Veer Narmad South Gujarat University, Surat
Masters of Science in Computer Application
(M.Sc.(Computer Application))
Under the Faculty of
Computer Science, Application and Information Technology

Name of Program:	Masters of Science in Computer Application (M.Sc.(Computer Application)) (Second Year (Sem-III and Sem-IV)
Abbreviation:	M.Sc.(Computer Application) : Post Graduate Program
Duration:	Two years
Program Outcome:	**PO1:** Ability to analyze a problem, identify and define the Computing requirements appropriate to its solution. **PO2:** Foster creativity and innovation in students, encouraging them to develop novel solutions to real-world problem. Enhancing the problem solving, logical, reasoning and analysis capabilities of a problem and integrate the ability with the coding using specific computer programming languages. **PO3:** Develop student's ability to analyze, evaluate and solve complex problems in the field of computer applications, using critical thinking and problem-solving skills. **PO4:** Design, implement and evaluate a computer-based system, processing, component or program to meet desired goal with the help of various programming languages, application software, packages, tools, databases, data analytics and representation of data on various platforms. **PO5:** An ability to apply design and development principles in construction of software systems of varying complexity using various algorithmic principles, modeling, coding and design of computer-based systems. **PO6:** Prepare the aspiring students to become computer professionals in applied areas who can work in corporate/software industry at entry to advanced level as well as independent developers. Overall, the program outcomes aim to produce graduates who are: (a) competent in computer application, development and design. (b) Adapt to changing technology and industry trends. (c) Can make significant contributions to the software applications coding, designing, database managements, testing, deployments and ready to adapt any upcoming technologies.
Program Specific Outcome:	**PSO1:** Provide students with a strong understanding of programming languages, algorithms and data structures, necessary for software development. (PO-1 & 2) **PSO2:** Equip with skills in database management, data modeling and data analysis to develop efficient and effective data-driven software solutions. **PSO3:** Develop expertise in field of software engineering, covering software design, testing and maintenance to ensure the production of high-quality software products. **PSO4:** Develop knowledge and skills in web development, including web design, client-side and server-side programming and web security.

PSO5: Develop students to capabilities for self-learning, skill development through self-practicing and problem solving abilities.

PSO6: Develop students to address and work on the real-world problems as an individual and as part of team. Understand the business problems and ability to work on their solutions by applying various software technologies.

PSO7: To enhance development skills at various level including problem analysis, data analysis, logical and critical analysis of the problems and implementing the solutions by imparting various recent and upcoming technologies.

PSO8: Enhance the passion among the students for updating knowledge, innovative ideas, upskilling and implementing the knowledge in applied areas and research areas by understanding the real world problems, addressing the real world problems and their possible solutions that lead to build a successful Professional career.

PO and PSO mapping:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
PO1								
PO2								
PO3								
PO4								
PO5								
PO6								

Medium of Instruction:

English

Program Structure:

Semester-wise Breakup of the course is given as follows :

Course Fees:

a. Admission Processing Fees: Rs. 500/- (Admission process fees for Semester-1)
b. Semester-1 Tuition Fees : Rs. 19,250/-
c. Semester-2 Tuition Fees : Rs. 19,250/-
d. Laboratory Utilization Fees: Rs.1500/- per semester.
[Value addition course (2-credit certification) in semester-3 and semester-4 are mandatory for students. Both these courses will be as per the SOP of certificate courses and their fees will be paid separately by the students as per the university certificate courses norms. It is not part of the Tuition Fees.]

Internal/External Assessment:

- **Internal assessment:** Continuous assessment process (Assignment, Attendance, Class Test, Viva)
- **External Assessment:**
- **Theory Exams:** University exam of 70 Marks (3 Hours duration) will be conducted for course-901 to course-905.
- **Practical Exam (Course Code: 906) :** At end of semester-III, combined Practical exam carrying 4 Credit (140 marks) will be conducted based on course codes : 901, 902 and 905.
  Duration of Practical Exams: 5 hours.
- **Project Exam (Course Code: 907) :** Project exam will evaluate performance of the student based on the project developed during the semester based on course - 903 and course-904.
- **Project Evaluation:** Project presentation and viva-voce. The evaluation of the project will be carried out by the examiner panel of minimum three examiners consist of one local examiner, one expert from the Information technology/software Industry or from any other university except V.N.S.G.U. and one external examiner from any other institute affiliated to University having minimum eleven years of teaching experience at graduation level.

Veer Narmad South Gujarat University, Surat

Program Structure: M.Sc.(Computer Application) (SEM – 3 and SEM – 4)

(w.e.f. Academic Year June, 2024-2025)

Masters in Computer Application (S.Y.M.Sc. (C.A.)) – Two Year Post Graduate Program

Program Structure	Semester-wise break up for the courses :

SEMESTER – 3

Course Code	Course Title	Course Category	Level of Course	Course Credits	Teaching Hours/week	
				Th.+Pra.	Theory	Practical/ Fieldwork /Project/ Internship
900	Value Addition Course [2-credit university approved certificate course]	Value Addition Course*	600-699 Advance level Technical	2	2	0
901	Data Intelligence and Visualization	Major Course	600-699 Advance level Technical	3	2	2 (Supervised Mode)
902	Robotic Process Automation (RPA)	Major Course	600-699 Advance level Technical	3	2	2 (Supervised Mode)
903-01 Or	Server side scalable Application	Major Course Elective-1	600-699 Advance level Technical	4	2	4 (Un-supervised mode)
903-02	Android based Sensors handling	Major Course Elective-II	600-699 Advance level Technical	4	2	4 (Un-supervised mode)
904-01 Or	Application Development using UI	Major Course Elective-I	600-699 Advance level Technical	4	2	4 (Un-supervised mode)
904-02	Cloud storage interaction using Android application	Major Course Elective-II	600-699 Advance level Technical	4	2	4 (Un-Supervised mode)
905	Big Data and Hadoop	Major Course	600-699 Advance level Technical	4	3	2 (2- supervised mode)
906	Practical (Based on Course Code: 901,902 & 905)	Major Course	600-699 Advance level Technical	4	-	8 (2 - supervised mode, 6- unsupervised mode)
907	Project (Based on Course Code: 903 & 904) (Part Time Project at Industry/corporate)	Major course (Skill Enhancement)	600-699	8	-	16 (Un-supervised mode)
Other Activities	The student is expected to participate in activities related to National Service Scheme (NCC), National Cadet Corps (NCC), adult education/literacy initiatives, mentoring school students, Elderly literacy program/ Environment preservation activities and other similar activities.			-	-	-
Total				32	10	30

Course Code	Course Title	Course Credit	University Exam Type	Exam Duration	External Marks	Internal Marks	Total Marks
900	Value Addition Course[#]	2	As mentioned in the course.*	-	70	30[#]	100
901	Data Intelligence and Visualization	3	Theory (Descriptive ,Short Questions and MCQ)	3 Hours	70	30	100
902	Robotic Process Automation (RPA)	3	Theory (Descriptive, Short Questions and MCQ)	3 Hours	70	30	100
903-01 OR 903-02	Server side scalable Application OR Android based Sensors handling	4	Theory (Descriptive .Short Questions and MCQ)	3 Hours	70	30	100
904-01 OR 904-02	Application Development using UI OR Cloud storage interaction using Android application	4	Theory (Descriptive .Short Questions and MCQ)	3 Hours	70	30	100
905	Big Data and Hadoop	4	Theory (Descriptive .Short Questions and MCQ)	3 Hours	70	30	100
906	Practical (Based on 901,902,905)	4	Practical	5 Hours	140	60	200
907	Project	8	Demonstration/ Presentation / Viva	5 Hours	70	30	100
Total		32			630	270	900
Minimum Passing Score : 40% in each individual head.							

For Practical and Project:

- Batch Size – 40 Maximum (Desirable). Maximum 45 students can be accommodated in a batch. Separate batch should be considered if the student strength exceed 45 numbers.
- Practical Course-906 includes Practical sessions for course-901, course-902 and course-905. **Minimum** Ten Practical hours(5 hours for course-902 and 5 hours for course-905) per week should be allocated per batch. Out of which 6 hours will be in supervised mode and balance hours in un-supervised mode.
- The journal should be certified by the concerned faculty and by the Head of the Department, failing which the student should not be allowed to appear for External Practical Examination. Student will submit softcopy of Minor Project duly certified by the internal guide.
- The students must carry out a part-time industrial project during the semester based on Course-903 and Course-904. Students are expected to analyse the project requirement, design, develop, code, test and deploy the project. The work will be carried out by the students in un-supervised mode. Minimum 8 hours per week should be allocated to the student for working on the project in un-supervised mode.

Value Addition Course: Student will opt any one course from the given choices by the institute/college of nature Value Addition Course from available pool of courses recognized by the University. The external and internal evaluation (For course code: 900) will be carried out by the institution/college based on the nature of course. The evaluation pattern may include any or combination of (i) MCQ exam (ii) Viva-Voce (iii) Presentation (iv) Project Demonstration.

***Certificate Course :** For this courses, the students will enrol for the course from the given university approved list of certificate courses offered by the respective college/department. The student will select and enrol separately for any of the offered list of courses at college/department/institute and obtain respective credits. The institute will evaluate the performance (preferably continuous evolution) as per the SOP of certificate courses and on successfully completion of the course, the student will be eligible to obtain respective credits for the course. These credits will be considered and reflect in student's mark-sheet as well as in ABC(Academic Bank of Credit). These courses are mandatory and student is required to obtain the specified credits in process to acquire the certificate/diploma/degree. [The student is required to pay separately for these courses as prescribed by the college.]

Marks: The scale on which the students will be evaluated for the Audit course. The evaluation methodology will be continuous evaluation and the score obtained will reflect in mark-sheet.

*The Un-supervised mode of Practical/Fieldwork/Internship/Project work will be carried out by the students independently for the allocated hours/week at computer Lab./place of internship/field/project place.

M.Sc.(Computer Application) Semester-3
Course Code: 900
Course Title: Value Addition Course

Course Code	900
Course Title	Value Addition Course
Credit	2
Category of Course	Value Addition Course
Level of Course	600-699 (Advance Level Technical)
Teaching per Week	2 Hrs (Any or Combination of Theory/Practical/Fieldwork/Internship/Project)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Student will select minimum one 2-credit course of category value addition out of the choices given by the college/institute. It will be mandatory for the student to opt minimum one 2-credit Value Addition Course out of the list of offered courses recognised by the University during semester-1. This course will be an Audit course. Student can enhance the knowledge in the selected field by obtaining higher degree of knowledge in the area.
Course Objective	Obtaining knowledge in all or any of the components/fields like (i) Understanding India (ii) Environmental Science/Education (iii) Digital/Technological solutions (iv) Health & Wellness, Yoga education, sports, and fitness are essential for holistic development and (v) Indian Knowledge System (vi) Artificial intelligence and Robotics. The course components should be among these six categories/fields and as per the Curriculum and Credit Framework for Undergraduate Programmes of the UGC. The purpose is to impart knowledge and understand the necessities of these aspects in life to make the healthy society and nation. It help in development of a dedicated and responsible citizen of the country by adding value to the life.
Pre-requisite	No prior knowledge in the field is essential.
Course outcome	CO1: Student select the area of Value addition as per his/her interest. The choices will be given by the institute/department. CO2: The student acquire basic and fundamental level of knowledge in the field that the student opted. CO3: Understand the insight of the area and possibility of to explore more in the field. CO4: Understand effective representation of problems, solutions and insights of the challenges and problems of the peer subject relevant to value addition. CO5: Learn to upskill and upgrade the knowledge in the area of selected subject.
Course Content and Implementation road-map.	(i) The university has categorised and prepared the list of the courses that can be offered as Value Addition Course. (ii) The institute/college/department can design and implement skill enhancement course by getting approval from the relevant apex body of the university considering the SOP of the certificate course policies of the University. (iii) The institutes/college/departments can select more than one course out of the given sets of courses and offer them to their students. (iv) The students can select any of the courses offered by the institute/college/department from the given choices and enrol for the course.

	(v) The institute/college/department will arrange appropriate resource person(s) for the course. (vi) The evaluation will be taken place at the college/institute/department based on the nature of the course. (vii) The institute/college/department will assess the student based on the nature of the course. The student will be granted the credits on successful completion of the course.
Reference Books	- The reference materials and books will be decided by the Institutes/Colleges/Departments based on the selected Courses. - Minimum five copies of relevant topics are recommended to keep in the library.
Teaching Methodology	Class Work/ Discussion/ Self-Study/ Seminars/ field works/ practical training/ field work and/or Assignments.
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Course Code: 901
Course Title: Data Intelligence and Visualization

Course Code	901
Course Title	Data Intelligence and Visualization
Credit	04
Category of Course	Major Course
Level of Course	600-699 (Advance Level Technical)
Teaching per Week	4 Hrs (2 hours of theory + 4 hours of Lab sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Purpose of a data visualization course is to learn how to effectively communicate complex data and information through visual representations. By understanding the principles of data visualization and design, students will be able to create meaningful and impactful visualizations that can be used to inform decision-making and drive business outcomes. Power BI is a business analytics service by Microsoft that allows users to create interactive visualizations and reports from a variety of data sources. It is designed to enable users to quickly and easily create powerful visualizations, dashboards, and reports that can be shared across an organization.
Course Objective	- To Understand the principles of data visualization: - To learn about the principles of effective data visualization, including visual perception, colour theory, and layout design. - To understand the features and functionalities of Power BI, including data modelling, data visualization, and dashboard design. - To learn how to use Power BI to import, clean, and transform data from various sources, and how to create interactive reports and dashboards. - To enhance data analysis skills. - To develop skills in data analysis, including data wrangling, exploratory data analysis, and statistical analysis. They will learn how to use data to identify trends, patterns, and insights, and how to use these insights to inform decision-making. - To work on Real-world applications. Practical, hands-on experience working with real-world datasets and scenarios. Applying the principles and skills learned in the course to solve real-world problems, such as identifying trends in sales data, analyzing customer behavior, or forecasting financial performance. - To emphasize the importance of effective communication in data visualization. To learn how to create visualizations that effectively communicate insights and information to a wide range of audiences, including executives, managers, and colleagues. To learn how to tell a compelling story with data, and how to use visualizations to influence decision-making.
Pre-requisite	Basic understanding of statistics, Familiarity with worksheet, Data analysis skills including data cleaning, transformation, and filtering. Basic programming skills, Knowledge of a programming language such as Python or R.
Course outcome	**CO1:** Ability to create effective visualizations: Students will be able to create effective and impactful visualizations that communicate insights and information in a clear and concise manner. They will understand the principles of effective data visualization and be able to apply them to real-world scenarios.

CO2: Mastery of Data visualization tools: Students will have a strong understanding of Power BI and its features, and be able to use the software to create interactive reports and dashboards. They will know how to connect to different data sources, transform and clean data, and create visualizations that can be shared across an organization.
CO3: Data analysis skills: Students will develop skills in data analysis, including data wrangling, exploratory data analysis, and statistical analysis. They will understand how to identify trends, patterns, and insights in data, and use these insights to inform decision-making.
CO4: Practical experience with real-world datasets: The course should provide practical, hands-on experience working with real-world datasets and scenarios. Students will apply the principles and skills learned in the course to solve real-world problems, such as identifying trends in sales data, analyzing customer behaviour, or forecasting financial performance.
CO5: Effective communication: Students will develop skills in effective communication with data.

Mapping between COs with PSOs

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

Course Content

Unit 1. Data Acquisition and Pre-processing:
1.1 Reading and writing data:
1.1.1 Data cleaning and pre-processing using Pandas and Numpy
1.1.2 Data transformation and normalization using Numpy
1.1.3 Exploratory Data Analysis
1.2 Data visualization using Matplotlib and Seaborn
1.2.1 Summary statistics and descriptive analysis
1.2.2 Outlier detection and treatment
1.3 Data Modelling and Analysis
1.3.1 Introduction to statistical models and inference
1.3.2 Linear regression and logistic regression
1.3.3 Time series analysis

Unit 2.
2.1 Data visualization (using seaborn, matplotlib and bookeh)
2.1.1 Working with Scatter plot, line char, bar chart and histogram using matplotlib, seaborn and bookeh
2.1.2 Adding widgets using bookeh (Buttons, CheckboxGroup,RadioGroup, slider)
2.2 Introduction to Business Intelligence (BI)
2.1.1 Stages of business intelligence (BI)
2.1.2 Use cases of BI and various BI tools
2.1.3 Overview of Data warehouse & concepts

Unit 3.
3.1 Introduction to Power BI
3.1.1 Introduction of Power BI and its Components
3.1.2 Building blocks and Architecture of Power BI
3.1.3 Introduction and installation of Power BI Desktop
3.2 Data sources in Power BI
3.2.1 Using files (excel, pdf, csv) as a data source

	3.2.2 Extracting data from folders, and databases 3.2.3 Working with Cloud SQL database and database sources 3.2.4 Connecting to Analysis Services 3.2.5 Working with Other data sources (OData, web, SharePoint) 3.3 Python script in Power BI 3.4 Introduction of power query editor **Unit 4.** 4.1 Data Transformation (Shaping and Combining Data) 4.1.1 Formatting and Transformation of data 4.1.2 Understanding of Data types 4.1.3 Data profiling for data quality check 4.1.4 Merge, Append and Group by(Aggregate) Query 4.2 Query settings 4.2.1 Duplicate & Reference tables 4.2.2 Transpose of data 4.2.3 Pivot & Un-pivot of data 4.2.4 Custom columns, Conditional columns 4.2.5 Replacing data from the tables 4.2.6 Split columns values 4.3 Move columns & sorting of data 4.3.1 Detect data type, count rows & reverse rows 4.3.2 Promote rows as column headers 4.3.3 Hierarchies in Power BI 4.3.4 Concept of M Query **Unit 5** 5.1 DAX (Data Analysis Expression) 5.1.1 Introduction of DAX 5.1.2 DAX syntax 5.1.3 DAX functions 5.1.4 Context in DAX 5.2 Calculated columns using DAX 5.2.1 Measures using DAX 5.2.2 Calculated tables using DAX 5.2.3 Learning about table, information, logical, text, iterator, 5.2.4 Time intelligence functions (YTD, QTD, MTD) 5.2.5 Cumulative values, calculated tables, and ranking and rank over groups 5.3 Date and time functions 5.3.1 identify poorly performing measures, relationships, and visuals 5.4 Data visualization 5.4.1 Understanding Power View and Power Map 5.4.2 Data visualization techniques 5.4.3 Page layout & Formatting 5.4.4 Desktop visualization 5.4.4.1 Formatting and customizing visuals 5.4.4.2 Visualization interaction 5.4.4.3 Custom visualization 5.4.5 Top-down and bottom-up analytics 5.4.6 Drill down, Drill through, Slicer [All Units carry Equal Weightage]

Reference Books	1)"Data Visualization Made Simple: Insights into Becoming Visual" by Kristen Sosulski (ISBN: 9780367257055) 2)"Data Visualization with Power BI" by Dan Clark (ISBN: 9781788297233) 3)"Power BI Essentials: An Introduction to Microsoft Power BI" by Basictech Information Services (ISBN: 9781539702831) 4)"Data Visualization: A Practical Introduction" by Kieran Healy (ISBN: 9780691181622) 5)"Mastering Microsoft Power BI: Expert techniques for effective data analytics and business intelligence" by Brett Powell (ISBN: 9781788297233) 6)"Data Analytics Made Accessible" by Anil Maheshwari (ISBN: 9780367353191) 7)"Data Analytics: An Essential Beginner's Guide to Data Mining, Data Collection, Big Data Analytics for Business Intelligence and Data Science" by Herbert Jones (ISBN: 9781724015361) 8)"Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython" by Wes McKinney (ISBN: 9781491957660) 9)"Data Smart: Using Data Science to Transform Information into Insight" by John W. Foreman (ISBN: 9781118661468) 10)"Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking" by Foster Provost and Tom Fawcett (ISBN: 9781449361327)
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Course Code: 902
Course Title: Robotic Process Automation (RPA)

Course Code	902
Course Title	Robotic Process Automation (RPA)
Credit	04
Level of Course	600-699 (Advance level technical)
Teaching per Week	4 Hrs (2 hours of theory + 4 hours of Lab sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Purpose of an RPA (Robotic Process Automation) course is to provide learners with the knowledge and skills necessary to create software robots that can automate repetitive and manual tasks. RPA technology uses software robots or "bots" to interact with applications, databases, and systems to perform tasks just like a human worker would. By automating these tasks, RPA can increase efficiency, reduce errors, and improve productivity. An RPA course typically covers topics such as RPA basics, UiPath RPA tools, automation workflow development, RPA governance, and management. Upon completing the course, learners should be able to design, develop, test, and deploy software robots using RPA technology. It is beneficial to optimize their business processes and increase their operational efficiency.
Course Objective	- Understand the basics of RPA - To Gain proficiency in RPA tools - To Learn to identify automation opportunities - To Understand the importance of governance - To Learn to manage and maintain RPA solutions
Pre-requisite	Basic Computer Skills, Understanding of Business processes, Familiarity of Programming concepts, Analytical Thinking
Course outcome	**CO1:** Ability to create basic UiPath automations: By the end of the course, students should be able to create basic UiPath automations using the UiPath Studio interface. Understanding different types of UiPath activities, variables, data types, and how to use them to create effective automation workflows. **CO2:** Understanding of UiPath best practices: To ensure that UiPath automations are efficient, reliable, and maintainable, it's important to follow best practices for automation design and development. The course should cover UiPath best practices for exception handling, logging, and error management. **CO3:** Knowledge of UiPath Orchestrator: UiPath Orchestrator is a web-based management platform that enables the deployment, monitoring, and management of UiPath automations. The course should cover the basics of using Orchestrator, including setting up robots, scheduling jobs, and managing queues. **CO4:** Experience with advanced UiPath features: UiPath offers a range of advanced features that can be used to build more complex and sophisticated automations. The course should cover some of these advanced features, such as UiPath Activities, Recording, Data Scraping, and Re-Framework. **CO5:** Practical experience with real-world scenarios: To be truly effective at RPA using UiPath, students need practical experience with real-world scenarios. The course should provide hands-on opportunities to create UiPath automations for common business processes, such as invoice processing, data entry, and report generation.

Mapping between COs and PSOs

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

Course Content

Unit 1.

1.1 Introduction to RPA
- 1.1.1 Concepts of RPA.
- 1.1.2 Benefits and limitations of RPA
- 1.1.3 Use cases of RPA

1.2 RPA Tools and Technologies
- 1.2.1 Overview of popular RPA tools (UiPath, Automation Anywhere and Blue Prism)
- 1.2.2 Key features and capabilities of RPA

[Practical Applications of Unit-1: Case study of three popular RPA tools.]

Unit-2.: UiPath

2.1 Overview of UiPath Studio and UiPath Orchestrator
2.2 Installing and setting up UiPath
- 2.2.1 UiPath Studio
- 2.2.2 User interface and features of UiPath Studio

2.3 Recording and editing automation workflows
2.4 Building automation workflows using UiPath activities and variables
2.5 Debugging and troubleshooting automation workflows
2.1 Ribbon
2.2 Universal Search Box
2.3 Activities Panel, Design Panel, Library Panel, Project Panel
2.4 Properties Panel, Outline Panel, Output Panel, Control Panel

[Practical Applications of Unit-2: Working with UiPath studio and various panels]

Unit 3. Workflow and Selectors

3.1 Types of workflow: Sequences, Flowcharts, State Machines
3.2 Variables : DataTypes and Usage, Managing Arguments
- 3.2.1 Using Data Scrapping with examples
- 3.2.2 Recording : About Recording, Recording Types
  - 3.2.2.1 Automatic Recording with examples (basic and Desktop)
  - 3.2.2.2 Automatic Recording with Web
  - 3.2.2.3 Manual Recording

3.3 Selectors :
- 3.3.1 Concepts of Selectors
- 3.3.2 Selectors with Wildcards
- 3.3.3 Full versus Partial Selectors
- 3.3.4 UiPath Explorer

[Practical Applications of Unit-2: Using variables and selectors.]

Unit 4. Automation and Activities:

4.1 Image and Text Automation
- 4.1.1 Mouse and Keyboard Activities
- 4.1.2 Text, OCR and Image Activities

4.2 Citrix Automation and Use case of Citrix Automation
4.3 Worksheet and Data Activities
4.4 Data Extraction from PDF and PDF activities

	4.5 Email automation and Activities 4.6 Debug workflow and Error handling [Practical Applications of Unit-4: Data extractions from PDF, Email and worksheets] **Unit 5. UiPath Orchestrator and Best Practices** 5.1 UiPath Orchestrator 5.1.1 User interface and features of UiPath Orchestrator 5.1.2 Setting up and managing robots 5.1.3 Creating and scheduling jobs 5.1.4 Monitoring and analyzing automation performance 5.2 UiPath Development Best Practices 5.2.1 Best practices for UiPath development 5.2.2 Designing automation workflows for reusability and scalability 5.2.3 Documenting automation workflows 5.2.4 Testing and validating automation workflows 5.3 Real-world examples of UiPath in various industries (e.g. finance, healthcare, retail) **[**Practical Application of Unit-5: Use of UiPath Orchestrator and development of best practices.] [All Units carry Equal Weightage]
Reference Books	1)"Robotic Process Automation: A Comprehensive Guide" by Ankur Kothari, Nandan Mullakara, and Raghu Nath, ISBN-10: 1838981082, Packt Publishing. 2)"Practical RPA: Moving Past the Hype to Realizing Business Value" by Edward Brooks, ISBN-10: 1484267653, Apress. 3)"Implementing Robotic Process Automation: A Practical Guide" by Neil Kolban, ISBN-10: 1801073650, Packt Publishing. 4)"The Enterprise Automation Playbook: A Playbook for RPA, Intelligent Automation, and Digital Transformation" by Sam Best, ISBN-10: 1119775739, Wiley. 5)"Mastering UiPath: Robotic Process Automation" by Martin Vayu, ISBN-10: 1801810927, Packt Publishing. 6)"UiPath RPA Developer: Build a Foundation in RPA" by Asha Kumar, ISBN-10: 1800563187, Packt Publishing. 7)"UiPath Cookbook: Over 130 actionable recipes to automate your enterprise processes with UiPath" by Alex Vaidya, ISBN-10: 1801817662, Packt Publishing. 8)"Mastering UiPath: Robotic Process Automation" by Martin Vayu, ISBN-10: 1801810927, Packt Publishing. 9)"UiPath Automation Projects: A beginner's guide to learning Robotic Process Automation with UiPath" by Rahul Mehta, ISBN-10: 1801073219, Packt Publishing. 10)"UiPath Process Mining: From Data to Value" by Marco Ramoni, Emanuele Cecchetti, and Andrew Spanyi, ISBN-10: 180107407X, Packt Publishing.
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Course Code: 903-01
Course Title: Server side scalable Application

Course Code	903-01
Course Title	Server side scalable Application
Credit	04
Category of Course	Major Course (Elective) (Student will select any one course under course code 903).
Level of Course	600-699 (Advance level - Technical)
Teaching per Week	4 Hrs (2 hours of theory + 4 hours of Lab sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Node. js is an open-source, cross-platform JavaScript runtime environment and library for running web applications outside the client's browser. It provides an event driven, non-blocking (asynchronous) I/O and cross-platform runtime environment for building highly scalable server-side application using JavaScript.
Course Objective	- Get user inputs via Command Line Arguments and store data using File system. - To learn how to develop scalable web applications using Express Framework and deploy them using Nginx. - To gain in-depth knowledge of REST APIs, implement testing, build applications using microservices architecture and write a real-time chat application using Socket IO.
Pre-requisite	Knowledge of Java Script, web applications and MongoDB/Any RDBMS.
Course outcome	**CO1:** Learn to get user inputs via Command Line Arguments and store data using File system. Also learn how to create the applications using Express Framework, whereas manage and deploy them using PM2 and Nginx. **CO2:** learn how to develop asynchronous Node.js applications using Call stack, Callback queue and Event Loop mechanism. **CO3:** Understanding to Create dashboard application using ES6. **CO4:** Learning client-server interaction using socket.io **CO5:** Learning testing the UI and data interaction.
Mapping between COs and PSOs:	
Course Content	**Unit 1.** **1.1**File System Module 1.1.1 Inputs from Users 1.1.2 Pass Multiple Arguments with Yargs 1.1.3 File System Module 1.1.4 Operations associated with File System Module 1.2 JSON Data, Http Server and Client 1.2.1 Sending and receiving events with EventEmitters

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

	1.2.2 Express Framework Run a Web Server using Express Framework 1.2.3 Routes Deploy application using PM2 and Nginx [Practical Applications of Unit-1: Build an API using express, read file with FS module, and deploy application using PM2 and Nginx] **Unit 2.** 2.1 Call Stack Callbacks, Callback Queue and Event 2.1.1 Loop Callback Abstraction 2.1.2 Callback Chaining 2.2 Promises 2.2.1 Promise Chaining 2.2.2 Request Package 2.2.3 Customizing HTTP Requests and Error handling with HTTP codes 2.2.4 Introduction to template engine (EJS) 2.3 Paragraph Development **[**Practical Application of Unit-2: HTML Page Using EJS Template and create an Application on retail store] **Unit 3. Application building using ES6** 3.1. ES6 variables 3.1.1 Functions with ES6 3.1.2 Import and Export withES6 3.1.3 Async/Await 3.2 Introduction to Babel 3.3 Rest API with ES6 3.3.1 Browsing HTTP Requests with Fetch 3.3.2 Processing Query String 3.4 Creating API using ES6 3.4.1 Transpilation 3.4.2 Building Dashboard API 3.4.3 Creating dashboard UI with EJS 3.4.4 ES6 Aside: Default Function Parameters 3.4.5 Data Validation and Sanitization **[**Practical Application of Unit-3: Building Dashboard application using ES6 concepts.] **Unit 4. Client-server interaction using socket.io** 4.1 Concepts of Web Sockets 4.1.1 Understanding Socket.io 4.1.2 Broadcasting Events 4.1.3 Sharing current Location 4.1.4 Event Acknowledgements 4.2 Form and Button States 4.2.1 Rendering Messages 4.2.2 Working with Time 4.2.3 Timestamps for determining Location of Messages 4.2.4 Storing Users, Rendering User List 4.2.5 Tracking Users Joining and Leaving 4.3 Deploying the Chat Application 4.4 Concepts of Redis and Building API with Redis [Practical Application of Unit-4: Develop a Realtime Chat Application using Socket.io , Build an API using Redis.]

	Unit 5. Testing Node.js application 5.1. Understanding mocha framework 5.2. Writing Tests and Assertions 5.2.1 Testing Asynchronous Code 5.2.2 Testing an Express Application 5.2.3 Setup and Teardown 5.2.4 Testing with Authentication 5.2.5 Understanding chai.js 5.3 Advanced Assertions 5.4 Mocking Libraries 5.5 Wrapping up User Tests 5.6 Setup Task Test Suite 5.7 Testing with Task Data [Practical Application of Unit-5: Using mocha and chai for testing the application.] [All Units carry Equal Weightage]
Reference Books	1."Node.js in Action" by Mike Cantelon, Marc Harter, TJ Holowaychuk, and Nathan Rajlich. ISBN-13: 978-1617292576. Publisher: Manning Publications. 2."Learning Node.js: A Hands-On Guide to Building Web Applications in JavaScript" by Marc Wandschneider. ISBN-13: 978-0134436540. Publisher: Addison-Wesley Professional. 3."Node.js Design Patterns: Master best practices to build modular and scalable server-side web applications" by Mario Casciaro. ISBN-13: 978-1785885587. Publisher: Packt Publishing. 4."Pro Node.js for Developers" by Colin J. Ihrig. ISBN-13: 978-1484219727. Publisher: Apress. 5."Node.js, MongoDB, and AngularJS Web Development" by Brad Dayley and Brendan Dayley. ISBN-13: 978-0134655536. Publisher: Addison-Wesley Professional. 6."Web Development with Node and Express: Leveraging the JavaScript Stack" by Ethan Brown. ISBN-13: 978-1491949306. Publisher: O'Reilly Media. 7."Hands-On Full Stack Web Development with Angular 6 and Laravel 5: Become fluent in both frontend and backend web development with Docker, Angular and Laravel" by Fernando Monteiro. ISBN-13: 978-1788833912. Publisher: Packt Publishing. 8."Beginning Node.js" by Basarat Ali Syed. ISBN-13: 978-1484201883. Publisher: Apress. 9."Node.js 8 the Right Way: Practical, Server-Side JavaScript That Scales" by Jim Wilson. ISBN-13: 978-1680501957. Publisher: Pragmatic Bookshelf. 10."Node.js for PHP Developers: Porting PHP to Node.js" by Daniel Howard. ISBN-13: 978-1491904430. Publisher: O'Reilly Media.
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments, Practical/Project
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Course Code: 903-02
Course Title: Android based Sensors handling

Course Code	903 – 02
Course Title	Android based Sensors handling
Credit	04
Category of Course	Major Course (Elective) (Student will select any one course among under course code 903).
Level of Course	600 – 699
Teaching Per Week	4 Hrs (2 hours of theory + 4 hours of Leb sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review/Revision	-
Implementation Year	A.Y. 2024-2025
Purpose of Course	It provides students with an in-depth understanding of how touch, multi-touch, and gesture recognition work on the Android platform. This course covers the touch-based input, including the different types of touch sensors, multi-touch gestures, and the android touch event system. The course also covers advanced topics such as gesture recognition, which is used to interpret complex touch-based inputs, and multi-window handling, which allows multiple applications to be displayed on the screen simultaneously.
Course Objective	• Understand the basics of touch-based input on Android devices, including the different types of touch sensors and the Android touch event system. • Implement multi-touch gestures such as pinch-to-zoom, rotate, and swipe in Android applications. • Use gesture recognition techniques to interpret complex touch-based inputs in Android applications. • Implement multi-window handling to allow multiple applications to be displayed on the screen at the same time. • Apply best practices for designing touch-based interfaces in Android applications.
Pre-requisite	Fundamental of Programming, Basic concepts of Android Application Development
Course Outcome	CO1: Understanding the basics of touch-based input on Android devices, including the different types of touch sensors and the Android touch event system. Use gesture recognition techniques to interpret complex touch-based inputs in Android applications. Apply best practices for designing touch-based interfaces in Android applications. CO2: Implement multi-window handling to allow multiple applications to be displayed on the screen at the same time. Implement multi-window handling to allow multiple applications to be displayed on the screen at the same time.

	CO3: Keep up-to-date with the latest trends and advancements in touch-based input, gesture recognition, and multi-window handling for Android devices. Understand the user experience implications of touch-based input, gesture recognition, and multi-window handling for Android applications. CO4: Apply best practices for touch-based interfaces including optimizing for different screen sizes and densities. Understand the differences between touch-based and non-touch-based interfaces and the impact of touch-based interfaces on user experience. CO5: able to prepare apps for publication on Play Store by completing necessary requirements, such as signing apps, testing apps, and configuring app metadata.
Mapping between COs and PSOs:	

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

Course Content	Unit 1 1.1. Android Touch and Multi-touch 1.1.1.Introduction to touch-based input on Android devices 1.1.2. Touch Events 1.1.3. The MotionEvent Object 1.1.4. Concepts of Touch Actions 1.1.5. Multiple Touches handling 1.1.6. Touch Event Listener implementation Unit 2 2.1 Detecting Common Gestures using the Android Gesture Detector Class 2.1.1. Common Gesture Detection implementation 2.1.2. Implementing the Listener Class 2.1.3. Creating the GestureDetectorCompat Instance 2.1.4. Implementing the onTouchEvent() Method 2.2 The GestureOverlayView Class 2.3 Detecting Gestures 2.4 Identifying Specific Gestures 2.5 Building and Running the Gesture Builder Application Unit 3 3.1. An Introduction to Android Multi-Window Support 3.1.1. Split-Screen, Freeform and Picture-in-Picture Modes 3.1.2. Entering Multi-Window Mode 3.2. Enabling Freeform Support 3.3. Enabling Multi-Window Support in an App 3.3.1. Specifying Multi-Window Attributes 3.3.2. Detecting Multi-Window Mode in an Activity 3.3.3. Receiving Multi-Window Notifications 3.4. Launching an Activity in Multi-Window Mode Unit 4 4.1. An Android Fingerprint Authentication 4.2. Configuring Device Fingerprint Authentication

	4.3. Adding the Fingerprint Permission to the Manifest File 4.4. Adding the Fingerprint Icon 4.5. Designing the User Interface 4.6. Accessing the Keyguard and Fingerprint Manager Services 4.7. Checking the Security Settings 4.8. Accessing the Android Keystore and KeyGenerator 4.9. Generating the Key 4.10. Implementing the Fingerprint Authentication Handler Class Unit 5 5.1. Signing and Preparing an Android Application for Release 5.2. The Release Preparation Process 5.3. Register for a Google Play Developer Console Account 5.4. Configuring the App in the Console 5.5. Enabling Google Play App Signing 5.6. Enabling ProGuard 5.7. Creating a Keystore File 5.8. Creating the Application APK File 5.9. Uploading Instant App APK Files
Reference Books	1) Android Application Development (With Kitkat Support), Author: Pradeep Kothari, Publisher:DreamTech Press.,ISBN:978-9351194095 2) Android Studio 3.0 Development Essentials: Android 8 Edition Author – Neil Smyth, Publisher: Payload Media, ISBN – 13: 978 – 1977540096 3) Fundamentals of Android App Development : Android Development for Beginners to Learn Android Technology, SQLite, Firebase and Unity, Author: Sujit Kumar Mishra, Publisher: BPB Publication, ISBN: 978-93-89845-204 4) Starting with Android: Android application development guide 1st Edition, Author: Dr. M. M. Sharma, Publisher :BPB Publication, ISBN: 978-9386551955 5) Android Programming for Beginners - Second Edition, Author:John Horton, Publisher: Image Short ISBN: 978-1789538502 6) Android 9 Development Cookbook, Author: Rick Boyer, Publisher: Packet Publishing, ISBN:978-1788991216 7) Professional Android – fourth Edition, Author: Reto Meier, Ian Lake, Publisher: Wrox, ISBN – 13:978-1118949528 8) Android Programming: Pushing the Limits 1st Edition, Author: Erik Hellman, Publisher: Wiley, ISBN – 13: 978-1118717370
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments, Practical/Project
Evaluation Method	30% Internal Assessment 70% External Assessment

M.Sc.(Computer Application) Semester-3
Course Code: 904-01
Course Title: Application Development using UI

Course Code	904-01
Course Title	Application Development using UI
Credit	04
Category of Course	Major Course (Elective) (Student will select any one course among under course code 904).
Level of Course	600-699 (Advance level technical course)
Teaching per Week	4 Hrs (2 hours of theory + 4 hours of Lab sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	ReactJS with Redux focuses and demonstrates how both of them can be used together to build extensive web applications. It also guides on how to develop responsive UIs to handle user interactions. The course also covers the GraphQL which is an open-source data query language and data manipulation language for APIs, and a query runtime engine.
Course Objective	- To Develop understanding of Web Development Architecture - To Create application using React components - To Perform Navigation using Routes - To Build Web Applications using React with Redux Program - To Async Actions using Redux-Saga Middleware - To Write Queries using GraphQL - To Execute Test Cases using Jest - To Deploy Applications using Docker and Nginx - To Build Mobile Applications using Native React
Pre-requisite	Fundamentals of Programming, concepts of Objects and classes, HTML, CSS and knowledge of Java Script. Knowledge of SQL.
Course outcome	CO1: Ability to build scalable and complex web applications: React.js is known for its ability to build reusable and scalable components, while Redux provides a predictable state management system. Learning both together can enable developers to build more complex and scalable web applications. Co2: Better management of application state: Redux provides a centralized store for application state, making it easier to manage and debug. By learning how to use Redux with React, developers can better manage the state of their application, leading to more maintainable and robust code. CO3: Understanding of functional programming principles: Redux follows functional programming principles, which can help developers write more concise and predictable code. Learning how to use Redux with React can help developers understand and apply these principles to their code. CO4: Improved debugging and error handling: Redux provides a clear separation between state and UI, making it easier to debug and handle errors in a React application. By learning how to use Redux with React, developers can improve their debugging and error handling skills. CO5: Knowledge of popular front-end frameworks and libraries: React and Redux are two of the most popular front-end frameworks and libraries, respectively. Learning how to use both together can provide developers with valuable knowledge and skills that can be applied to other front-end development projects.

Mapping between COs and PSOs			PS01	PS02	PS03	PS04	PSO5	PS06		
		CO1								
		CO2								
		CO3								
		CO4								
		CO5								

Course Content

Unit 1.

1.1 Building Blocks of Web Application Development:
- 1.1.1 Single-page and Multi-page Applications, Different Client-side Technologies
- 1.1.2 MVC Architecture
- 1.1.3 Introduction to React, Installation of React JSX and its use case
- 1.1.4 DOM, Virtual DOM and its working
- 1.1.5 ECMAScript, Difference between ES5 and ES6.
- 1.1.6 NPM Modules

1.2 React Elements:
- 1.2.1 Render Function, Components, Class Component, Component Constructor, Functional Components, Multiple Components,
- 1.2.2 Props: Props with Class based Component, Props with Function based Component, States,
- 1.2.3 Component Lifecycle

1.3 React Events,
1.4 React Forms, Different Form Concepts.
1.5 Styling in React and Inline Styling

[Practical Applications of Unit-1: Build Music shop application using Unit-1]

Unit 2.

2.1 Routing: react-router, Features of react-router, Configuration of routing using react-router
2.2 Navigation using Links
- 2.2.1 404 page (Not found Page)
- 2.2.2 URL Parameters
- 2.2.3 Nested Routes
- 2.2.4 Implementing styles using NavLink

2.3 Application Programming Interface
- 2.3.1 Build a REST API using json-server
- 2.3.2 API consumption in React application using Fetch method

[Practical Application of Unit-2: Build a dynamic Music Store application using Routing and API connectivity]

Unit 3. Redux and Saga-Middleware

3.1 Redux: Need of Redux, Redux Architecture, Redux Action,
- 3.1.1 Redux Reducers, Redux Store, Principles of Redux
- 3.1.2 Pros of Redux

3.2 NPM Packages required to work with Redux
3.3 Async Operations:
- 3.3.1 Need of Async operations
- 3.3.2 Async Workflow
- 3.3.3 Action Creators and How to write Action Creators
- 3.3.4 Handling Async Actions via Reducers

3.4 Middleware: Redux-Saga

	3.4.1 Generators in Redux-Saga 3.4.2 Saga Methods() 3.4.3 Building a Product List 3.4.4 Debugging application using Redux Devtools **[**Practical Application of Unit-3: Building an application to list the food items using React and Redux. Building News application using React, Redux, and promise middleware. Building a Product list application using Redux-Saga Middleware.] **Unit 4. GraphQL:** 4.1 Understanding GraphQL: 4.1.1 Cons of Rest API Pros of GraphQL 4.1.2 Frontend backend communication using GraphQL 4.1.3 Type system 4.2 GraphQL datatypes 4.3 Modifiers Schemas 4.3.1 GraphiQL tool 4.3.2 Express framework 4.3.3 NPM libraries to build server side of GraphQL 4.4 Build a GraphQL API 4.5 Apollo client 4.5.1 NPM libraries to build client side of GraphQL 4.5.2 Setup Apollo client [Practical Application of Unit-4: Build a GraphQL API and execute queries using GraphiQL tool Fetch Space Launch Data using Apollo-GraphQL] **Unit 5. Open Source Testing Framework** 5.1 Understanding Jest 5.1.1 Setup Testing environment 5.1.2 Add Snapshot testing 5.2 Integrate Test Reducers 5.2.1 Create Test Components 5.2.2 Push Application on Git 5.3 Understanding Nginx 5.3.1 Deploy App on Nginx 5.3.2 Create Docker for React Application [Practical Application of Unit-5: Testing application using Jest Application and Deployment via Nginx and Docker] [All Units carry Equal Weightage]
Reference Books	1)"React: Up & Running: Building Web Applications" by Stoyan Stefanov and Kirupa Chinnathambi (ISBN: 978-1491931820, Publisher: O'Reilly Media) 2)"Learning React: A Hands-On Guide to Building Web Applications Using React and Redux" by Kirupa Chinnathambi (ISBN: 978-0134843551, Publisher: Addison-Wesley Professional) 3)"React Design Patterns and Best Practices" by Michele Bertoli (ISBN: 978-1786464538, Publisher: Packt Publishing) 4)"React Cookbook: Over 66 hands-on recipes that cover UI development, animations, component architecture, routing, databases, testing, and debugging

	with React" by Carlos Santana Roldán (ISBN: 978-1783980727, Publisher: Packt Publishing) 5)"Full-Stack React Projects: Modern web development using React 16, Node, Express, and MongoDB" by Shama Hoque (ISBN: 978-1788835534, Publisher: Packt Publishing) 6)"Learning Redux" by Daniel Bugl (ISBN: 978-1786462398, Publisher: Packt Publishing) 7)"Hands-On Redux for React Native: A Practical Guide to Building Real-Time, Scalable Mobile Applications" by Spencer Carli (ISBN: 978-1788997414, Publisher: Packt Publishing) 8)"Learning GraphQL: Declarative Data Fetching for Modern Web Apps" by Eve Porcello and Alex Banks (ISBN: 978-1492030713, Publisher: O'Reilly Media) 9)"GraphQL API Design" by Matthew Mahoney (ISBN: 978-1484242698, Publisher: Apress) 10)"Fullstack GraphQL Applications with GRANDstack: Modernize Legacy Systems and Build Scalable GraphQL APIs with GraphQL, React, Apollo, and Neo4j" by William Lyon (ISBN: 978-1492090909, Publisher: O'Reilly Media) 11)"Testing JavaScript Applications: A Comprehensive Guide to the Jest Testing Framework" by Lucas da Costa and Felipe N. Moura (ISBN: 978-1484250464, Publisher: Apress) 12)"Nginx: From Beginner to Pro" by Rahul Soni and Dipankar Sarkar (ISBN: 978-1484216576, Publisher: Apress) 13)"Mastering Nginx: A complete guide to Nginx setup, configuration, and deployment" by Dimitri Aivaliotis and Tim Butler (ISBN: 978-1786466174, Publisher: Packt Publishing)
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Course Code: 904-02
Course Title: Cloud storage interaction using Android Applications

Course Code	904 – 02
Course Title	Cloud storage interaction using Android application
Credit	04
Category of Course	Major Course (Student will select any one course among under course code 904).
Level of Course	600 – 699
Teaching Per Week	4 Hrs (2 hours of theory + 4 hours of Leb sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review/Revision	-
Implementation Year	A.Y. 2024-2025
Purpose of Course	The purpose is to provide a secure and reliable way for Android applications to store and retrieve data from a remote server over the internet. Cloud storage allows for seamless synchronization and collaboration between different devices and users. With cloud storage, developers can provide users with access to their data from anywhere, on any device, and at any time.
Course Objective	• To introduce students to the concepts of cloud storage and how it is used in Android application development. • To provide an overview of different cloud storage services available for Android development, such as Google Cloud Storage, Amazon S3, and Microsoft Azure. • To teach students how to use APIs provided by cloud storage services in their Android applications, including methods for uploading and downloading files, and managing data. • To understand how to manage data in cloud storage, including organizing data, applying access controls, and managing data versions. • To integrate cloud storage with mobile applications, including how to authenticate users, store user data securely, and manage data synchronization between devices.
Pre-requisite	Fundamental of Programming, Basic concepts of Android Application Development.
Course Outcome	CO1: Understand the concepts of cloud storage and how it is used in Android application development. Identify and use different cloud storage services available for Android development, such as Google Cloud Storage, Amazon S3, and Microsoft Azure. CO2: Implement cloud storage APIs in Android applications, including methods for uploading and downloading files, and managing data. CO3: Manage data in cloud storage, including organizing data, applying access controls, and managing data versions.

	Integrate cloud storage with mobile applications, including authenticating users, storing user data securely, and managing data synchronization between devices CO4: Learn how to upload, download and manage files in Google Cloud Storage using Android applications. Understand the differences between various storage options available in Google Cloud Storage and choose the best option for different types of data. CO5: Design and implement secure and scalable cloud storage solutions for Android applications. Implement cloud storage in Android applications using Google Cloud Platform.
Mapping between COs and PSOs	
Course Content	Unit 1 1.1. Overview of cloud storage 1.1.1. Comparison of other storage solution with google cloud 1.1.2. Benefits of using Google cloud storage for mobile application data 1.2. Network infrastructure and architecture of Google cloud 1.3. IaaS, Paas and Saas 1.4. Signing with Google cloud 1.5. Google Cloud Resources 1.6. Google cloud Identity Unit 2 2.1. Google cloud console 2.2. Installing and configuring cloud SDK 2.3. Cloud shell 2.4. Google cloud API 2.5. Cloud Console Mobile app 2.6. Google cloud compute 2.6.1. Exploring IaaS with cloud compute 2.6.2. Exploring PaaS with App Engine 2.7. Event driven programs with cloud function Unit 3 3.1. Cloud Storage options 3.2. Structured and Unstructured storage 3.3. Unstructured storage using Cloud Storage 3.4. SQL Managed Services 3.5. Exploring cloud SQL 3.6. NoSQL Managed Services 3.7. NoSQL Document storage on Cloud Unit 4 4.1. Introduction to API 4.2. Purpose of API 4.3. Using Apigee 4.4. Using Google Cloud storage API to upload and Download data

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

	Unit 5 5.1. Accessing Cloud Storage using the Android Storage Access Framework 5.2. The Storage Access 5.3. Working with the Storage Access 5.4. Deploying application with google cloud 5.5. Handling Intent Results
Reference Books	1) Beginning Mobile Application Development in the Cloud, Author: Richard Rodger, Publisher: Wrox, ISBN: 978-1118034699 2) Exploring Cloud Computing, Author: Andreas Wittig and Michael Wittig, Publisher: Manning, ISBN: 978-1617294877 3) Visualizing Google Cloud: 101 Illustrated References for Cloud Engineers and Architects, Author: Priyanka Vergadia, Publisher: Wiley, ISBN: 978-1119816324 4) Android Application Development (With Kitkat Support), Author: Pradeep Kothari, Publisher:DreamTech Press.,ISBN:978-9351194095 5) Android Studio 3.0 Development Essentials: Android 8 Edition Author – Neil Smyth, Publisher: Payload Media, ISBN – 13: 978 – 1977540096 6) Google Cloud Platform All-In-One Guide, Author: Praveen Kukreti, Publisher: BPB Publication, ISBN: 978-9355513328 7) Google Cloud Platform for Architects, Author: Vitthal Srinivasan, Janani Ravi and Judy Raj, Publication: Packt, ISBN: 9781788834308 8) Professional Android – fourth Edition, Author: Reto Meier, Ian Lake, Publisher: Wrox, ISBN – 13:978-1118949528
Teaching Methodology	Class Work, Discussion, Self-study, Seminars and Assignments
Evaluation Method	30% Internal Assessment 70% External Assessment

M.Sc.(Computer Application) Semester-3
Course Code: 905
Course Title: Big Data and Hadoop

Course Code	905
Course Title	Big Data and Hadoop
Credit	04
Category of Course	Major Course
Level of Course	600-699 (Foundation / Introductory)
Teaching per Week	4 Hrs (2 hours of theory + 4 hours of Lab sessions)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	This course is designed to provide students with an understanding of Big Data and Hadoop technology. The course will cover the fundamentals of Big Data and Hadoop, including Hadoop Architecture, Hadoop Distributed File System (HDFS), MapReduce programming model, and Hadoop Ecosystem components. Students will also learn how to install, configure and manage Hadoop clusters.
Course Objective	- Explain the concept of Big Data and its challenges. - Understand Hadoop Architecture and its components. - Configure and manage Hadoop clusters. - Understand Hadoop Distributed File System (HDFS) and its components. - Develop and execute MapReduce programs on Hadoop clusters. - Understand and work with Hadoop Ecosystem components such as Hive, Pig, and HBase. - Perform data analysis using Hive and Pig. - Understand the basics of data ingestion, data processing, and data visualization.
Pre-requisite	Understanding of computer programming, Familiarity with Linux/Unix commands and shell scripting, Understanding of database concepts and SQL.
Course outcome	**CO1:** Students should be able to demonstrate knowledge of big data and its underlying technologies, including Hadoop, MapReduce, and Hive. This includes understanding the various components of a Hadoop cluster, the MapReduce programming model, and the role of Hive in processing large datasets. **CO2:** Ability to design and implement solutions for processing and analysing large datasets using Hadoop, MapReduce, and Hive. This includes designing data models, writing MapReduce programs, and creating Hive queries to analyse data. **CO3:** Understanding of Data Warehousing concepts and techniques, including data modelling, data integration, and data aggregation. They should also be able to apply these concepts in the context of big data solutions using Hadoop, MapReduce, and Hive. **CO4:** Ability to Work with Big Data Tools and Technologies including Hadoop Distributed File System (HDFS), Hadoop YARN, and Apache Pig. They should also be familiar with tools for data analysis and visualization, such as Apache Spark and Tableau. **CO5:** Collaboration and Communication Skills with others in a team environment, and communicate their ideas and solutions clearly and effectively to technical and non-technical stakeholders.

Mapping between COs and PSOs

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

Course Content

Unit 1: Introduction to Big Data and Hadoop

1.1 Introduction to Big Data
- 1.1.1 Overview of Big Data
- 1.1.2 Characteristics of Big Data
- 1.1.3 Big Data Technologies and Tool

1.2 Hadoop Architecture
1.3 Hadoop Ecosystem Components
1.4 Understanding Hadoop Distributed File System (HDFS)
1.5 Hadoop Installation and Configuration
- 1.5.1 Installing and Configuring Hadoop on a Cluster
- 1.5.2 Configuring Hadoop Parameters
- 1.5.3 Monitoring Hadoop Cluster

Unit 2: Data Ingestion, Processing and Visualization

2.1 Introduction of NoSQL
- 2.1.1 Concepts, purpose and types of NoSQL databases
- 2.1.2 Advantages of NoSQL
- 2.1.3 Comparison of SQL vs NoSQL

2.2 Introduction of MongoDB
- 2.2.1 Concepts and architecture
- 2.2.2 Using JSON
- 2.2.3 Creating or generating a unique key
- 2.2.4 Support for Dynamic Queries
  - 2.2.4.1 Storing Binary Data, Replication, Shading
  - 2.2.4.2 Updating information in –place
  - 2.2.4.3 Data types in Mongo DB
- 2.2.5 MongoDB Query Language

Unit 3: MapReduce Programming Model

3.1 Understanding MapReduce Programming Model
3.2 MapReduce Programming Basics
- 3.2.1 Input and Output Formats in MapReduce
- 3.2.2 Mapper and Reducer Functions
- 3.2.3 Combiner Functions
- 3.2.4 Partitioners in MapReduce

3.3 Advanced MapReduce Programming Concepts
- 3.3.1 Hadoop Streaming
- 3.3.2 Multi-Stage MapReduce Jobs
- 3.3.3 MapReduce Joins, Counters, Optimization

3.4 Data Locality in MapReduce
- 3.4.1 MapReduce Compression
- 3.4.2 MapReduce Sort and Shuffle

Unit 4: Hadoop Ecosystem Components:

4.1 Hive Architecture
- 4.1.1 Comparison with Traditional RDBMS

	4.1.2 HiveQL Data Types 4.1.3 Working with Tables and Databases 4.1.4 HiveQL Operators and Functions 4.2 Data Loading and Manipulation 4.2.1 Loading Data into Hive Tables 4.2.2 Creating and Managing Partitioned Tables 4.2.3 Altering and Dropping Tables 4.2.4 Data Manipulation using HiveQL **Unit 5: Data Analysis using Hive** 5.1 Hive Optimization Techniques 5.1.1 Hive Query Optimization 5.1.2 Partitioning and Bucketing 5.1.3 Indexing in Hive 5.1.4 Joins and Subqueries Optimization 5.2 User-Defined Functions (UDFs) 5.2.1 Overview of UDFs in Hive 5.2.2 Developing and Using UDFs in Hive 5.2.3 Hive Transactions and Concurrency 5.3 Concurrency Control in Hive 5.4 Locking in Hive [All Units carry Equal Weightage]
Reference Books	1)"React: Up & Running: Building Web Applications" by Stoyan Stefanov and Kirupa Chinnathambi (ISBN: 978-1491931820, Publisher: O'Reilly Media) 2)"Learning React: A Hands-On Guide to Building Web Applications Using React and Redux" by Kirupa Chinnathambi (ISBN: 978-0134843551, Publisher: Addison-Wesley Professional) 3)"React Design Patterns and Best Practices" by Michele Bertoli (ISBN: 978-1786464538, Publisher: Packt Publishing) 4)"React Cookbook: Over 66 hands-on recipes that cover UI development, animations, component architecture, routing, databases, testing, and debugging with React" by Carlos Santana Roldán (ISBN: 978-1783980727, Publisher: Packt Publishing) 5)"Full-Stack React Projects: Modern web development using React 16, Node, Express, and MongoDB" by Shama Hoque (ISBN: 978-1788835534, Publisher: Packt Publishing) 6)"Learning Redux" by Daniel Bugl (ISBN: 978-1786462398, Publisher: Packt Publishing) 7)"Hands-On Redux for React Native: A Practical Guide to Building Real-Time, Scalable Mobile Applications" by Spencer Carli (ISBN: 978-1788997414, Publisher: Packt Publishing) 8)"Learning GraphQL: Declarative Data Fetching for Modern Web Apps" by Eve Porcello and Alex Banks (ISBN: 978-1492030713, Publisher: O'Reilly Media) 9)"GraphQL API Design" by Matthew Mahoney (ISBN: 978-1484242698, Publisher: Apress) 10)"Fullstack GraphQL Applications with GRANDstack: Modernize Legacy Systems and Build Scalable GraphQL APIs with GraphQL, React, Apollo, and Neo4j" by William Lyon (ISBN: 978-1492090909, Publisher: O'Reilly Media)

