A Laboratory Manual For

Software Engineering
(22413)

Semester –IV

Maharashtra State

Board of Technical Education, Mumbai

(Autonomous) (ISO 9001 : 2015) (lSO/IEC 27001 : 2013)

Maharashtra State Board of Technical Education,
(Autonomous) (ISO 9001 : 2008) (lSO/IEC 27001 : 2005)
4th Floor, Government Polytechnic Building, 49, Kherwadi, Bandra (East), Mumbai
400051.
(Printed on December, 2018)

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that Mr. / Ms:………………………………………………………………............. Roll No............................,…………. of Fourth Semester of Diploma in…………………………….. of Institute...........……………………………………………………… (Code:……….......) has completed the term work satisfactorily in course - **Software Engineering (22413)** for the academic year 20…. - 20…… as Prescribed in curriculum

Place:…………………… Enrollment No:…………………………………

Date: …………………….. Exam. Seat No:………………………………...

Subject Teacher Head of Department Principal

Sr. No.	CO	PrO	Name of Experiment	Performance date	Actual Date	Mark(25)	Remark
1	CO1	PrO1	Write a Problem Statement to define a title of the project with bounded scope of project				
2	CO1	PrO2	Select relevant process model to define activities and related task set for assighned project				
3	CO2	PrO3	Gather Application Specific requirementsfor assimilate into RE model				
4	CO2	PrO4	Prepare broad SRS (Software requitement Software) for the above selected projects				
5	CO2	PrO5	Prepare USE Cases and and Draw Use Case Diagram using modelling Tool				
6	CO3	PrO6	Develop the activity diagram to represent flow from one activity to another for software development				
7	CO3	PrO7	Develop data Designs using DFD Decision Table & ER				
8	CO3,CO4	PrO8	Draw class diagram,sequence diagram, Collaboration Diagram, State Transition Diagram for the assigned project				
9	CO3,CO4	PrO9	Write Test Cases to Validate requirements of assigned project from SRS Document				
10	CO4	PrO10	Identify the risk involved in the project and prepare RMMM plan				
11	CO4	PrO11	Evaluate Size of the project using function point metric for the assigned project				
12	CO4	PrO12	Estimate cost of the project using COCOMO and COCOCMOII for the assigned project				
13	CO3,CO4	PrO13	Use CPM/PERT for scheduling the assigned project				
14	CO3,CO4	PrO14	Use timeline Charts or Gantt Charts to track progress of the assigned project				
15	CO5	PrO15	Prepare SQA Plan that facilitates various attributes of quality of the process				
16	CO5	PrO16	Prepare SQA Plan that facilitates various attributes of quality of the product				

PRATICAL -1

Write Problem statement to define the Project Title with bounded scope of the project.

PROJECT(Write your project title):
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

To analyze the basic requirement of software product and to generate problem statement and to analyze the bounded scope of the software product.

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] COMPETENCY AND PRACTICAL SKILLS

This practical is expected to develop the following skills.

- Write a program statement while deciding title of the project text editors.
- You can use Microsoft World, Notepad etc. for writing problem statement.

IV] RELEVANT COURSE OUTCOMES:-

Co 1:- select suitable process model for software development.

Co 2:- prepare software requirement specification.

V] PRACTICAL OUTCOMES:-

Write a problem statement to define the perfect title the perfect title with bounded scope of the project.

VI] RELEVANT AFFECTIVE DO MAIN RELATED OUTCOMES:-

1. Follow precaution measured.

2. Demonstrate working as a leader is a team member.
3. Follow ethical practices.

VII] MINIMUM THEROTICAL BACKGROUND:-

A problem statement is clear consist description of the issue that needs to be addressed by a problem solving team. It is used to center and focus the team at the beginning keep the team on track during the effects delivered on outcomes that solves the problem statement.

- **How to write problem statement: -** Write down your vision to in order to decided what must, be done while, solving. The pattern it’s important to understand the vision. It’s an initial starting point for a project. A problem statement expressed the words that will be used to keep the efforts focused and it should represent a solvable problem. The file can be used to spark the discussion about the problem.

1) How to write a program statement.
2) How to get it started.
3) Who does the problem affect?
4) What are the boundaries of the problem?
5) When the issue does occurs.
6) Where is the issue programming.
7) Why it is important to fix the program.

VIII] PROBLEM STATEMENTS FOR A PROJECT

1. Describe ideal state of affairs.
2. Explain your problem
3. Explain your problem financial cost
4. Backup your assertions
5. Purpose your solution
6. Explain the benefits of solution
7. Conclude by summarizing the problems and solutions
8. Write down the thesis statements

IX] RESOURCES REQUIRED:-

1) Hardware personal computer(i3-i5 preferable)
2) Operating System Windows 7/ windows 8/ windows 10/ linux or any other
3) Software tools:- Any UML tools

X] LIST OF STUDENTS AND TEAM MEMBERS

1)……………………………………………………………………………………………..
2)………………………………………………………………………………………………
3)………………………………………………………………………………………………
4)………………………………………………………………………………………………

XI] CONCLUSION

…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………

XII] EXERCISE:

1. Describe the characteristics of software.
2. Write note on development framework.

…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………

XIII] REFERENCES:-

2) http://www.google.com
3) http://softwarebaba.com
4) Software Engineering by press man

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -2

Select relevant process model of define activities & related tasks set of assignment project.

PROJECT(Write your project title):
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

To select and analyze the relevant model for assigned project and to define activities and related tasks set for assigned project.

II] RELEVANT PROGRAM OUTCOMES.

Po1 - Basic Knowledge.

Po2 - Discipline Knowledge.

Po3 - Individual & Teamwork.

Po4 - Communication.

Po5 - Lifelong learning.

III] RELEVANT OUTCOME SOURCE.

Select suitable process model for software development.

IV] PRACTICAL OUTCOMES.

Write a program statement to define project. Project title with bounded scope the project.

V] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES.

1. Follow precautionally measures.

2. Demonstrate working as a leader or two members.

3. Follow ethical practices.

VI] MINIMUM THEORATICAL BACKGROUND.

PROCESS MODEL:-

A software process model is define simplified process representation of software process each methods represent a process from a specific perspective.

WATERFALL MODEL:-

The waterfall model is also called as ' the linear-sequential model ' or ' classic life ' cycle model. The software development starts with requirements gathering phase. Then progress through analysis, design, coding, testing & maintenance.

