

GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)**Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)**
Semester-III**Course Title: Summer Internship-I**
(Course Code: 4330001)

Diploma programme in which this course is offered	Semester in which offered
All Branches of Diploma Engineering(Except Automobile, Bio Medical, ICT, Power)	Third

1. RATIONALE

Idea of Embedded Internships- AICTE has made 7-10 weeks summer internships mandatory in the new curriculum which will equip the students with practical understanding and training about industry practices in a suitable industry or organization. To make education holistic, sports, physical activities, values and ethics have been embedded in the curriculum.

We must agree that all Branches of Diploma Engineering are changing rapidly. New technologies are adding fast which effects can be seen in our society. Summer internship is a good option by which students to get flavor of such emerging technology and familiar with industry environment to identify scope and focus of their career development opportunities. Main objective of summer internship is hand-on practice to expose students for thinking about professional career by observing, understanding working mechanism of ongoing work of industry and to obtain various types of skills throughout internship program.

This two week mandatory internship is to equip the students with practical knowledge and provide them exposure to real time industrial environments. Further, in these internships, the option is provided to do internship in Government Agencies/ skill centers/ social sector/ Govt. initiated social schemes/ NGOs etc. The duration of internship will be two weeks. It will be after completion of 2nd Semester and before the commencement of Semester 3rd. Any options from following can be chosen by the students:

- Offline internship in industry** - Student is supposed to produce joining letter for starting and relieving letter once the internship is over in case of Offline internship in any industry.
- Online internships** – Student can select from any of approved /supported / recommended by the All India Council of Technical education for Internship (like Internshala/ NEAT/ Gujarat Knowledge Society Initiative etc.) or Approved by the state government or University approved
- A Mini Project** - On some suitable topic related to respective branch. It can be small fabrication / experimental results/ simulations/ Application development / Design and / or Analysis of System(s) etc. depending on the branch of the student. Preferably a single student should carry out a mini-project.

2. COMPETENCY

The purpose of this course is to help the student to attain flavor of the following industry identified competency through summer internship experiences:

- **Develop multiple types of skills such as planning, communication, collaboration, decision making / Problem solving and management skills along with selected technical knowledge.**

3. COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with the identified competency are to be developed in the student for the achievement of the following COs:

- Learn and adopt the engineer's role and responsibilities with ethics.
- Get exposure to the industrial environment for professional activities.
- Get possible opportunities to learn, understand and sharpen the technical skills required for technical advancement.
- Develop managerial skills required for professional career.
- Attain skill for writing technical report and prepare poster for presentation.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P/2)	Examination Scheme				Total Marks
L	T	P		Theory Marks		Practical Marks		
			C	CA	ESE	CA	ESE	
0	0	0	1	0	0	25	25	50

- Offline internship in industry:** CA will be carried out based on submitted progress card by Industry resource person and ESE / Assessment will be carried out by institute resources person.
- Online internships:** CA will be carried out based on submitted certificate and ESE/ Assessment will be carried out by institute resources person.
- A Mini Project:** CA will be carried out based on project work by institute resources person.

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

List of Documents to be prepared for Submission:

- Detail report duly signed and approved by the internal/external mentor
- Presentation softcopy approved by the internal/external mentor
- Poster of summer internship activities approved by the internal/external mentor.

Sample forms for Registration and Evaluation of Summer Internship-I –SI-I are given below:

- Both forms are mandatory to be filled at the commencement and completion of SI respectively.
- It is mandatory to file and map SI-I Registration and Evaluation with respective forms of SI-II (Later in Semester 5) so that students get enough exposure of industry / technology. (Mapping doesn't mean same industry/ company/ project-it can be independent/ different also.)
- Mapping will be done to ease CA and ESE Evaluations.
- A Seminar / Webinar can be arranged so that students coming from different industry / institute / project background can share experiences and learnings to their peers / all students of the same department.
- Attached formats for Registration, Completion and Evaluation are suggestive. But, adhering to these formats is anticipated.

Summer Internship-I Registration Form

Note: Students needs to submit this registration form after finalizing mode of internship.

Student Details											
Enrollment Number											
Student Name											
Student Details	Mobile Number :										
	Email Address:										
Branch											
Code of the Institute	Name of the Institute										
Mentor Details (Institute)	Name:										
	Designation:										
	Mobile No:										
	Email Address:										
Industry Details	Name:										
	Address:										
	Email:										
	Phone:										
	Website:										
Mentor Details (Industry)	Name:										
	Designation:										
	Mobile No:										
	Email Address										
Mode of Internship Carried Out	Online / Offline/ Mini Project										
Title of the Project/ Internship carried out											
Nature of Work Carried Out	Web Design / Application development (Web / Mobile), Experimental results/ simulations/ Analysis of System(s) etc...										
	Other please Specify_____										

Student Signature

Faculty Signature

Summer Internship-I -Suggested Letter for Completion

[Company or Institute letter head]

No:

Date

TO WHOM SO EVER IT MAY CONCERN

This is to certify that, Mr. /Mrs. _____

Enrollment No. _____ Student of _____

Has successfully completed a two week Internship in the field of _____

From the date: _____ to date: _____.

[90% Attendance is mandatory for completion of Internship]

During the period of his/her summer internship program with us, He / She were exposed to following different processes and were found sincere and hardworking.

1. _____
2. _____
3. _____
4. _____

Mentor Signature

Head of Department

Stamp

Stamp

Summer Internship-I -Evaluation Rubrics for Institute Evaluation Rubrics (Institute)

Enrollment No: _____

Branch: _____

Name of the Students: _____

Date of Evaluation: _____

Internal Evaluation – 25 Marks PA(I) (To be carried out by the mentor in consultation with Industry) Minimum Passing Marks: 13					
Parameter	Excellent	Good	Average	Not up the level of Satisfaction	Obtained Marks
Mark range	4-5	3-4	2-3	Below 2	
Knowledge acquisition in specific domain. 5 marks					
Skill and attitude attainment in specific domain. 5 marks					
Feedback and suggestions given are incorporated? 5 marks					
Quality of the prepared report and poster. 5 marks					
Quality of the presentation. 5 marks					
Total Marks Obtained Out of 25 PA(I)					

Signature: _____

Institute Resource Examiner Name: _____

Suggested Evaluation Rubrics for Industry Evaluation Rubrics (Industry)

Enrollment No: _____

Branch: _____

Name of the Students: _____

Date of Evaluation: _____

External Evaluation – 25 Marks ESE(V) (To be carried out by the Industry Supervisor) Minimum Passing Marks: 13					
Parameter	Excellent	Good	Average	Not up the level of Satisfaction	Obtained Marks
Mark range	4-5	3-4	2-3	Below 2	
Student regularity during the Internship period and proactiveness/responsiveness towards the given tasks (5 Marks)					
Work Plan, Execution and quality of work in forms of Outcome achieved (5 Marks)					
Engineering Tools and Techniques (5 Marks)					
Quality of poster design and presentation (5 Marks)					
Quality of the report and Skill (5 Marks)					
Total Marks Obtained Out of 25 ESE(V)					

Signature: _____

Industry resource/ Examiner Name: _____

Common Note:

- 1) For Summer Internship / Projects / Seminar etc. Evaluation is based on work done, quality of report, performance in viva-voce, presentation etc. The internal / external assessment is based on the student's performance in viva-voce /work record respectively.
- 2) In case Industry Supervisor is not available / Institute Mentor/ Faculty can fill up both.

5. AFFECTIVE DOMAIN OUTCOMES

The following affective Domain Outcomes (ADOs) are embedded in many of the above mentioned COs. More could be added to fulfill the development of this course competency.

- a) Work as a leader/a team member as role of Engineer.
- b) Practice environmentally friendly methods and processes.
Follow safety precautions and ethical practices.

6. SUGGESTED STUDENT ACTIVITIES

Following are the suggested student-related curricular, **co-curricular** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should perform following activities and prepare reports and give presentation in front of students and faculty members. They should also collect/record physical evidences for their (student's) portfolio which may be useful for their placement interviews:

- a) Perform various tasks given by industry resources person during offline internship.
- b) Perform various tasks given during online internship.
- c) Perform various task required to complete mini project work under guidance of faculty member.
- d) Summer Internship program Interns are required to give a presentation before review committee consisting of a group of academic staff members.
- e) The review committee gives feedback and suggests possible improvements in the work.
- f) At the end of the program all the Summer Internship program Interns make a poster presentation of the work carried out. The poster presentation is open to the public. It is also evaluated by faculty members.
- g) A completion certificate will be issued to all Summer Internship program Interns only after the completion of internship tenure.

7. SOFTWARE / LEARNING WEBSITES

An internship is a short term work program usually offered to students by companies and institutes who require staff for assistance at junior levels. Thus for the students undergoing internship a professional learning experience is provided to benefit them in their skills as well as career. It will brush existing skills and provide exposure to new skills. Generally it is provided at entry level in the industry.

Here is a suggestive list for reference only.

- <https://www.internshala.com>
- <https://swayam.gov.in>
- <https://nptel.ac.in/>
- <https://neat.aicte-india.org/>
- <https://www.edx.org/>
- <https://www.coursera.org/>
- <https://www.udemy.com/>
- <https://www.linkedin.com>
- <https://www.stumags.com>
- <https://www.letsintern.com>
- <https://www.internship.com>
- <https://www.glassdoor.com>

8. PO-COMPETENCY-CO MAPPING

Semester III	Summer Internship (Course Code:4330001)						
	POs						
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
Competency	Use principles of basic electronics to maintain various electronics circuits And equipment						
CO1) Learn and adopt the engineer's role and responsibilities with ethics.	2	1	1	1	1	1	1
CO2) Get exposure to the industrial environment for professional activities.	1	1	1	1	1	1	1
CO3) Get possible opportunities to learn understand and sharpen the technical skills required for technical advancement.	2	1	2	2	1	1	1
CO4) Develop managerial skills required for professional career.	1	1	2	1	1	1	1
CO5) Attain skill for writing technical report and prepare poster for presentation.	1	1	-	1	1	1	1

Legend: '3' for high, '2' for medium, '1' for low and '-' for no correlation of each CO with PO.

9. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Sr. No.	Name and Designation	Institute	Contact No.	Email
1	Jiger P. Acharya	GP, Ahmedabad	9429462026	jigeracharya@gmail.com
2	Alpeshkumar R. Thaker	GP, Ahmedabad	9879709675	alpeshrthaker@gmail.com
3	Umang D. Shah	GP, Ahmedabad	9427686364	umang.shah111gp@gmail.com

BoS Resource Persons

Sr. No.	Name and Designation	Institute	Contact No.	Email
1	Shri U. V. Buch- BoS Member and Subject in-charge (EC)	G P Ahmedabad	9825346992	uvbuch@gmail.com

GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)**Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)**

Semester-III

Course Title: Scripting Language - Python

(Course Code: 4330701)

Diploma programme in which this course is offered	Semester in which offered
Computer Engineering	Third

1. RATIONALE

Python is an open-source, high-level, general-purpose programming language used for software development. It is one of the most popular programming languages in the world today and known for its simplicity as well as rich library. It is widely used programming language in various domains, such as Automation, Server-side Web Development, Tools Development, Game Programming, Blockchain, Data Science, Artificial Intelligence, Machine Learning, Big Data etc. It's relatively easy to learn to use and incredibly versatile.

This course aims to teach the basics of Python programming. The course focuses on how to use the building blocks of Python programming to solve different problems. At the end of the course, students will be able to develop simple applications using Python programming. This knowledge will provide a solid foundation for exploring advanced applications of Python programming in the different domains mentioned above.

