Final NEW

(7)

ದಾವಣಗೆರೆ ವಿಶ್ವವಿದ್ಯಾನಿಲಯ

DAVANGERE UNIVERSITY

nirf

ದಾವಣಗೆರೆ ವಿಶ್ವವಿದ್ಯಾಲಯವು ಇಂಡಿಯನ್ ಸಾರಿನಲ್ಲಿ 'ರಾಜ್ಯ ಸಾರ್ವಜನಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ' ವಿಭಾಗದಲ್ಲಿ NIRF ಶ್ರೇಯಾಂಕ 51 ರಿಂದ 100 ರೊಳಗೆ ಸ್ಥಾನದಲ್ಲಿದೆ

Ranked at #51-100 in the 'State Public University' Category by the NIRF Ranking 2024

NAAC ನಿಂದ "B+" ಗ್ರೇಡ್ನೊಂದಿಗೆ ಮಾನ್ಯತೆ ಪಡೆದಿದೆ

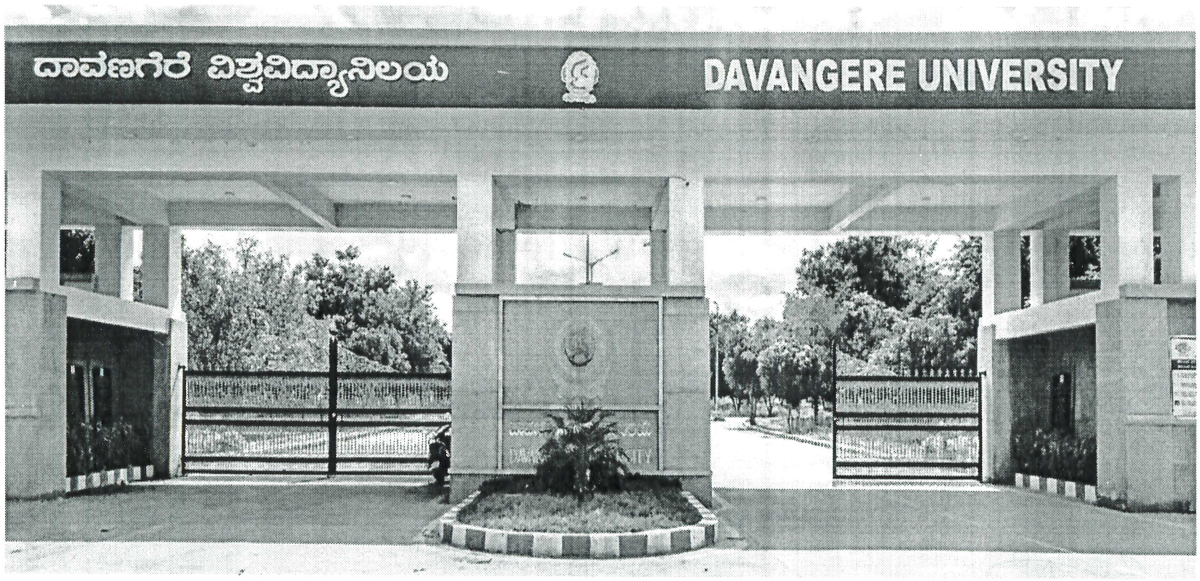
Accredited by NAAC with "B+" Grade

Syllabus for **Bachelor of Computer Science (B.Sc.)**

5th and 6th Semester

[As per SEP (State Education Policy): 2024-25]

WEF: 2026-27 & onwards



DEPARTMENT OF STUDIES IN COMPUTER SCIENCE, DAVANAGERE UNIVERSITY, DAVANAGERE – 577007

BOS Chairman
Dept. of Computer Science
Davangere University
Shivagangotri, Davangere

Prof. M. Govindappa
Dean-Science & Technology
Davangere University
Shivagangotri, Davangere-577007