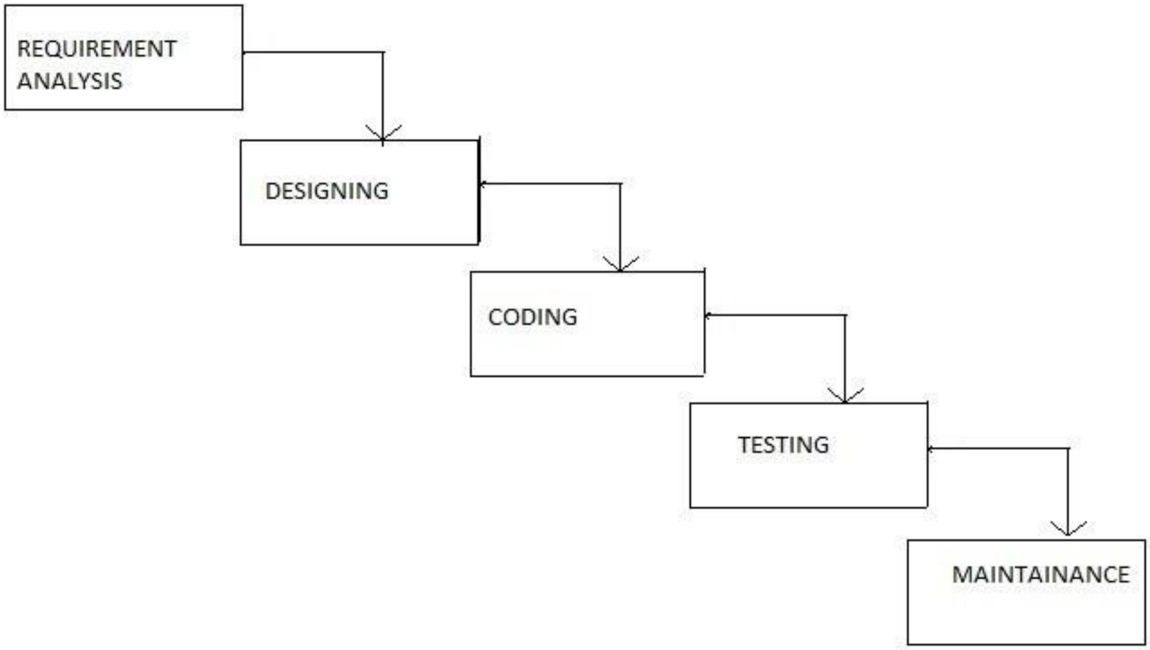


Fig: Water fall Model

ADVANTAGES OF WATERFALL MODEL.

1. It is very simple to understand.
2. For implementation of small systems waterfall model is useful.

DISADVANTAGES OF WATERFALL MODEL.

1. It is not useful for large projects.
2. It is very difficult to modify systems requirement if the middle of the development process.
3. It is not suitable for projects in which requirement software are not clear initially.

We have used waterfall model for the projects because requirements of Projects because requirements of projects are very well known. Clear and fixed our projects is very small to be implemented. In our project there is no need of customer involvement in the project. Development cycle product definition of project are not changed the frequently. In our project there is no need of our participation in all phase. Software the waterfall model is suitable process model for our project.

VII] CONCLUSION:

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

VIII] EXERCISE:

1. Explain waterfall model.
2. Explain the basic process frame work activities.
3. Select relevant process model of define activities & related tasks set of your project.

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……

……

……

……

……

……

……

……

……

……

……

……

IX] REFERENCES:-

1. Textbook of software engineering by pressman.
2. www.thesoftwareexperts.com
3. www.google.com

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -3

Gather Application Specific requirements for assimilate into RE model

PROJECT(Write your project title):
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

Gather Application Specific requirements for assimilate into RE model. The functional requirement is organized in two section; Requirement of your project and requirement of your personal computer.

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] COMPETENCY AND PRACTICAL SKILLS

This practical is expected to develop the following skills.

- Gather application specific requirement for your project.
- You can use Microsoft World, Notepad etc. for writing problem statement.

IV] RELEVANT COURSE OUTCOMES:-

Co 1:- select suitable process model for software development.

Co 2:- prepare software requirement specification.

V] PRACTICAL OUTCOMES:-

Gathered functional and nonfunctional requirement for given project.

VI] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES:-

4. Follow the project title.

5. Collect functional and nonfunctional requirement.

VII] MINIMUM THEROTICAL BACKGROUND:-

1. Functional Requirement: ……………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ………….

2. Nonfunctional requirement:
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………
   ……………………………………………………………………………………………

VIII] RESOURCES REQUIRED:-

1. Hardware personal computer(i3-i5 preferable)
2. Operating System Windows 7/ windows 8/ windows 10/ linux or any other

IX] CONCLUSION:

……
……
……

X] EXERCISE:

1. Gather and write application specific requirement for your any micro project.

2. What is requirement elicitation write in detail?

………

………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………
………

………

………
………
………
………
………
………
………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

XI] REFERENCES:-

5) http://www.google.com
6) http://softwarebaba.com
7) Software Engineering by press man

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -4

Prepare broad SRS (Software requirement Software) for the above selected projects

PROJECT(Write your project title):
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

Prepare broad SRS (Software requirement Software) for the above selected projects

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] PRACTICAL OUTCOME

Prepared broad software requirement specification for the above defined project.

IV] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES:-

1. Follow the project title.
2. Demonstrate working at a leader/team member.

V] MINIMUM THEROTICAL BACKGROUND (Related with your project):-

- PURPOSE:
  …………………………………………………………………………………………
  …………………………………………………………………………………………
  …………………………………………………………………………………………
- SCOPE OF PROJECT:
  …………………………………………………………………………………………
  …………………………………………………………………………………………
  …………………………………………………………………………………………
  …………………………………………………………………………………………
- THE OVERALL DESCRIPTION:
  …………………………………………………………………………………………
  …………………………………………………………………………………………

………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………
………………………………………………………………………………………………

- SECURITY REQUIREMENT:
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
- RELIABILITY:
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
- EXTENSIBILITY:
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………
  ………………………………………………………………………………………………

VI] RESOURCES REQUIRED:-

1. Hardware personal computer(i3-i5 preferable)
2. Operating System Windows 7/ windows 8/ windows 10/ linux or any other

VII] CONCLUSION:

………………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

VIII] REFERENCES:-

a. http://www.google.com
b. http://softwarebaba.com
c. Software Engineering by press man

Marks obtained			Sign of Teacher
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -5

Prepare USE Cases and Draw Use Case Diagram using modelling Tool

PROJECT(Write your project title):
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………
…………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

A use case diagram software and system engineering term that describe how a user uses a system to accomplish particular goal. A use case act as a software modeling technique that defines the features to be implemented and the resolution of any errors that may be encountered.

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] PRACTICAL OUTCOME

To prepare use case and draw use case diagram using software modelling tool.

IV] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES:-

1. Follow the project title.
2. Follow properly method.
3. Demonstrate working at a leader/team member.
4. Use current tool for practical.

V] MINIMUM THEROTICAL BACKGROUND (Related with your project):-

- Use case diagram:
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………

- Actor:
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
- Use case:
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
- Relationship:
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………
  ……………………………………………………………………………………………

VI] RESOURCES REQUIRED:-

1. Hardware personal computer(i3-i5 preferable)
2. Operating System Windows 7/ windows 8/ windows 10/ linux or any other

VII] CONCLUSION:

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………

VIII] REFERENCES:-

d. http://www.google.com
e. http://softwarebaba.com
f. Software Engineering by press man

IX] PRACTICAL RELATED QUESTION:

1. Describe relationship between use cases.
2. Explain categories of actors.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -6

Develop the activity diagram to represent flow from one activity to another for software development

PROJECT(Write your project title):

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

Develop the activity diagram to represent flow from one activity to another for software development

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] PRACTICAL OUTCOME

To prepare activity diagram for represent flow.