2. COMPETENCY

The course content should be taught and implemented with the aim of developing different types of skills so that students are able to acquire the following competencies:

- **Develop simple applications using scripting language Python.**

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 Course Outcomes (COs):

- a) Develop programs to solve the given simple computational problems.
- b) Apply control flow structures to solve the given problems.
- c) Implement data structures lists, tuples, sets and dictionaries to solve the given problems.
- d) Apply modular programming approach to solve given problems using user-defined functions.
- e) Perform string manipulation and file operations to solve a given problem.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T/2+P/2)	Examination Scheme				Total Marks
L	T	P		Theory Marks		Practical Marks		
			C	CA*	ESE	CA	ESE	
3	-	4	5	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 subcomponents of the COs. Some of the **PrOs** marked “*” are compulsory, as they are crucial for that particular CO at the ‘Precision Level’ of Dave’s Taxonomy related to ‘Psychomotor Domain’.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
1	Environment Setup Install and configure the Python environment. Run basic Python commands to verify the Python environment.	1	02
2	Input-Output Write a program to read your name, contact number, email, and birthdate and print those details on the screen.	1	02
3	Variables, operators, Expressions i. Write a program to convert temperature from Celsius to Fahrenheit. Equation to convert Celsius to Fahrenheit: $F = (9/5) * C + 32.$ ii. Write a program to compute the slope of a line between two points (x1, y1) and (x2, y2). $\text{Slope} = \frac{y_2 - y_1}{x_2 - x_1}$ iii. Write a program to calculate simple and compound interest. $\text{Simple Interest} = \frac{P * R * T}{100}$ $\text{Compound Interest} = P * \left(1 + \frac{R}{100 * n}\right)^{n * T}$	1	08

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
	<p>iv. Write a program to get change values in Quarter, Dime, Nickels and Pennies, and calculate the value of change in Dollars. Consider Quarter = 0.25 \$, Dime = 0.10 \$, Nickels = 0.05 \$ and Penny = 0.01 \$.</p> <p>v. Write a program to find a maximum of given three numbers (Use ternary operator).</p> <p>vi. Write a program to calculate area and volume of Sphere. $\text{Area of Sphere} = 4 \pi r^2$ $\text{Volume of Sphere} = \frac{4}{3} \pi r^3$</p> <p>vii. Write a program that computes the real roots of a given quadratic equation (Use math library). $\text{Discriminant } \Delta = b^2 - 4 a c$ $\text{Real Roots} = \frac{-b \pm \sqrt{\Delta}}{2 a}$</p> <p>viii. Write a program to determine the length of ladder required to reach a given height when leaned against the house. The height and the angle of the ladder are given as inputs (Use math Library).</p>		
4	<p>Decision-Making Structures</p> <p>i. A year is a Leap year if it is divisible by 4, unless it is a century year that is not divisible by 400 (1800 and 1900 are not leap years, 1600 and 2000 are leap years). Write a program that calculates whether a given year is a leap year or not.</p> <p>ii. Many companies pay time-and-a-half for any hours worked above 40 hours in a given week. Write a program to input the number of hours worked and hourly rate and calculate the total wages for the week.</p> <p>iii. The Body Mass Index (BMI) is calculated as a person's weight (in kg), divided by the square of the person's height (in meters). If the BMI is between 19 and 25, the person is healthy. If the BMI is below 19, then the person is underweight. If the BMI is above 25, then the person is overweight. Write a program to get a person's weight (in kgs) and height (in cms) and display a message whether the person is healthy, underweight or overweight. $BMI = \frac{\text{Weight in kg}}{(\text{Height in m})^2}$</p>	2	04

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required										
	iv. Write a program to read the marks and assign a grade to a student. Grading system: A (≥ 90), B (80-89), C (70-79), D (60-69), E (50-59), F (< 50). (Use the Switch case)												
5	<p>Loops</p> <p>i. Write a program to read n numbers from users and calculate the average of those n numbers.</p> <p>ii. Write a program that prompts the user to enter 10 integers and displays all the combinations of picking two numbers from the 10.</p> <p>iii. Write programs to print below patterns:</p> <table border="1" data-bbox="347 768 805 992"> <tbody> <tr> <td style="text-align: center;">*</td> <td style="text-align: center;">1</td> </tr> <tr> <td style="text-align: center;">* *</td> <td style="text-align: center;">1 2</td> </tr> <tr> <td style="text-align: center;">* * *</td> <td style="text-align: center;">1 2 3</td> </tr> <tr> <td style="text-align: center;">* * * *</td> <td style="text-align: center;">1 2 3 4</td> </tr> <tr> <td style="text-align: center;">* * * * *</td> <td style="text-align: center;">1 2 3 4 5</td> </tr> </tbody> </table> <p>iv. Write a program that displays an ASCII character table from ! to ~. Display the ASCII value of a character in decimal and hexadecimal. Display five characters per line.</p> <p>v. Write a program to sum the following series: $\frac{1}{3} + \frac{3}{5} + \frac{5}{7} + \frac{7}{9} + \frac{9}{11} + \frac{11}{13} + \dots + \frac{95}{97} + \frac{97}{99}$</p> <p>vi. A positive integer is called a perfect number if it is equal to the sum of all of its positive divisors, excluding itself. For example, 6 is the first perfect number, because $6 = 3 + 2 + 1$, the next is $28 = 14 + 7 + 4 + 2 + 1$. There are four perfect numbers that are less than 10,000. Write a program to find these four numbers.</p>	*	1	* *	1 2	* * *	1 2 3	* * * *	1 2 3 4	* * * * *	1 2 3 4 5	2	06
*	1												
* *	1 2												
* * *	1 2 3												
* * * *	1 2 3 4												
* * * * *	1 2 3 4 5												
6	<p>Lists</p> <p>i. Write a program to perform the below operations on the list:</p> <ul style="list-style-type: none"> • Create a list. • Add/Remove an item to/from a list. • Get the number of elements in the list. • Access elements of the list using the index. • Sort the list. • Reverse the list. <p>ii. Write a program to read n numbers from a user and print:</p> <ul style="list-style-type: none"> • Number of positive numbers. • Number of negative numbers. • Number of zeros. 	3	05										

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
	<ul style="list-style-type: none"> ● Number of odd numbers. ● Number of even numbers. ● Average of all numbers. <p>iii. Write a program that counts the occurrences of each digit in a string. The program counts how many times a digit appears in the string. For example, if the input is "12203AB3", then the output should output 0 (1 time), 1 (1 time), 2 (2 times), 3 (2 times).</p> <p>iv. Write a program to eliminate duplicate values in the list.</p> <p>v. Write a program to randomly fill in 0s and 1s into a 4x4 2-dimension list, print the list and find the rows and columns with the most number of 1s.</p>		
7	<p>Tuples, Sets and Dictionaries</p> <p>i. Write a program to perform below operations on tuple:</p> <ul style="list-style-type: none"> ● Create a tuple with different data types. ● Print tuple items. ● Convert tuple into a list. ● Remove data items from a list. ● Convert list into a tuple. ● Print tuple items. <p>ii. Write a program to perform below operations on set:</p> <ul style="list-style-type: none"> ● Create two different sets with the data. ● Print set items. ● Add/remove items in/from a set. ● Perform operations on sets: union, intersection, difference, symmetric difference, check subset of another set. <p>iii. Write a program to perform below operations on dictionary:</p> <ul style="list-style-type: none"> ● Create a dictionary. ● Print dictionary items. ● Add/remove key-value pair in/from a dictionary. ● Check whether a key exists in a dictionary. ● Iterate through a dictionary. ● Concatenate multiple dictionaries. <p>iv. Write a program that is given a dictionary containing the average daily temperature for each day of the week, and prints all the days on which the average temperature was between 40 and 50 degrees.</p> <p>v. Write a program to repeatedly prompt the user to enter the capital of a state. Upon receiving the user's input, the program reports whether the answer is correct. Assume the</p>	3	05

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
	states and their capitals are stored in dictionaries as key-value pairs.		
8	<p>Function</p> <p>i. Write a program that defines a function (shuffle) to scramble a list into a random order, like shuffling a deck of cards.</p> <p>ii. Write a program that defines a function to return a new list by eliminating the duplicate values in the list.</p> <p>iii. Write a program to print Fibonacci sequence up to n numbers using recursion. Fibonacci sequence is defined as below:</p> $\text{Fibonacci Sequence} = 1\ 1\ 2\ 3\ 5\ 8\ 13\ 21\ \dots$ <p>where n^{th} term $x_n = x_{n-1} + x_{n-2}$</p> <p>iv. Write a program that defines a function to determine whether input number n is prime or not. A positive whole number $n > 2$ is prime, if no number between 2 and \sqrt{n} (inclusive) evenly divides n. If n is not prime, the program should quit as soon as it finds a value that evenly divides n.</p> <p>v. Write a program that defines a function to find the GCD of two numbers using the algorithm below. The greatest common divisor (GCD) of two values can be computed using Euclid's algorithm. Starting with the values m and n, we repeatedly apply the formula: $n, m = m, n \% m$ until m is 0. At that point, n is the GCD of the original m and n (Use Recursion).</p> <p>vi. Write a program that lets the user enter the loan amount, number of years, and interest rate, and defines a function to calculate monthly EMI, total payment and display the amortization schedule for the loan.</p>	4	06
9	<p>Modules</p> <p>i. Write a program that defines functions (mean and deviation), that computes mean and standard deviation of given numbers. The formula for the mean and standard deviation of n numbers is given as:</p> $\text{mean} = \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n}$	4	06

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
	$deviation = \sqrt{\frac{\sum_{i=1}^n (x_i - mean)^2}{n - 1}}$ <p>ii. Write a program that plays the popular scissor-rock-paper game. (A scissor can cut a paper, a rock can knock a scissor, and a paper can wrap a rock.) The program randomly generates a number 0, 1, or 2 representing scissor, rock, and paper. The program prompts the user to enter a number 0, 1, or 2 and displays a message indicating whether the user or the computer wins, loses, or draws.</p> <p>iii. Write a program to print the dates of all the Sundays in a given year.</p> <p>iv. Write a program to display a graph for ReLU (Rectified Linear Unit) function. ReLU function is defined as below: $y = \max(0, x)$ Consider the range of x from -5 to 5.</p> <p>v. Write a program to create a list representing the results of 100 students in a test, where each element represents a student's marks (between 0 to 10), and display a histogram for the result.</p> <p>vi. Create a user defined module with simple functions for: addition, subtraction, multiplication, division, modulo, square, factorial. Write a program to import the module and access functions defined in the module.</p>		
10	<p>String Processing</p> <p>i. Write a program to check whether a given string is palindrome or not.</p> <p>ii. Write a program to read a string containing letters, each of which may be in either uppercase or lowercase, and return a tuple containing the number of vowels and consonants in the string.</p> <p>iii. Write a program to read a date in the format DD/MM/YYYY and print the same date in MM-DD-YYYY format.</p> <p>iv. Write a program that checks whether two words are anagrams. Two words are anagrams if they contain the same letters. For example, <i>silent</i> and <i>listen</i> are anagrams.</p> <p>v. Write a program that allows users to enter six-digit RGB color codes and converts them into base 10. In this format, the first two hexadecimal digits represent the amount of red, the second two the amount of green, and the last two the</p>	5	06

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
	<p>amount of blue. For example: If a user enters FF6347, then the output should be Red (255), Green (99) and Blue (71).</p> <p>vi. Numerologists claim to be able to determine a person's character traits based on the "numeric value" of a name. The value of a name is determined by summing up the values of the letters of the name, where "a" is 1 "b" is 2 "c" is 3 and so on up to "z" being 26. For example, the name "Python" would have the value $16 + 25 + 20 + 8 + 15 + 14 = 98$. Write a program that calculates the numeric value of a name provided as input.</p>		
11	<p>File Handling</p> <p>i. Write a program to perform the below operations on files:</p> <ul style="list-style-type: none"> • Create a text file and write a string to it. • Read an entire text file. • Read a text file line by line. • Write a string to a file. • Write a list of strings to a file. • Count the number of lines, words in a file. <p>ii. Write a program that reads a text file and counts the occurrences of each alphabet in the file. The program should prompt the user to enter the filename.</p> <p>iii. Write a program that reads a text file and displays all the numbers found in the file.</p> <p>iv. Write an automated censor program that reads the text from a file and creates a new file where all of the four-letter words have been replaced by "****". You can ignore punctuation, and you may assume that no words in the file are split across multiple lines.</p> <p>v. Write a program that reads a text file and calculates the average word length and sentence length in that file.</p> <p>vi. Write a program that reads two strings stored in two different text files and prints a string containing the characters of each string interleaved. Remove white spaces from both strings before string interleaving. For example, Two strings "Hello World" and "Sky is the Limit" should generate output "HSeklyliosWtohreLDimit"</p>	5	06
	Total		56

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	Correctness of the program.	30
2	Readability and documentation of the program/Quality of input and output displayed (messaging and formatting).	10
3	Code efficiency.	20
4	Debugging ability.	20
5	Execution of the program/answer to the sample questions.	20
Total		100

6. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment/instruments and Software required to develop PrOs are given below with broad specifications to facilitate procurement of them by the administrators/management of the institutes. This will ensure the proper conduct of practicals in all institutions across the state in a proper way so that the desired skills are developed in students.