	11)"Testing JavaScript Applications: A Comprehensive Guide to the Jest Testing Framework" by Lucas da Costa and Felipe N. Moura (ISBN: 978-1484250464, Publisher: Apress) 12)"Nginx: From Beginner to Pro" by Rahul Soni and Dipankar Sarkar (ISBN: 978-1484216576, Publisher: Apress) 13)"Mastering Nginx: A complete guide to Nginx setup, configuration, and deployment" by Dimitri Aivaliotis and Tim Butler (ISBN: 978-1786466174, Publisher: Packt Publishing)
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Course Code: 906
Course Title: Practical

Course Code	906
Course Title	Practical
Credit	04
Category of Course	Major Course
Level of Course	600-699 (Advance Level Technical)
Teaching per Week	22 hours of Lab work per week (8 hours Supervised mode + 14 Hours Unsupervised mode)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	To develop a comprehensive understanding of Robotic Process Automation (RPA) and its application in automating repetitive tasks, improving efficiency, and reducing human error in data processing and analysis. Develop proficiency in data analytics techniques to extract insights and patterns from large datasets, enabling informed decision-making and strategic planning. Acquire skills in data visualization tools and techniques to effectively communicate complex information and trends, aiding in data-driven storytelling and presentation. Gain hands-on experience in managing and analyzing big data using technologies like Hadoop and Spark, enabling efficient processing of large volumes of structured and unstructured data. To apply the knowledge and skills acquired in data analytics, data visualization, big data, and RPA to real-world projects, demonstrating the ability to tackle complex data-related challenges and deliver actionable insights.
Course Objective	- Develop a comprehensive understanding of Robotic Process Automation (RPA) technologies and their applications, including the ability to identify suitable processes for automation and implement RPA solutions. - Acquire hands-on experience with Big Data technologies, such as Hadoop and Spark, to effectively manage, process, and analyze large volumes of structured and unstructured data. - Develop proficiency in using data analytics tools and techniques to extract meaningful insights from complex datasets, including data cleaning, visualization, and statistical analysis. - Enhance problem-solving and critical thinking skills by applying RPA, Big Data, and Data Analytics concepts to real-world scenarios, including identifying business opportunities, optimizing processes, and making data-driven decisions. - Foster collaboration and teamwork by engaging in practical projects that require the integration of RPA, Big Data, and Data Analytics, working effectively in multidisciplinary teams to achieve project goals and deliver actionable outcomes.
Pre-requisite	Practical knowledge of computer languages like Python, C and operating systems like Unix/Linux. Also having basic knowledge about cloud.
Course outcome	Develop a solid understanding of the concepts, principles, and applications of Robotic Process Automation (RPA), Big Data, and Data Visualization. Apply RPA techniques to automate repetitive tasks, streamline processes, and improve operational efficiency in various domains. Utilize Big Data technologies and analytics techniques to manage, process, and derive valuable insights from large and complex datasets.

<table>
<tr><td></td><td>Create visually compelling and meaningful data visualizations to effectively communicate data-driven insights and patterns.
Demonstrate proficiency in using industry-leading RPA tools, Big Data platforms, and data visualization software.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><td></td><td>PS01</td><td>PS02</td><td>PS03</td><td>PS04</td><td>PSO5</td><td>PS06</td></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Course Content</td><td>Practical work is based on Course Code: 901, 902 and 905. The practical work will be carried out by the students during the computer lab allocated in supervised and un-supervised mode as specified.</td></tr>
<tr><td>Teaching Methodology</td><td>Class Work, Discussion, Self-Study, Seminars and/or Assignments</td></tr>
<tr><td>Evaluation Method</td><td>30% Internal assessment.
70% External assessment.
1) Internal Assessment : The internal examination will be carried out at institute/college level. Duration of internal examination will be of five hours consists of the course code 901,902,905. Purpose of the evaluation is to assess the understanding and proficiency regarding the tools, technologies and software used during the practical sessions throughout the semester. Internal marks will be evaluated based on five criteria. : (i)Practical work carried out during whole semester (Practical Journals)(15% weightage) (ii) Attendance and active participation during the semester and active involvement during allocated lab. hours (10%) (iii) Problem solving capabilities (35%) (iv) Understanding about the problems and solution oriented approach (30%) (v) viva-voce (10%).

2) External Assessment : The external examination will be carried out at institute/college by examiner panel. Duration of practical examination will be of five hours consists of the course code 901,902,905. Purpose of the evaluation is to assess the understanding and proficiency regarding the tools, technologies and software used during the practical sessions throughout the semester. Assessment of the examination will be based on following five criteria. : (i) Practical Journals (15% weightage) (ii) Attendance and active participation during the semester and active involvement during allocated lab. hours (10% weightage) (iii) Problem solving capabilities (35% weightage) (iv) Understanding about the problems and solution oriented approach (30%) (v) viva-voce (10% weightage).</td></tr>
</table>

M.Sc.(Computer Application) Semester-3
Course Code: 907
Course Title: Project

Course Code	907
Course Title	Project
Credit	08
Category of Course	Major Course
Level of Course	600-699 (Advance Level Technical)
Teaching per Week	16 hours of Lab work. (Unsupervised mode)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Enhance problem-solving and critical-thinking skills by addressing challenges and implementing innovative solutions in real-world scenarios. Acquire practical knowledge in testing, debugging, and optimizing Node.js, React.js, and Redux applications for performance and scalability. Prepare students for industry demands by equipping them with the skills and experience necessary to develop robust and modern web applications using Node.js, React.js, and Redux. To address and meet the challenges related to data synchronization, scalability, and security in cloud-based Android applications. Acquire practical knowledge in testing, debugging, and optimizing Android applications that utilize cloud services and Socket.io. Prepare students for industry demands by equipping them with the skills and experience necessary to develop robust and interactive Android applications that leverage cloud services and real-time communication capabilities.
Course Objective	1) To gain practical experience in designing and developing web applications or Android-based applications from start to finish, including requirements gathering, user interface design, implementation, and deployment. 2) To apply programming languages, frameworks, and tools relevant to web development (such as HTML, CSS, JavaScript, and popular frameworks like React.js, Angular, or Django) or Android app development (Java, Kotlin) to create robust and functional applications. 3) To develop proficiency in utilizing databases and backend technologies to store and retrieve data, ensuring seamless functionality and data integrity in web or Android applications. 4) To enhance problem-solving skills by identifying and implementing appropriate solutions to overcome challenges encountered during the development process, such as handling user input, managing data, and ensuring application security. 5) Collaborate effectively within teams to deliver projects, demonstrating effective communication, teamwork, and project management skills, as well as the ability to meet project deadlines and deliver high-quality outcomes.
Pre-requisite	Knowledge of project development life cycle and collaborating various tools, software, API, frameworks and integrate them.
Course outcome	These objectives aim to equip students with practical skills, knowledge, and experience in web development and Android application development,

	enabling them to confidently apply their learning in real-world scenarios and prepare them for future career opportunities in the respective domains.
Mapping between Cos with PSOs	