IV] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES:-

- Follow the project title.
- Follow properly method.
- Demonstrate working at a leader/team member.
- Use current tool for practical.

V] MINIMUM THEROTICAL BACKGROUND (Related with your project):-

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………

VI] RESOURCES REQUIRED:-

- Hardware personal computer(i3-i5 preferable)
- Operating System Windows 7/ windows 8/ windows 10/ linux or any other

VII] CONCLUSION:

……………………………………………………………………………………………
……………………………………………………………………………………………
……………………………………………………………………………………………

VIII] REFERENCES:-

- http://www.google.com
- http://softwarebaba.com
- Software Engineering by press man

IX] PRACTICAL RELATED QUESTION:

1. Draw activity diagram for above defined project

……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………
……………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -7

Develop data Designs using DFD, Decision Table & ER diagram

PROJECT(Write your project title):

……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………
……………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

There are several notation as well as languages exists for the purpose of representation software design. Some of them help to describe a structural organization, some are used to represent behavior of system.

- DFD (Data Flow Diagram).
- ERD (Entity Relationship Diagram)
- Decision Table.

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] PRACTICAL OUTCOME

We develop data Designs using DFD Decision Table & ER diagram

IV] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES:-

- Follow the project title.
- Follow properly method.
- Demonstrate working at a leader/team member.
- Use current tool for practical.

V] MINIMUM THEROTICAL BACKGROUND (Related with your project):-

Level 0 DFD:

……………………………………………………………………………………………………
……………………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

Level 1 DFD:

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

Level 2 DFD:

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

ER Diagram:

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

Decision Table :

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

………

………

………

VI] RESOURCES REQUIRED:-

- Hardware personal computer(i3-i5 preferable)
- Operating System Windows 7/ windows 8/ windows 10/ linux or any other

VII] CONCLUSION:

………

………

………

VIII] REFERENCES:-

- http://www.google.com
- http://softwarebaba.com
- Software Engineering by press man

IX] PRACTICAL RELATED QUESTION:

1. Explain term attribute with example.
2. Explain the symbol of DFD.

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -8

Draw class diagram, sequence diagram, Collaboration Diagram, State Transition Diagram for the assigned project.

PROJECT(Write your project title):

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

I] PRATICAL SIGNIFICANCE

There are several notation as well as languages exists for the purpose of representation software design. Some of them help to describe a structural organization, some are used to represent behavior of system.

- Class diagram
- Sequence diagram
- Collaboration diagram
- State transition diagram.

II] RELEVANT PROGRAM OUTCOMES:-

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III] PRACTICAL OUTCOME

We draw class diagram, sequence diagram, Collaboration Diagram, State Transition Diagram for the assigned project.

IV] RELEVANT AFFECTIVE DOMAIN RELATED OUTCOMES:-

- Follow the project title.
- Follow properly method.
- Demonstrate working at a leader/team member.
- Use current tool for practical.

V] MINIMUM THEROTICAL BACKGROUND (Related with your project):-

- Class diagram :
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
- Sequence diagram
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ……………………………………………………………………………
- Collaboration diagram:
  ………………………………………………………………………………………
  ………………………………………………………………………………………
  ………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

…………………………………………………………………………

- State transition diagram:

.………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

…

VI] RESOURCES REQUIRED:-

- Hardware personal computer(i3-i5 preferable)
- Operating System Windows 7/ windows 8/ windows 10/ linux or any other

VII] CONCLUSION:

……

……………………………………………………………………………………………

……………………………………………………………………………………………

VIII] REFERENCES:-

- http://www.google.com
- http://softwarebaba.com
- Software Engineering by press man

IX] PRACTICAL RELATED QUESTION:

1. Draw symbol used in:

- Class diagram.
- Sequence diagram.
- Collaboration diagram.
- State transition diagram.

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

Marks obtained			**Sign of Teacher**
Process Related(7.5)	**Product Related(17.5)**	**Total(25)**	

PRATICAL -9

Write test cases to validate requirements of assigned project from SRS

I. Practical Significance

SRS is a document that is created by the development team in collaboration with business analysts and environment/data teams. This practical is useful for writing test cases from SRS(**Software requirements specification)**

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication.

III. Competency and Practical skills

Apply basic and fundamental knowledge of writing test cases.

IV. Relevant Course Outcomes

Basic Concept of writing test cases.

V. Practical Outcomes (PrOs)

Write test cases to validate requirements.

VI. Relevant Affective domain related Outcomes

1. Follow safety practices
2. Practice good Housekeeping
3. Demonstrate working as a leader/team member
4. Follow ethical practices