S. No.	Equipment Name with Broad Specifications	PrOs. No.
1	Computer with latest configuration with Windows/Linux/Unix Operating System.	All
2	Text Editor (VS Code, Sublime Text, Atom, Vim or any other editor) or Python IDE (IDLE, PyCharm, PyDev, Spyder or any other IDE)	All
3	Python Interpreter (Versions: 3.6.x or higher)	All
4	Jupyter Notebook (Optional)	All

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) Practice good housekeeping.
- b) Demonstrate working as a team leader/member.
- c) Maintain tools and equipments.
- d) Adhere to ethical practices.
- e) Follow safety 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 1st year
- ii. 'Organization Level' in 2nd year.
- iii. 'Characterization Level' in 3rd year.

8. UNDERPINNING THEORY

The major underpinning Theory is formulated as given below and only higher level UOs of Revised Bloom's taxonomy are mentioned for the development of the COs and competency in the students by the teachers. (Higher level UOs automatically include lower level UOs in them). 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)	Topics and Sub-topics
Unit-1: Fundamentals of Python	1.a Install and configure Python 1.b Explain general structure of python program 1.c Develop programs using variables, operators and input-output functions	1.1 Introduction to Python, History of Python, Python Features, Python Applications 1.2 Installing Python 1.3 Basic Structure of Python program 1.4 Keywords and Identifiers 1.5 Data types and Variables 1.6 Type Casting 1.7 Input-Output functions: input, print 1.8 Operators
Unit-2: Control Flow Structures	2.a Explain different types of Control Structures 2.b Develop programs using Decision making Structures 2.c Develop programs using Loops	2.1 Introduction to Control Structures Decision Making Structures 2.2 if, if-else statements 2.3 Nested if-else and if-elif-else statements 2.4 switch statement Loops 2.5 for loop 2.6 while loop 2.7 Nested loops 2.8 break, continue and pass statements
Unit-3: Lists, Tuples, Sets and Dictionaries	3.a Develop program using Structured types – Tuples, Lists, Sets and Dictionaries	3.1 Lists and operations on Lists 3.2 Tuples and operations on Tuples 3.3 Sets and operations on Sets 3.4 Dictionaries and operations on Dictionaries

Unit	Unit Outcomes (UOs)	Topics and Sub-topics
Unit-4: Functions and Modules	4.a Develop programs using function 4.b Develop programs using recursion 4.c Use Built-in functions in programs 4.d Develop programs using rand, math, datetime modules	4.1 Introduction to Python User defined Function 4.2 Passing parameters to a function and returning values from a function 4.3 Recursion 4.4 Standard Library: Built-in Functions 4.5 Modules and Packages <ul style="list-style-type: none"> ● rand module - Random numbers generators ● math module – Mathematical functions ● datetime module - Date and time functions ● matplotlib module – Plotting functions 4.6 Create and import custom user defined module
Unit-5: String Processing and File Handling	5.a Develop programs to create and access strings 5.b Use built-in functions to manipulate strings 5.c Develop programs using text files	5.1 Introduction to String 5.2 Access String elements using index operator 5.3 String functions <ul style="list-style-type: none"> ● Basic functions: len, max, min ● Testing functions: isalnum, isalpha, isdigit, isidentifier, islower, isupper, and isspace ● Searching functions: endswith, startswith, find, rfind, count ● Manipulation functions: capitalize, lower, upper, title, swapcase, replace, lstrip, rstrip, strip ● Formatting functions: format, center, ljust, rjust 5.4 Introduction to Text files 5.5 File Handling functions: <ul style="list-style-type: none"> ● Basic functions: open, close ● Reading file: read, readline, readlines ● Writing file: write, append, writelines

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	Fundamentals of Python	7	2	4	4	10
II	Control Flow Statements	9	2	4	8	14
III	Lists, Tuples, Sets and Dictionaries	8	4	4	6	14
IV	Functions and Modules	9	4	4	8	16
V	String Processing and File Handling	9	4	4	8	16
Total		42	16	20	34	70

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

Note: This specification table provides general guidelines to assist student 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 above table.

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, the 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 the following activities in groups and prepare small reports (of 1 to 5 pages for each activity). For micro project reports should be as per the suggested format. For other activities, students and teachers together can decide the format of the report. Students should also collect/record physical evidence such as photographs/videos of the activities for their (student's) portfolio, which will be useful for their placement interviews:

- Students are encouraged to use Jupyter Notebook for lab activities and projects.
- Undertake micro-projects in teams.
- Make a list of the Python-based applications or software.
- Students are encouraged to register themselves in various MOOCs such as: Swayam, edx, Coursera, Udemy etc. to further enhance their learning.
- Encourage students to participate in different coding competitions like hackathons, online competitions on codechef etc.
- Encourage students to form a coding club at institute level and can help the slow learners.
- <https://www.codechef.com>, in this website, very elementary programs are available, students are expected to solve those programs using Python programming.
- <https://code.org/>, an hour of code may be organized and students are encouraged to participate.

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies that 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) Guide students to do personalized learning so that students can understand the course material at their own pace.
- e) Encourage students to do group learning by sharing so that teaching can easily be enhanced.
- f) 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.
- g) With respect to **section No.10**, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- h) Diagnose Essential missed learning concepts that will help the students.
- i) Using the knowledge gained from this course, instruct students on how to address environmental and sustainability issues.

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 dated work diary consisting of individual contribution 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 *14-16 (sixteen) student engagement hours* during the course. The student ought to submit 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:

- a) Automating Emails: Develop a Python project to send emails automatically.
- b) Alarm Clock: Develop a Python project for an alarm clock.
- c) Unit Conversion Calculator: Develop a Python project to convert between commonly used units.
- d) Scraping Google Results: Develop a Python project that can scrape results from google based on some query.
- e) Weather app: Develop a Python project that takes the city name and returns the weather information of that city using web scraping.

- f) Article Reader: Develop a Python project that automatically reads the article from the provided link.
- g) Audio Book: Develop a Python project that can be used to convert a pdf into an audiobook.
- h) URL shortener: Develop a Python script that shortens a given URL.
- i) Tic-Tac-Toe: Develop a Python project for tic-tac-toe game.
- j) Desktop Notifier in Python: Develop a Python project to generate pop-up notification messages on desktop.
- k) Morse Code Encryption/Decryption program: Develop a Python project that should be able to both convert an English message file into Morse code, and a Morse code file into English.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	Introduction to Computer Science Using Python: A Computational Problem–Solving Focus	Charles Dierbach	John Wiley & Sons (25 January 2013) ISBN-10: 0470555157 ISBN-13: 978-0470555156
2	Python Programming: An Introduction to Computer Science	John Zelle	Franklin, Beedle & Associates Inc; Pap/Cdr edition (1 December 2003) ISBN-10 : 1590280288 ISBN-13 : 978-1887902991
3	Introduction to Programming Using Python	Liang Y. Daniel	Pearson Education; First edition (26 February 2017) ISBN-10: 9332551847 ISBN-13: 978-9332551848
4	Core Python Programming	R. Nageswara Rao	Dreamtech Press (1 September 2021), Delhi; ISBN-10: 9390457157 ISBN-13: 978-9390457151
5	Head First Python	Paul Barry	Shroff/O'Reilly; Second edition (1 December 2016). ISBN-10: 9789352134823 ISBN-13: 978-9352134823
6	Learning Python	Lutz M	Shroff; Fifth edition (1 January 2013) ISBN-10:9351102017 ISBN-13:9351102014-978

14. SOFTWARE/LEARNING WEBSITES

- a) <https://www.python.org>
- b) <https://www.learnpython.org>
- c) <https://www.python-course.eu>
- d) <https://nptel.ac.in>
- e) <https://www.youtube.com>
- f) <https://www.edx.org>
- g) <https://www.coursera.org/in>
- h) <https://www.udemy.com>
- i) <https://www.codecademy.com>

15. PO-COMPETENCY-CO MAPPING

Semester III	Scripting Language - Python (Course Code: 4330701)									
	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 and Testing	PO 5 Engineering practices for society, sustainability and environment	PO 6 Project Management	PO 7 Life-long learning	PSO 1	PSO 2	PSO 3 (If needed)
Competency	Develop simple applications using Python to solve the given problem.									
CO a) Develop programs to solve the given simple computational problems.	3	2	1	2	-	-	1			
CO b) Apply control flow structures to solve the given problems.	3	2	2	2	-	2	2			
CO c) Implement data structures lists, tuples, sets and dictionaries to solve the given problems.	2	1	2	2	-	2	2			
CO d) Apply modular programming approach to solve the given problems using user-defined functions.	2	2	3	2	-	3	3			
CO e) Perform string manipulation and file operations to solve the given problems.	2	2	3	2	-	2	2			

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	Contact No.	Email
1	Shri K N Raval, Head of the Computer Department	DTE	9428011250	raval.kamlesh@gmail.com
2	Smt. Manisha Mehta Head of Computer Department	Government Polytechnic - Gandhinagar	9879578273	manishamehtain@gmail.com
3	Smt. Jasmine Kargathala, Lecturer in Computer Engineering	Government Girls Polytechnic - Ahmedabad	9824799620	jdaftary@gmail.com
4	Shri Kartik Detroja Lecturer in Computer Engineering	Government Polytechnic - Porbandar	9972419091	detroja.kartik@gmail.com

GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)

Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021) Semester-III

Course Title: Relational Database Management Systems (Course Code: 4330702)

Diploma programme in which this course is offered	Semester in which offered
Computer Engineering	Third

1. RATIONALE

The aim of this course is to get broad understanding of the basic concepts of database management system used for business, scientific and engineering application which stored centralised. The students will develop the skills to develop manage & retrieve data from different perspective using Structured Query Language (SQL) in ORACLE (centralized storage) so there is no need of storing data in files and paper. This will turn reduce of paper wastage. By the end of this course the students will be able to write simple and advanced PL/SQL code blocks, use advanced features such as cursors and bulk fetches and database designing with normalization. Hence students will be able to design database which will be helpful to them in the designing phase of project in the upcoming semester.