Registrar
Davangere University
Shivagangotri, Davangere

DAVANGERE UNIVERSITY

Department of Studies in Computer Science

Shivagangotri, Davangere – 577 007

Bachelor of Science (B.Sc.) Semester Scheme of SEP

Curriculum Structure for Undergraduate Programme for 2024-25 batches onwards

Sem	Title of the Paper	Teaching Hours/ week	Semester End Exam	Internal Assessment	Total Marks	Credits	Duration of the Exam
V	Computer Network and Mobile Computing	4	80	20	100	3	3
	Python Programming	4	80	20	100	3	3
	Python Programming Lab	4	40	10	50	2	3
	Elementary Research Methodology	2	40	10	50	2	2
	Total	**14**	**240**	**60**	**300**	**10**	-
VI	Operating System Concepts	4	80	20	100	3	3
	Web Technologies	4	80	20	100	3	3
	Web Technologies Lab	4	40	10	50	2	3
	Minor Project	4	40	10	50	2	3
	Total	**16**	**240**	**60**	**300**	**10**	-

Note: Question paper pattern and scheme of valuation of theory and practical examination is same as that of previous year (I-VI Semester).

BOS Chairman
Dept. of Computer Science
Davangere University
Shivagangotri, Davangere

Prof. M. Govindappa
Dean-Science & Technology
Davangere University
Shivagangotri, Davangere-577007

Registrar
Davangere University
Shivagangotri, Davangere

B.Sc. V Semester

Course Title: Computer Network and Mobile Computing	
Course Code: B.Sc.	**Total Teaching Hours: 56**
IA Marks: 20	**Teaching Hours /Week: 04**
Exam Marks: 80	**Examination Hours: 03**
Course Credits: 03	

Course Learning Objectives:

1. Understand the fundamental concepts of data communication and computer networks.
2. Explain layered network architectures and reference models.
3. Describe data transmission techniques and data link layer mechanisms.
4. Analyze network, transport, and application layer protocols.

Course Outcomes:

1. Understand basic concepts of computer networking
2. Explain network models and protocols
3. Analyze data transmission and routing techniques
4. Understand transport and application layer protocols

Unit-I	**14Hrs**

Introduction to Data Communication – Data, Information, Signals, Data flow. Computer Networks – Definition, objectives, advantages, and applications. Types of Networks, Network Topologies, Transmission Modes, Network Devices – Hub, Switch, Router, Bridge, Repeater, Gateway, Modem. Layered Architecture – Need and advantages. OSI Reference Model, TCP/IP Model. OSI vs TCP/IP models.

Unit-II	**14 Hrs.**

Network Addressing – MAC address, IP address, Port address. Introduction to IPv4 and IPv6. Transmission Media – Guided (Twisted pair, Coaxial cable, Optical Fiber) and Unguided (Radio waves, Microwave, Infrared). Analog and Digital Transmission. Switching Techniques – Circuit switching, Packet switching, Message switching. Data Link Layer – Functions and services. Error Detection and Correction – Parity, Checksum, CRC. Flow Control – Stop-and-Wait and Sliding Window protocols.

Unit-III	**14 Hrs.**

Routing Algorithms – Shortest path, Distance vector, Link state routing.
Congestion control – Causes and techniques. Transport Layer – Services and functions.
Transport protocols – UDP and TCP (features and comparison).
Application Layer – DNS, HTTP, FTP, SMTP, POP, Telnet.

Unit-IV	**14 Hrs.**

Introduction to Mobile Computing – Definition, characteristics, and applications. Mobile Computing Architecture, Wireless Communication Basics – Wireless transmission, challenges. Mobile Devices and Operating Systems – Smartphones, tablets. Mobile Networks – Cellular networks, Wi-Fi, Bluetooth. Mobile Computing Issues – Security, mobility management, energy constraints. Future trends in Mobile Computing.

Text Books:

1. Andrew S. Tanenbaum – *Computer Networks*, Pearson Education.
2. Behrouz A. Forouzan – *Data Communications and Networking*, McGraw-Hill.
3. Jochen Schiller – *Mobile Communications*, Pearson Education.

Reference:

1. "Data Communications and Computer Networks" — Stallings, William (Pearson)
2. "Computer Networking: A Top-Down Approach" — Kurose & Ross (Addison-Wesley)

B.Sc. V Semester

Course Title: Python Programming	
Course Code: B.Sc.	**Total Teaching Hours: 56**
IA Marks: 20	**Teaching Hours /Week: 04**
Exam Marks: 80	**Examination Hours: 03**
Course Credits: 03	

Course Learning Objectives:

1. Understand the basic syntax, semantics, and structure of Python programming language.
2. Develop Python programs using fundamental programming concepts like variables, data types, operators, and control flow.
3. Use functions, modules, and libraries to write reusable and modular Python code.
4. Work with data structures such as lists, tuples, dictionaries, and sets effectively.

Course Outcomes:

1. Write and execute Python programs to solve computational problems.
2. Demonstrate understanding of Python syntax and semantics including expressions, statements, and control structures.
3. Design and implement modular programs using functions and Python standard libraries.
4. Manipulate various Python data structures to organize and store data efficiently.

Unit-I	**14 Hrs.**

Introduction to Python: History of Python and Features, Installing Python and Setting up Environment, Writing and Running Python Scripts, Python Syntax, Indentation, and Comments, Variables, Data Types, and Type Conversion, Input and Output Operations. Control Structures: Conditional Statements (if, if-else, nested if), Looping Statements (for, while, nested loops), Loop Control Statements (break, continue, pass), Iterators and Generators.

Unit-II	**14 Hrs.**

Functions: Defining and Calling Functions, Function Arguments and Return Values, Lambda Functions, Recursive Functions, Built-in Modules (math, random, date time, etc.), Creating and Importing User-defined Modules. Strings: creating and sorting strings, accessing string characters, the str () function, Operations on strings: Concatenation, Comparison, Slicing and Joining, Traversing, String Methods.

Unit-III	**14 Hrs.**

Data Structures in Python: Lists: Operations, Slicing, Methods, Tuples: Characteristics, Operations, Sets: Operations, Methods, Dictionaries: Keys, Values, Methods, File Handling: Reading and Writing Files (Text and Binary), File Methods (open, close, read, write, append). Exception Handling: Introduction to Errors and Exceptions, Handling Exceptions using try, except, using else, finally Blocks.

Unit-IV	**14 Hrs**

NumPy (Numerical Python): Introduction to NumPy, Creating and Manipulating NumPy Arrays, Indexing, Slicing, and Iterating, Mathematical and Statistical Functions, Working with Multidimensional Arrays. Pandas (Data Analysis): Introduction to Pandas, Creating and Manipulating Data Frames and Series, Importing and Exporting Data (CSV, Excel), Matplotlib (Data Visualization): Introduction to Matplotlib, Creating Line Plots, Bar Charts, Histograms, Customizing Plots (Labels, Titles, Legends), Subplots and Multiple Graphs, Plotting with Pandas.

Text Book:

1. Python Programming: A Modern Approach- V.K. Jain (BPB Publications)
2. Programming in Python – Dr. Pooja Sharma (BPB Publications)
3. Core Python Programming – Dr. R. Nageswara Rao (Dreamtech Press)

Reference Books:

1. Python: The Complete Reference – Martin C. Brown (McGraw-Hill Education)
2. Python Crash Course – Eric Matthes (No Starch Press)
3. Learning Python – Mark Lutz (O'Reilly Media)

B.Sc. V Semester

Course Title: Elementary Research Methodology	
Course Code: B.Sc.	**Total Teaching Hours: 32**
IA Marks: 10	**Teaching Hours/Week: 02**
ExamMarks:40	**ExaminationHours:02**
Course Credits: 02	

Course Learning Objectives

1. Explain the meaning, objectives, significance, and types of research.
2. Describe the steps involved in the research process.
3. Define a research problem and justify the need for its clear formulation
4. Identify and differentiate various types of research designs
5. Explain the role of Intellectual Property Rights (IPR) in research and development.
6. Select appropriate methods for data collection based on research objectives
7. Prepare a structured research report with proper layout and format

Course Outcomes:

1. Students who complete this course will be able to understand and comprehend the basics in research methodology and apply them in research/ project work.
2. This course will help them to select an appropriate research design.
3. With the help of this course, students will be able to take up and implement a research project/ study.
4. The course will also enable them to collect the data, edit it properly and analyze it accordingly. Thus, it will facilitate students' prosperity in higher education.
5. The students will develop skills in qualitative and quantitative data analysis and presentation.
6. Students will be able to demonstrate the ability to choose methods appropriate to research objectives.

UNIT-I	08 Hrs.

Introduction-Meaning, Objectives, Types of Research, Significance of Research, Research Methods versus Methodology, Research and Scientific Method, Research Process, Criteria of Good Research. **Defining the Research Problem:** Research Problem, Selecting the Problem, Necessity of Defining the Problem, Technique Involved in Defining a Problem.

UNIT-II	08 Hrs.

Research Design- Meaning, need for Research Design, features of a Good Design, Important Concepts relating to Research Design. Cluster Analysis: Introduction, Clustering algorithms, **Scientific Body in Research:** Ethical and scientific issues in research. A brief idea about the DST, ICMR, CSIR and UGC. Role of IPR (Intellectual Property Rights) in Research and Development.

UNIT-III	08 Hrs.

Data Collection-Introduction, Experiments and surveys, Collection of Primary and Secondary Data, selection of appropriate method for data collection. **Data Preparation**: Data Preparation process, Missing values and Outliers, types of Analysis, Statistics in research.

UNIT-IV	08 Hrs.

Testing of Hypothesis- Hypothesis, Basic Concepts Concerning Testing the Hypotheses, Test Statistic and Critical region, critical value and Decision Rule, Procedure for Hypothesis Testing. **Interpretation and Report Writing**- Meaning of Interpretation, Technique of Interpretation, Precautions in Interpretation, Significance of Report Writing, Layout of the Research Report, Types of Reports, Oral Presentation, Mechanics of Writing a Research Report.

Text Books:

1. Research Methodology: A step-by-step guide for beginners, Ranjit Kumar, Sage publications.
2. Engineering Research Methodology a Practical Insight for Researchers by Dipankar Deb, RajeebDey, Valentina E. Balas.
3. Kothari C.K. (2004) 2/e, Research Methodology – Methods and Techniques (New Age International, New Delhi).
4. Montgomary, Douglas C. (2007) 5/e, Design and Analysis of Experiments (Wiley India).

Reference:

1. Montgomary, Douglas C. &Runger, George C. (2007) 3/e, Applied Statistics & probability for Engineers (Wiley India).
2. MLA (Modern Language Association) Handbook for Writers of Research Papers, 7th edition, 2009.
3. How to Write and Publish a Scientific Paper, Cambridge University Press.
4. Conducting Research Literature Reviews: From the Internet to Paper. Sage Publications.
5. Citation Analysis in Research Evaluation (Information Science and Knowledge Management) by Henk F. Moed.

B.Sc VI Semester

Course Title: Python Programming Lab	
Course Code: B.Sc.	**Total Teaching Hours: 56**
IA Marks : 10	**Teaching Hours /Week: 04**
Exam Marks: 40	**Examination Hours: 03**
Course Credits:02	

PART-A

1. Write a Python program to create a calculator.
2. Write Python programs to demonstrate the following: i. input () ii. print () iii. 'sep' (attribute) iv. 'end' (attribute) v. replacement Operator ({})
3. Demonstrate the following control transfer statements in Python with suitable examples.
   i. break ii. continue iii. Pass.
4. Program, using user-defined functions to find the area of a rectangle, square, circle and triangle by accepting suitable input parameters from the user.
5. Write a Python program to calculate the factorial of a given number using a recursive function.
6. Write a Python program to explore string functions.
7. Write a program to create a list with N elements. Find all unique elements in the list. If an element appears only once in the list, add it to the unique list.
8. Consider a tuple t1 = (1,2,5,7,9,2,4,6,8,10). Write a program to perform the following operations:
   a) Print half the values of the tuple in one line and the other half in the next line.
   b) Print another tuple whose values are even numbers in the given tuple.
   c) Concatenate a tuple t2 = (11,13,15) with t1.
   d) Return the maximum and minimum value from this tuple.
9. Write a function that takes a sentence as input from the user and calculates the frequency of each letter. Use a variable of dictionary type to maintain the count.
10. Write a Python program for Bubble Sort.

PART-B

1. Write a Python program to check if a given number is a prime number or not.
2. Write a program to create a text file and compute the number of characters, words and lines in a file
3. Write a program using user defined exception class that will ask the user to enter a number until he guesses a stored number correctly. To help them figure it out, a hint is provided whether their guess is greater than or less than the stored number using user-defined exceptions.
4. Create an array using NumPy and perform Operations on the array.
5. Create a Data Frame from excel sheet using Pandas and perform Operations on Data Frames.
6. Write a Python program to draw a line chart and a bar chart using Matplotlib.
7. Write a Python program to draw a histogram and a pie chart using Matplotlib.
8. Write a Python program to check if the given number is a palindrome or not.
9. Write a Python program to find the sum of n natural numbers.
10. WAP in Python for Binary Search.

B.Sc. VI Semester

Course Title: Operating System Concepts	
Course Code: B.Sc.	**Total Teaching Hours: 56**
IA Marks: 20	**Teaching Hours /Week: 04**
Exam Marks: 80	**Examination Hours: 03**
Course Credits: 03	

Course Learning Objectives:

1. Understand the fundamental concepts and components of operating systems.
2. Explain the role and functions of an operating system in managing hardware and software resources.
3. Describe different types of operating systems such as batch, time-sharing, distributed, and real-time systems.
4. Understand process management including process states, scheduling algorithms, and inter-process communication.

Course Outcomes:

1. Describe the structure, services, and types of operating systems. Illustrate the concepts of processes, threads, and CPU scheduling algorithms.
2. Implement solutions to synchronization problems and inter-process communication challenges.
3. Explain memory management schemes such as paging, segmentation, and virtual memory

Unit-I	14 Hrs

Introduction to Operating System: Definition, History and Examples of Operating System, Computer System Organization, Types of Operating System, Functions of Operating System, System Calls, Operating System Structure. Process Management: Process Concept- Process Definition, Process State, Process Control Block, Threads, Process Scheduling- Multiprogramming, Scheduling Queues, CPU Scheduling, Context Switch, Operations on Processes- Creation and Termination of Processes, Inter Process communication (IPC)- Definition and Need for Inter process Communication, IPC Implementation Methods- Shared Memory and Message Passing.

Unit-II	14 Hrs

Multithreaded Programming: Introduction to Threads, Types of Threads, Multithreading- Definition, Advantages, Multithreading Models, Thread Libraries, Threading Issues. CPU Scheduling: Basic concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-processor scheduling, Real-Time CPU Scheduling.

Unit-III	14 Hrs

Process Synchronization: Introduction, Race Condition, Critical Section problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization – Readers and Writers Problem. Deadlocks: System Model, Deadlocks Characterization, Methods of Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

Unit-IV	14 Hrs

Virtual Memory: Introduction to Virtual Memory, Demand Paging, Page Replacement, Page Replacement Algorithms, Allocation of frames, Thrashing. File System: File Concepts – Attributes, Operations and Types of Files, File System, File Access methods, Directory structure, Protection, File System Implementation – File System Structure, Allocation Methods, Free Space Management, Secondary Storage Structure, Protection.

Text Books:

1. Abraham Silberschatz and Peter Baer Galvin, Greg Gagne," Operating System Principles", Seventh edition.

Reference Books:

1. Milan Milonkovic, Operating System Concepts & Design, II Edition, McGRaw Hill 1992.
2. Stallings, Operating Systems, Pearson Edition.
3. Milan Milonkovic, Operating System Concepts & Design, II Edition, McGRaw Hill 1992.
4. Stallings, Operating Systems, Pearson Edition.
5. Tanenbaum, Operating System Concepts, Pearson Education
6. Nutt : Operating System, 3/e Pearson Education 2004

B.Sc. VI Semester

Course Title: Web Technologies	
Course Code: B.Sc.	**Total Teaching Hours: 56**
IA Marks: 20	**Teaching Hours /Week: 04**
Exam Marks: 80	**Examination Hours: 03**
Course Credits: 03	

Course Learning Objectives:

1. Understand the fundamentals of web technologies and the architecture of the World Wide Web.
2. Learn the basics of HTML, CSS, and JavaScript for building interactive web pages.
3. Develop skills to design responsive and visually appealing web pages using modern web standards.
4. Understand client-side and server-side scripting concepts.

Course Outcomes:

1. Understand basics of web technology.
2. Recognize the different client-side technologies and tools like HTML, CSS, JavaScript.
3. Learn Java Servlets and JDBC.
4. Web Technology for mobiles and understand web security.

Unit-I	**14 Hrs.**

Introduction to Internet, WWW and Web 2.0, Web browsers, Web protocols and Web servers, Web Design Principles and Web site structure, Client-server technologies, Client-side tools and technologies, Server-side Scripting, URL, MIME, search engine, HTTP protocol. Introduction to HTML, HTML5 Basics tags, formatting tags in HTML, HTML5 Page layout and Navigation concepts, Semantic Elements in HTML, List, types of list tags, table and form tags in HTML, multimedia basics, images, embedding audio and video clips on webpage.

Unit-II	**14 Hrs.**

XML Syntax, XML Tree, Elements, Attributes, Namespace, Parser, XSLT, DOM, DTD, Schema. Introduction to CSS, CSS syntax, CSS selectors, CSS Background, Cursor, CSS text fonts, CSS-List Tables, CSS Box Modeling, Display Positioning, Floats, JavaScript Data Types and Variables, JavaScript Operators, Conditional Statements, Looping Statements, JavaScript Functions, Number.

Unit-III	**14 Hrs.**

Strings, Arrays, Objects in JavaScript, Window and Frame objects, Event Handling in JavaScript, Exception Handling, Form Object and DOM, JSON, Browser Object Model. Common Gateway Interface (CGI), Lifecycle of Servlets, deploying a Servlet, The Servlets API, Reading Servlets parameters, reading initialization parameters, Handling HTTP Request & Responses, Using Cookies and sessions, connecting to a database using JDBC.

Unit-IV	**14 Hrs.**

Authentication Techniques, Design Flaws in Authentication, Implementation Flaws in Authentication, Securing Authentication, Path Traversal Attacks, Injecting into Interpreted Contexts, SQL Injection, NoSQL Injection, XPath Injection, LDAP Injection, XML Injection, HTTP Injection, Mail Service Injection.

Text Books:

1. Kogent Learning Solutions Inc, HTML5insimplesteps, Dream-tech Press, A beginner's guide to HTML, NCSA,14th May,2003
2. Murray, Tom/Lynchburg, Creating a Web Page and Web Site College, 2002

Reference:

1. Web Designing & Architecture-Educational Technology Centre, University of Buffalo
2. Steven M. Schafer HTML, XHTML, andCSSBible,5ed, Wiley India
3. John Duckett, Beginning HTML, XHTML, CSS, and JavaScript, Wiley India
4. Ian Pouncey, Richard York, Beginning CSS: Cascading Style Sheets for Web Design, Wiley India
5. Kogent Learning, Web Technologies: HTML, Javascript, Wiley India

B.Sc. VI Semester

Course Title: Web Technologies Lab	
Course Code: B.Sc.	Total Teaching Hours: 56
IA Marks : 10	Teaching Hours /Week: 04
Exam Marks: 40	Examination Hours: 03
Course Credits: 02	

PART-A

1. Design web pages for your college containing the college name and logo, a list using href, and list tags.
2. Create a class timetable using a table tag.
3. Create a Photo Gallery page for your college/department.
4. Write an HTML code to design a student registration form for your college admission.
5. Design a web page that includes multi-media data (Image, Audio, Video, GIFs etc).
6. Create a web page using a frame.
7. Write code in HTML to develop a webpage having two frames to divide the webpage into two equal rows and then divide the row into equal columns; fill each frame with a different background colour.
8. Write CSS code to use inline CSS to format your ID Card.
9. Using HTML and CSS, create a display of text called **"Hello India!"** on top of an image of the India Map using an overlay.
10. Create a feedback form to rate facilities in your college.

PART-B

1. Write a JavaScript program to perform basic arithmetic operations.
2. JavaScript program to check prime number.
3. JavaScript program to implement JavaScript object concept.
4. JavaScript program to reverse the given number.
5. JavaScript program to create array and inserting data into array.
6. JavaScript program to validate an email address.
7. JavaScript Program to Generate Fibonacci Series
8. Write a program for printing system date & time using **SERVLET**.
9. Write a server-side **SERVLET** program for accepting number from HTML file and display.
10. Write a program to create the **Life-Cycle Servlet Application**.

B.Sc. VI Semester

Course Title: Minor Project	
Course Code: B.Sc.	**Total Teaching Hours:56**
IA Marks : 10	**Teaching Hours /Week: 04**
Exam Marks: 40	**Examination Hours: 03**
Course Credits: 02	

COURSE DESCRIPTION

The **Minor Project** course is designed to provide students with hands-on experience in applying fundamental concepts of Computer Science to solve simple real-world problems. This course emphasizes problem identification, basic system design, implementation, documentation, and presentation skills within a limited scope suitable for a semester-based evaluation.

COURSE OBJECTIVES

The objectives of this course are to enable students to:

- Understand and define a simple computational problem
- Apply basic programming and logical skills to develop a solution
- Design a simple system using standard methodologies
- Implement the solution using appropriate tools and technologies
- Document and present the project work effectively

COURSE OUTCOMES

Upon successful completion of the course, students will be able to:

CO1: Identify and formulate a problem statement relevant to Computer Science

CO2: Design a basic system architecture for a given problem

CO3: Implement a mini project using standard programming tools

CO4: Analyze and present results obtained from the project

CO5: Demonstrate communication skills during project presentation and viva voce

TEACHING AND LEARNING METHODS

- Self-learning under faculty guidance
- Practical implementation in the laboratory
- Periodic reviews and demonstrations
- Documentation and presentation practice

GENERAL PROJECT STRUCTURE

The mini project report shall contain the following chapters:

1. Title Page
2. Introduction
3. Problem Statement
4. Objectives of the Project
5. Advantages
6. Limitations
7. Applications
8. Scope of the Project
9. System Requirements (Hardware and Software)
10. Methodology / Working Procedure
11. System Architecture / Block Diagram
12. Module Description
13. Implementation Details
14. Results and Discussion
15. Future Enhancements
16. Conclusion
17. References

EVALUATION SCHEME- 50 MARKS (IA-10 + Exam-40 MARKS)

Component	Marks
Problem Statement & Objectives	5
Project Design (Methodology & Architecture)	10
Implementation / Logic	10
Output / Results	5
Documentation & Report	10 (IA)
Viva Voce	10
Total	**50 Marks**

PROJECT NATURE

- The project shall be of **minor-project level**.
- Projects may be **individual or group-based** (maximum 4 students per group)

BCS Chairman
Dept. of Computer Science
Davangere University
Shivagangotri, Davangere

Prof. M. Govindappa
Dean-Science & Technology
Davangere University
Shivagangotri, Davangere-577007

Registrar
Davangere University
Shivagangotri, Davangere