GUJARAT TECHNOLOGICAL UNIVERSITY

Bachelor of Engineering
Subject Code: 3150711
SOFTWARE ENGINEERING
Semester V

Type of course: NA

Prerequisite: Object Oriented Programming fundamental, UML

Rationale:
- To study Software Development Life Cycle, Development models and Agile Software development.
- To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
- To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
- To learn the process of improving the quality of software work products.
- To gain the techniques and skills on how to use modern software testing tools to support software testing projects.
- To expose Software Process Improvement and Reengineering

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks				Total Marks
L	T	P	C	Theory Marks		Practical Marks		
				ESE (E)	PA (M)	ESE (V)	PA (I)	
3	0	2	4	70	30	30	20	150

Content:

Sr. No.	Content	Total Hrs	% Weightage
1	**Introduction to Software and Software Engineering** The Evolving Role of Software, Software: A Crisis on the Horizon and Software Myths, Software Engineering: A Layered Technology, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Process Models, Agile Process Model, Component-Based Development, Process, Product and Process.	05	14%
2.	**Agile Development** Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools.	03	08%

w.e.f. AY 2018-19

3	**Managing Software Project** Software Metrics (Process, Product and Project Metrics), Software Project Estimations, Software Project Planning (MS Project Tool), Project Scheduling & Tracking, Risk Analysis &Management (Risk Identification, Risk Projection, Risk Refinement , Risk Mitigation).	04	10%
4	**Requirement Analysis and Specification** Understanding the Requirement, Requirement Modeling, Requirement Specification (SRS), Requirement Analysis and Requirement Elicitation, Requirement Engineering.	03	8%
5	**Software Design** Design Concepts and Design Principal, Architectural Design, Component Level Design (Function Oriented Design, Object Oriented Design), User Interface Design, Web Application Design.	04	10%
6.	**Software Coding & Testing** Coding Standard and coding Guidelines, Code Review, Software Documentation, Testing Strategies, Testing Techniques and Test Case, Test Suites Design, Testing Conventional Applications, Testing Object Oriented Applications, Testing Web and Mobile Applications, Testing Tools (Win runner, Load runner).	05	15%
7	**Quality Assurance and Management** Quality Concepts and Software Quality Assurance, Software Reviews (Formal Technical Reviews), Software Reliability, The Quality Standards: ISO 9000, CMM, Six Sigma for SE, SQA Plan.	03	10%
8	**Software Maintenance and Configuration Management** Types of Software Maintenance, Re-Engineering, Reverse Engineering, Forward Engineering, The SCM Process, Identification of Objects in the Software Configuration, Version Control and Change Control	03	10%
9.	**DevOps:** Overview, Problem Case Definition, Benefits of Fixing Application Development Challenges, DevOps Adoption Approach through Assessment, Solution Dimensions, What is DevOps?, DevOps Importance and Benefits, DevOps Principles and Practices, 7 C's of DevOps Lifecycle for Business Agility, DevOps and Continuous Testing, How to Choose Right DevOps Tools, Challenges with DevOps Implementation, Must Do Things for DevOps, Mapping My App to DevOps -	04	10%

	Assessment, Definition, Implementation, Measure and Feedback		
10	**Advanced Topics in Software Engineering** Component-Based Software Engineering, Client/Server Software Engineering, Web Engineering, Reengineering, Computer-Aided Software Engineering, Software Process Improvement, Emerging Trends in software Engineering.	02	5%

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	U Level	A Level	N Level	E Level	C Level
15	**25**	**10**	**10**	**5**	**5**

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Reference Books:

1. Roger S.Pressman, Software Engineering- A practitioner's Approach, McGraw-Hill International Editions
2. Ian Sommerville, Software engineering, Pearson education Asia
3. Pankaj Jalote, Software Engineering – A Precise Approach Wiley
4. Behhforoz & Frederick Hudson, Software Engineering Fundamentals, OXFORD
5. Rajib Mall, Fundamentals of software Engineering, Prentice Hall of India.
6. Deepak Gaikwad, Viral Thakkar, DevOps Tools from Practitioner's ViewPoint, Wiley
7. Merlin Dorfman (Editor), Richard H. Thayer (Editor) ,Software Engineering
8. Robert C. "Uncle Bob" Martin , Clean Architecture: A Craftsman's Guide to Software Structure and Design

Course Outcome:

After learning the course the students should be able to:

1. Prepare SRS (Software Requirement Specification) document and SPMP (Software Project Management Plan) document.
2. Apply the concept of Functional Oriented and Object Oriented Approach for Software Design.
3. Recognize how to ensure the quality of software product, different quality standards and software review techniques.
4. Apply various testing techniques and test plan in.
5. Able to understand modern Agile Development

List of Experiments:

(Pl. Note: List of Experiments and Tutorials should be as per theory covered in the class, below mentioned practical are just for the reference purpose)

Tutorial-1

Study the complete Software Development Life Cycle (SDLC) and analyze various activities conducted as a part of various phases. For each SDLC phase, **identify** the objectives and **summaries** outcomes.

Tutorial-2

Consider any project to be developed in any technology as a Software Architect or Project Manager. **Construct** Software Requirement Specification (SRS) document for the project.

Tutorial-3

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:

- 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)

(All other cost drivers have nominal rating).

Now answer the following:

- **Solve** the problem by **Applying** Basic and intermediate COCOMO

  o Find Project Type?
  o Find Project Size?
  o Find Initial Effort Estimation?
  o Find Adjusted Effort Estimation?
  o Find schedule?
  o 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?

Tutorial-4:

Function Point: http://conferences.embarcadero.com/article/32094#Bonus .

Analyze the case study and **identify** the error and **solve** it. At the end, need to **assess** calculation part of effort using FP oriented estimation model.

Tutorial-5

Consider the following Java code segment:

```java
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;
}
```

1. Guarantees that all independent execution path is exercised at least once;
2. Guarantees that both the true and false side of all logical decisions are exercised;
3. Executes the loop at the boundary values and within the boundaries.

Sketch out Design control flow diagram and **Apply** Cyclomatic complexity for given Code. **Identify** numbers of Independence path require for testing.

Tutorial 6:-

Subject Project: For below mentioned Systems and other systems assign a mini-project two a group of students to prepare Software documents mentioned as A to E

1. Library Information System
2. Villager Telephone System
3. Waste Management Inspection Tracking System (WMITS)
4. Flight Control System

5. Ambulance Dispatching System

A. Development of Software Requirements Specification (SRS)
B. Function oriented design using SA/SD
C. Object-oriented design using UML
D. Test case design
E. Implementation using Java and testing

Tutorial 7:

Study of any any two Open source tools in DevOps for Infrastructure Automation, Configuration Management ,Deployment Automation, Performance Management, Log Management. Monitoring. (Behat , Watir, Chef, Supergiant, SaltStack, Docker, Hudson etc)

Design based Problems (DP)/Open Ended Problem:

- Assume that you are Software Architect or Project Manager in organization. You have been assigned the task of constructing a website for a specific company with your team. Design and priorities the test cases using test case templates for this project.

- For Natural Language Processing (NLP) applications, estimate project failure rate.

- Design and develop an open source method of detecting the DIFFERENCESS between two Engineering designs for the same problem.

List of Open Source Software/learning website:

- www.en.wikipedia.org/wiki/Software_engineering
- www.win.tue.nl
- www.rspa.com/spi
- www.onesmartclick.com/engsineering/software-engineering.html
- www.sei.cmu.edus
- https://www.edx.org/school/uc-berkeleyx
- https://devops.com/most-popular-open-source-devops-tools/
- https://www.guru99.com/devops-tutorial.html

Various Web Based SE Tools

- Software:-Rational Rose, Microsoft Visio, Enterprise resource planning
- Project Management Tools
- SCM Tools
- SQA Tools
- Analysis and Design Tools
- User Interface Development Tools
- Object-Oriented Software Engineering Tools
- Testing Tools