2. COMPETENCY

The course should be taught and implemented with the aim to develop various types of skills so that students are able to acquire following competency:

- **Design, Develop and manage databases for simple applications using Structured Query Language (SQL) and PL/SQL in ORACLE.**

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:

- a) Perform queries on datasets using SQL*Plus
- b) Perform joins, subqueries and nested queries on multiple tables using SQL*plus
- c) Apply rules on datasets using SQL*Plus constraints
- d) Apply various Normalization techniques.
- e) Write PL/SQL block using concept of Cursor Management, Error Handling, Package and Triggers

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T/2+P/2)	Examination Scheme			
				Theory Marks		Practical Marks	
L	T	P	C	CA	ESE	CA	ESE

3	-	4	5	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) that are the sub-components of the COs. *Some of the PrOs marked ‘*’ are compulsory, as they are crucial for that particular CO at the ‘Precision Level’ of Dave’s Taxonomy related to ‘Psychomotor Domain’.*

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
1	Implement SQL queries to perform various DDL Commands. (Create minimum 5 tables with different data types and operate upon them)	I	02
2	a. Implement SQL queries to perform various DML Commands. (Insert minimum 10 rows using different insert methods, edit and remove data using update and delete commands) b. Retrieve data using SELECT command and various SQL operators.	I	04
3	Perform queries for TCL and DCL Commands	I	02
4	Implement SQL queries using Date functions like add-months, months-between, round, nextday, truncate etc	II	02
5	Implement SQL queries using Numeric functions like abs, ceil, power, mod, round, trunc, sqrt etc. and Character Functions like initcap, lower, upper, ltrim, rtrim, replace, substring, instr etc.	II	02
6	Implement SQL queries using Conversion Functions like to-char, to-date, to-number and Group functions like Avg, Min, Max, Sum, Count, Decode etc.	II	02
7	Implement SQL queries using Group by, Having and Order by clause	II	04
8	Implement SQL queries using simple Case Operations and using Group functions and Case operations for getting summary data	II	02
9	Implement SQL queries using Set operators like Union, unionall, Intersect, Minus etc.	II	02
10	Retrieve data spread across various tables or same table using various Joins.	II	02
11	Retrieve data from multiple tables using Subqueries (Multiple, Correlated) (write minimum 3 level subquery)	II	04
12	Perform queries to Create, alter and update views	III	02
13	Implement Practical-1 again with Domain Integrity, Entity Integrity and Referential Integrity constraints.	III	02
14	Perform queries to Create synonyms, sequence and index	III	02
15	Implement PL/SQL programs using control structures	IV	02
16	Implement PL/SQL programs using Cursors	IV	04
17	Implement PL/SQL programs using exception handling.	IV	02

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
18	Implement user defined procedures and functions using PL/SQL blocks	IV	04
19	Perform various operations on packages.	IV	02
20	Implement various triggers	IV	04
21	Draw E-R Diagram of the given problem statements.	V	06
22	Practices on Normalization – using any database perform various normal forms.	V	04
Total			62

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	Problem Analysis	20
2	Development of the Solution	20
3	Testing of the Solution	10
4	Record observations correctly	20
5	Interpret the result and conclude	30
Total		100

6. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

- a. Hardware: Computer Systems with minimum PIV processor (or equivalent) and 1 GB RAM.
- b. Software: SQL/PLSQL supporting software. (e.g. Oracle, SQLServer, MySQL)

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 fulfill the development of this competency.

- a) Work as a leader/a team member.
- b) Follow ethical practices.
- c) Practice environment friendly methods and processes. (Environment related)

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 1st year
- ii. 'Organization Level' in 2nd year.
- iii. 'Characterization Level' in 3rd year.

9. 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
Unit – I Introduction to Database System and SQL commands	1a. Differentiate the terms: Data, Information, Records, Fields, Metadata, Data warehouse, Data dictionary	1.1 Concepts and Definitions: Database and database systems and database environment 1.2 Data, Information, Data Item or Fields, Records, Files, Metadata, Data dictionary and it's components, 1.3 Schemas, Sub-schemas, and Instances
	1b. DBMS Data types, Creating Tables (DDL), Managing Tables (DML) with SQL,	1.4 Data types 1.5 Database Language commands: Data Definition Language (DDL): CREATE, ALTER, TRUNCATE, DROP 1.6 Database Language: Data Manipulation Language (DML): INSERT, SELECT, UPDATE, DELETE
	1c. Describe & practice Transaction Control Data Control Language	1.7 Transactional Control: Commit, Save point, Rollback 1.8 DCL Commands: Grant and Revoke
Unit – II SQL In built functions and Joins	2a. Execute various SQL operators and Functions	2.1 Operators Arithmetic, Comparison, Logical SQL functions- Single row function i. Single row function. ii. Date functions (add-months, months-between, round, truncate). iii. Numeric Functions (abs, power, mod, round, trunc, sqrt) iv. Character Functions (initcap, lower, upper, ltrim, rtrim, replace, substring, instr) v. Conversion Functions (to-char, to-date, to-number)
	2b. Perform queries on 'Group by', 'Having' and 'Order by' clause	2.2 Groupby, Having and Order by clause
	2c. Implement 'Joins'	2.3 Joins: Simple, Equi-join, Non-equi, Self-Joins, Outer-joins.

Unit	Unit Outcomes (UOs) (4 to 6 UOs at Application and above level)	Topics and Sub-topics
	2d. Perform different types of sub queries	2.4 Subqueries - Multiple, Correlated 2.5 Implementation of Queries using SQL Set operators: Union, union all, Intersect, Minus
Unit– III Database Integrity Constraints & Objects	3a. Describe with examples Domain Integrity and Entity Integrity constraint	3.1 Domain Integrity constraints: Not null, Check 3.2 Entity Integrity constraints: Unique, Primary key.
	3b. Describe with examples Referential Integrity constraint	3.3 Referential Integrity constraints: Foreign key, referenced key, on delete cascade
	3c. Database Objects	3.4 Views – Create, Alter, Drop views 3.5 Synonym: Create, Drop synonym 3.6 Sequences: Create, alter, Drop sequences 3.7 Index: Unique and composite – Create, Drop
Unit– IV PL/ SQL and Triggers	4a. Describe the fundamentals of the PL/SQL programming language	4.1 Basics of PL / SQL 4.1.1 Data types 4.2 Advantages of PL/SQL over SQL
	4b. Use different Control Structures 4c. Write and execute PL/SQL programs in SQL*Plus	4.3 Control Structures: Conditional, Iterative, Sequential
	4d. Describe & Implement Concepts of exception handling	4.4 Exceptions: Predefined Exceptions, User defined exceptions
	4e. Implement cursor, procedure and function in Package	4.5 Cursors: Static (Implicit & Explicit), Dynamic 4.6 Procedures & Functions
	4f. Describe the various types of triggers 4g. Write, code, test and debug various types of triggers	4.7 Fundamentals of Database Triggers 4.8 Creating Triggers 4.9 Types of Triggers: Before, after for each row, for each statement
Unit– V Normalization	5a. Describe different Normal Forms 5b. Solve problems of normalization 5c. Describe advantages and disadvantages of Normalization	5.1 Basics of Normalization 5.2 Normal Forms 4.2.1 First Normal Form(1NF) 4.2.2 Second Normal Form(2NF) 4.2.3 Third Normal Form(3NF) 5.3 Advantages and disadvantages of Normalization

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.

10. SUGGESTED SPECIFICATION TABLE FOR QUESTIONPAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Introduction to Database System and SQL commands	10	7	5	4	16
II	SQL In built functions and Joins	8	5	4	5	14
III	Database Integrity Constraints & Objects	8	5	4	5	14
IV	PL/ SQL and Triggers	10	6	5	5	16
V	Normalization	6	4	3	3	10
Total		42	27	21	22	70

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

Note: This specification table provides general guidelines to assist student 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 above table.

11. 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:

- i. Prepare seminar presentations explaining the organization of database in various live systems like banking, insurance, online booking etc.
- ii. Prepare power point presentation for different database objects.
- iii. Prepare case study explaining the need for converting a large table to many smaller tables using 1NF, 2NF, 3NF.
- iv. Design database which can be used in the course on .Net programming

12. 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/sub topics.
- 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**.
- f) Guide students on how to address issues on environ and sustainability

13. 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 is 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 dated work diary consisting of individual contribution 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 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:

- Do related activities such as prepare E-R Model, Relational Model, do Normalization, Create Tables, insert data, Delete Data, Query database, create stored procedures, etc.
 - a) **Effective Library Management System:** Book, student and faculty record with issuance, retrieval and fine details
 - b) **Payroll:** Useful for all organizations as every organization definitely has employees. No of days worked, salary, leaves etc. can be kept.
 - c) **University:** Details of department and its courses. Track of students, teachers and staff can also be kept.
 - d) **IT TRAINING GROUP DATABASE PROJECT:** This can be clearly indicated with an ER diagram. Entities, Relationships and Key constraints should be clearly indicated. The company has 50 instructors and can handle 500 trainees for each training session. There are 20 advanced technology courses in all that company offers. Each course is well handled by a team of 5 or more trainers. Maximum of 3 teaching teams is assigned to every instructor. They might also be asked to do some research. One advanced technology course is handled by per trainee per session.
 - e) **Employee database:** ‘Create’ employee table, ‘Select’ and display an employee matching a given condition, ‘Delete’ duplicate records, delete rows using triggers, insert and update records, find net salary, etc.
 - f) Case Study: **Visitor Management database**
 - g) Case Study: **Students Academic database**
 - h) Case Study: **Inventory Management System database**
 - i) Case study: **Bank Operations database**
 - j) Case Study: **Bus Operator (Roadways)**

14. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	SQL/PL/SQL	Bayross, Ivan	BPB, NewDelhi, 2010.
2	Database Systems Concepts, design and	Singh, S.K.	Pearson Education, New Delhi,2011

S. No.	Title of Book	Author	Publication with place, year and ISBN
	Applications 2/e		
3	An Introduction to Database Systems	Date, C. J.	Pearson Education, New Delhi,2006
4	Database System Concepts,	Korth, Henry	McGrawHill,Delhi,2011
5	Introduction to Database Systems	ITLESL.	Pearson Education, New Delhi,2010

15. SOFTWARE/LEARNING WEBSITES

- DBMS: <http://nptel.iitm.ac.in/video.php?subjectId=106106093>
- SQL Plus Tutorial: <http://holowczak.com/oracle-sqlplus-tutorial/>
- Database Tutorials:<http://www.roseindia.net/programming-tutorial/Database-Tutorials>
- SQL Basic Concepts: <http://www.w3schools.com/sql/>
- SQL Tutorial : <http://beginner-sql-tutorial.com/sql.htm>

16. PO-COMPETENCY-CO MAPPING

Semester III	Relational Database Management Systems (Course Code:									
	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	PSO 1	PSO 2	PSO 3 (If needed)
Competency Design, Develop and manage databases for simple applications using Structured Query Language (SQL) and PL/SQL in ORACLE										
Course Outcomes CO a) Perform queries on datasets using SQL*Plus	3	-	-	-	1	-	1	-	-	
CO b) Perform joins, sub queries and nested queries on multiple tables using SQL*plus	3	2	1	2	1	-	1	-	-	
CO c) Apply rules on datasets using SQL*Plus constraints	2	3	1	-	2	2	1	1	-	
CO d) Apply various Normalization techniques.	3	1	1	1	1	2	1	1	-	
CO e) Perform PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers.	3	2	2	1	1	1	1	-	-	

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

17. COURSE CURRICULUM DEVELOPMENT COMMITTEE

GTU Resource Persons

S. No.	Name and Designation	Institute	Contact No.	Email
1	Prof. B.H. Kantewala , Head Department of Computer Engineering	CU Shah Polytechnic, Surendranagar	9428000592	bhkantevala@yahoo. com
2	Prof. M. P. Mehta , Head Department of Computer Engineering	Govt. Polytechnic, Gandhinagar	9879578273	manishamehtain@g mail.com
3	Prof. Sachin D. Shah , Lecturer Department of Computer Engineering	RCTI, Ahmedabad	9427955671	sachindshah@yahoo. com
4	Miss Dhara H. Wagh Lecturer Department of Computer Engineering	Govt. Polytechnic, Gandhinagar	9427465614	dhara.wagh1791@g mail.com
5	Ms. Pravina R. Mehta , Lecturer Department of Computer Engineering.	Govt. Polytechnic, Himatnagar	9825446175	pravina6mehta@gm ail.com

GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)

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

Semester-III

Course Title: Basics of Operating System

(Course Code: 4330703)

Diploma programme in which this course is offered	Semester in which offered
Computer Engineering	Third

1. RATIONALE

As a core subject of Computer Engineering, this course enables to understand importance of Operating System, its functionalities to manage resources of Computer and Peripherals, program development and its execution. Every student of computer science must therefore understand basic structure of an operating system. After learning this subject student will be able to discriminate between various types of operating systems, its processor, processes, and memory and file management. The subject also emphasizes on Linux utilities and scripting.