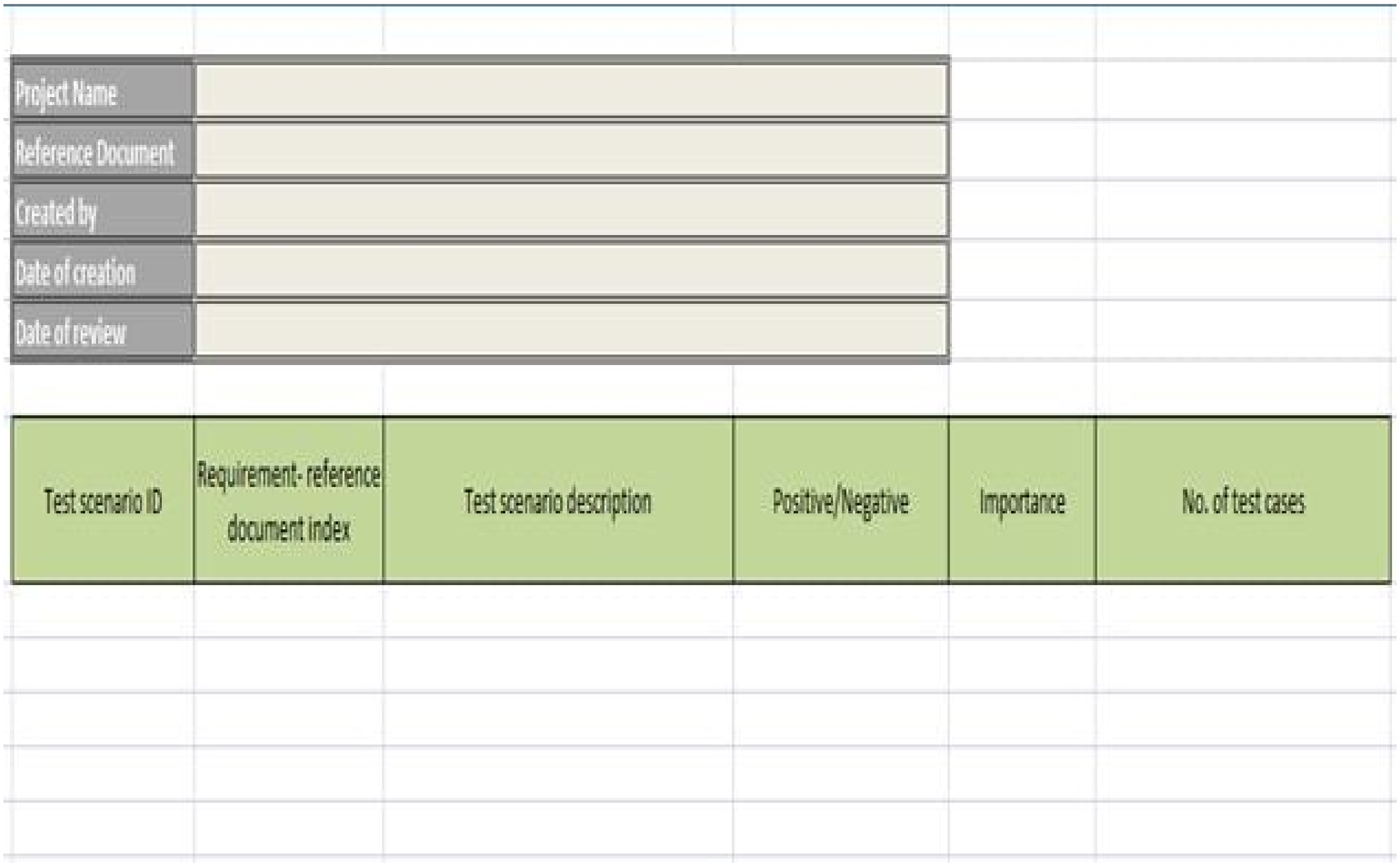
VII. Minimum Theoretical Background

What is an SRS?

SRS is a document that is created by the development team in collaboration with business analysts and environment/data teams. Typically, this document once finalized will be shared with the QA team via a meeting where a detailed walkthrough is arranged.

VIII. Diagrams / Experimental set-up /Work Situation

To create a template for Test scenarios/cases:



Project Name	
Reference Document	
Created by	
Date of creation	
Date of review	

Test scenario ID	Requirement- reference document index	Test scenario description	Positive/Negative	Importance	No. of test cases

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System	Any desktop with basic configuration	One computer system for each student	

2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

What do we need to get started?

- The correct version of the SRS document
- Clear instructions on who is going to work on what and how much time have they got.
- A template to create test scenarios
- Other information on- who to contact in case of a question or who to report in case of a documentation inconsistency

X Precaution

Handle computer system with care

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1. What is SRS?
2. Explain benefits of SRS document.
3. Explain different fields in test case writing.
4. Write test cases for testing login functionality of any web application,

XIV Exercise

(Space for Answer)

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……..…..……………………………………………………………………………………

…………………………..

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……..…..………………………………………………………………………………………

………………………..

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

XV References/ Suggestions for further Reading

https://en.wikipedia.org/wiki/Software_requirements_specification#References

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	Product Related (17.5)	Total(25)	

PRATICAL -10

Identify risks involved in the project and prepare RMMM plan.

I. Practical Significance

Risk is an expectation of loss, a potential problem that may or may not occur in the future. It is generally caused due to lack of information, control or time.A possibility of suffering from loss in **software** development process is called a **software risk**.This practical is useful for identifying risks involved in the project.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

Apply basic and fundamental knowledge to identify risk.

Relevant Course Outcomes

Apply the RMMM plan to identify risk.

IV. Practical Outcomes (PrOs)

Identify risk & calculate risk probabilities along with qualitative risk impacts for software under development.

V. Relevant Affective domain related Outcomes

1. Follow safety practices
2. Practice good Housekeeping
3. Demonstrate working as a leader/team member
4. Follow ethical practices

VI. Minimum Theoretical Background

What is a risk?

- The risk denotes the uncertainty that may occur in the choices due to past actions and risk is something which causes heavy losses.

- Risk management refers to the process of making decisions based on an evaluation of the factors that threats to the business.

  Various **activities** that are carried out for **risk management** are-

  • Risk identification

  • Risk projection

  • Risk refinement

  • Risk mitigation, monitoring and management.

VII. Diagrams / Experimental set-up /Work Situation

-Not Applicable-

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System	Any desktop with basic configuration	One computer system for each student	
2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

Prepare Risk Table & RMMM plan Template for Risk table
➢ Template for Risk table:
No. Risk Category Probability Impact RMMM

➢ Risk Table Construction
• List all risks in the first column of the table
• Classify each risk and enter the category label in column two
• Determine a probability for each risk and enter it into column three

- Enter the severity of each risk (negligible, marginal, critical, and catastrophic)
in column four.

X Precaution

Handle computer system with care

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1. **What are the software Risks?**
2. **What is risk management in software development?**
3. **Explain RMMM and RMMM plan.**

XIV Exercise

(Space for Answer)

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

………………………………….…..……………………………………………

………………………………………………………………

XV References/ Suggestions for further Reading

https://www.test-institute.org/What_Is_Software_Risk_And_Software_Risk_Management...

<table>
<tr><th colspan="3">Marks Obtained</th><th>Dated Signature of Teacher</th></tr>
<tr><th>Process Related(7.5)</th><th>Product Related (17.5)</th><th>Total(25)</th><td rowspan="2"></td></tr>
<tr><td></td><td></td><td></td></tr>
</table>

PRATICAL -11

Evaluate size of the project using Function point metric for the assigned project.

I. Practical Significance

Since ‘functionality’ cannot be measured directly, it must be derived indirectly using other direct measures. Function-oriented metrics uses a measure called the *function point.* Function points are derived using an empirical relationship based on countable (direct) measures of software's information domain and assessments of software complexity. This practical is useful for evaluating size of the project.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

1.Apply basic and fundamental knowledge of calculating size of the project.

2.Use different techniques to Evaluate size of the project

IV. Relevant Course Outcomes

Evaluate size of the project using Function point metric.

V. Practical Outcomes (PrOs)

Calculate size of the project using function point metric for software under development.

