external queries, manages 4 internal logical files, and interface with 4 different legacy system. All of these data are average complexity, and overall system is relatively simple. Compute FP for the system.

		Exercise 2(A): Compute function point value for a project with the following domain characteristics: • Number of user input: 5 • Number of user output: 5 • Number of user enquires: 6 • Number of files: 5 • Number of external interfaces: 5 • Assume that all the complexity adjustment values are simple. Where Σ (Fi) =25 **Exercise 3(A):** Given the following values, compute function point when all complexity adjustment factor (CAF) and weighting factors are average. • User Input = 50 • User Output = 40 • User Inquiries = 35 • User Files = 6 • External Interface = 4 **Exercise 4(B):** Compute the function point for the following data • Number of user inputs = 24 • Number of user outputs = 46 • Number of inquiries = 8 • Number of files = 4 • Number of external interfaces = 2 • all complexity adjustment factor (CAF) is Significant and weighting factors are average. **Exercise 5(C):** Calculate the function point, for software application with multiple Processing Factors 5, 1, 0, 4, 3, 5, 4, 3, 4, 5, 2, 3, 4, 2 by using following given Date: • The number of EI(Avg): 22, • The number of EO(Low): 45, • The number of EI(High): 06, • The number of ILF(Avg): 05, • The number of ELF(Low): 02,
12	11	**Using COCOMO model, estimate the effort and development time based on given project details** **Exercise 1(A):** Consider a software project using semi-detached mode with 300 KLOC. find out effort estimation, development time, and person estimation. **Exercise 2(A):** Consider project was estimated with a size of 30 KLOC. Calculate the Effort, scheduled time for Embedded project.

Exercise 3(B): Consider a project having 30,000 lines of code

- Consider a semi-detached mode software
- Consider an embedded software with critical area hence reliability is high.
- Estimate Cost and schedule for different phases.

Exercise 4(B): Consider a project to develop a full screen editor. The major components identified are (1) Screen edit, (2) Command Language Interpreter, (3) File input and output, (4) Cursor movement and (5) Screen movement. The sizes for these are estimated to be 4K, 2K, 1K, 2K and 3K delivered source code lines. Use COCOMO model to determine Cost and schedule estimates for different phases.

Exercise 5(C): Considering your immense expertise in software development, the absolute beginners Inc. Has recently allotted you a mega project. The goal of the project is to create a database of all Hindi films released since 2000. The software would allow one to generate a list of top ten hit films, top ten flop films, best comedy films, and so on. Using your prior experience you have decided the approximate sizes of each module of the software as follow:

- Data entry (0.9 KDSI)
- Data update (0.7 KDSI)
- Query (0.9 KDSI)
- Report generation and display (2 KDSI)

Also take into consideration the following cost drivers with their ratings:
Cost Drivers

- Storage constraints (low)
- Experience in developing similar software (high)
- Programming capabilities of the developers (high)
- Application of software engineering methods (high)
- Use of software tools (high)

Solve the problem by applying basic and intermediate COCOMO model

- Find project type.
- Find project size.
- Find initial effort estimation.
- Find adjusted effort estimation.
- Find schedule.
- Find minimum size of the team you would require to develop this system.

Assuming that your client would pay Rs. 50,000 per month of development, how much would be the likely billing?

13	12	**Draw a control flow diagram and apply cyclomatic complexity for the given codes. Be sure about following points.**
		• Guarantees that all independent execution path is exercised at least once.
		• Guarantees that both the true and false side of all logical decisions are exercised.
		• Executes the loop at the boundary values and within the boundaries.
		• Identify numbers of independence path require for testing.
		Exercise 1(A)
		`void main(){`
		`  int i,j,k;`
		`  readln (i,j,k);`
		`  if( (i < j) || ( i > k) ){`
		`  writeln("then part");`
		`  if (j < k)`
		`  writeln ("j less then k");`
		`  else writeln ("j not less than k");}`
		`  else writeln("else Part");`
		`}`
		Exercise 2(A)
		`i = 0;`
		`n=4; //N-Number of nodes present in the graph`
		`while (i<n-1) do`
		`  j = i + 1;`
		`  while (j<n) do`
		`  if A[i]<A[j] then`
		`  swap(A[i], A[j]);`
		`  end do;`
		`  i=i+1;`
		`end do;`

Exercise 3(B)

```
public class CyclomaticComplexityDemo {
public static void main(String[] args) {
// TODO Auto-generated method stub
        int var1 = 10;
        int var2 = 9;
        int var3 = 8;
        int var4 = 7;
        if (var1 == 10){
                if(var2 > var3){
                        var2 = var3;
                }
                else{
                        if (var3 > var4){
                                var3 = var4;
                        }
                        else{
                                var4 = var1;
                        }
                }
        }
        else{
        var1=var4;
        }
        System.out.println("Printing value for var1, var2, var3, and
var4"+var1+" "+var2+" "+var3+" "+var4);
        }
}
```

Exercise 4(C)

```
public Hashtable countAlphabet(String aString)
{
        Hashtable table = new Hashtable();
        If (aString.length > 4000)
                return table;
        StringBuffer buffer = new StringBuffer(aString);
        While (buffer.length() > 0){
                 String firstChar = buffer.substring(0, 1);
                Integer count = (Integer)table.get(firstChar);
                if (count == null){
                        count = new Integer(1);
                }
                else{
                         count = new Integer(count.intValue() + 1);
                }
        table.put(firstChar, count);
        buffer.delete(0, 1);
        }
        Return table;
}
```