2. COMPETENCY

The course content should be taught and implemented with the aim to develop various types of related skills leading to the achievement of the following competency

- **Manage operations of Operating Systems.**

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:

The practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry-oriented COs associated with the above-mentioned competency:

- a) Differentiate operating systems based on their features.
- b) Apply scheduling algorithms to calculate turnaround time and average waiting time.
- c) Interpret various memory management techniques.
- d) Apply File management techniques.
- e) Execute basic Linux commands and Shell scripts.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (CI+T/2+P/2)	Examination Scheme				
CI	T	P		Theory Marks		Practical Marks		Total Marks
3	0	2	C	CA	ESE	CA	ESE	
			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: CI-ClassRoom Instructions; 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) that are the sub-components of the COs. *Some of the PrOs marked ‘*’ are compulsory, as they are crucial for that particular CO. These PrOs need to be attained at least at the ‘Precision Level’ of Dave’s Taxonomy related to ‘Psychomotor Domain’.*

Sr.No	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Req																				
1	Compare windows and Linux OS. (latest version)	I	02																				
2	Solve below given example with SJF, FCFS and Round robin algorithm. Draw Gantt chart. <table border="1" style="margin: 10px auto;"> <thead> <tr> <th>Process</th> <th>Arrival Time</th> <th>Execution Time</th> <th>Service Time</th> </tr> </thead> <tbody> <tr> <td>P0</td> <td>0</td> <td>5</td> <td>0</td> </tr> <tr> <td>P1</td> <td>1</td> <td>3</td> <td>5</td> </tr> <tr> <td>P2</td> <td>2</td> <td>8</td> <td>8</td> </tr> <tr> <td>P3</td> <td>3</td> <td>6</td> <td>16</td> </tr> </tbody> </table>	Process	Arrival Time	Execution Time	Service Time	P0	0	5	0	P1	1	3	5	P2	2	8	8	P3	3	6	16	II	02
Process	Arrival Time	Execution Time	Service Time																				
P0	0	5	0																				
P1	1	3	5																				
P2	2	8	8																				
P3	3	6	16																				
3	Process requests are given as: 25 K , 50 K , 100 K , 75 K <div style="text-align: center; margin: 10px 0;"> <table style="border-collapse: collapse; margin: 0 auto;"> <tr> <td style="padding: 0 10px;">50 K</td> <td style="padding: 0 10px;">75 K</td> <td style="padding: 0 10px;">150 K</td> <td style="padding: 0 10px;">175 K</td> <td style="padding: 0 10px;">300 K</td> </tr> <tr> <td style="text-align: center;">○</td> <td style="text-align: center;">○</td> <td style="text-align: center;">○</td> <td style="text-align: center;">○</td> <td style="text-align: center;">○</td> </tr> </table> </div> Solve above example using following algorithms: <ol style="list-style-type: none"> 1. First fit 2. Best fit 3. Worst fit 	50 K	75 K	150 K	175 K	300 K	○	○	○	○	○	III	02										
50 K	75 K	150 K	175 K	300 K																			
○	○	○	○	○																			
4	Page replacement algorithms <ol style="list-style-type: none"> a. First in First out (FIFO) - Consider page reference string 1, 3, 0, 3, 5, 6, 3 with 3 page frames. Find the number of page faults. b. Least Recently Used – Consider the page reference string 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2 with 4 page frames. Find number of page faults. 	III	04																				
5	Disk Scheduling Algorithms	IV	04																				

	1. Scan: Suppose the requests to be addressed are- 82,170,43,140,24,16,190. And the Read/Write arm is at 50, and it is also given that the disk arm should move “ towards the larger value ”. 2. CScan: Suppose the requests to be addressed are- 82,170,43,140,24,16,190. And the Read/Write arm is at 50, and it is also given that the disk arm should move “ towards the larger value ”.		
6	Test and run basic unix commands.	V	02
7	Test and run Advanced unix commands.	V	02
8	Test commands related with File editing with Vi, Vim, gedit, gcc.	V	02
9	Create a shell script to read from command line and print “Hello”.	V	02
10	Create a Shell script to read and display content of a file. And append content of one file to another	V	02
11	Create a Shell script to accept a string in lower case letters from a user, & convert to upper case letters.	V	02
12	Create a Shell script to add two numbers.	V	02
	Total		28

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	Correctness of solution/answer	30
2	Interpret and Solve various algorithms	30
3	Debugging ability	20
4	Program execution/answer to sample questions	20
Total		100

6. MAJOR EQUIPMENT/ INSTRUMENTS AND SOFTWARE REQUIRED

These major equipment/instruments and Software required to develop PrOs are given below with broad specifications to facilitate procurement of them by the administrators/management of the institutes. This will ensure conduction of practical in all institutions across the state in proper way so that the desired skills are developed in students.

S. No.	Equipment Name with Broad Specifications	PrO. No.
1	Computer with latest configuration with windows or UNIX OS	All
2	Linux based Operating system	All

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 1st year
- ii. 'Organization Level' in 2nd year.
- iii. 'Characterization Level' in 3rd year.

8. UNDERPINNING THEORY

The major Underpinning Theory is formulated as given below and only higher level UOs of *Revised Bloom's taxonomy* are mentioned for development of the COs and competency in the students by the teachers. (Higher level UOs automatically includes lower level UOs in them). 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)	Topics and Sub-topics
Unit –1: Introduction of Operating System	1a. Explain needs of Operating system 1b. Classify types of Operating System 1c. Describe OS services – User point of view and System point of view.	1.1 Fundamental Goals of Operating system 1.2 Overview of Operating Systems i. Multi programming ii. Time Sharing iii. Real Time iv. Multithreading v. Distributed 1.3 Operating System services 1.4 Case Study ii. Linux iii. Latest Windows Operating System 1.5 Generations of Operating System
Unit– 2: Process Management	2a. Define process model 2b. Describe process Life Cycle	Concepts of Process 2.1 Overview of the Process & threads 2.2 Process Life Cycle/ Process States 2.3 Process Control Block

	<p>2c. Compare different process scheduling algorithm.</p> <p>2d. Compare different schedulers</p> <p>2e. Describe Critical Section & mutual exclusion</p> <p>2f. Identify conditions for Deadlock</p> <p>2g. Solve Deadlock conditions using Resource allocation graph</p>	<p>Process Scheduling</p> <p>2.4 Scheduling Criteria</p> <p>2.5 Scheduling Algorithms</p> <ol style="list-style-type: none"> i. First Come First Serve ii. Shortest Job First iii. Round Robin <p>2.6 Overview of Schedulers</p> <p>2.7 Scheduling Queues</p> <p>2.8 Context Switch</p> <p>Process Synchronization</p> <p>2.9 Critical Section</p> <p>2.10 Mutual Exclusion</p> <p>Deadlock</p> <p>2.11 Conditions for Deadlock</p> <p>2.12 Resource allocation graph</p>
Unit– 3: Memory Management	<p>3a. Describe memory management</p> <p>3b. Differentiate Contiguous and Non- contiguous memory allocation</p> <p>3c. Differentiate primary and secondary memory</p> <p>3d. Apply different page replacement algorithms for memory allocation</p>	<p>Memory Management</p> <p>3.1 Logical and physical address map</p> <p>3.2 Swapping</p> <p>Memory Allocation</p> <p>3.3 Contiguous memory allocation</p> <ol style="list-style-type: none"> i. Fixed and variable partition ii. Internal and External Fragmentation and compaction iii. Memory relocation and protection mechanism iv. Allocation techniques – First Fit, Best Fit and Worst Fit <p>3.4 Non Contiguous Memory allocation</p> <ol style="list-style-type: none"> i. Overview of Paging ii. Address translation using basic method of paging iii. Overview of Segmentation iv. Page replacement algorithm – FIFO, LRU
Unit–4: File Management System	<p>4a. Apply file management concepts in Operating System</p> <p>4b. Explain Directory structure of Operating System</p> <p>4c. Describe physical disk structure.</p> <p>4d. Discuss Allocation methods of directory system.</p>	<p>Files System</p> <p>4.1 Files Attributes</p> <p>4.2 File Operations</p> <p>4.3 File Types</p> <p>Directory System</p> <p>4.4 Directory Structures</p> <p>4.5 Protection</p> <p>4.6 Allocation Methods – Contiguous, Linked</p> <p>Secondary Storage Structure</p> <p>4.7 Disk Structure</p>

	4e. Compare different disk scheduling algorithms.	4.8 Disk Scheduling Algorithm – SCAN, CSCAN
Unit-5: Linux Basics	5a. Test and Execute basic Linux commands 5b. Test and Execute shell commands in different shell scripts 5c. Develop shell scripts in 'Unix/Linux'	5.1 Linux Introduction 5.2 Basic architecture of Unix/ Linux 5.3 Introduction to shell and commands 5.4 Commands: pwd, cd, mkdir, rmdir, ls, cat, cp, rm, mv, wc, split, cmp, comm, diff, head, tail, grep, sort 5.5 Editing files with "vi", "vim", "gedit", "gcc" Linux Basic shell scripts 5.6 Read using command line argument 5.7 the logical operators 5.8 Evaluate expression using test and [..] 5.9 Branching : if, case 5.10 Basic of Looping : while, for

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	Introduction of Operating System	06	04	06	00	10
II	Process Management	12	06	10	04	20
III	Memory Management	10	06	06	04	16
IV	File Management System	06	04	06	00	10
V	Linux Basics	08	02	04	08	14
Total		42	22	32	16	70

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

Note: This specification table provides general guidelines to assist student 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 from 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 small reports (of 1 to 5 pages for each activity). For micro project report should be as per suggested format, for other activities students and teachers together can decide the format of the report. Students should also collect/record physical evidences such as photographs/videos of the activities for their (student's) portfolio which will be useful for their placement interviews:

- a) Undertake micro-projects in teams.
- b) Prepare charts to explain use/process of the identified topic.
- c) <https://boonsuen.com/process-scheduling-solver>, this website gives output for various process scheduling algorithms, students are expected to solve examples and crosscheck with output.
- d) An hour of problem solving for various process and disk scheduling algorithms may be organized and students are encouraged to participate
- e) Students are encouraged to register themselves in various MOOCs such as: Swayam, edx, Coursera, Udemy etc to further enhance their learning.
- f) List different versions of Linux and windows operating system
- g) Encourage students to form a coding club at institute level and can help the slow learners

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES

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/sub topics.
- b) Guide student(s) in undertaking micro-projects.
- c) Managing Learning Environment
- d) Diagnosing Essential Missed Learning concepts that will help for students.
- e) Guide Students to do Personalized learning so that students can understand the course material at his or her pace.
- f) Encourage students to do Group learning by sharing so that teaching can easily be enhanced.
- g) **'CI' in section No. 4** means different types of teaching methods that are to be employed by teachers to develop the outcomes.
- h) 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.
- i) With respect to **section No.10**, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- j) Guide students on how to address issues on environment and sustainability using the knowledge of this course

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 (group of 3 to 5). However, **in the fifth and sixth semesters**, 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 dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total work load on each student due to the micro-project should be about **16 (sixteen) student engagement hours** (i.e., about one hour per week) during the course. The students ought to submit micro-project by the end of the semester (so that they develop the industry-oriented COs).

