સૌરાષ્ટ્ર યુનિવર્સિટી

એકેડેમિક વિભાગ

યુનિવર્સિટી કેમ્પસ, યુનિવર્સિટી રોડ, રાજકોટ-૩૬૦૦૦૫

ફોન નં.(૦૨૮૧)૨૫૭૮૫૦૧ એક્સટે. નં.૨૦૨, ૩૦૪ ફેક્સ નં.(૦૨૮૧)૨૫૭૬૩૪૭ ઈ-મેઈલ : academic@sauuni.ac.in

નં.એકે/કોમ્પ્યુટર સાયન્સ/2613745 /૨૦૨૬

તા.06 /૦૮/૨૦૨૬

B.C.A.

પરિપત્ર:-

સૌરાષ્ટ્ર યુનિવર્સિટીની ફેકલ્ટી ઓફ કોમ્પ્યુટર સાયન્સ હેઠળની સ્નાતક કક્ષાના B.C.A. ના અભ્યાસક્રમ ચલાવતી સર્વે સંલગ્ન કોલેજોના આચાર્યશ્રીઓને આથી જાણ કરવામાં આવે છે કે, ચેરપર્સનશ્રી, કોમ્પ્યુટર સાયન્સ વિષયની અભ્યાસ સમિતિ તથા ડીનશ્રી, ફેકલ્ટી ઓફ કોમ્પ્યુટર સાયન્સ દ્વારા રજુ કરાયેલ B.C.A (Honours) સેમેસ્ટર-૭ અને ૦૮ તથા B.C.A (Honours With Research) સેમેસ્ટર-૭ અને ૦૮ નો SOP મુજબનો અભ્યાસક્રમ અધિકાર મંડળોની બહાલીની અપેક્ષાએ મંજુરી આપવા માન.કુલપતિ સાહેબને ભલામણ કરેલ જે માન.કુલપતિશ્રીએ મંજુર કરેલ છે. જેથી સંબંધિત તમામે તે મુજબ તેની ચુસ્તપણે અમલવારી કરવી.

(મુસદ્દો કુલસચિવશ્રીએ મંજુર કરેલ છે.)

સહી/-
(ડૉ. એ. કે. જાડેજા)
કુલસચિવ

બિડાણ:- ઉક્ત અભ્યાસક્રમ (સોફ્ટ કોપી)

રવાના કર્યું

ઈ. ચા. એકેડેમિક ઓફીસર

પ્રતિ,

(૧) અધ્યક્ષશ્રી, કોમ્પ્યુટર સાયન્સ ભવન, સૌરાષ્ટ્ર યુનિવર્સિટી, રાજકોટ.

(૨) સૌરાષ્ટ્ર યુનિવર્સિટીની ફેકલ્ટી ઓફ કોમ્પ્યુટર હેઠળની B.C.A.ના વિષય ચલાવતી સ્નાતક કક્ષાની સર્વે સંલગ્ન કોલેજોના આચાર્યશ્રીઓ તરફ

(૩) ડીનશ્રી, ફેકલ્ટી ઓફ કોમ્પ્યુટર સાયન્સ

નકલ જાણ અર્થે રવાના:-

૧. માન.કુલપતિશ્રી/કુલસચિવશ્રીના અંગત સચિવ

નકલ રવાના (યોગ્ય કાર્યવાહી અર્થે):-

૧. પરીક્ષા વિભાગ
૨. પી.જી.ટી.આર.વિભાગ
૩. જોડાણ વિભાગ

Saurashtra University

Rajkot

Curriculum of

4th Year UG Programme

Bachelor of Computer Application (Honours)

(Semester-7 and Semester–8)

To be effective from June– 2026

The BCA Honours programme in the 4th year should focus mainly on advanced practical knowledge, industry exposure, employability skills, digital innovation, and professional development in modern computing environments. The program should maintain continuity with the academic and evaluation structure already followed by Saurashtra University in Semester 5 and Semester 6 so that students and affiliated colleges can easily adapt to the new curriculum framework.

The BCA Honours programme should consist of 22 credits in Semester VII and 22 credits in Semester VIII, making a total of 44 credits in the 4th year. Each semester should contain three Major/Core subjects of 4 credits each, one Minor/Elective subject of 4 credits, and one Internship/On-Job Training (OJT) component of 6 credits as per guidelines.

The evaluation pattern should remain aligned with the existing BCA structure of Saurashtra University. Each course may be evaluated through Continuous Comprehensive Evaluation (CCE) and Semester End Examination (SEE) as per university norms. The CCE component may include attendance, assignments, seminars, presentations, practical performance, analytical activities, project implementation, and mid-semester tests. Students should secure minimum passing marks separately in CCE and SEE as prescribed by the university.

The Internship/On-Job Training (OJT) component should be compulsory for Honours students. Students should complete internship training, industry-oriented project work, startup-based training, or practical implementation activities in IT companies, startups, educational software firms, digital business organizations, or approved institutional setups. The OJT evaluation may include industry certificate submission, internship diary, weekly progress report, project implementation, documentation, presentation, and viva voce examination.

The overall objective of the BCA Honours programme should be to develop industry-ready graduates with practical implementation skills, analytical abilities, startup orientation, freelancing capabilities, digital innovation awareness, and strong employability in modern IT and software industries.

SEE (50 marks) Paper setting guidelines for all the semester

Question Paper contains 5 questions (each of 10 marks). Every question will be asked from respective unit as specified in the syllabus of each course. (i.e., Question-1 from Unit No.1 and remaining questions from their respective units). Every question is divided in three parts like (a), (b) and (c). Part (a) contains three objective type questions (not MCQ) like definition, reason, answer in one line, answer in one word etc., each of one mark and no internal option. Part (b) contains two questions each of two marks and student will attempt any one out of two. Part (c) contains two questions each of five marks and student will attempt any one out of two.

Saurashtra University **BCA Semester- 7, 8** **Time:02:00**				**Total marks: 50**	
Q.1	(a)	Attempt the following		Unit-1	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			
Q.2	(a)	Attempt the following		Unit-2	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			
Q.3	(a)	Attempt the following		Unit-3	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			

Q. 4	(a)	Attempt the following		Unit - 4	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			
Q.5	(a)	Attempt the following		Unit-5	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –41: UI/UX Design and Interactive Systems	4	50	50	100
2	Major	CS –42: Cloud Tools, DevOps and Deployment Services	4	50	50	100
3	Major	CS –43: Mobile Application Development with Flutter	4	50	50	100
4	Minor	CS –44: Practical Based on CS-4, CS-43	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –45: Internship / Industry Project	6	0	150	150
		Total	**22**			**550**

BCA SEM 8						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –46: Blockchain Applications and Smart Systems	4	50	50	100
2	Major	CS –47: Data Science using Python	4	50	50	100
3	Major	CS –48: React JS & Express JS	4	50	50	100
4	Minor	CS –49: Practical Based on CS-47, CS-48	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –50: Internship / Industry Project	6	0	150	150
		Total	**22**			**550**

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –41: UI/UX Design and Interactive Systems	4	50	50	100
2	Major	CS –42: Cloud Tools, DevOps and Deployment Services	4	50	50	100
3	Major	CS –43: Mobile Application Development with Flutter	4	50	50	100
4	Minor	CS –44: Practical Based on CS-4, CS-43	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –45: Internship / Industry Project	6	0	150	150
		Total	**22**			**550**

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –41: UI/UX Design and Interactive Systems

Objectives:

- To provide students with practical knowledge of User Interface (UI) design, User Experience (UX) principles, responsive web technologies, and modern front-end development practices.
- Focuses on designing interactive, user-friendly, and responsive digital interfaces using modern tools and front-end frameworks.
- To develop creative, analytical, and implementation skills required in current web and application development industries.

Prerequisites:

- Basic Knowledge of:
  - Computer applications, internet technologies, introductory web technologies.
- Basic familiarity of:
  - HTML/CSS Concepts, website structures, user interaction systems

Unit No	Topic	Details
1	UI / UX Design Principles	● Introduction to UI and UX ▪ Concept and importance of UI design ▪ Concept and importance of UX design ▪ Difference between UI and UX ▪ Evolution of modern digital interface systems ● User-Centered Design (UCD) ▪ Principles of user-centered Design ▪ Understanding user requirements ▪ User behavior and interaction patterns ▪ User journey concepts ● Visual Design Principles ▪ Layout and alignment techniques ▪ Typography fundamentals ▪ Color theory and combinations ▪ White space and visual hierarchy ▪ Imagery and iconography ● Responsive Design Concepts ▪ Responsive interface principles ▪ Mobile-first design approach ▪ Multi-device compatibility concepts
2	User Research and Analysis	● User Research Methods ▪ Introduction ▪ Qualitative and quantitative research methods ▪ Observation and survey methods

		▪ User interview techniques ● User Analysis Techniques ▪ User personas ▪ User behavior analysis ▪ Requirement gathering methods ▪ User expectation mapping ● Information Architecture ▪ Introduction ▪ Content organization techniques ▪ Navigation structures ▪ Sitemap planning ● User Data Analytics ▪ User feedback collection methods ▪ Basic analytics concepts ▪ Using user data for design decisions ▪ Interface improvement strategies
3	UI/UX Design Tools and Prototyping	● UI/UX Design Tools ▪ Introduction to Figma ▪ Adobe XD basics ▪ Canva for interface design ▪ Collaborative design platforms ● Wire framing Concepts ▪ Low-fidelity wireframes ▪ High-fidelity wireframes ▪ Interface structure planning ▪ Grid Systems and alignment ● Prototyping Techniques ▪ Interactive prototyping ▪ Clickable interfaces ▪ Responsive prototype development ▪ Mobile application prototyping ● Collaborative Design ▪ Team-based design workflow ▪ Design sharing and collaboration ▪ Feedback integration techniques
4	Visual Design and Interactive UI Components	● Advanced Visual Design ▪ Advanced layout techniques ▪ Visual balance and composition ▪ Modern design trends ▪ Dark mode interfaces ● UI Components ▪ Buttons and navigation systems

		▪ Cards and panels ▪ Forms and input systems ▪ Dashboard components ● Design Patterns and Frameworks ▪ Design systems ▪ Component libraries ▪ Reusable interface patterns ▪ Accessibility standards ● Motion Design and Interaction ▪ Animation principles ▪ Interactive transitions ▪ Motion design basics ▪ Multi-platform interface design
5	UI/UX Evaluation and Iteration	● Usability Testing ▪ Introduction ▪ Testing methods and techniques ▪ User Observation methods ▪ Task-based usability testing ● Heuristic Evaluation ▪ Heuristic principles ▪ Interface evaluation techniques ▪ Problem Identification Methods ▪ UI improvement strategies ● Feedback Analysis ▪ User feedback interpretation ▪ Data-driven design improvements ▪ Communication and presentation strategies ● Iterative Design Process ▪ Continuous improvement techniques ▪ A/B testing basics ▪ UI performance metrics ▪ Design refinement workflow

Reference Books

(1) Don’t Make Me Think – Steve Krug
(2) The Design of Everyday Things – Don Norman
(3) UI/UX Design Essentials – Daniel Walter Scott
(4) Responsive Web Design with HTML5 and CSS – Ben Frain
(5) Learning Web Design – Jennifer Niederst Robbins
(6) The Principles of Beautiful Web Design, Third Edition, Jason Beaird, O’Reilly, 2014
(7) UX/UI Design 2021-22 Tutorial the Complete Step by Step Guide to UX/UI Design and Best Practices for designers with no experience, Tom Mulligan

Course Outcomes:

After successful completion of this course, students will be able to:

- Understand the principles of UI and UX design
- Design user-friendly and responsive digital interfaces
- Conduct user research and usability analysis
- Develop wireframes, prototypes and modern UI frameworks
- Apply visual design principles and modern UI frameworks
- Evaluate and improve user experiences using testing and feedback techniques

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –42: Cloud Tools, DevOps and Deployment Services

Objectives:

- To provide students with conceptual and industry-oriented knowledge of cloud computing environments, collaborative software platforms, DevOps workflows, deployment strategies, hosting systems, and modern software delivery practices used in contemporary IT industries.
- Focuses on understanding cloud-based collaboration systems, deployment models, version control concepts, cloud hosting environments, and modern application management practices.

Prerequisites:

- Basic Knowledge of:
  - computer networks, internet technologies, operating systems
- Basic understanding of:
  - website hosting, Software applications and web technologies

Unit No	Topic	Details
1	Introduction to Cloud Computing and Collaborative Systems	● Introduction to Cloud Computing ▪ Concept and importance of cloud computing ▪ Evolution of cloud-based systems ▪ Characteristics of cloud environments ▪ Advantages and limitations of cloud environments ● Cloud Service Models ▪ Infrastructure as a Service (IaaS) ▪ Platform as a Service (PaaS) ▪ Software as a Service (SaaS) ▪ Comparison of cloud service models ● Cloud Deployment Models ▪ Public cloud ▪ Private cloud ▪ Hybrid cloud environments ▪ Community cloud concepts ● Collaborative Cloud Systems ▪ Cloud-based collaboration platforms ▪ Team collaboration platforms ▪ Online document sharing systems ▪ Real-time communication platforms ● Cloud Storage Systems ▪ Cloud file storage ▪ Cloud backup systems ▪ Data Synchronization concepts ▪ Advantages of cloud storage

2	Version Control Systems and Collaborative Development	● Introduction to Version Control ▪ Concept and importance of version control ▪ Centralized and distributed version control systems ▪ Advantages of version management ● Git Fundamentals ▪ Introduction to Git ▪ Repository concepts ▪ Version tracking concepts ▪ Commit and update workflow ● Branching and Collaboration ▪ Branch management concepts ▪ Merge concepts and workflows ▪ Conflict management concepts ▪ Team collaboration workflow ● GitHub and Collaborative Platforms ▪ Introduction to GitHub ▪ Repository hosting concepts ▪ Pull requests and code review concepts ▪ Open-source collaboration environments
3	DevOps Workflow and Containerization Concepts	● Introduction to DevOps ▪ Concept and importance of DevOps ▪ DevOps lifecycle ▪ Collaboration between development and operations teams ▪ Benefits of DevOps practices ● Continuous Integration and Deployment ▪ Introduction to CI/CD concepts ▪ Automated build processes ▪ Deployment pipeline concepts ▪ Continuous delivery workflows ● Containerization Concepts ▪ Introduction to containers ▪ Virtualization and containerization ▪ Docker fundamentals ▪ Container lifecycle concepts ▪ Advantages of containerized applications ● Monitoring and Logging ▪ Application monitoring basics ▪ Logging systems ▪ Performance tracking concepts ▪ Error tracking concepts
4	Hosting, Deployment and	● Hosting Concepts ▪ Shared hosting systems

	Cloud Application Management	▪ VPS hosting concepts ▪ Cloud hosting systems ▪ Hosting service providers ● Domain and DNS Systems ▪ Domain registration concepts ▪ DNS configuration basics ▪ Domain mapping techniques ▪ URL management concepts ● Deployment Practices ▪ Frontend deployment concepts ▪ Backend deployment basics ▪ Deployment lifecycle ▪ Deployment troubleshooting concepts ● Cloud Application Management ▪ Application maintenance concepts ▪ Cloud resource management ▪ Backup and recovery concepts ▪ Hosting management systems ● Security Concepts ▪ SSL certificates ▪ Cloud security basics ▪ Authentication concepts ▪ Data protection concepts
5	Modern Cloud Practices and Industry Applications	● Modern Cloud Practices ▪ Cloud scalability concepts ▪ Server less computing basics ▪ Cloud-based automation systems ▪ Multi-platform integration concepts ● Industry Applications ▪ Cloud systems in business environments ▪ Enterprise collaboration systems ▪ Educational cloud platforms ▪ Cloud workflow systems ● Deployment Strategies ▪ Agile deployment concepts ▪ Deployment planning strategies ▪ Continuous improvement strategies ▪ Software release strategies ● Emerging Cloud Technologies ▪ Edge computing basics ▪ AI-integrated cloud systems ▪ Future Trends in cloud technologies

		● Case Studies and Industry Trends ▪ Real-world cloud application examples ▪ Case studies of cloud-based organizations ● Cloud adoption challenges and opportunities

Reference Books

(1) Cloud Computing – Rajkumar Buyya
(2) The DevOps Handbook – Gene Kim
(3) Docker Deep Dive – Nigel Poulton
(4) AWS Fundamentals – Michael Wittig
(5) DevOps for Beginners – Viktor Farcic
(6) Cloud Computing Concepts – Thomas Erl

Course Outcomes:

After successful completion of this course, students will be able to:

- Understand the concepts and importance of cloud computing systems.
- Explain cloud service models and deployment models.
- Understand collaborative software development and version control workflows.
- Analyze DevOps concepts and deployment strategies used in software industries.
- Understand hosting, deployment, and cloud application management concepts.
- Identify modern trends and applications of cloud-based systems.

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –43: Mobile Application Development with Flutter		

Objectives:
• To provide students with practical knowledge of cross-platform mobile application development using Flutter framework and modern mobile UI technologies.
• Focuses on mobile interface design, widget-based application development, navigation systems, data handling, Firebase integration, API connectivity, and deployment concepts used in modern mobile application environments.
Prerequisites:
• Basic Knowledge of:
  ◦ Programming concepts, Web technologies, Databases
• Basic understanding of:
  ◦ Object-oriented programming, APIs, Responsive interface concepts		

Unit No	Topic	Details
1	Introduction to Flutter and Dart Programming	● Introduction to Mobile Applications ▪ Concept and importance of mobile applications ▪ Types of mobile applications ▪ Native and cross-platform applications ▪ Mobile application development lifecycle ● Introduction to Flutter ▪ What is Flutter? ▪ Features and advantages of Flutter ▪ Flutter framework architecture ▪ Working of Flutter and Dart ▪ Widget-based application development ● Flutter Development Environment ▪ Flutter SDK installation ▪ Android Studio and VS Code integration ▪ Flutter project structure ▪ Organizing files and folders ▪ Hot Reload concept ▪ Themes in Flutter applications ● Introduction to Dart Programming ▪ Variables and data types ▪ Operators and expressions ▪ Decision-making statements ▪ Looping statements ▪ Functions and collections ▪ Object-oriented programming concepts
2	Flutter Widgets,	● Introduction to Widgets ▪ Stateless widgets

	Layouts and Navigation Systems	▪ Stateful widgets ▪ Widget tree concepts ▪ Widget build visualization ● Layout Systems ▪ Single child widgets ▪ Multiple child widgets ▪ Rows and columns ▪ Containers and padding ▪ Responsive layout concepts ▪ Advanced layout systems ● Gestures and User Interaction ▪ Introduction to gestures ▪ User interaction handling ▪ Buttons and input widgets ▪ Form handling and validation ● Navigation and State Management ▪ Navigation concepts ▪ Routing systems ▪ Multi-screen applications ▪ Drawer and bottom navigation ▪ Ephemeral state management ▪ Application state concepts
3	Database Systems and Firebase Integration	● Local Database Concepts ▪ Introduction to SQLite ▪ Local data storage concepts ▪ Data persistence techniques ▪ CRUD operation concepts ● Firebase Integration ▪ Introduction to Firebase and Firebase project setup ▪ Cloud Firestore basics ▪ Real-time database concepts ▪ Firebase authentication systems ● Internationalization and Packages ▪ Internationalization concepts ▪ Using intl package ▪ External package import concepts ▪ Flutter package management ● Testing and Development Tools ▪ Introduction to testing in Flutter ▪ Widget testing basics ▪ Application debugging concepts

		▪ Flutter DevTools basics ▪ Flutter SDK utilities
4	API Integration and Advanced Mobile Features	● REST API Concepts ▪ Introduction to APIs, REST API architecture ▪ HTTP request and response, JSON parsing techniques ● API Integration ▪ Accessing web services ▪ Product service API concepts ▪ Data fetching and display ▪ Error handling concepts ● Asynchronous Programming ▪ Future and async concepts ▪ Asynchronous data loading ▪ Background processing basics ● Multimedia and Device Features ▪ Image and multimedia integration ▪ Camera and gallery access basics ▪ Location and map integration concepts ▪ Push notification basics
5	Flutter Animation, Deployment and Real-World Applications	● Flutter Animation ▪ Introduction to animation concepts ▪ Animation-based classes, Animation workflow in Flutter ▪ Transition and motion effects ● Packages and Plugins ▪ Introduction to packages ▪ Types of packages ▪ Using Dart packages ▪ Flutter plugin concepts ● Deployment Concepts ▪ APK generation ▪ Android application deployment ▪ Introduction to iOS application deployment ▪ Application publishing concepts ● Real-World Mobile Applications ▪ Educational mobile applications ▪ E-commerce mobile systems ▪ Healthcare and productivity applications Modern trends in mobile computing

Reference Books

(1) Flutter for Beginners – Alessandro Biessek
(2) Beginning Flutter – Marco Napoli

(3) Flutter Complete Reference – Alberto Miola
(4) Dart Programming Language Specification – Dart Team
(5) Mobile Application Development using Flutter – Frank Zammetti
(6) Firebase Essentials – John Horton

Course Outcomes:

After successful completion of this course, students will be able to:

- Understand the concepts and architecture of Flutter applications.
- Develop cross-platform mobile applications using Flutter and Dart.
- Design responsive mobile interfaces using widgets and layouts.
- Implement navigation, forms, and state management in Flutter applications.
- Integrate Firebase, SQLite, and REST APIs in mobile applications.
- Develop and deploy basic real-world mobile applications.

CS –44: Practical Based on CS-4, CS-43	
CCE- Continuous and comprehensive Evaluation as follow • The continuous Comprehensive Evaluation (CCE) for each subject will be conducted by the teacher of that subject. The teacher will decide how the evaluation will be done. Usually CCE includes things like class participation, case studies and presentation, assignments, tutorials, small test (announced or surprised), quizzes and attendance or a mix of these. • Students must submit their work for internal evaluation on time to time. • Another part of CCE is the mid-term exam, which is compulsory for all students. This exam will be conducted internally by the college.	50 Marks
SEE – Semester End Examination as per the following • Practical exams may be scheduled before or after the theory examinations. Exam is conducted by Saurashtra University using external examiner (3 hours duration) • Students must prepare a practical notebook/book for the final practical examination. (The practical book serves as a record of all practical work, observations, procedures and results performed during the semester in lab. It is essential for evaluation during the final practical examination)	50 Marks

CS –44: Practical Based on CS-41, CS-43	**Total Marks - 100**	
Topics	**CCE**	**SEE**
CS-41:UI/UX Design and Interactive Systems	25	25
CS-43:Mobile Application Development with Flutter	25	25

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

<table>
<tr><th colspan="4">CS –45: Internship / Industry Project</th></tr>
<tr><td colspan="4">Objectives:
• To apply theoretical concepts learned in previous semesters to practical software development.
• To gain hands-on experience in using programming languages, tools, and technologies.
• To understand and implement the complete Software Development Life Cycle (SDLC).
• To enhance problem-solving and analytical skills by designing solutions to real-life problems.
• To develop teamwork and collaboration skills through group-based project execution.
• To improve project planning and time management abilities.
• To practice documentation and reporting skills as per industry standards.
• To prepare students for major projects, internships, or professional roles in software development.</td></tr>
<tr><th>No.</th><th>Topic</th><th colspan="2">Guidelines for Internship</th></tr>
<tr><td>1</td><td>General</td><td colspan="2">• Students must get prior approval from the college before starting the internship / project topic.
• Choose an Internship/project topic
• Projects can be individual or group-based (2 Students)(as per college policy)
• Submit a project proposal with the title, objective, scope, tools/technologies to be used, and team members.
• Get the proposal approved by the project guide/faculty coordinator.
• Use any suitable programming language or platform
• Project must be developed in any Company / Organization /Institute or computer laboratory of college.
• Students must maintain a daily activity log during the internship.</td></tr>
<tr><td>2</td><td>Documentation</td><td colspan="2">Maintain a project report including the following:
• Title Page
• Certificate
• Acknowledgment
• Table of Contents
• Introduction & Objective
• System Analysis (Problem Definition, Feasibility Study)
• System Design (DFD, ER Diagram, etc.)
• Data Dictionary
• Implementation (Screenshots)
• Testing (Test cases and results)
• Conclusion & Future Scope</td></tr>
<tr><td>3</td><td>Submission</td><td colspan="2">• Submit a soft copy (CD/ Pen drive) and hard copy of the project report. Project report must be printed on both side of the page.
• Project must be submitted before two weeks of commencement of theory exam
• During the project viva examination project must be run.</td></tr>
<tr><td>4</td><td>Assessment Criteria</td><td>SEE –Exam is conducted by Saurashtra University using external examiner (examiner will evaluate the executable copy, hard copy of the project and take the viva voce)</td><td>150 Marks</td></tr>
</table>

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 8						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –46: Blockchain Applications and Smart Systems	4	50	50	100
2	Major	CS –47: Data Science using Python	4	50	50	100
3	Major	CS –48: React JS & Express JS	4	50	50	100
4	Minor	CS –49: Practical Based on CS-47, CS-48	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –50: Internship / Industry Project	6	0	150	150
		Total	**22**			**550**

CS –46: Blockchain Applications and Smart Systems

Objectives:

- To provide students with conceptual understanding of blockchain technology, distributed digital systems, smart applications, crypto-currency ecosystems, and modern decentralized platforms used in contemporary digital environments.
- Focuses on blockchain fundamentals, smart contracts, digital ledger systems, business applications of blockchain, and emerging smart technologies.
- To introduce students to modern decentralized computing concepts and future-oriented digital systems without excessive mathematical or cryptographic complexity.

Prerequisites:

- Basic knowledge of:
  - Computer networks, internet technologies, databases
- Basic understanding of:
  - Online transactions, digital platforms and security concepts

Unit No	Topic	Details
1	Fundamentals of Blockchain Technology	● Introduction to Blockchain ▪ Concept and importance of blockchain technology ▪ Evolution of blockchain systems ▪ Features of decentralized systems ▪ Difference between centralized and decentralized systems ● Distributed Ledger Technology ▪ Concept of distributed ledgers ▪ Structure of blockchain networks ▪ Blocks and chain mechanisms ▪ Peer-to-peer network concepts ● Blockchain Characteristics ▪ Transparency and immutability ▪ Security and trust mechanisms ▪ Decentralization concepts ▪ Consensus mechanisms overview ● Types of Blockchain ▪ Public blockchain ▪ Private blockchain ▪ Consortium blockchain ▪ Hybrid blockchain systems
2	Cryptocurrency and Digital Transactions	● Introduction to Cryptocurrency ▪ Concept and importance of cryptocurrency ▪ Evolution of digital currencies

		▪ Bitcoin and Ethereum overview ▪ Cryptocurrency wallets ● Blockchain Transactions ▪ Digital transaction workflow ▪ Transaction verification concepts ▪ Mining concepts and processes ▪ Blockchain transaction lifecycle ● Digital Payment Systems ▪ Cryptocurrency payment systems ▪ Digital wallets and exchanges ▪ Advantages and challenges of digital currencies ▪ Financial technology (FinTech) concepts ● Security and Challenges ▪ Security concerns in digital transactions ▪ Cryptocurrency risks and fraud awareness ▪ Regulatory challenges ▪ Ethical concerns in digital finance
3	Smart Contracts and Decentralized Applications	● Smart Contracts ▪ Concept and importance of smart contracts ▪ Features of smart contracts ▪ Working mechanism of smart contracts ▪ Advantages and limitations ● Decentralized Applications (DApps) ▪ Introduction to decentralized applications ▪ Difference between traditional and decentralized applications ▪ Components of DApps ▪ Use cases of decentralized systems ● Blockchain Automation Concepts ▪ Automated digital agreements ▪ Workflow automation using smart contracts ▪ Smart transaction systems ▪ Digital verification systems ● Smart Systems ▪ Introduction to smart systems ▪ Smart governance concepts ▪ Smart identity systems ▪ Smart asset management
4	Blockchain Applications in industry	● Blockchain in Business ▪ Supply chain management systems ▪ Blockchain in banking and finance ▪ Healthcare applications ▪ Educational certification systems

		● Blockchain in Governance ▪ Digital identity systems ▪ E-voting concepts ▪ Smart governance applications ▪ Public service systems ● Enterprise Blockchain Applications ▪ Enterprise resource management ▪ Secure digital documentation ▪ Blockchain-based authentication process ▪ Data integrity management ● Emerging Business Opportunities ▪ Blockchain startups ▪ Digital asset management ▪ NFT concepts and applications
5	Emerging Smart Technologies and Future Trends	● Emerging Blockchain Technologies ▪ Blockchain and Artificial Intelligence integration ▪ Blockchain and IoT systems ▪ Cloud and blockchain integration ▪ Smart city concepts ● Future of Decentralized Systems ▪ Web 3.0 concepts ▪ Future of digital ownership ▪ Decentralized internet concepts ▪ Future digital ecosystems ● Challenges and Limitations ▪ Scalability challenges ▪ Energy consumption concerns ▪ Regulatory and legal issues ▪ Technology adoption challenges

ReferenceBooks

(1) Blockchain Basics – Daniel Drescher
(2) Mastering Blockchain – Imran Bashir
(3) Blockchain Revolution – Don Tapscott
(4) Introducing Ethereum and Solidity – Chris Dannen
(5) Blockchain Technology Explained – Alan Norman
(6) Cryptocurrency and Blockchain Fundamentals – Arun J.

Course Outcomes:

After successful completion of this course, students will be able to:

- Understand the concepts and working principles of blockchain systems.
- Explain distributed ledger technologies and decentralized networks.
- Understand cryptocurrency ecosystems and smart contract concepts.
- Analyze blockchain applications in business and digital industries.
- Identify emerging smart systems and decentralized applications.
- Understand future trends and challenges of blockchain technologies.

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –47: Data Science using Python

Objectives:

- Understanding the fundamental concepts of Data Science and the data analytics lifecycle.
- Familiarize students with statistical methods, measures of central tendency, and dispersion.
- Enable students to explore, analyze, and visualize data effectively using Python libraries.
- Understand the principles of correlation, exploratory data analysis (EDA), and statistical quality control.

Prerequisites:

- Basic Computer Skills.
- Fundamental Programming Concepts (Python basics).
- Basic understanding of Mathematics and Statistics.

Unit No	Topic	Details
1	Introduction & Exploring Data	• Introduction to Data Science & Life Cycle • Definition of Data Science • Importance of Data Science • Applications of Data Science • Data Science Life Cycle • Roles of a Data Scientist • Data Science vs Data Analytics • Python ecosystem for Data Science (NumPy, Pandas overview). • Explore Data using Python • Understanding Characteristics of the dataset • Size, Shape, value counts • Data Types and change data types.
2	Descriptive Statistics	• (Merged Measures of Central Tendency & Dispersion) Arithmetic Mean, Median, Mode. • Relationship between mean, median and mode. Computation of the measures for grouped and ungrouped data. • Range, mean deviation and standard deviation • coefficient of variation and its use • Quartiles and Inter quartile range • Quintiles, Deciles and Percentiles • Skewness and Kurtosis and their uses.
3	Conducting EDA using Python	▪ Data Analysis using Python ▪ Handling missing data ▪ Computing metrics ▪ Analysis & Interpretation of the data and connected visualization.

4	Understanding Correlation	▪ Introduction to Covariance, ▪ Understanding correlation ▪ calculating correlation using Pandas ▪ Interpreting a correlation, Matrix.
5	Statistical Quality Control	▪ Nature of Control Limits ▪ Purpose of Control Charts ▪ Control Charts for Variables ▪ Control Charts for Attributes.

Reference Books

1. Hands-On Data Science and Python Machine Learning, Kane Frank, Packt Publishing Limited.
2. Data Science and Machine Learning Interview Questions Using Python a Complete Question Bank to Crack Your Interview, Vishwanathan Narayanan, BPB Publications.
3. Python for Data Science, Tony F Charles.
4. Data Structures using Python, Shriram K. Vasudevan, OUP India.
5. Python Data Science Handbook, Jake Vanderplas, O'Reilly Media.

Course Outcomes:

- Understand the data science lifecycle and explore datasets using Python.
- Calculate and interpret measures of central tendency and dispersion.
- Perform Exploratory Data Analysis (EDA) and handle missing or messy data.
- Compute and analyze correlations between different data variables.
- Implement statistical quality control charts for data-driven decision-making.

CS –48: React JS & Express JS

Objectives:

- Articulate what React is and why it is useful.
- Explore various attributes of web development.
- Explore the basic architecture of a React application.
- Explore various web development techniques of this Java Script and would learn new techniques based on industry requirement.
- Gain a deep understanding of JSX and Hooks.

Prerequisites:

- JavaScript, HTML, CSS and OOPs,

Unit No	Topic	Details
1	Express JS & Java Script	• Java Script • Java Script Overview & Basics • Variable, Conditional Statements, Loops in JS, • Functions, Arrays & Events in JS • ES6 Overview & Basics • ES6 Classes, functions & Promises • Express JS • Setting up an app with ExpressJS, Routing in ExpressJS, Connecting views with templates, configurations and error handling.
2	Introduction to JSX & REACT JS	• Introduction: What is ReactJS? Installation or Setup, Hello World Program, Create a first app, folder structure • Components: Creating components, Basic components, Nesting components, functional component, class component • Introduction to JSX: JSX Programs • Props: ReactJS Props, React State, Destructing Props and State, setState, methods as Props.
3	Form Handling, components and fragments	• Event Handling: Event Handling and Binding event handlers • Rendering: Conditional Rendering and List Rendering, List and keys, Index as Key Anti-pattern • Introduction: Basic form handling • Components: Components Life Cycle Methods, Components Mounting Lifecycle methods, Components Updating Lifecycle methods, Pure Components • Fragments

4	Memo, Refs, Props and Context	• Memo • Introduction to Refs: Refs, Refs with Class Components, Forwarding Refs and Portals • Components: Higher Order Components • Props Again! Rendering Props and Context • HTTP: HTTP and React, GET and React, POST and React.
5	Introduction to Hooks and its implementation	• Introduction: React Hooks introduction, useState Hook, useState Previous state, useState with object, useState with array. • useEffect: useEffect Hook, useEffect after render, Condionally run effects, run effects only once, useEffect with cleanup, useEffect with incorrect dependency. • Fetching data: Fetching data with useEffect, useContext Hook • useReducer Hook: useReducer – simple state and action, complex state and action, multiple useReducers • useContext: useContext, useReducer, Fetching data with useReduer, useState vs useReducer

Reference Books

1. Learning React, Martin Bean, Kirupa Chinna thambi Pearson Addison Wesley
2. ReactJS Notes for Professional,Goal Kicker,Web site ebook,
3. The Road to React_Your journey to master plain yet pragmatic React,Lean Pub Book, Robin Wieruch -Independently Published (2020)

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –49: Practical Based on CS-47, CS-48	
CCE- Continuous and comprehensive Evaluation as follow • The continuous Comprehensive Evaluation (CCE) for each subject will be conducted by the teacher of that subject. The teacher will decide how the evaluation will be done. Usually CCE includes things like class participation, case studies and presentation, assignments, tutorials, small test (announced or surprised), quizzes and attendance or a mix of these. • Students must submit their work for internal evaluation on time to time. • Another part of CCE is the mid-term exam, which is compulsory for all students. This exam will be conducted internally by the college.	50 Marks
SEE – Semester End Examination as per the following • Practical exams may be scheduled before or after the theory examinations. Exam is conducted by Saurashtra University using external examiner (3 hours duration) • Students must prepare a practical notebook/book for the final practical examination. (The practical book serves as a record of all practical work, observations, procedures and results performed during the semester in lab. It is essential for evaluation during the final practical examination)	50 Marks

CS –49: Practical Based on CS-47, CS-48	**Total Marks - 100**	
Topics	**CCE**	**SEE**
CS-47:Data Science using Python	25	25
CS– 48: REACTJS & EXPRESSJS	25	25

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

<table>
<tr><th colspan="4">CS –50: Internship / Industry Project</th></tr>
<tr><td colspan="4">Objectives:
To apply theoretical concepts learned in previous semesters to practical software development.
To gain hands-on experience in using programming languages, tools, and technologies.
To understand and implement the complete Software Development Life Cycle (SDLC).
To enhance problem-solving and analytical skills by designing solutions to real-life problems.
To develop teamwork and collaboration skills through group-based project execution.
To improve project planning and time management abilities.
To practice documentation and reporting skills as per industry standards.
To prepare students for major projects, internships, or professional roles in software development.</td></tr>
<tr><th>No.</th><th>Topic</th><th colspan="2">Guidelines for Internship</th></tr>
<tr><td>1</td><td>General</td><td colspan="2">Students must get prior approval from the college before starting the internship / project topic.
Choose an Internship/project topic
Projects can be individual or group-based (2 Students)(as per college policy)
Submit a project proposal with the title, objective, scope, tools/technologies to be used, and team members.
Get the proposal approved by the project guide/faculty coordinator.
Use any suitable programming language or platform
Project must be developed in any Company / Organization /Institute or computer laboratory of college.
Students must maintain a daily activity log during the internship.</td></tr>
<tr><td>2</td><td>Documentation</td><td colspan="2">Maintain a project report including the following:
Title Page
Certificate
Acknowledgment
Table of Contents
Introduction & Objective
System Analysis (Problem Definition, Feasibility Study)
System Design (DFD, ER Diagram, etc.)
Data Dictionary
Implementation (Screenshots)
Testing (Test cases and results)
Conclusion & Future Scope</td></tr>
<tr><td>3</td><td>Submission</td><td colspan="2">Submit a soft copy (CD/ Pen drive) and hard copy of the project report. Project report must be printed on both side of the page.
Project must be submitted before two weeks of commencement of theory exam
During the project viva examination project must be run.</td></tr>
<tr><td>4</td><td>Assessment Criteria</td><td>SEE –Exam is conducted by Saurashtra University using external examiner (examiner will evaluate the executable copy, hard copy of the project and take the viva voce)</td><td>150 Marks</td></tr>
</table>

BCA SEM 7: CS–41:UI/UX Design and Interactive Systems

Note:

- The practical exercises of this subject shall be performed using standard computer laboratory facilities and modern UI/UX design tools.
- Students may use software/tools/platforms such as Figma, Canva, Adobe XD, HTML5, CSS3, Bootstrap, Tailwind CSS, JavaScript, and standard web browsers for practical implementation and frontend development activities.
- Open-source, free, or equivalent software/tools may be used wherever applicable.
- The practical work should focus on interface design, responsive frontend systems, prototyping, usability evaluation, and frontend implementation concepts.

Minimum Following Exercises Should be Performed by the Students During the Semester

Unit – I: UI / UX Design Principles

1. Study and analysis of modern website interfaces based on UI and UX principles.
2. Design of typography combinations for educational and business applications.
3. Preparation of color palettes and visual themes for digital interfaces.
4. Development of visual hierarchy layouts using spacing, typography, and alignment principles.
5. Preparation of responsive layout structures for desktop and mobile interfaces.
6. Design of user journey maps for online registration or e-commerce systems.
7. Comparative analysis of good and poor interface designs.
8. Design of responsive navigation structures and menu systems.
9. Accessibility evaluation of selected websites and applications.
10. User-centered interface planning for digital platforms.

Unit – II: User Research and Analysis

11. Creation of user personas for educational or business applications.
12. Preparation of online survey forms for collecting user feedback.
13. Conduct of user interviews and preparation of analysis reports.
14. Development of requirement analysis documents based on user feedback.
15. Preparation of sitemap and information architecture layouts for websites.
16. Content organization and navigation flow analysis for digital systems.
17. Observation and analysis of user interaction patterns with selected interfaces.
18. Preparation of usability and user behavior study reports.
19. Comparative analysis of frontend navigation systems.
20. Preparation of UX research documentation for selected applications.

Unit – III: UI / UX Design Tools and Prototyping

21. Development of low-fidelity wireframes using modern UI/UX tools.
22. Design of high-fidelity interface prototypes using Figma or similar tools.

23. Creation of responsive mobile application prototypes.
24. Development of interactive clickable prototypes for frontend systems.
25. Design of dashboard interface prototypes for analytical applications.
26. Creation of e-commerce frontend UI prototypes.
27. Development of reusable UI component libraries.
28. Collaborative interface design activity using shared platforms.
29. Responsive prototype development for multiple screen sizes.
30. Complete UI prototype preparation for real-world applications.

Unit – IV: Visual Design and Interactive UI Components

31. Design of responsive forms and input interface systems.
32. Development of modern navigation bars and interactive menus.
33. Creation of dashboard cards, panels, and frontend UI components.
34. Design of dark mode interface systems.
35. Development of animated buttons and interactive frontend elements.
36. Preparation of reusable frontend component libraries.
37. Accessibility improvement activity for selected interfaces.
38. Design of multi-platform interface layouts for desktop and mobile devices.
39. Implementation of motion design and transition effects in frontend systems.
40. Development of interactive frontend mini applications integrating advanced UI components.

Unit – V: UI / UX Evaluation and Iteration

41. Conduct of usability testing for selected websites or applications.
42. Preparation of heuristic evaluation reports for digital interfaces.
43. User feedback collection and interface improvement analysis.
44. Comparative A/B testing of frontend interface designs.
45. Preparation of UI redesign proposals based on usability findings.
46. Interface optimization activity based on accessibility standards.
47. Analysis of frontend responsiveness and performance.
48. Preparation of UI evaluation presentations and reports.
49. Iterative redesign of interfaces based on testing observations.
50. Development and presentation of a final mini project integrating wireframing, prototyping, responsive frontend design, usability evaluation, and interface optimization concepts covered in the course.

Saurashtra University
B.C.A. (Honours) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7: CS–43:Mobile Application Development With Flutter

Note:

- The practical exercises of this course shall be performed using standard computer laboratory facilities and modern mobile application development tools and platforms.
- Students may use software/tools/platforms such as Flutter SDK, Dart, Android Studio, VS Code, Android Emulator, Firebase, SQLite, REST APIs, and related mobile development tools for practical implementation and application development activities.
- Open-source, free, educational, or equivalent software/tools may be used wherever applicable.
- The practical work should focus on mobile user interface development, widget-based application design, responsive layouts, state management, database integration, Firebase services, REST API connectivity, application testing, and cross-platform mobile application development practices.

Unit – I: Introduction to Flutter and Dart Programming

1. Installation and configuration of Flutter SDK and development tools.
2. Creation and execution of basic Flutter applications.
3. Study and analysis of Flutter project architecture and file structure.
4. Demonstration of Hot Reload and Flutter themes.
5. Implementation of Dart programming examples using variables and operators.
6. Development of conditional and looping statement programs in Dart.
7. Implementation of function-based Dart applications.
8. Demonstration of object-oriented programming concepts in Dart.
9. Development of simple console-based Dart applications.
10. Emulator/device setup and testing activity.
11. Suggested Tools / Platforms:
12. Flutter SDK, Dart, Android Studio, VS Code, Android Emulator.

Unit – II: Flutter Widgets, Layouts and Navigation Systems

13. Design of mobile interfaces using stateless and stateful widgets.
14. Development of responsive layouts using rows, columns, and containers.
15. Creation of form-based mobile applications.
16. Implementation of button and gesture handling systems.
17. Development of multi-screen navigation systems.
18. Design of drawer and bottom navigation interfaces.
19. Implementation of form validation techniques.
20. Development of responsive mobile dashboards.
21. State management demonstration activity.
22. Development of mini UI-based Flutter applications.
23. Suggested Tools / Platforms:
24. Flutter SDK, Dart, Android Studio, VS Code.

Unit – III: Database Systems and Firebase Integration

25. SQLite database integration activity.
26. Development of CRUD operation applications.
27. Firebase project creation and configuration activity.
28. Integration of Firebase authentication systems.
29. Cloud Firestore database integration activity.
30. Data storage and retrieval using Firebase services.
31. Internationalization and localization activity using intl package.
32. External package integration demonstration.
33. Widget testing and debugging activity.
34. Development of Firebase-integrated mini applications.
35. Suggested Tools / Platforms:
36. Flutter SDK, Firebase Console, SQLite, Cloud Firestore, Android Studio.

Unit – IV: API Integration and Advanced Mobile Features

37. REST API integration in Flutter applications.
38. JSON data parsing and display activity.
39. Development of applications using HTTP requests and responses.
40. Asynchronous data loading activity using Future and async concepts.
41. Error handling implementation in Flutter applications.
42. Image and multimedia integration activity.
43. Camera and gallery access demonstration.
44. Location and map integration basics.
45. Push notification demonstration activity.
46. Development of API-based Flutter mini applications.
47. Suggested Tools / Platforms:
48. Flutter SDK, REST APIs, JSON Tools, Android Emulator.

Unit – V: Flutter Animation, Deployment and Real-World Applications

49. Animation and transition effect implementation in Flutter applications.
50. Package and plugin integration activity.
51. APK generation and deployment demonstration.
52. Development of educational or business mobile applications.
53. Creation of e-commerce mobile application interfaces.
54. Development of productivity or healthcare mobile systems.
55. Responsive mobile application optimization activity.
56. Case study analysis of modern Flutter applications.
57. Preparation of mobile application project documentation.
58. Development and presentation of a final Flutter-based mobile application integrating widgets, Firebase, APIs, animation, and deployment concepts covered in the course.

Saurashtra University

Rajkot

Curriculum of

4th Year UG Programme

Bachelor of Computer Application
(Honours with research)

(Semester-7 and Semester–8)

To be effective from June– 2026

The BCA Honours with research programme in the 4th year should focus mainly on advanced practical knowledge, industry exposure, employability skills, digital innovation, and professional development in modern computing environments. The programme should maintain continuity with the academic and evaluation structure already followed by Saurashtra University in Semester 5 and Semester 6 so that students and affiliated colleges can easily adapt to the new curriculum framework.

The BCA Honours with research programme should consist of 22 credits in Semester VII and 22 credits in Semester VIII, making a total of 44 credits in the 4th year. Each semester should contain three Major/Core subjects of 4 credits each, one Minor/Elective subject of 4 credits, and one Internship/On-Job Training (OJT) component of 6 credits as per guidelines.

The evaluation pattern should remain aligned with the existing BCA structure of Saurashtra University. Each course may be evaluated through Continuous Comprehensive Evaluation (CCE) and Semester End Examination (SEE) as per university norms. The CCE component may include attendance, assignments, seminars, presentations, practical performance, analytical activities, project implementation, and mid-semester tests. Students should secure minimum passing marks separately in CCE and SEE as prescribed by the university.

The Internship/On-Job Training (OJT) component should be compulsory for Honours with research students. Students should complete internship training, industry-oriented project work, startup-based training, or practical implementation activities in IT companies, startups, educational software firms, digital business organizations, or approved institutional setups. The OJT evaluation may include industry certificate submission, internship diary, weekly progress report, project implementation, documentation, presentation, and viva voce examination.

The overall objective of the BCA Honours with research programme should be to develop industry-ready graduates with practical implementation skills, analytical abilities, startup orientation, freelancing capabilities, digital innovation awareness, and strong employability in modern IT and software industries.

SEE (50marks) Paper setting guidelines for all the semester

Question Paper contains 5 questions (each of 10 marks).Every question will be asked from respective unit as specified in the syllabus of each course. (i.e., Question-1 from Unit No.1 and remaining questions from their respective units). Every question is divided in three parts like (a), (b) and (c). Part (a) contains three objective type questions (not MCQ) like definition, reason, answer in one line, answer in one word etc., each of one mark and no internal option. Part (b) contains two questions each of two marks and student will attempt any one out of two. Part (c) contains two questions each of five marks and student will attempt any one out of two.

Saurashtra University **BCA Semester- 7, 8**					
Time:02:00				**Total marks: 50**	
Q.1	(a)	Attempt the following		Unit-1	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			
Q.2	(a)	Attempt the following		Unit-2	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			
Q.3	(a)	Attempt the following		Unit-3	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			

Q. 4	(a)	Attempt the following		Unit - 4	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			
Q.5	(a)	Attempt the following		Unit-5	03
		(1)			
		(2)			
		(3)			
	(b)	Attempt any one of the following			02
		(1)			
		(2)			
	(c)	Attempt any one of the following			05
		(1)			
		(2)			

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –41: UI/UX Design and Interactive Systems	4	50	50	100
2	Major	CS –42: Research Methodology In Computing	4	50	50	100
3	Major	CS –43: Mobile Application Development with Flutter	4	50	50	100
4	Minor	CS –44: Practical Based on CS-41, CS-43	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –45: Internship / Industry Project / Research project	6	0	150	150
		Total	**22**			**550**

BCA SEM 8						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –46: Advanced Research Methods	4	50	50	100
2	Major	CS –47: Data Science using Python	4	50	50	100
3	Major	CS –48: React JS & Express JS	4	50	50	100
4	Minor	CS –49: Practical Based on CS-47, CS-48	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –50: Internship / Industry Project / Research project	6	0	150	150
		Total	**22**			**550**

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –41: UI/UX Design and Interactive Systems	4	50	50	100
2	Major	CS –42: Research Methodology In Computing	4	50	50	100
3	Major	CS –43: Mobile Application Development with Flutter	4	50	50	100
4	Minor	CS –44: Practical Based on CS-4, CS-43	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –45: Internship / Industry Project / Research project	6	0	150	150
		Total	**22**			**550**

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –41: UI/UX Design and Interactive Systems

Objectives:

- To provide students with practical knowledge of User Interface (UI) design, User Experience (UX) principles, responsive web technologies, and modern frontend development practices.
- Focuses on designing interactive, user-friendly, and responsive digital interfaces using modern tools and frontend frameworks.
- To develop creative, analytical, and implementation skills required in current web and application development industries.

Prerequisites:

- Basic Knowledge of :
  - Computer applications, internet technologies, introductory web technologies.
- Basic familiarity of :
  - HTML/CSS Concepts, website structures, user interaction systems

Unit No	Topic	Details
1	UI / UX Design Principles	● Introduction to UI and UX ▪ Concept and importance of UI design ▪ Concept and importance of UX design ▪ Difference between UI and UX ▪ Evolution of modern digital interface systems ● User-Centered Design (UCD) ▪ Principles of user-centered Design ▪ Understanding user requirements ▪ User behavior and interaction patterns ▪ User journey concepts ● Visual Design Principles ▪ Layout and alignment techniques ▪ Typography fundamentals ▪ Color theory and combinations ▪ White space and visual hierarchy ▪ Imagery and iconography ● Responsive Design Concepts ▪ Responsive interface principles ▪ Mobile-first design approach ▪ Multi-device compatibility concepts
2	User Research and Analysis	● User Research Methods ▪ Introduction ▪ Qualitative and quantitative research methods ▪ Observation and survey methods

		▪ User interview techniques ● User Analysis Techniques ▪ User personas ▪ User behavior analysis ▪ Requirement gathering methods ▪ User expectation mapping ● Information Architecture ▪ Introduction ▪ Content organization techniques ▪ Navigation structures ▪ Sitemap planning ● User Data Analytics ▪ User feedback collection methods ▪ Basic analytics concepts ▪ Using user data for design decisions ▪ Interface improvement strategies
3	UI/UX Design Tools and Prototyping	● UI/UX Design Tools ▪ Introduction to Figma ▪ Adobe XD basics ▪ Canva for interface design ▪ Collaborative design platforms ● Wireframing Concepts ▪ Low-fidelity wireframes ▪ High-fidelity wireframes ▪ Interface structure planning ▪ Grid Systems and alignment ● Prototyping Techniques ▪ Interactive prototyping ▪ Clickable interfaces ▪ Responsive prototype development ▪ Mobile application prototyping ● Collaborative Design ▪ Team-based design workflow ▪ Design sharing and collaboration ▪ Feedback integration techniques
4	Visual Design and Interactive UI Components	● Advanced Visual Design ▪ Advanced layout techniques ▪ Visual balance and composition ▪ Modern design trends ▪ Dark mode interfaces ● UI Components ▪ Buttons and navigation systems

		▪ Cards and panels ▪ Forms and input systems ▪ Dashboard components ● Design Patterns and Frameworks ▪ Design systems ▪ Component libraries ▪ Reusable interface patterns ▪ Accessibility standards ● Motion Design and Interaction ▪ Animation principles ▪ Interactive transitions ▪ Motion design basics ▪ Multi-platform interface design
5	UI/UX Evaluation and Iteration	● Usability Testing ▪ Introduction ▪ Testing methods and techniques ▪ User Observation methods ▪ Task-based usability testing ● Heuristic Evaluation ▪ Heuristic principles ▪ Interface evaluation techniques ▪ Problem Identification Methods ▪ UI improvement strategies ● Feedback Analysis ▪ User feedback interpretation ▪ Data-driven design improvements ▪ Communication and presentation strategies ● Iterative Design Process ▪ Continuous improvement techniques ▪ A/B testing basics ▪ UI performance metrics ▪ Design refinement workflow

Reference Books

(1) Don’t Make Me Think – Steve Krug
(2) The Design of Everyday Things – Don Norman
(3) UI/UX Design Essentials – Daniel Walter Scott
(4) Responsive Web Design with HTML5 and CSS – Ben Frain
(5) Learning Web Design – Jennifer Niederst Robbins
(6) The Principles of Beautiful Web Design, Third Edition, Jason Beaird, O’Reilly, 2014
(7) UX/UI Design 2021-22 Tutorial the Complete Step by Step Guide to UX/UI Design and Best Practices for designers with no experience, Tom Mulligan

Course Outcome:

After successful completion of this course, students will be able to:

- Understand the principles of UI and UX design
- Design user-friendly and responsive digital interfaces
- Conduct user research and usability analysis
- Develop wireframes, prototypes and modern UI frameworks
- Apply visual design principles and modern UI frameworks
- Evaluate and improve user experiences using testing and feedbacktechniques

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –42: Research Methodology In Computing

Objectives:

- To provide students with fundamental knowledge of research methodology, research design, problem identification, analytical thinking, and scientific approaches used in computing and digital technology domains.
- Focuses on research processes, data collection methods, literature review techniques, research ethics, analytical interpretation, and documentation practices required for computing-related research and project development.

Prerequisites:

- Basic understanding of:
  - Computer applications, analytical thinking
- Basic familiarity with:
  - Project work, Data handling, Documentation practices

Unit No.	Topic	Detail
1	Introduction to Research and Computing Research	● Introduction to Research ▪ Concept and importance of research ▪ Objectives of research ▪ Characteristics of good research ▪ Types of research ● Research in Computing ▪ Computing and technology-based research ▪ Applied research in software and digital systems ▪ Industry-oriented research concepts ▪ Innovation and problem-solving approaches ● Research Process ▪ Steps in research methodology ▪ Problem identification ▪ Research planning concepts ▪ Research workflow ● Research Terminology ▪ Variables and hypotheses ▪ Research assumptions ▪ Scope and limitations ▪ Research outcomes
2	Research Problem Identification and Literature Review	● Research Problem Identification ▪ Selection of research topics ▪ Problem statement preparation ▪ Research objectives formulation ▪ Research questions ● Literature Review ▪ Concept and importance of literature review ▪ Sources of literature ▪ Research journals and conference papers ▪ Online research database ● Research Gap Identification ▪ Comparative study methods ▪ Identifying research gaps

		▪ Analytical review techniques ▪ Critical evaluation of literature ● Citation and Referencing ▪ Importance of referencing ▪ Citation styles overview ▪ Bibliography preparation ▪ Plagiarism awareness
3	Research Design and Data Collection Methods	● Research Design ▪ Exploratory research ▪ Descriptive research ▪ Experimental research ▪ Case study methods ● Data Collection Methods ▪ Primary and Secondary data ▪ Surveys and questionnaires ▪ Observation methods ▪ Interview techniques ● Sampling Techniques ▪ Introduction to sampling ▪ Random sampling ▪ Non-random sampling ▪ Sample size concepts ● Data Organization ▪ Data classification, Data tabulation ▪ Data validation concepts ▪ Data preparation for analysis
4	Research Analysis, Ethics and Documentation	● Data Analysis Concepts ▪ Introduction to data analysis ▪ Analytical interpretation methods ▪ Comparative analysis ▪ Graphical representation basics ● Research Ethics ▪ Importance of research ethics ▪ Ethical research practices ▪ Academic integrity ▪ Responsible use of digital resources ● Plagiarism and Publication Ethics ▪ Concept of plagiarism ▪ Plagiarism prevention ▪ Copyright and intellectual property basics ▪ Ethical publication practices ● Technical Documentation ▪ Research report structure ▪ Abstract presentation techniques ▪ Conclusion and future scope writing

5	Modern Computing Research and Case Studies	● Emerging Areas of Computing Research ▪ Artificial Intelligence applications ▪ Data analytics and visualization research ▪ Cloud and distributed systems research ▪ Blockchain and smart system research ● Research Project Planning ▪ Proposal preparation ▪ Research implementation planning ▪ Timeline preparation ▪ Resource management ● Case Studies in Computing Research ▪ Analysis of computing research projects ▪ Industry-oriented research examples ▪ Innovation-driven research case studies ● Future Trends in Computing Research ▪ Digital transformation research ▪ Smart systems and automation research ▪ Future opportunities in computing research

Reference Books:

- Research Methodology – C. R. Kothari
- Research Methods for Computer Science – Greg Anderson
- Practical Research Methods – Catherine Dawson
- Research Methodology in Computer Applications – Suresh Basandra
- The Craft of Research – Wayne C. Booth
- Research and Publication Ethics – Shailaja Rego

Course Outcome:

After successful completion of this course, students will be able to:

- Understand the concepts and importance of research in computing.
- Identify and formulate research problems in digital and computing domains.
- Apply research methodologies and data collection techniques.
- Conduct literature review and analytical studies.
- Understand research ethics and responsible research practices.
- Prepare research proposals, reports, and technical documentation.

Suggested Activities

Students may be encouraged to perform:

- literature review preparation,
- research paper analysis,
- research proposal writing,
- analytical report preparation,
- citation and referencing activities,
- seminar presentations,
- and mini research-based studies.

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –43: Mobile Application Development with Flutter		

Objectives:
• To provide students with practical knowledge of cross-platform mobile application development using Flutter framework and modern mobile UI technologies.
• Focuses on mobile interface design, widget-based application development, navigation systems, data handling, Firebase integration, API connectivity, and deployment concepts used in modern mobile application environments.
Prerequisites:
• Basic Knowledge of:
  ◦ Programming concepts, Web technologies, Databases
• Basic understanding of:
  ◦ Object-oriented programming, APIs, Responsive interface concepts		

Unit No	Topic	Details
1	Introduction to Flutter and Dart Programming	● Introduction to Mobile Applications ▪ Concept and importance of mobile applications ▪ Types of mobile applications ▪ Native and cross-platform applications ▪ Mobile application development lifecycle ● Introduction to Flutter ▪ What is Flutter? ▪ Features and advantages of Flutter ▪ Flutter framework architecture ▪ Working of Flutter and Dart ▪ Widget-based application development ● Flutter Development Environment ▪ Flutter SDK installation ▪ Android Studio and VS Code integration ▪ Flutter project structure ▪ Organizing files and folders ▪ Hot Reload concept ▪ Themes in Flutter applications ● Introduction to Dart Programming ▪ Variables and data types ▪ Operators and expressions ▪ Decision-making statements ▪ Looping statements ▪ Functions and collections ▪ Object-oriented programming concepts
2	Flutter Widgets,	● Introduction to Widgets ▪ Stateless widgets

	Layouts and Navigation Systems	▪ Stateful widgets ▪ Widget tree concepts ▪ Widget build visualization ● Layout Systems ▪ Single child widgets ▪ Multiple child widgets ▪ Rows and columns ▪ Containers and padding ▪ Responsive layout concepts ▪ Advanced layout systems ● Gestures and User Interaction ▪ Introduction to gestures ▪ User interaction handling ▪ Buttons and input widgets ▪ Form handling and validation ● Navigation and State Management ▪ Navigation concepts ▪ Routing systems ▪ Multi-screen applications ▪ Drawer and bottom navigation ▪ Ephemeral state management ▪ Application state concepts
3	Database Systems and Firebase Integration	● Local Database Concepts ▪ Introduction to SQLite ▪ Local data storage concepts ▪ Data persistence techniques ▪ CRUD operation concepts ● Firebase Integration ▪ Introduction to Firebase and Firebase project setup ▪ Cloud Firestore basics ▪ Realtime database concepts ▪ Firebase authentication systems ● Internationalization and Packages ▪ Internationalization concepts ▪ Using intl package ▪ External package import concepts ▪ Flutter package management ● Testing and Development Tools ▪ Introduction to testing in Flutter ▪ Widget testing basics ▪ Application debugging concepts ▪ Flutter DevTools basics

		▪ Flutter SDK utilities
4	API Integration and Advanced Mobile Features	● REST API Concepts ▪ Introduction to APIs, REST API architecture ▪ HTTP request and response, JSON parsing techniques ● API Integration ▪ Accessing web services ▪ Product service API concepts ▪ Data fetching and display ▪ Error handling concepts ● Asynchronous Programming ▪ Future and async concepts ▪ Asynchronous data loading ▪ Background processing basics ● Multimedia and Device Features ▪ Image and multimedia integration ▪ Camera and gallery access basics ▪ Location and map integration concepts ▪ Push notification basics
5	Flutter Animation, Deployment and Real-World Applications	● Flutter Animation ▪ Introduction to animation concepts ▪ Animation-based classes, Animation workflow in Flutter ▪ Transition and motion effects ● Packages and Plugins ▪ Introduction to packages ▪ Types of packages ▪ Using Dart packages ▪ Flutter plugin concepts ● Deployment Concepts ▪ APK generation ▪ Android application deployment ▪ Introduction to iOS application deployment ▪ Application publishing concepts ● Real-World Mobile Applications ▪ Educational mobile applications ▪ E-commerce mobile systems ▪ Healthcare and productivity applications ▪ Modern trends in mobile computing

Reference Books

(1) Flutter for Beginners – Alessandro Biessek
(2) Beginning Flutter – Marco Napoli
(3) Flutter Complete Reference – Alberto Miola
(4) Dart Programming Language Specification – Dart Team

(5) Mobile Application Development using Flutter – Frank Zammetti
(6) Firebase Essentials – John Horton

Course Outcome:

After successful completion of this course, students will be able to:

- Understand the concepts and architecture of Flutter applications.
- Develop cross-platform mobile applications using Flutter and Dart.
- Design responsive mobile interfaces using widgets and layouts.
- Implement navigation, forms, and state management in Flutter applications.
- Integrate Firebase, SQLite, and REST APIs in mobile applications.
- Develop and deploy basic real-world mobile applications.

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –44: Practical Based on CS-41, CS-43	
CCE- Continuous and comprehensive Evaluation as follow • The continuous Comprehensive Evaluation (CCE) for each subject will be conducted by the teacher of that subject. The teacher will decide how the evaluation will be done. Usually CCE includes things like class participation, case studies and presentation, assignments, tutorials, small test (announced or surprised), quizzes and attendance or a mix of these. • Students must submit their work for internal evaluation on time to time. • Another part of CCE is the mid-term exam, which is compulsory for all students. This exam will be conducted internally by the college.	50 Marks
SEE – Semester End Examination as per the following • Practical exams may be scheduled before or after the theory examinations. Exam is conducted by Saurashtra University using external examiner (3 hours duration) • Students must prepare a practical notebook/book for the final practical examination. (The practical book serves as a record of all practical work, observations, procedures and results performed during the semester in lab. It is essential for evaluation during the final practical examination)	50 Marks

CS –44: Practical Based on CS-41, CS-43	**Total Marks - 100**	
Topics	**CCE**	**SEE**
CS-41:UI/UX Design and Interactive Systems	25	25
CS-43:Mobile Application Development with Flutter	25	25

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

<table>
<tr><th colspan="4">CS –45: Internship / Industry Project / Research project</th></tr>
<tr><td colspan="4">Objectives:
• To apply theoretical concepts learned in previous semesters to practical software development.
• To gain hands-on experience in using programming languages, tools, and technologies.
• To understand and implement the complete Software Development Life Cycle (SDLC).
• To enhance problem-solving and analytical skills by designing solutions to real-life problems.
• To develop teamwork and collaboration skills through group-based project execution.
• To improve project planning and time management abilities.
• To practice documentation and reporting skills as per industry standards.
• To prepare students for major projects, internships, or professional roles in software development.</td></tr>
<tr><th>No.</th><th>Topic</th><th colspan="2">Guidelines for Internship</th></tr>
<tr><td>1</td><td>General</td><td colspan="2">• Students must get prior approval from the college before starting the internship / project topic.
• Choose an Internship/project topic
• Projects can be individual or group-based (2 Students)(as per college policy)
• Submit a project proposal with the title, objective, scope, tools/technologies to be used, and team members.
• Get the proposal approved by the project guide/faculty coordinator.
• Use any suitable programming language or platform
• Project must be developed in any Company / Organization /Institute or computer laboratory of college.
• Students must maintain a daily activity log during the internship.</td></tr>
<tr><td>2</td><td>Documentation</td><td colspan="2">Maintain a project report including the following:
• Title Page
• Certificate
• Acknowledgment
• Table of Contents
• Introduction & Objective
• System Analysis (Problem Definition, Feasibility Study)
• System Design (DFD, ER Diagram, etc.)
• Data Dictionary
• Implementation (Screenshots)
• Testing (Test cases and results)
• Conclusion & Future Scope</td></tr>
<tr><td>3</td><td>Submission</td><td colspan="2">• Submit a soft copy (CD/ Pen drive) and hard copy of the project report. Project report must be printed on both side of the page.
• Project must be submitted before two weeks of commencement of theory exam
• During the project viva examination project must be run.</td></tr>
<tr><td>4</td><td>Assessment Criteria</td><td>SEE –Exam is conducted by Saurashtra University using external examiner (examiner will evaluate the executable copy, hard copy of the project and take the viva voce)</td><td>150 Marks</td></tr>
</table>

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 8						
Sr. No	**Type of Course**	**Subject**	**Credit**	**CCE**	**SEE**	**Total**
1	Major	CS –46: Advanced Research Methods	4	50	50	100
2	Major	CS –47: Data Science using Python	4	50	50	100
3	Major	CS –48: React JS & Express JS	4	50	50	100
4	Minor	CS –49: Practical Based on CS-47, CS-48	4	50	50	100
5	Dissertation/ Research Project/OJT	CS –50: Internship / Industry Project / Research project	6	0	150	150
		Total	**22**			**550**

CS –46: Advanced Research Methods

Objectives:

- To provide students with advanced knowledge of research methodologies, analytical techniques, experimental approaches, and applied computing research practices used in modern digital and computing environments.
- Focuses on advanced research design, quantitative and qualitative analysis techniques, experimental validation, research interpretation, analytical tools, technical reporting, and publication-oriented research practices required for higher studies and computing-related research activities.

Prerequisites:

- Basic understanding of:
  - research methodology, data analysis concepts

Unit No.	Topic	Detail
1	Advanced Research Design and Analytical Approaches	● Advanced Research Concepts ▪ Advanced research methodologies ▪ Interdisciplinary research concepts ▪ Applied computing research approaches ▪ Research innovation concepts ● Research Design Techniques ▪ Advanced research design models ▪ Experimental and analytical research approaches ▪ Comparative research methods ▪ Design of research frameworks ● Research Variables and Validation ▪ Independent and dependent variables ▪ Research assumptions and limitations ▪ Research validation concepts ▪ Reliability and consistency concepts ● Research Planning ▪ Advanced research planning techniques ▪ Timeline and workflow preparation ▪ Resource planning concepts ▪ Research implementation workflow

2	Quantitative and Qualitative Research Techniques	● Quantitative Research Techniques ▪ Quantitative data analysis concepts ▪ Sampling and survey techniques ▪ Statistical interpretation basics ▪ Numerical data interpretation ● Qualitative Research Techniques ▪ Qualitative analysis methods ▪ Observation and interview techniques ▪ Case study analysis methods, Content analysis concepts ● Mixed Research Approaches ▪ Mixed methodology concepts ▪ Comparative analytical approaches ▪ Integrated research techniques ▪ Cross-disciplinary research methods ● Data Organization and Validation ▪ Research data organization ▪ Data consistency and reliability ▪ Data validation methods ▪ Analytical data preparation
3	Experimental Design and Data Interpretation	● Experimental Research Methods ▪ Experimental design concepts ▪ Controlled and uncontrolled experiments ▪ Hypothesis validation techniques ▪ Experimental workflow systems ● Data Interpretation Techniques ▪ Comparative analysis methods ▪ Trend and pattern analysis ▪ Analytical interpretation techniques ▪ Result evaluation methods ● Statistical and Graphical Representation ▪ Statistical representation basics ▪ Graphs and visualization techniques ▪ Data presentation methods ▪ Interpretation of analytical results ● Research Findings and Conclusions ▪ Interpretation of research outcomes ▪ Research conclusion techniques ▪ Future scope identification ▪ Recommendation preparation methods

4	Research Tools, Technical Writing and Publication Systems	● Digital Research Tools ▪ Research databases and digital libraries ▪ Reference management systems ▪ Online research resources ▪ Digital analytical tools ● Research Software Concepts ▪ Spreadsheet analysis tools ▪ Data visualization platforms, Statistical tool concepts ▪ Documentation and formatting tools ● Technical Writing and Reporting ▪ Technical research writing techniques ▪ Research paper structure, ▪ Abstract and summary preparation ▪ Technical documentation systems ● Research Publication Systems ▪ Journal and conference publications ▪ Research indexing concepts ▪ Citation and referencing systems ▪ Ethical publication practices
5	Applied Computing Research and Emerging Research Trends	● Applied Computing Research ▪ Research in software systems ▪ Educational and business computing research ▪ Data-driven computing research ▪ Industry-oriented research concepts ● Emerging Research Areas ▪ Artificial Intelligence research trends ▪ Cloud and distributed systems research ▪ IoT and smart systems research ▪ Blockchain and digital platform research ● Research Project Implementation ▪ Research project planning ▪ Research implementation workflow ▪ Resource and timeline management ▪ Research presentation techniques ● Case Studies and Future Research Opportunities ▪ Analysis of computing research projects ▪ Innovation-driven research case studies ▪ Emerging digital research trends ▪ Future opportunities in computing research

Reference Books

(1) Research Methodology – C. R. Kothari
(2) