

## GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)

### Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)

Semester - III

#### Course Title: Fundamentals of Software Development

(Course Code: 4331604)

Diploma programme in which this course is offered	Semester in which offered
Information Technology	Third

#### 1. RATIONALE

One of the important product in the field of Information Technology is a Software. The software has changed every aspect of our life and made available everything of our finger tips. It has revolutionize every area of human life like education, health, defense and security, finance and business, travel, social life, politics, entertainment and so on. Thus, a software increases the responsibilities for the developer. The software development process in the industry is also changing rapidly. This course guides the students to analyze, design, implement and test the software product with proper documentation during the courses in higher semesters of diploma programme.

#### 2. COMPETENCY

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

- **Analyze problems and prepare the software solutionwith documentation.**

#### 3. COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with this competency are to be developed in the student to display the following COs:

At the end of the course, the student will develop adequate skills ofdocumentation and will understand the application of concepts in software development.

- a) Explain software development activities.
- b) Select appropriate software process model for software project development.
- c) Prepare software requirement specification(SRS) document for a software project.
- d) Organize software project development schedule.
- e) Prepare a design of the software with user interface.
- f) Apply testing on software product with proper test cases.

#### 4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T/2+P/2)	Examination Scheme				Total Marks
				Theory Marks		Practical Marks		
L	T	P	C	CA	ESE	CA	ESE	
3	-	2	4	30*	70	25	25	150

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

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

## 5. SUGGESTED PRACTICAL EXERCISES

The following practical outcomes (PrOs) are the sub-components of the COs. These PrOs need to be attained to achieve the COs.

To perform all the practicals more effectively, the faculty members are advised to prepare the group of 3-4 students and assign one software project with 3 to 4 modules/major functionalities. Based on that project, the group will perform all the practicals. The example of software projects for practicals: student information system, attendance management, result management, online exam, inventory control, payroll system, sales management, library management, leave management(employee), medical store-shop-any store management, hotel management, patient management, Medical Laboratory Management etc. The project can be any known online or offline information management system.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
1	Select a software project and identify the process model with proper justification.	II	02
2	Collect the functional requirements for the project. (Questionnaires/ stakeholders' interview questions)	III	04
3	Analyze functional and non-functional requirement and prepare SRS document for the project.	III	04
4	Prepare GANTT chart for selected system.	IV	02
5	Prepare PERT chart for selected system.	IV	02
6	Design DFD(context, level-1/2) for function oriented design.	V	02
7	Design data dictionary of the selected system.	V	02
8	Prepare User-case diagrams of the selected system for object oriented design.	V	02
9	Prepare Activity diagrams of the selected system for object oriented design.	V	02
10	Design appropriate User Interface based on the type of project.	V	02
11	Prepare the test cases to test the functionalities of the modules.	VI	04
	<b>Total</b>		<b>28</b>

### Note

i. More **Practical Exercises** can be designed and offered by the respective course teacher to develop the industry relevant skills/outcomes to match the COs. The above table is only a suggestive list.

ii. The following are some **sample** 'Process' and 'Product' related skills (more may be added/deleted depending on the course) that occur in the above listed **Practical Exercises** of this course required which are embedded in the COs and ultimately the competency.

S. No.	Sample Performance Indicators for the PrOs	Weightage in %
1	Identify suitable approach to develop software.	10
2	Prepare proper SRS document.	25
3	Design DFD and use-cases for software product.	25
4	Design user interface depends on the software type.	20
5	Prepare the test cases for the modules.	20
<b>Total</b>		<b>100</b>

## 6. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

These major equipment with broad specifications for the PrOs is a guide to procure them by the administrators to usher-in uniformity of practicals in all institutions across the state.

S. No.	Equipment Name with Broad Specifications	PrO. No.
1	Online or offline tool for documentation.	All
2	Online or offline tools to draw the various diagrams. For example, Microsoft Visio, StarUMLForeUI, Smartdraw	4,5,6,8,9,10

## 7. AFFECTIVE DOMAIN OUTCOMES

The following **sample** Affective Domain Outcomes (ADOs) are embedded in many of the above mentioned COs and PrOs. More could be added to fulfil the development of this competency.

- a) Work as a leader/a team member.
- b) Follow ethical practices.

The ADOs are best developed through the laboratory/field based exercises. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- i. 'Valuing Level' in 1<sup>st</sup> year
- ii. 'Organization Level' in 2<sup>nd</sup> year.
- iii. 'Characterization Level' in 3<sup>rd</sup> year.

## 8. UNDERPINNING THEORY

Only the major Underpinning Theory is formulated as higher level UOs of *Revised Bloom's taxonomy* in order development of the COs and competency is not missed out by the students and teachers. If required, more such higher level UOs could be included by the course teacher to focus on attainment of COs and competency.

Unit	Unit Outcomes (UOs) (4 to 6 UOs at Application and above level)	Topics and Sub-topics
<b>Unit – I About Software Development</b>	1.1 Software and Software Development	1.1.1 Software – definition, characteristics 1.1.2 Characteristics of Web-based application 1.1.3 Software engineering – A layered technology 1.1.4 Software Myths 1.1.5 Software Process framework and umbrella activities
<b>Unit – II Software Life cycle models</b>	2.1 Select Software process model for project development  2.2 Agile Development	2.1.1 Waterfall model 2.1.2 Incremental Model 2.1.3 Prototyping Model 2.1.4 Spiral Model 2.1.5 Rapid Application Development(RAD)  2.2.1 Agile Process & Principles 2.2.2 Comparison of Agile development with traditional models 2.2.3 Extreme Programming (XP) Model 2.2.4 Scrum Model
<b>Unit– III Software Requirement Analysis</b>	3.1 Identify software requirements  3.2 Prepare Software Requirement Specifications (SRS)	3.1.1 Requirement gathering –collect requirements from stakeholders 3.1.2 Analyze the requirements  3.2.1 Software Requirement Specifications – Importance of SRS – Users of SRS – Characteristics of good & bad SRS 3.2.2 Types of Requirements in SRS – Functional Requirements – Non-functional Requirements
<b>Unit– IV Software Project Management</b>	4.1 Prepare schedule for software development activities	4.1.1 Responsibility of software project Manager – Job responsibility – Necessary skill to manage software projects 4.1.2 Scheduling – Work breakdown structure – Activity network and critical path Method – PERT chart

		<ul style="list-style-type: none"> <li>– Gantt chart</li> <li>– Project Monitoring and control</li> </ul> <p>4.1.3 Risk Management</p> <ul style="list-style-type: none"> <li>– Risk Identification: Project, Technical, Business</li> <li>– Risk Assessment</li> <li>– Risk Mitigation</li> </ul>
<b>Unit– V Software Design</b>	<p>5.1 Prepare software design</p> <p>5.2 Design proper User Interface for the software</p>	<p>5.1.1 Software design process</p> <ul style="list-style-type: none"> <li>– Design activities</li> <li>– Design methodologies</li> </ul> <p>5.1.2 Cohesion and Coupling</p> <p>5.1.3 Approach of software design</p> <ul style="list-style-type: none"> <li>– Function oriented design</li> <li>– Data Flow Diagram with levels, its shortcomings, Data dictionary</li> <li>– Object oriented design</li> <li>– Prepare use-case and activity diagrams of the system</li> </ul> <p>5.2.1 User Interface design</p> <ul style="list-style-type: none"> <li>– Characteristics of good UI</li> <li>– Types of UI: command-based, menu-based, direct manipulation</li> </ul>
<b>Unit– VI Software Coding and Testing</b>	<p>6.1 Follow coding standards</p> <p>6.2 Test the software with proper test cases</p>	<p>6.1.1 Coding standards and guidelines</p> <p>6.1.2 Code review</p> <ul style="list-style-type: none"> <li>– Code Walkthrough</li> <li>– Code Inspection</li> </ul> <p>6.2.1 Testing</p> <ul style="list-style-type: none"> <li>– Test cases and test suit</li> <li>– Verification, Validation</li> <li>– Unit testing</li> <li>– Black-box testing</li> <li>– White-box testing</li> </ul>

**Note:** The UOs need to be formulated at the 'Application Level' and above of Revised Bloom's Taxonomy' to accelerate the attainment of the COs and the competency.

### 9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	About Software Development	02	02	02	00	04
II	Software Life cycle models	10	04	08	04	16
III	Software Requirement Analysis	06	02	04	04	10
IV	Software Project Management	08	04	04	06	14
V	Software Design	10	02	06	08	16
VI	Software Coding and Testing	06	02	04	04	10
<b>Total</b>		<b>42</b>	<b>16</b>	<b>28</b>	<b>26</b>	<b>70</b>

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

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

#### 10. SUGGESTED STUDENT ACTIVITIES

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

- a) Give a seminar on any relevant topics.
- b) Suggest a small existing software/website to students and discuss on betterment of its user-interface design.
- c) Visit a software company and observe their documents and discuss their practices adopted for software development.

#### 11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

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

- a) Massive open online courses (**MOOCs**) may be used to teach various topics/subtopics.
- b) Guide student(s) in undertaking micro-projects.
- c) '**L**' in **section No. 4** means different types of teaching methods that are to be employed by teachers to develop the outcomes.
- d) About **20% of the topics/sub-topics** which are relatively simpler or descriptive in nature is to be given to the students for **self-learning**, but to be assessed using different assessment methods.
- e) With respect to **section No.11**, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.

#### 12. SUGGESTED MICRO-PROJECTS

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

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

student ought to submit a micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects is given here. This has to match the competency and the COs. Similar micro-projects could be added by the concerned course teacher:

- 1: Study the SRS document of project from any software company and enlist functional as well as non-functional requirements.
- 2: Prepare class and other UML diagrams of any known system.
- 3: Prepare the user interface using online tools of the known system.
- 4: Search any automated test case generator and prepare test suits for the system.

### 13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	Software Engineering: A Practitioner's Approach	Roger S. Pressman	Tata McGraw Hill, New Delhi 978-9353165710
2	Fundamentals of Software Engineering	Rajib Mall	PHI Learning Private Limited, New Delhi 978-9388028028
3	Software Engineering	Ian Sommerville	Pearson Education, India 978-9332582699
4	Object - Oriented Modeling and Design With UML, 2e	Michael Blaha, James Rumbaugh	Pearson Education, India 978-8131711064

### 14. SOFTWARE/LEARNING WEBSITES

- a) <https://nptel.ac.in/courses/106101061/>
- b) <https://www.mhhe.com/engcs/compsci/pressman/student/olc/cases.mhtml>
- c) <http://www.foreui.com/> (for User interface)
- d) <https://www.smartdraw.com/> (for all types of diagrams)
- e) <https://online.visual-paradigm.com/diagrams/features/uml-tool/>

### 15. PO-COMPETENCY-CO MAPPING

Semester III	Fundamentals of Software Development (Course Code: 4331604)						
	POs and PSOs						
Competency & Course Outcomes	PO 1 Basic & Discipline specific knowledge	PO 2 Problem Analysis	PO 3 Design/ development of solutions	PO 4 Engineering Tools, Experimentation & Testing	PO 5 Engineering practices for society, sustainability & environment	PO 6 Project Management	PO 7 Life-long learning
<b>Competency</b>	<i>Analyze problems and prepare the software solution with documentation.</i>						
<b>Course Outcomes</b>							
CO a) Explain software development activities.	2	-	3	-	-	2	2
CO b) Select appropriate software process model for software project development.	3	2	3	2	-	2	3
CO c) Prepare software	3	3	3	3	-	3	3

requirement specification (SRS) document for a software project.							
CO d) Prepare software project development schedule.	2	2	3	3	-	3	3
CO e) Prepare design for the software product including user interface.	2	3	3	3	2	3	3
CO f) Apply testing on software product with proper test cases.	2	2	3	3	-	3	3

Legend: '3' for high, '2' for medium, '1' for low or '-' for the relevant correlation of each competency, CO, with PO/ PSO

## 16. COURSE CURRICULUM DEVELOPMENT COMMITTEE

### GTU Resource Persons

S. No.	Name and Designation	Institute	Email
1	Mr. Chintan A Gajjar	Dr S & S S Ghandhy College of Engineering & Technology, Surat	<a href="mailto:cagajjar.it@gmail.com">cagajjar.it@gmail.com</a>
2	Mr. Dhaval R Gandhi	Dr S & S S Ghandhy College of Engineering & Technology, Surat	<a href="mailto:dhavalgandhi88@gmail.com">dhavalgandhi88@gmail.com</a>
3	Mr. Umang D Shukla	Dr S & S S Ghandhy College of Engineering & Technology, Surat	<a href="mailto:umang.shuklait@gmail.com">umang.shuklait@gmail.com</a>