VI. Relevant Affective domain related Outcomes

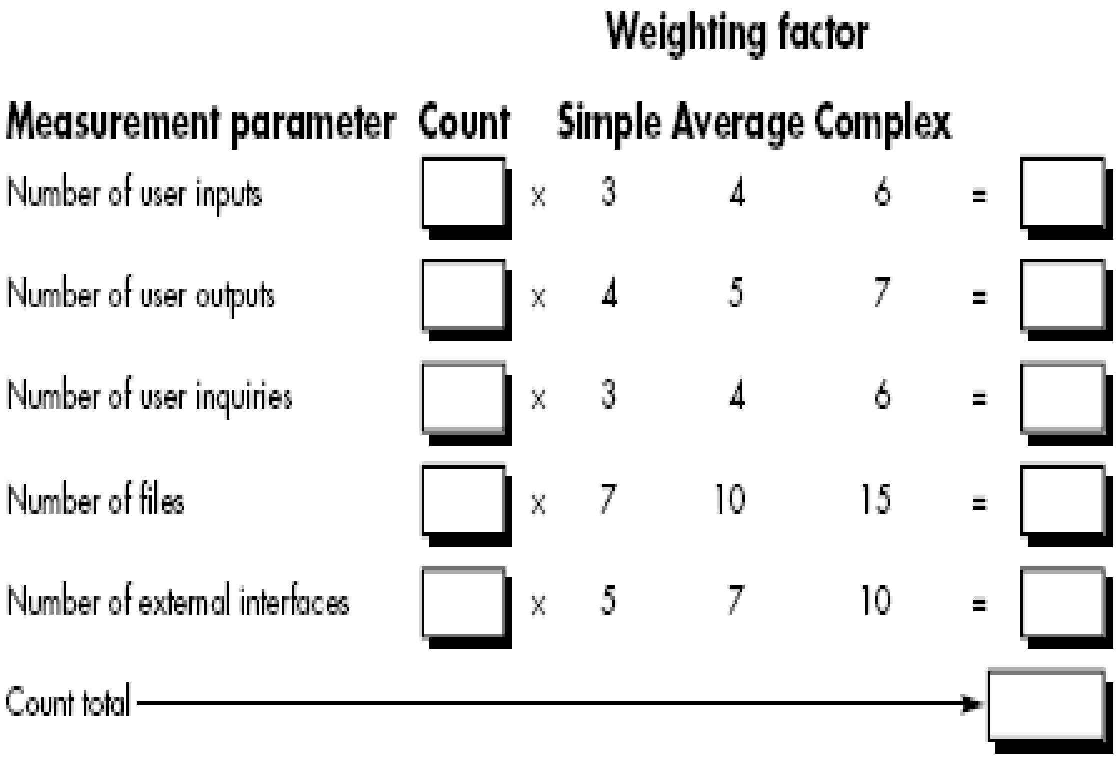
1. Follow safety practices
2. Practice good Housekeeping
3. Demonstrate working as a leader/team member
4. Follow ethical practices

VII. Minimum Theoretical Background

Function points are computed by completing the table shown in Figure (A). Five information domain characteristics are determined and counts are provided in the appropriate table location. Information domain values are defined in the following manner:

- **Number of user inputs.** Each user input that provides distinct application oriented data to the software is counted. Inputs should be distinguished from inquiries, which are counted separately.
- **Number of user outputs.** Each user output that provides application-oriented information to the user is counted. In this context output refers to reports, screens, error messages, etc. Individual data items within a report are not counted separately.
- **Number of user inquiries.** An inquiry is defined as an on-line input that results in the generation of some immediate software response in the form of an on-line output. Each distinct inquiry is counted.
- **Number of files.** Each logical master file (i.e., a logical grouping of data that may be one part of a large database or a separate file) is Counted.
- **Number of external interfaces.** All machine-readable interfaces (e.g., data files on storage media) that are used to transmit information to another system are counted.

VII Diagrams / Experimental set-up /Work Situation



Measurement parameter	Count		Weighting factor Simple	 Average	 Complex		
Number of user inputs	□	x	3	4	6	=	□
Number of user outputs	□	x	4	5	7	=	□
Number of user inquiries	□	x	3	4	6	=	□
Number of files	□	x	7	10	15	=	□
Number of external interfaces	□	x	5	7	10	=	□
Count total →							□

FIGURE (A) Computing function point

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks

1	Computer System	Any desktop with basic configuration	One computer system for each student	
2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

Once these data have been collected, a complexity value is associated with each count. Organizations that use function point methods develop criteria for determining whether a particular entry is simple, average, or complex.Nonetheless,the determination of complexity is somewhat subjective.

To compute function points (FP), the following relationship is used:

$$\text{FP} = \text{count total} * [0.65 + 0.01 * \Sigma(\mathit{Fi})] \qquad (1)$$

Where,count total is the sum of all FP entries obtained from Figure (A).

X Precaution

Handle computer system with care

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1. **What is estimation techniques?**
2. **What are Function points in software engineering?**
3. **How many Information Domain Values are used for Function Point Computation?**

XIV Exercise

(Space for Answer)

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

……………………………………..….…………………………………………………………

……………………………………………………………………..

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

………………………………………..….…………………………………………………

………………………………………………………………………..

XV References/ Suggestions for further Reading

https://www.tutorialspoint.com/estimation_techniques/estimation_techniques_function_points.htm.

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	**Product Related (17.5)**	**Total(25)**	

PRATICAL -12

Estimate cost of the project using COCOMO/COCOMO II approach for the assigned project.

I. Practical Significance

COCOMO is one of the most widely used software estimation models in the world. This model is developed in 1981 by Barry Boehm to give an estimate of the number of man-months it will take to develop a software product. COCOMO stands for "Constructive Cost Model". This practical is useful for estimating cost of the project using COCOMO/COCOMO II approach.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

Apply basic and fundamental knowledge to calculate size using COCOMO technique.

IV. Relevant Course Outcomes

Evaluate the size of the given software using COCOMO.

V. Practical Outcomes (PrOs)

Estimate cost of the project using COCOMO/COCOMO II approach for the assigned project.

VI. Relevant Affective domain related Outcomes

1. Follow safety practices
2. Practice good Housekeeping
3. Demonstrate working as a leader/team member
4. Follow ethical practices

VII. Minimum Theoretical Background

COCOMO has three different models that reflects the complexity-

- Basic model
- Intermediate model
- Detailed model

COCOMO applies to three classes of software projects:

1) Organic mode:

In this mode, relatively small, simple software projects with a small team are handled.

Such a team should have good application experience to less rigid requirements.

2) Semi-detached projects:

In this class an intermediate projects in which teams with mixed experience level are handled .Such projects may have mix of rigid and less than rigid requirements.

3) Embedded projects:

In this class, projects with tight hardware, software and operational constraints are handled.

1.**Basic COCOMO** computes software development effort (and cost) as a function of program size. Program size is expressed in estimated thousands of source lines of code (KLOC).

The basic COCOMO equations take the form

Effort Applied (E) = $a_b(\text{KLOC})^{b}{}_{b}$ [man-months]

Development Time (D) = $c_b(\text{Effort Applied})^{d}{}_{b}$ [months]

People required (P) = Effort Applied / Development Time [count]

Where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for project.

2. **Intermediate COCOMO** computes software development effort as function of program size and a set of "cost drivers" that include subjective assessment of product,hardware, personnel and project attributes.

3**.Detailed COCOMO** incorporates all characteristics of the intermediate version with an assessment of the cost driver's impact on each step (analysis, design, etc.) of the software engineering process.

The Six phases of detailed COCOMO are:-

- planning and requirements
- system design
- detailed design
- module code and test
- integration and test
- Cost Constructive model

VII Diagrams / Experimental set-up /Work Situation

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System	Any desktop with basic configuration	One computer system for each student	
2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

The basic COCOMO model gives an approximate estimate of the project parameters. The basic COCOMO estimation model is given by the following expressions:

Effort = a1 x (KLOC)a 2 PM Tdev = b1 x (Effort)b 2 Months

Where,

• KLOC is the estimated size of the software product expressed in Kilo LinesofCode

• a1, a2, b1, b2 are constants for each category of software products,

• Tdev is the estimated time to develop the software, expressed in months,

• Effort is the total effort required to develop the software product, expressed in person months (PMs).

Example: Assume that the size of an organic type software product has been estimated to be 32,000 lines of source code. Assume that the average salary of software engineers be Rs. 15,000/- per month. Determine the effort required to develop the software product and the nominal development time.

=From the basic COCOMO estimation formula for organic software: Effort = 2.4 x (32)1.05 = 91 PM Nominal development time = 2.5 x (91)0.38 = 14 months

Cost required to develop the product = 14 x 15,000 = Rs. 210,000/-

X Precaution

Handle computer system with care.

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1. **Explain COCOMO II Model.**
2. **What is cost estimation?**
3. **What is LOC in cost estimation?**

XIV Exercise

(Space for Answer)

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

……………………………………………………………………………………………

……………………………………………………………………………………………

…………………………………..….…..…………………………………………………

……………………………………………………………………..

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

XV References/ Suggestions for further Reading

http://www.mhhe.com/engcs/compsci/pressman/information/olc/COCOMO.html

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	Product Related (17.5)	Total(25)	

PRATICAL -13

Use CPM/PERT for scheduling the assigned project.

I. Practical Significance

To understand Project management software for scheduling, cost control and budget management, resource allocation, collaboration software, communication, quality management and documentation or administration systems which are used to deal with the complexity of large projects .

A Timeline chart& project table helps in planning, how long that project might take and that can really save time in the end and more so if you are on a deadline This practical is useful for scheduling the project using CPM/PERT technique.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

Apply basic and fundamental knowledge for scheduling the project.

IV. Relevant Course Outcomes

Apply the project scheduling technique for identified project.

V. Practical Outcomes (PrOs)

Use CPM/PERT for scheduling the assigned project.

VI. Relevant Affective domain related Outcomes

1. Follow safety practices
2. Practice good Housekeeping
3. Demonstrate working as a leader/team member
4. Follow ethical practices

VII. Minimum Theoretical Background

- Introduction

Basically, PERT (Programed Evaluation Review Technique) and CPM (Critical Path Method) are project management techniques, which have been created out of the need of Western industrial and military establishments to plan, schedule and control complex projects.

•Planning, Scheduling (or organizing) and Control are considered to be basic Managerial functions, and CPM/PERT has been rightfully accorded due importance in the literature on Operations Research and Quantitative Analysis. PERT/CPM provided a focus around which managers could brain-storm and put their ideas together. It proved to be a great communication medium by which thinkers and planners at one level could communicate their ideas, their doubts and fears to another level. Most important, it became a useful tool for evaluating the performance of individuals and teams.

VII Diagrams / Experimental set-up /Work Situation

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System	Any desktop with basic configuration	One computer system for each student	

2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

Calculating Critical Path is a simple 4-step process.

Step 1: Find Activities

Step 2: Build Schedule Network Diagram

Step 3: Find all Possible Paths

Step 4: Calculate Duration for Each Path

Students should define activities planned for their case study & use PERT &CPM Method to draw Timeline chart & project table.

X Precaution

Handle computer system with care

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1.What is the project scheduling?

2.What are scheduling techniques?

3.Can a project have two critical paths?

XIV Exercise

(Space for Answer)

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

…………………………………………………………………………………………

XV References/ Suggestions for further Reading

https://www.researchgate.net/publication/313250131_PERT_and_CPM

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	Product Related (17.5)	Total(25)	

PRATICAL -14

Use Timeline charts/Gantt charts to track progress of the project.

I. Practical Significance

A **Gantt chart** is a visual view of tasks scheduled over time. **Gantt charts** are used for planning **projects** of all sizes and they are a useful way of showing what work is scheduled to be done on a specific day. They also help you view the start and end dates of a **project** in one simple view. This practical is useful for tracking progress of the project.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

Apply basic and fundamental knowledge of gnatt chart to track progress of the project.

IV. Relevant Course Outcomes

Apply basic and fundamental knowledge of gnatt chart to track progress of the project.

V. Practical Outcomes (PrOs)

Apply basic and fundamental knowledge of gnatt chart to track progress of the project.

VI. Relevant Affective domain related Outcomes

1. Follow safety practices
2. Practice good Housekeeping
3. Demonstrate working as a leader/team member
4. Follow ethical practices

VII. Minimum Theoretical Background

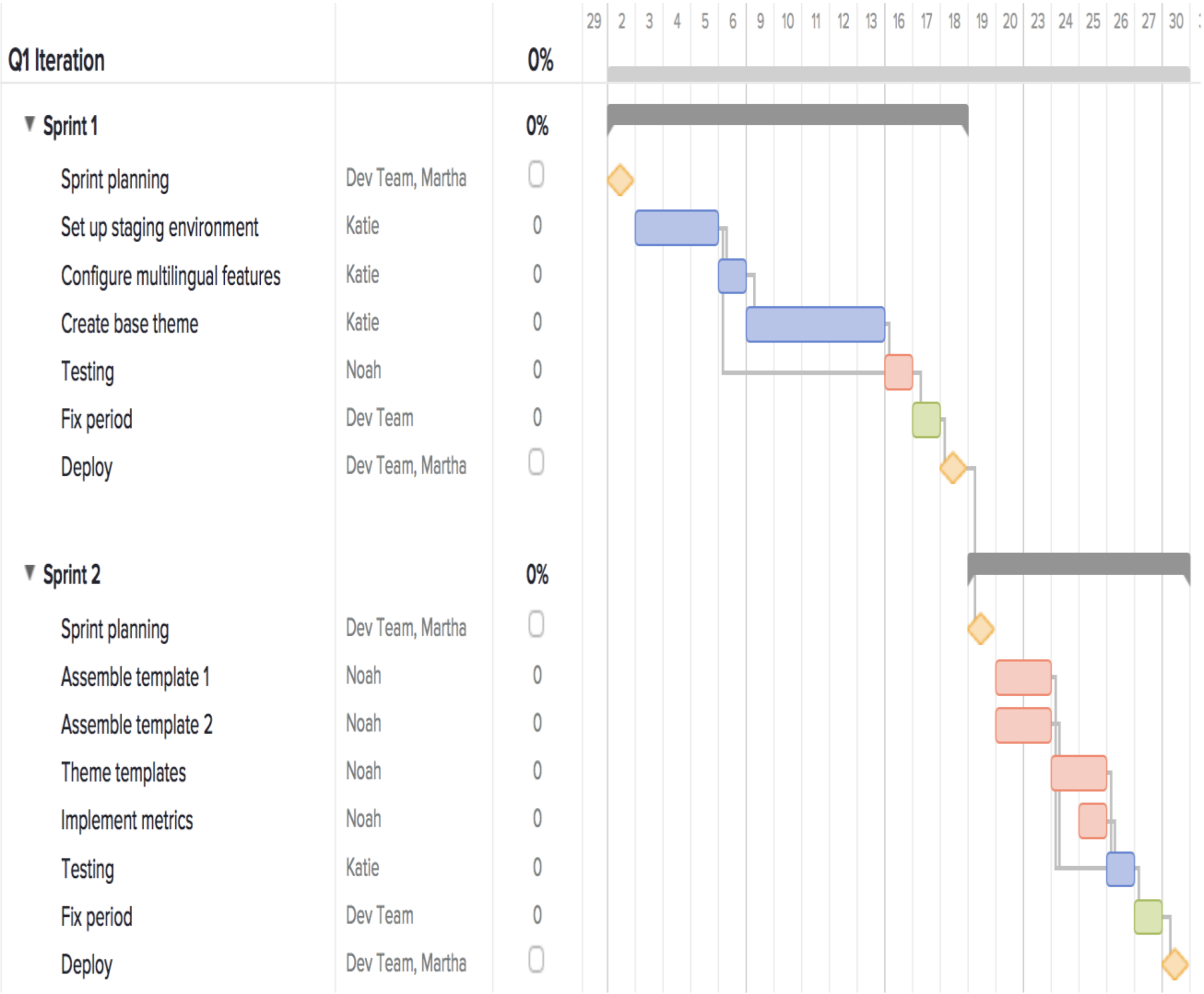
A **Gantt chart** is a type of bar **chart** that illustrates a series of tasks or activities on a **timeline**. Each task is a small step that must be finished as part of completing a larger objective. The tasks are arranged in a cascading order on the **timeline**, based on their start date.

Elements of a gantt chart

Gantt charts may seem complicated at first, but we can start by breaking them down into 4 sections.

- **Group and task names** A project is made up of several tasks, and related tasks can be organized into groups.
- **Group and task bars** group and task bars that correspond to the group and task names. Each bar represents when the task will start and end.
- **Milestones** A milestone is an important goal, event, or deliverable in your project, such as a kickoff meeting or major deadline. Using milestones in your project plan can help you monitor progress and identify potential delays. Milestones are signified by a gold diamond on the Gantt chart.
- **Dependencies** A dependency links tasks together to ensure work gets done in the right order. a dependency shows up as a light gray line connecting tasks on your gantt chart.

VII Diagrams / Experimental set-up /Work Situation



Q1 Iteration		0%
▼ Sprint 1		0%
Sprint planning	Dev Team, Martha	□
Set up staging environment	Katie	0
Configure multilingual features	Katie	0
Create base theme	Katie	0
Testing	Noah	0
Fix period	Dev Team	0
Deploy	Dev Team, Martha	□
▼ Sprint 2		0%
Sprint planning	Dev Team, Martha	□
Assemble template 1	Noah	0
Assemble template 2	Noah	0
Theme templates	Noah	0
Implement metrics	Noah	0
Testing	Katie	0
Fix period	Dev Team	0
Deploy	Dev Team, Martha	□

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System	Any desktop with basic configuration	One computer system for each student	

2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

How to Create a Gantt Chart

1.Define the project settings, such as its start date, end date and scheduling mode. The most common scheduling mode is forwards from the project start date. In this mode the default is for tasks to start as soon as possible, which means that the whole project finishes at the earliest possible date.

2.Define the project calendar. This sets the number of working days in the week, the number of working hours in the day, and so on.

3.Enter or edit task names and durations.

4.Set up a global resources list and assign resources to tasks. Although you can often define the resources as you need them, it is usually quicker to start by setting up a global resources list from which you can then select resources to assign to the various project tasks.

5.Create links to specify the dependencies between the project tasks.

6.Set constraints on the tasks as necessary.

7.Make final adjustments to the project plan.

8.Once the project has actually started, inspect it at regular intervals to detect potential problems or scheduling conflicts and make any corrections required.

X Precaution

Handle computer system with care.

XI Resources Used

Sr. No.	Name of Resource	Specification

XII Conclusion

XIII Practical Related Questions

1.What is Gantt chart?

2.Is a Gantt chart a timeline?

3.What are the Advantages of Gantt Charts?

XIV Exercise

(Space for Answer)

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

……

……

……

……

……

……

……

……

……

……

……

……

……

……

……..……..………………………………………………………………………………………………

…………………………

……

……

……

……

………

XV References/ Suggestions for further Reading

https://www.tutorialspoint.com/management_concepts/gantt_chart_tool.htm

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	Product Related (17.5)	Total(25)	

PRATICAL -15

Prepare SQA plan that facilitates various attributes of quality of process.

I. Practical Significance

Software quality assurance (SQA) is a process that ensures that developed **software** meets and complies with **defined** or standardized **quality** specifications. SQA is an ongoing process within the **software** development life cycle (SDLC) that routinely checks the developed **software** to ensure it meets desired **quality** measures.

Software Quality Assurance Plan. The purpose of this **Software Quality Assurance Plan** (SQAP) is to define the techniques, procedures, and methodologies that will be used at the Center for Space Research (CSR) to assure timely delivery of the **software** that meets specified requirements within project resources. This practical is useful for preparing SQA plan facilitating various attributes of process.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

Apply basic and fundamental knowledge of project management and quality assurance in software development.

Relevant Course Outcomes

Apply project management and quality assurance in software development.

Practical Outcomes (POs)

Apply project management and quality assurance in software development.

IV. Relevant Affective domain related Outcomes

1. Follow safety practices
2. Practice good Housekeeping

3. Demonstrate working as a leader/team member
4. Follow ethical practices

V. Minimum Theoretical Background

SQA auditor is the person who reviews and checks the project management activities are executed to the highest possible **standard**. Only through the result of this review, the Management Board can evaluate the quality of your project handling.

How to implement the quality assurance?

1) Develop SQA Plan

2) Define the standards/methodology

3) Review the process

Software **Quality Attributes** are: Correctness, Reliability, Adequacy, Learnability,Robustness, Maintainability, Readability, Extensibility, Testability, Efficiency,Portability. - Independence of the actual application of the software system.

VII Diagrams / Experimental set-up /Work Situation

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks

1	Computer System	Any desktop with basic configuration	One computer system for each student	
2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

Software Quality Assurance (SQA) is an umbrella activity that is applied through the software process. A SQA plan provides a road map for establishing software quality assurance. It is developed by the SQA group and the project team, the plan serves as a template for SQA activities that are instituted for each software project.

Following are the steps to develop SQA

1] Initiate SQA process

2] Planning

3] Tracking

4] Training

5] Review and Inspection

6] Issue Resolution

7] Testing

8] Audit

9] Improvement process

X Precaution

Handle computer system with care

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1. **Explain various attributes of quality of process.**
2. **How software quality is assured?**
3. **What is software quality plan?**
4. **Explain different steps to develop SQA plan.**

XIV Exercise

(Space for Answer)

……

……

……

……

……

……

……

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

XV References/ Suggestions for further Reading

https://www.energy.gov/sites/prod/files/cioprod/documents/csr_sqa_plan.pdf

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	Product Related (17.5)	Total(25)	

PRATICAL -16

Prepare SQA plan that facilitates various attributes of quality of product.

I. Practical Significance

Software quality assurance (SQA) is a process that ensures that developed **software** meets and complies with **defined** or standardized **quality** specifications. SQA is an ongoing process within the **software** development life cycle (SDLC) that routinely checks the developed **software** to ensure it meets desired **quality** measures.

Software Quality Assurance Plan. The purpose of this **Software Quality Assurance Plan** (SQAP) is to define the techniques, procedures, and methodologies that will be used at the Center for Space Research (CSR) to assure timely delivery of the **software** that meets specified requirements within project resources. This practical is useful for preparing SQA plan facilitating various attributes of quality of product.

II. Relevant Programs Outcomes (POs)

Po1 - Basic Knowledge
Po2 – Discipline Knowledge
Po3 – Life Long Knowledge
Po4 – Engineering tool.
Po5 – Communication

III. Competency and Practical skills

Apply basic and fundamental knowledge of project management and quality assurance in software development.

Relevant Course Outcomes

Apply basic and fundamental knowledge of project management and quality assurance in software development.

IV. Practical Outcomes (POs)

Apply basic and fundamental knowledge of project management and quality assurance in software development.

V. Relevant Affective domain related Outcomes

1.Follow safety practices

2.Practice good Housekeeping

3.Demonstrate working as a leader/team member

4.Follow ethical practices

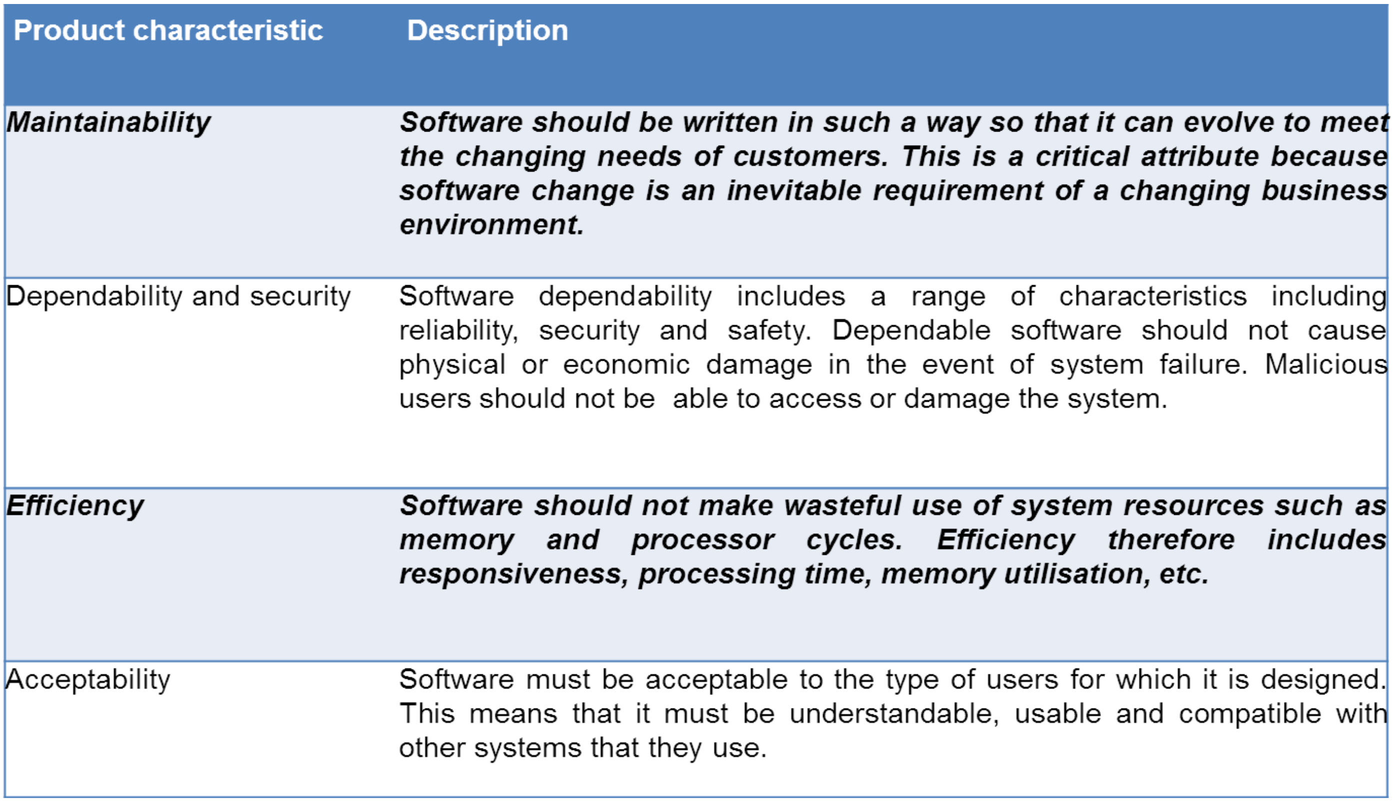
VI. Minimum Theoretical Background

- **Measure**
  - provides a quantitative indication of the size of some product or process attribute, E.g., Number of errors
- **Metric**
  - is a quantitative measure of the degree to which a system, component, or process possesses a given attribute *(IEEE Software Engineering Standards 1993) : Software Quality* - E.g., Number of errors found per person hours expended

Product metrics are measures of the software product at any stage of its development, from requirements to installed system. Product metrics may measure:

1. How easy is the software to use
2. How easy is the user to maintain
3. The quality of software documentation

VII Diagrams / Experimental set-up /Work Situation



Product characteristic	Description
Maintainability	***Software should be written in such a way so that it can evolve to meet the changing needs of customers. This is a critical attribute because software change is an inevitable requirement of a changing business environment.***
Dependability and security	Software dependability includes a range of characteristics including reliability, security and safety. Dependable software should not cause physical or economic damage in the event of system failure. Malicious users should not be able to access or damage the system.
Efficiency	***Software should not make wasteful use of system resources such as memory and processor cycles. Efficiency therefore includes responsiveness, processing time, memory utilisation, etc.***
Acceptability	Software must be acceptable to the type of users for which it is designed. This means that it must be understandable, usable and compatible with other systems that they use.

VIII Resources Required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System	Any desktop with basic configuration	One computer system for each student	
2	Operating System	Windows 7/Windows 8/ Windows 10/LINUX or any other		
3	Software Tool	Any UML tool		

IX Procedure

X Precaution

Handle computer system with care

XI Resources Used

Sr. No.	**Name of Resource**	**Specification**

XII Conclusion

XIII Practical Related Questions

1.What are the activities of quality assurance?

2.What is difference between process and product?

XIV Exercise

(Space for Answer)

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……………………………………………………………………………………………

……..…..………………………………………………………………………………

………………………

XV References/ Suggestions for further Reading

https://www.energy.gov/sites/prod/files/cioprod/documents/csr_sqa_plan.pdf

Marks Obtained			Dated Signature of Teacher
Process Related(7.5)	Product Related (17.5)	Total(25)	