A suggestive list of micro-projects is given here. This should relate highly with competency of the course and the COs. Similar micro-projects could be added by the concerned course teacher:

- a) A Comparative Analysis of Operating System: case study of Windows Operating and Mac OS. Comparing factors like booting process, user interface, handling system resources, device management, file management, security.
- b) A Comparative Analysis of Operating System: case study of Windows Operating and Linux based OS. Comparing factors like booting process, user interface, handling system resources, device management, file management, security.
- c) Comparing features of Windows 7, Windows 8 and Windows 10. Also show newly added functionality in each version.
- d) Case study on different Disk scheduling algorithms. Describe working of each algorithm.
- e) Case study on different Process scheduling algorithms. Describe working of each algorithm.
- f) Case Study on different page replacement algorithms. Describe working of each algorithm.
- g) Case study on fragmentation in operating system.
- h) Case Study in shell script on how to ask for user input until valid input is given. Also illustrates how to check if no characters were input, check if input exceeds 30 characters in length, check if input lacks both first and last name, exit if too many wrong answers.
- i) Animate the Disk scheduling algorithms.
- j) Animate the Process scheduling algorithms.
- k) Animate the Page replacement algorithms.
- l) Case study any one scheduling algorithms in the cloud.
- m) Case study on any one cloud operating system.
- n) Case study on any one real time operating system.
- o) Case study on any one mobile operating system.
- p) Case study on any one server operating system.
- q) Case study on any one distributed operating system.
- r) Case study on any one network operating system.
- s) Case study on any one time sharing or multitasking operating system.
- t) Case study on any one batch processing system.
- u) Case study on shell commands in detail.
- v) Case study on advanced shell command.
- w) Case study on any operating system used in smart gadgets.

13. SUGGESTED LEARNING RESOURCES

Sr. No.	Title of Book	Author	Publication with place, year and ISBN
1	Operating System Concepts, 9 th Edition	Abraham Silberschatz, Peter B Galvin, Gerg Gagne	WILEY, 2016, ISBN - 978-8126554270
2	Unix Concepts And Application, 4 th Ed	Sumitabha Das	MGH, ISBN – 0-07-063546-3
3	Modern Operating System 3 rd Ed	Andrew Tanenbaum, Herbert Bos	2015,Pearson, ISBN – 9780133591620
4	Operating System, 2 nd Ed	Milan Milenkovic	2014, MGH, ISBN-13: 978-0-07-463272-7
5	Linux –Application and Administration	Ashok Kumar Harnal	2009, TMH, ISBN-13: 978-0070680104

14. SUGGESTED LEARNING WEBSITES

- a) <https://boonsuen.com/process-scheduling-solver>
- b) <http://cpuburst.com/ganttcharts.html>
- c) <https://codepen.io/faso/pen/zqWGQW>
- d) <https://www.tutorialspoint.com>
- e) www.w3schools.com
- f) <https://nptel.ac.in/courses/106106144>
- g) <https://nptel.ac.in/courses/106105214>
- h) <https://nptel.ac.in/courses/106102132>

15. PO-COMPETENCY-CO MAPPING

Semester II	Basics of Operating System (Course Code:)									
	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	PSO 1	PSO 2	PSO 3 (If needed)
Competency Develop shell scripts in 'Unix/Linux' using Operating System Concepts.										
Course Outcomes										
CO a) Differentiate Operating Systems based on features	2	-	-	-	-	-	1			
CO b) Interpret various aspects of Process management	2	2	2	-	-	-	1			
CO c) Interpret various memory management techniques	2	2	2	-	-	-	1			
CO d) Interpret various File management techniques.	2	2	2	-	-	-	1			
CO e) Execute basic Linux commands and Shell scripts.	3	2	2	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	Contact No.	Email
1	Shri P. P. Kotak Principal	Government Polytechnic, Rajkot	9825469617	Kotakp2003@yahoo.com
2	Smt. M. P. Mehta HOD	Government Polytechnic, Gandhinagar	9879578273	manishamehtain@gmail.com
3	Smt. Avani S Galathiya	R C Technical Institute, Ahmedabad	9904126016	asgalathiya@gmail.com
4	Smt. Soniya S Dadhania	R C Technical Institute, Ahmedabad	9974006746	soniyasdadhania@gmail.com

GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)**Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)**

Semester – III

Course Title: Data Structures and Algorithms

(Course Code: 4330704)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering	Third

1. RATIONALE

Development of application systems and software that use underlying architecture of machines efficiently and effectively requires the ability to use and manipulate various types of Data Structures and other constructs. This being a fundamental ability which is language neutral, yet requires use of a language for its implementation. This is a basic course which goes along with other programming courses to develop an integrated ability to efficient software development, hence this course is very important for computer engineers.

2. COMPETENCY

The course content should be taught and implemented with the aim to develop various types of related skills leading to the achievement of the following competency

- **Implement various types of algorithms using Data Structures.**

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:

The practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry-oriented COs associated with the above-mentioned competency:

- Perform basic operations on arrays and strings.
- Demonstrate algorithms to insert and delete elements from the stack and queue data structure.
- Apply basic operations on the linked list data structure.
- Illustrate algorithms to insert, delete and searching a node in tree.
- Apply different sorting and searching algorithms to the small data sets.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (CI+T/2+P/2)	Examination Scheme				Total Marks
L	T	P		Theory Marks		Practical Marks		
			C	CA	ESE	CA	ESE	
3	0	4	5	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: CI-ClassRoom Instructions; 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) that are the sub-components of the COs. Some of the PrOs marked '**' are compulsory, as they are crucial for that particular CO. These PrOs need to be attained at least at the 'Precision Level' of Dave's Taxonomy related to 'Psychomotor Domain'.

S.No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1	Define various terms such as algorithm, various approaches to design an algorithm, time complexity, space complexity, best case, average case and worst-case time complexity etc.	I	02
2	Implement array using row major order and column major order	I	02
3	Implement Sequential search algorithms.	I	02
4	Implement Binary search algorithms.	I	02
5	Implement various string algorithms.	II	02
6	Implement push and pop algorithms of stack using array	III	02
7	Implement recursive functions.	III	02
8	Implement insert algorithms of queue using array.	III	02
9	Implement delete algorithms of queue using array.	III	02
10	Implement simple structure programs using pointers.	IV	02
11	Implement insertion of node in the beginning of the list in singly linked list.	IV	02
12	Implement insertion of node at the end of list in singly linked list.	IV	02
13	Implement insertion of node in sorted linked list.	IV	02
14	Implement insertion of node at any position in linked list.	IV	02

15	Implement counting no of node algorithm in singly linked list.	IV	02
16	Implement searching of a node algorithm in singly linked list.	IV	02
17	Implement delete a node algorithm in singly linked list.	IV	02
18	Implement construction of binary search tree.	V	02
19	Implement inorder, preorder and postorder traversal methods in binary search tree.	V	04
20	Implement searching algorithm in binary search tree.	V	02
21	Implement Bubble sort algorithm.	VI	02
22	Implement Selection sort algorithm.	VI	02
23	Implement Quick Sort algorithm.	VI	02
24	Implement Insertion sort algorithm.	VI	02
25	Implement Shell sort algorithm.	VI	02
26	Implement Merge Sort algorithm.	VI	02
27	Solve hash table example using division method, method square method, folding method. (paper work only)	VI	02
	Total		56

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	Correctness of program	30
2	Readability and documentation of the program/Quality of input and output displayed (messaging and formatting)	10
3	Code efficiency	20
4	Debugging ability	20
5	Program execution/answer to sample questions	20
Total		100

6. MAJOR EQUIPMENT/ INSTRUMENTS AND SOFTWARE REQUIRED

These major equipment/instruments and Software required to develop PrOs are given below with broad specifications to facilitate procurement of them by the administrators/management of the institutes. This will ensure conduction of practical in all

institutions across the state in proper way so that the desired skills are developed in students.

S. No.	Equipment Name with Broad Specifications	PrO. No.
1	Computer with latest configuration with windows or unix os	All
2	C/C++/Python Compiler	All

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) Follow safety practices.
- b) Practice good housekeeping.
- c) Demonstrate working as a leader/a team member.
- d) Maintain tools and equipment
- e) 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 1st year
- ii. 'Organization Level' in 2nd year.
- iii. 'Characterization Level' in 3rd year.

8. UNDERPINNING THEORY

The major Underpinning Theory is formulated as given below and only higher level UOs of *Revised Bloom's taxonomy* are mentioned for development of the COs and competency in the students by the teachers. (Higher level UOs automatically includes lower level UOs in them). 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)	Topics and Sub-topics
Unit – I Basic Concepts of Data Structures	1a. Represent the data in relevant memory	1.1 Data Structure Basic Concepts 1.2 Types of data structures
	1b. Differentiate primitive and non-primitive data structures	1.3 Primitive and non-primitive data structures
	1c. List key features of an algorithm	1.4 Introduction to Algorithms 1.5 Key features of an algorithm
	1d. Define time complexity and space complexity	1.6 Analysis Terms (for the definitions purpose only): a. Time Complexity b. Space Complexity c. Asymptotic Notations: Big Oh Notation, Big Omega Notation, Big Theta Notation, Best case Time Complexity, Average case Time Complexity, Worst case Time Complexity
	1e. Implement programs to represent array in row major and column major order	1.7 Array: i. Row Major Arrays ii. Column Major Arrays 1.8 Overview of different array operations.
	1f. Design and Implement search algorithms	1.9 Searching an element into an array: i. Linear Search ii. Binary Search
Unit– II Strings	2a. Describe representation of a strings	2.1 String representation: Reading and Writing Strings
	2b. Develop algorithms to implement various operations on string	2.2 String operations : Finding length of a string, Converting Characters of a string into upper case and lower case, Concatenation of two strings to form a new string, Appending, Reversing a string, Copying a string, Comparing strings, Insertion, Substring, Deletion

Unit– III Stack and Queues	3a. Define linear and non-linear data structures 3b. Implement algorithms to push an element into stack, pop an element from the stack.	3.1 Linear and Non-Linear Data Structures 3.2 Stack : Array representation of Stack, PUSH- POP Operations on Stack, Implementation of Stack, Application of Stack, Infix, Prefix and Postfix Forms of Expressions, Recursive Functions (Factorial, greatest common divisor, Fibonacci series)
	3c.. Implement different operations on a Queue	3.3 Queue: Array representation of Queue Operations on a Queue (Add an element, delete an element, display all elements of a queue), Implementation of a Queue, Limitation of a Single Queue
	3c. Differentiate circular and simple queue	3.4 Concepts of Circular Queue 3.5 Applications of a queue 3.6 Differentiate circular queue and simple queue
Unit– IV Linked List	4a. Define linked list	4.1 Pointers Revision 4.2 Revision of Structure 4.3 Revision of structure using pointers 4.4 Dynamic Memory Allocation 4.5 Linked list Presentation 4.6 Types of Linked List
	4b. Implement algorithms to perform various operations on a singly linked list	4.7 Basic operations on singly linked list : Insertion of a new node in the beginning of the list, at the end of the list, after a given node, before a given node, in sorted linked list, Deleting the first and last node from a linked list, Searching a node in Linked List, Count the number of nodes in linked list
	4c. Distinguish circular linked list and singly linked list	4.8 Concepts of circular linked list 4.9 Difference between circular linked list and singly linked list
	4d. Distinguish Doubly linked list and singly linked list	4.10 Doubly linked list: Representation 4.11 Difference between Doubly linked list and singly linked list
	4e. List applications of the linked list	4.12 Applications of the linked list
Unit– V Trees	5a. Define non-linear data structure	5.1 Non-linear data structures: Tree, Graph
	5b. Define basic terms of a tree data structure 5c. Convert general tree to binary tree	5.2 Basic Terms: General Tree, Forest, Binary trees, level number, degree, in-degree and out-degree, root node, leaf node, directed edge, path, depth Binary tree: Complete Binary Tree, Strict Binary Tree, Conversion of General Tree to Binary Tree

	5d. Implement basic operations on a binary tree 5e. Demonstrate the traversal of a binary tree	5.3 Binary Search Tree : Insertion of a node in binary tree, Deletion of a node in binary tree, Searching a node in binary tree 5.4 Binary Tree Traversal : Inorder, Preorder, Postorder
	5f. List applications of tree	5.5 Applications of binary tree
Unit– VI Sorting and Hashing	6a. Arrange data in ascending and descending orders using appropriate sorting algorithm	6.1 Sorting Methods : a. Bubble Sort, b. Selection Sort, c. Quick Sort, d. Insertion Sort, e. Merge Sort, f. Radix Sort
	6b. Apply various hashing techniques	6.2 Hashing Concepts 6.3 Hash functions: Division Method, Middle Square Method, Folding Method

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	Basic Concepts of Data Structures	4	4	3	0	7
II	Strings	4	2	2	3	7
III	Stack and Queues	8	2	6	6	14
IV	Linked List	9	4	8	2	14
V	Trees	7	4	4	6	14
VI	Sorting and Hashing	10	2	6	6	14
Total		42	18	29	23	70

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

Note: This specification table provides general guidelines to assist student 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 from 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 small reports (of 1 to 5 pages for each activity). For micro project report should be as per suggested format, for other activities students and teachers together can decide the format of the report. Students should also collect/record physical evidences such as photographs/videos of the activities for their (student's) portfolio which will be useful for their placement interviews:

- a) Students are encouraged to learn Visual Language programming like scratch, snap etc.
- b) Undertake micro-projects in teams.
- c) Prepare charts to explain use/process of the identified topic.
- d) <https://www.codechef.com/> , in this website very elementary programs are available, students are expected to solve those programs
- e) <https://code.org/>, an hour of code may be organized and students are encouraged to participate
- f) Students are encouraged to register themselves in various MOOCs such as: Swayam, edx, Coursera, Udemy etc to further enhance their learning.
- g) List the applications which are developed using C
- h) Encourage students to participate in different coding competitions like hackathon, online competitions on codechef etc.
- i) Encourage students to form a coding club at institute level and can help the slow learners

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/sub topics.
- b) Guide student(s) in undertaking micro-projects.
- c) Managing Learning Environment
- d) Diagnosing Essential Missed Learning concepts that will help for students.
- e) Guide Students to do Personalized learning so that students can understand the course material at his or her pace.
- f) Encourage students to do Group learning by sharing so that teaching can easily be enhanced.
- g) **'CI' in section No. 4** means different types of teaching methods that are to be employed by teachers to develop the outcomes.
- h) 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.
- i) With respect to **section No.11**, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- j) Guide students on how to address issues on environment and sustainability using the knowledge of this course

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 (group of 3 to 5). However, **in the fifth and sixth semesters**, 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 dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total work load on each student due to the micro-project should be about **16 (sixteen) student engagement hours** (i.e., about one hour per week) during the course. The students ought to submit micro-project by the end of the semester (so that they develop the industry-oriented COs).

A suggestive list of micro-projects is given here. This should relate highly with competency of the course and the COs. Similar micro-projects could be added by the concerned course teacher:

- a) Develop a C/C++/Python Program that evaluate the given arithmetic expression using stack.
- b) Develop a C/C++/Python Program that maintain a queue of persons. In this queue user can add a person, delete a person and search a person.
- c) Develop a C/C++/Python Program that perform banking operations like withdraw cash, deposit cash and mini statement using appropriate data structure.
- d) Develop a C/C++/Python Program for process management algorithm by using appropriate data structure.
- e) Develop a C/C++/Python Program for print spooler using appropriate Data structure.
- f) Develop a C/C++/Python Program for Telephone Directory system. In this user can adding, searching, modifying, listing, and deleting records through the use of appropriate data structure.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	An Introduction to Data Structures with Applications	Jean-Paul Tremblay & Paul G. Sorenson	Tata McGraw Hill
2	Data and File Structures using C	Thareja, Reema	Oxford University Press New
3	Data Structures	Chitra, A Rajan, P T	Tata McGraw Hill, New delhi,
4	Data Structures using C & C++	Tenen Baum	Prentice-Hall International
5	Classic Data Structures	Samanta, D.	PHI Learning, New Delhi
6	Data Structures using C	ISR D Group	McGraw Hill, New Delhi
7	Data Structures: A Pseudo-code approach with C	Gilberg & Forouzan	Thomson Learning

14. SUGGESTED LEARNING WEBSITES

- a) <https://www.programiz.com/dsa>
- b) <https://nptel.ac.in/courses/106102064> (Introduction to data structures and algorithms, IIT Delhi)
- c) <https://nptel.ac.in/courses/106106133> (Programming, Data structures and Algorithms, IIT Madras)
- d) <https://www.codecademy.com/learn/linear-data-structures>
- e) <https://www.udacity.com/course/data-structures-and-algorithms-in-python--ud513>
- f) <https://www.edx.org/learn/data-structures>

15. PO-COMPETENCY-CO MAPPING

Semester III	Data Structures and Algorithms (Course Code:4330704)									
	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	PSO 1	PSO 2	PSO 3 (If needed)
Competency Implement various types of algorithms using Data Structures										
Course Outcomes										
a) Perform basic operations on arrays and strings	2	1	2	-	-	-	-			
b) Demonstrate algorithms to insert and delete elements from the stack and queue data structure	3	1	2	-	-	-	-			
c) Apply basic operations on the linked list data structure.	3	-	2	-	-	-	-			
d) Illustrate algorithms to insert, delete and searching a node in tree.	2	1	1	-	-	-	-			
e) Apply different sorting and searching algorithms to the small data sets.	3	2	2	-	-	-	-			

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	Contact No.	Email
1	Shri P. P. Kotak Principal	Government Polytechnic Rajkot	8200601748	kotakp2003@yahoo.com
2	Smt. M. P. Mehta Head of the Department	Government Polytechnic, Gandhinagar	9879578273	manishamehtain@gmail.com
3	Smt. Rashmika K. Vaghela Lecturer	K.D.Polytechnic, Patan	982446419 9	rashmivaghela.rv@gmail.com
4	Ms. Kumundrini B Prajapati	Government Polytechnic, Gandhinagar	9974543026	kumundrini13187@gmail.com

GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)**Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)**

Semester-III

Course Title: Responsive Web Page Design

(Course Code: 4330705)

Diploma programme in which this course is offered	Semester in which offered
Computer Engineering	Third

1. RATIONALE

In this technological era, different devices being used to view the online content and websites. While designing webpage, it is required to maintain visual consistency spanning all the platforms. Bootstrap is a free and open-source web development framework. Bootstrap is a sleek, intuitive, and powerful, mobile first front-end framework for faster and easier web development. It's designed to ease the web development process of responsive, mobile-first websites Bootstrap provide in build classes and files to make more fun in designing and give new look more and design. It is used to convert html page design into RWD (Responsive Web Design) which is easy to learn and have many designs related functionalities. It uses HTML, jQuery and JavaScript to provide cross platform web design solution.

Course Objectives:

- Bootstrap CSS Contains global CSS classes for typography, tables, grids, forms, buttons, images etc.
- High built-in Support for layout, grids, fluid grids, and responsive and mobile first web designs.
- Bootstrap Components which contain various reusable components including Icons, Dropdowns, Navbars, Breadcrumbs, Popovers, Alerts, and many more.
- Various jQuery and JavaScript Plug-ins which added one by one in webpages to enhance user experience.

This course will give basic knowledge and skills for client-side web UI frameworks, in particular Bootstrap. You will learn about grids and responsive design, Bootstrap CSS and JavaScript components. Thus this course aims to help the developers to build the websites faster without worrying about the basic commands and functions.

2. COMPETENCY

The purpose of this course is to help the student to attain the following industry identified competency through Bootstrap Framework experiences:

- **Use Bootstrap Framework to build aesthetic responsive web pages that work uniformly in different devices (desktops, tablets and mobile) and operative platforms.**

3. COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with the identified competency are to be developed in the student for the achievement of the following COs:

- Prepare environment for Bootstrap framework for first time use.
- Build different web pages layouts adhering to all platforms and sizes.
- Apply reusable bootstrap components to design effective user-friendly web pages.
- Develop interactive features rich web pages using Bootstrap jQuery plug-ins.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P/2)	Examination Scheme				
L	T	P		Theory Marks		Practical Marks		Total Marks
			C	CA	ESE	CA	ESE	
0	0	2	1	0	0	25	25	50

(*): For this practical only course, 25 marks under the practical CA has two components i.e. the assessment of micro-project, which will be done out of 10 marks and the remaining 15 marks are for the assessment of practical. This is designed to facilitate attainment of COs holistically, as there is no theory ESE.

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. Some of the PrOs marked '**' are compulsory, as they are crucial for that particular CO at the 'Precision Level' of Dave's Taxonomy related to 'Psychomotor Domain'.

S. No	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
1	Install Bootstrap framework and understand various tags, attributes of HTML and other necessary files to make responsive web page.	I	02*
2	Design web page that shows department name, college name at center of web page by using bootstrap framework and without using bootstrap framework.	I	01*
3	Display student information content on responsive web page by using container and container-fluid classes.	II	01*
4	Use offset column, reordering column and Nesting column to create responsive web page for given format.	II	01
5	Create responsive web page of your class time table by using	II	01*

S. No	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
	bootstrap grid system.		
6	Shows at least five to six co-curricular/Extra activities of student that includes multiple images and short description of each activity on responsive web page using responsive images with different styles and responsive tables with 3 to 4 different styles such as hover state when mouse over, different color of each row, table with striped row etc.	II	02*
7	Use bootstrap typography to create responsive web page on given blog topic.	II	01*
8	Design responsive web page for student registration form using bootstrap form layout, form control, bootstrap buttons.	II	01*
9	Create various types of menus using bootstrap menu components such as right aligned dropdown menu, drop up menu, adding headers of each item etc. and glyphicons components.	III	01
10	Design responsive web page that shows odd (sem1, sem3, sem5) and even (sem2, sem4, sem6) semester consider as menu, courses of each semester as submenu using button groups and button toolbar component.	III	01*
11	Use different bootstrap input groups components to create responsive webpage for job application or any other kind of application.	III	01*
12	Use Navigation tabs/Pills to create responsive web page for summarize of all individual units of any one course.	III	01*
13	Use Nav bar component to create responsive fixed to top menu design includes logo, menu, drop down menu, form input elements such as sign-up button, search mechanism etc. And fixed to bottom menu design contains footer information.	III	01*
14	Create responsive web page of education website using bootstrap breadcrumb, pagination, labels/badge, Jumbotron/page header, thumbnail components.	III	01
15	Following tasks to be performed using bootstrap progress bars component. A) Creating progress bars B) Adding label to progress bars C) Creating multi-color, stripped and animated progress bars D) Changing value of progress bar dynamically using JavaScript	III	01
16	Use media, rounded media, Nested media object to create responsive web page for all family members in hierarchical order.	III	01*

S. No	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
17	Create responsive webpage of various service provider information using bootstrap panel component, List group component, various alert messages and show message after bootstrap alert has been closed.	III	01
18	Design a smooth page transition between homepage, about and contact us page using bootstrap transition plugin.	IV	01*
19	Design a webpage with different modal dialog for "Save record confirmation", "Delete record confirmation" using model dialog plugin of bootstrap.	IV	01*
20	Design news story page to demonstrate usage of Scrollspy for multiple section, Tooltip for different photos, Collapsible and popover plugins of bootstrap.	IV	02*
21	Design animated photo gallery page using Carousel bootstrap plugin with minimum seven photos.	IV	01*
22	Develop advanced responsive website with minimum Five to seven pages on any of following domains (Tourism, Hospitality, Airline reservation, Medical, E commerce etc.)	IV	04*
	Minimum 22 Practical Exercises		28 Hrs.

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	Use of creative and innovative approach to implement practical.	15
2	Use bootstrap framework to build efficient responsive websites	15
3	Effective utilization of different components and plugins to design web pages more user friendly.	20
4	Verify practical implementation for desired output.	30
5	Interpret the result and conclude	20
	Total	100

6. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

These major equipments 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	Computer system with operating system and browser that supports JavaScript.	All
2	HTML IDEs and Code Editors Open-source tools like Visual Studio Code, Notepad++	All
3	Open-source jQuery and Bootstrap Framework	All

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 course competency.

- a) Work as a leader/a team member.
- b) Follow ethical practices.
- c) Practice environmentally friendly methods and processes.
- d) Follow safety precautions.

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 1st year
- ii. 'Organization Level' in 2nd year.
- iii. 'Characterization Level' in 3rd year.

8. UNDERPINNING THEORY

The major underpinning theory is given below based on the higher level UOs of Revised Bloom's taxonomy that are formulated for development of the COs and competency. If required, more such 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 different levels)	Topics and Sub-topics
Unit – I: Introduction to jQuery and Bootstrap Framework	1a. Understand basics of jQuery 1b. Discuss importance of Bootstrap framework. 1c. Setting up environment to use Bootstrap framework. 1d. Design first basic responsive page based on Mobile First Strategy.	1.1 Introduction to jQuery 1.2 Importance of jQuery, jQuery syntax, jQuery selectors, jQuery Events 1.3 Bootstrap Framework 1.4 Why Bootstrap? 1.5 History of Bootstrap 1.6 Advantages of Bootstrap Framework 1.7 Responsive web page 1.8 How to remove Responsiveness 1.9 Major Features of Bootstrap

Unit	Unit Outcomes (UOs) (4 to 6 UOs at different levels)	Topics and Sub-topics
		1.10 Mobile-First Strategy 1.11 Setting up Environment 1.12 How to apply Bootstrap to Applications
Unit– II: Bootstrap Grid	2a. Describe the mechanism of bootstrap grid structure. 2b. Design the web pages using bootstrap grid and structure the content in rows and columns format. 2c. Apply different form layout, buttons and text formatting while designing web pages. 2d. Use caret class in web page design.	2.1 Bootstrap Grid 2.2 Apply Bootstrap Grid, Container, Offset Column, Reordering Columns 2.3 Advantages of Bootstrap Grid 2.4 Display responsive Images 2.5 Bootstrap Typography, use Typography 2.6 Bootstrap Tables, Form Layout, Button 2.7 Display images in different styles like Circle shape etc. 2.8 Carets Classes, hide or show the text in Bootstrap
Unit– III: Bootstrap Components	3a. Discuss different bootstrap components. 3b. Use different bootstrap components like Glyphicons, Dropdown Menu Button Groups and Button Toolbar to effective page design. 3c. Design webpages using bootstrap card, Navigation Pills and Tab Components. 3d. Apply various input, navigation, progress and alerts related bootstrap components to make functionality rich web page.	3.1 Bootstrap Components, Advantages of Bootstrap Components, The different types of Bootstrap Components 3.2 Glyphicons Component, Use Glyphicons Component 3.3 Bootstrap Dropdown Menu Component 3.4 Bootstrap Card Component 3.5 Button Groups and Button Toolbar, Use Button Groups and Button Toolbar 3.6 Different Input Groups Components 3.7 Navigation Pills & Tabs Components, Use Navigation Pills and Tabs Components 3.8 Navbar Component, build a Responsive Navbar 3.9 Add Forms and other controls to Navbar, Fix the position of navbar 3.10 Breadcrumb Component 3.11 Pagination Component, apply Pagination in Application 3.12 Labels/Badge Components, Jumbotron/Page Header Components, Thumbnail Component 3.13 Alerts & Dismissible Alerts, How to Create Progress Bar, Media Objects Component 3.14 Bootstrap List Group Component,

Unit	Unit Outcomes (UOs) (4 to 6 UOs at different levels)	Topics and Sub-topics
		Bootstrap Panel Component
Unit-IV: Bootstrap Plug-Ins	4a. Understand importance of various bootstrap plug-ins 4b. Use different model dialog box give different popup windows, messages on dialog mode. 4c. Apply Scrollspy, Tooltip, popover plug-in, alert and button plug-ins while designing page to make it more interactive. 4d. Design home screen more attractive by using Carousel plug-in.	4.1 Why bootstrap plug-ins 4.2 Use of bootstrap plug-ins 4.3 Transition plug-in 4.4 Modal dialog box 4.5 The different properties, methods and events of model dialog box 4.6 Scrollspy plug-in, tab plug-in, use of tab plug-in, Drop Down plug-in 4.7 Tooltip plug-in, Use of Button plug-in, Methods and events of tooltip plug-in 4.8 popover plug-in, alert and button plug-ins 4.9 Collapse plug-in, different types of properties, methods and events of collapse plug-in 4.10 Carousel plug-in, affix plug-in

9. SUGGESTED SPECIFICATION TABLE FOR QUESTIONPAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Introduction to jQuery and Bootstrap Framework	3				
II	Bootstrap Grid	7				
III	Bootstrap Components	9				
IV	Bootstrap Plug-Ins	9				
Total		28				

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 to 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 slightly vary from above table.

10. SUGGESTED STUDENT ACTIVITIES

Other than the 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 perform following activities in group and prepare reports of about 5 pages for each activity. They should also collect/record physical evidences for their (student's) portfolio which may be useful for their placement interviews:

- a) Identify tools used for web page development and present its features.
- b) Undertake course “HTML, CSS, and Javascript for Web Developers” available on courseraonline platform. (<https://www.coursera.org/learn/html-css-javascript-for-web-developers>).
- c) Undertake course “Front-End Web UI Frameworks and Tools: Bootstrap” available on coursera online platform. (<https://www.coursera.org/learn/bootstrap-4>) or any other such site.
- d) Undertake course “Bootstrap 5 From scratch with 13 Projects | 4 Sass Projects” available on Udemy online platform. (<https://www.udemy.com/course/learn-by-building-bootstrap-5-from-scratch-with-8-projects>) or any other such site.

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/sub topics.
- b) Guide student(s) in undertaking micro-projects.
- c) **‘P’ in section No. 4** means different types of instructions 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.10**, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- f) Guide students for open-source HTML editors, components and plugins.
- g) Motivate students to visit as many websites as they can to increase the design knowledge and creativity.

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 (group of 3 to 5). However, **in the fifth and sixth semesters**, 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 dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The duration of the microproject should be about **12-14 (twelve to fourteen) student engagement hours** during the course. The students ought to submit 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:

- a) Design mobile first responsive departmental website

- b) Develop any domain specific responsive website (Food, Automobiles, Educational, Business etc.)
- c) Develop a responsive website for NGO or society working to save environment to make awareness towards environment to save soil, water, trees and other major resources of our environment.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	Bootstrap	Jake Spurlock	O'Reilly Media, Inc. ISBN: 9781449343910
2	Bootstrap Reference Guide	Jacob Lett	Bootstrap Creative. ISBN:1732205833
3	Bootstrap 4 Cookbook	AjdinImsirovic	Packt Publishing Limited. ISBN:178588929X
4	Mastering Bootstrap 4 - Second Edition	Benjamin Jakobus	Packt Publishing Limited. ISBN:1788834909
5	Bootstrap 4 By Example	Silvio Moreto	Packt Publishing Limited ISBN:1785288873
6	HTML & CSS: The Complete Reference	Thomas Powell	Tata McGraw Hills, New Delhi, ISBN: 9780070701946

14. SOFTWARE/LEARNING WEBSITES

- <https://www.geeksforgeeks.org/bootstrap/>
- <https://www.getbootstrap.com>
- <https://www.w3schools.com/bootstrap/>
- <https://www.websitesetup.org/bootstrap-tutorial-for-beginners/>
- <https://www.tutorialrepublic.com/twitter-bootstrap-tutorial/>
- <https://www.tutorialspoint.com/bootstrap/index.htm>
- [12 Awesome CSS3 Features That You Can Finally Start Using - Tutorialzine](#)
- [Twitter Bootstrap Tutorial for Beginners \(ieatcss.com\)](#)
- [CSS3 - Tutorial \(tutorialspoint.com\)](#)
- [CSS3 Tutorial - An Ultimate Guide for Beginners \(tutorialrepublic.com\)](#)

15. PO-COMPETENCY-CO MAPPING

Semester III	Responsive Web Page Design (Course Code: 4330705)						
	POs						
Competency & Course Outcomes	PO 1 Basic & Discipline specific	PO 2 Problem Analysis	PO 3 Design/development	PO 4 Engineering Tools, Experiment-	PO 5 Engineering practices for society	PO 6 Project Management	PO 7 Life-long

	knowledge		t of solutions	tation&Testing	sustainability & environment		learning
Competency	Use Advanced CSS and Bootstrap Framework to build aesthetic responsive web pages that work uniformly in different devices (desktops, tablets and mobile) and operative systems.						
Course Outcomes							
CO a) Prepare environment for Bootstrap framework for first time use.	2	-	-	1	-	-	-
CO b) Build different webpages layouts adhering to all platforms and sizes.	2	1	1	1	-	-	1
CO c) Apply reusable bootstrap components to design effective user-friendly web pages.	2	2	2	2	1	-	1
CO d) Develop interactive features rich web pages using Bootstrap jQuery plug-ins.	2	1	1	2	1	1	1

Legend: '3' for high, '2' for medium, '1' for low and '-' for no correlation of each CO with PO.

16. COURSE CURRICULUM DEVELOPMENT COMMITTEE

GTU Resource Persons

Sr. No.	Name and Designation	Institute	Contact No.	Email
1	Shri P P Kotak, Principal	GP Rajkot	8200601748	Kotakp2003@yahoo.com
2	Jiger P. Acharya	GP Ahmedabad	9429462026	jigeracharya@gmail.com
3	Alpeshkumar R. Thaker	GP Ahmedabad	9879709675	alpeshrthaker@gmail.com
4	Umang D. Shah	GP Ahmedabad	9427686364	umang.shah111gp@gmail.com