	PS01	PS02	PS03	PS04	PSO5	PS06
CO1						
CO2						
CO3						
CO4						
CO5						

Course Content	During the semester, the student will work on project development based on selected electives among the course code-903 and course code-904. The students will work in a team or individual to develop a reasonable size project using API, frameworks, tools , applications, databases and software that they learn so far, preferably as part-time project. In house project will be admissible subject to prior consent from the concerned faculty from the institute.
Reference Books	1) "The Pragmatic Programmer: Your Journey to Mastery" by Andrew Hunt and David Thomas (Addison-Wesley Professional) 2) "Clean Code: A Handbook of Agile Software Craftsmanship" by Robert C. Martin (Prentice Hall) 3) "Web Development with Node and Express: Leveraging the JavaScript Stack" by Ethan Brown (O'Reilly Media) 4) "JavaScript: The Good Parts" by Douglas Crockford (O'Reilly Media) 5) "Head First Design Patterns" by Eric Freeman, Elisabeth Robson, Bert Bates, and Kathy Sierra (O'Reilly Media) 6) "Learning React: Functional Web Development with React and Redux" by Alex Banks and Eve Porcello (O'Reilly Media) 7) "Android Programming: The Big Nerd Ranch Guide" by Bill Phillips and Chris Stewart (Big Nerd Ranch Guides) 8) "Effective Java" by Joshua Bloch (Addison-Wesley Professional) 9) "Android Studio 4.1 Development Essentials – Kotlin Edition" by Neil Smyth (eBookFrenzy) 10) "Building Microservices: Designing Fine-Grained Systems" by Sam Newman (O'Reilly Media)
Teaching Methodology	Industrial visit, Discussion, Self-Study, Seminars and/or Assignments - The students must prepare documentation of the project completed as per the guidelines given by the institute. - At the end of the semester, the students have to submit the project reports in bounded form along with the softcopy to the institution. - Project completion certificate issued by the institute is mandatory for appearing in project presentation and viva - Voce. - The project presentation and viva-Voce will be conducted as per the university exam schedule. Workload for teachers : Guiding five students as an internal guide for project work will considered as one hour workload per week.
Evaluation Method	30% Internal assessment. 70% External assessment. 1) Internal Assessment : The internal examination will be carried out at institute/college level based on the project developed using course code 903 and 904. Purpose of the evaluation is to assess the understanding and proficiency regarding the tools, technologies and software used during the project development and active involvement throughout the semester. Internal marks will be evaluated based on five criteria. : (i)Project work

	carried out during whole semester (Project Report)(15% weightage) (ii) Attendance and active participation during the semester and active involvement during allocated lab. hours (10%) (iii) Problem solving capabilities (25%) (iv) Understanding about the problems and solution oriented approach (30%) (v) viva-voce/Presentation (20%). 2) External Assessment : The external examination will be carried out at institute/college by examiner panel. The panel will consists of three examiners including (i) one local examiner (from same college/institute), (ii) one external examiner preferably from institute offering M.Sc.(C.A.) program/ from institute having minimum of fifteen years of teaching experience and one expert from software industry having minimum five years of experience in software industry. Duration of project examination will be based on as per the actual need of the presentation and viva-voce. Purpose of the evaluation is to assess the understanding and proficiency regarding the tools, technologies and software used during the practical sessions for the purpose of project development throughout the semester. Assessment of the examination will be based on following five criteria. : (i) Project report (softcopy) (15% weightage) (ii) Attendance and active participation during the semester and active involvement during allocated lab. Hours for development of project. (10% weightage) (iii) Problem solving capabilities (35% weightage) (iv) Understanding about the problems and solution oriented approach (30%) (v) viva-voce (10% weightage).

Veer Narmad South Gujarat University, Surat

Program Structure: M.Sc.(Computer Application) (SEM –4)

(w.e.f. Academic Year June, 2024-2025)

Masters in Computer Application (M.Sc. (C.A.)) –Post Graduate Program

SEMESTER – 4

Course Code	Course Title	Course Category	Level of Course	Course Credits	Teaching Hours/week	
				Th.+Pra.	Theory	Practical/ Fieldwork /Project/ Internship
1000	Value Addition Course [2-credit university approved certificate course]	Value Addition Course*	600-699 Advance level Technical	2	2	0
1001	Project (Full time Project at Industry/corporate)	Major course (Skill Enhancement / Internship)	600-699	30	-	Industrial Project full time.
Other Activities	The student is expected to participate in activities related to National Service Scheme (NCC), National Cadet Corps (NCC), adult education/literacy initiatives, mentoring school students, Elderly literacy program/ Environment preservation activities and other similar activities.			-	-	-
Total				32	10	30

Course Code	Course Title	Course Credit	University Exam Type	Exam Duration	External Marks	Internal Marks	Total Marks
1000	Value Addition Course[#]	2	As per the Nature of course*	-	70	30[#]	100
1001	Project (Full time Project at Industry/corporate)	30	Theory (Descriptive ,Short Questions and MCQ)	3 Hours	420	180	600
Total		32			490	210	700

Minimum Passing Score : 40% in each individual head.

*The external and internal evaluation (For course code: 1000) will be carried out by the institution/college based on the nature of course. The evaluation pattern may include any or combination of (i) MCQ exam (ii) Viva-Voce (iii) Presentation (iv) Project Demonstration.

For Project:

- The journal should be certified by the concerned faculty and by the Head of the Department, failing which the student should not be allowed to appear for External Practical Examination. Student will submit softcopy of Project duly certified by the internal guide.
- The students must carry out a full-time industrial project during the semester based on technical knowledge acquired during past semesters. Students are expected to analyse the project requirement, design, develop, code, test and deploy the project. The work will be carried out by the students in un-supervised mode. Minimum 8 hours per week should be allocated to the student for working on the project in un-supervised mode. Students who pursue the full time project at corporate/industry/software organization are required to report to the project supervisor at the institute/college once in a week and submit the progress report.

[#]**Value Addition Course:** Student will opt any one course from the given choices by the institute/college of nature Value Addition Course from available pool of courses recognized by the University.

***Certificate Course :** For Certificate courses, the students will enrol for the course from the given university approved list of certificate courses offered by the respective college/department. The student will select and enrol separately for any of the offered list of courses at college/department/institute and obtain respective credits. The institute will evaluate the performance (preferably continuous evolution) as per the SOP of certificate courses and on successfully completion of the course, the student will be eligible to obtain respective credits for the course. These credits will be considered and reflect in student's mark-sheet as well as in ABC(Academic Bank of Credit). These courses are mandatory and student is required to obtain the specified credits in process to acquire the certificate/diploma/degree. [The student is required to pay separately for these courses as prescribed by the college.]

 # Marks: The scale on which the students will be evaluated for the course. The evaluation methodology will be continuous evaluation and the score obtained will reflect in mark-sheet.

M.Sc.(Computer Application) Semester-4
Course Code: 1000
Course Title: Value Addition Course

Course Code	1000
Course Title	Value Addition Course
Credit	2
Category of Course	Value Addition Course
Level of Course	600-699 (Foundation / Introductory)
Teaching per Week	2 Hrs (Any or Combination of Theory/Practical/Fieldwork/Internship/Project)
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Student will select minimum one 2-credit course of category value addition out of the choices given by the college/institute. It will be mandatory for the student to opt minimum one 2-credit Value Addition Course out of the list of offered courses recognised by the University and offered by the college/institution. Student can enhance the knowledge in the selected field by obtaining higher degree of knowledge in the area.
Course Objective	Obtaining knowledge in all or any of the components/fields like (i) Understanding India (ii) Environmental Science/Education (iii) Digital/Technological solutions (iv) Health & Wellness, Yoga education, sports, and fitness are essential for holistic development and (v) Indian Knowledge System (vi) Artificial intelligence and Robotics. The course components should be among these six categories/fields and as per the Curriculum and Credit Framework for Undergraduate Programmes of the UGC. The purpose is to impart knowledge and understand the necessities of these aspects in life to make the healthy society and nation. It help in development of a dedicated and responsible citizen of the country by adding value to the life.
Pre-requisite	No prior knowledge in the field is essential.
Course outcome	CO1: Student select the area of Value addition as per his/her interest. The choices will be given by the institute/department. CO2: The student acquire basic and fundamental level of knowledge in the field that the student opted. CO3: Understand the insight of the area and possibility of to explore more in the field. CO4: Understand effective representation of problems, solutions and insights of the challenges and problems of the peer subject relevant to value addition. CO5: Learn to upskill and upgrade the knowledge in the area of selected subject.
Course Content and Implementation road-map.	(i) The university has categorised and prepared the list of the courses that can be offered as Value Addition Course. (ii) The institute/college/department can design and implement skill enhancement course by getting approval from the relevant apex body of the university considering the SOP of the certificate course policies of the University. (iii) The institutes/college/departments can select more than one course out of the given sets of courses and offer them to their students.

	(iv) The students can select any of the courses offered by the institute/college/department from the given choices and enrol for the course. (v) The institute/college/department will arrange appropriate resource person(s) for the course. (vi) This is an audit course, hence the evaluation will be taken place at the college/institute/department based on the nature of the course. (vii) The institute/college/department will assess the student based on the nature of the course. The student will be granted the credits on successful completion of the course.
Reference Books	- The reference materials and books will be decided by the Institutes/Colleges/Departments based on the selected Courses. - Minimum five copies of relevant topics are recommended to keep in the library.
Teaching Methodology	Class Work/ Discussion/ Self-Study/ Seminars/ field works/ practical training/ field work and/or Assignments.
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-4
Course Code: 1001
Course Title: Project

Course Code	1001
Course Title	Project
Credit	30
Category of Course	Industrial/corporate full time project
Level of Course	600-699 (Foundation / Introductory)
Project Duration	15 (Including project work, examination, preparation, internal/external presentation. etc.)
Review / Revision	-
Implementation Year:	A.Y. 2024-2025
Purpose of Course	Student will work on industrial/corporate project and implement the technical knowledge being a part of the project team. Student can enhance the knowledge in the selected field by obtaining higher degree of knowledge in the area.
	- The students must prepare documentation of the project completed as per the guidelines given by the institute. - At the end of the semester, the students have to submit the project reports in bounded form along with the softcopy to the institution. - Project completion certificate issued by the institute is mandatory for appearing in project presentation and viva - Voce. - The project presentation and viva-Voce will be conducted as per the university exam schedule. - Workload: Guiding five students as an internal guide for project work will considered as one hour workload per week.
Evaluation Method	30% Internal assessment. 70% External assessment.

M.Sc.(Computer Application) Semester-3
Theory Exam Question Paper Style

Course Code	Course Code 901 to 905 (All Theory External Exams)
Question Paper suggestive marks distribution:	1) 20% - Objective (consists of 2 marks short questions) 2) 20% - MCQ (Consists of MCQ questions of 1 marks each) 3) 20% - Understanding and technical skills about the subject. (Four Questions of five marks each.) 4) 40% - Descriptive questions including short-notes, long questions, case studies, problem solving abilities assessment etc.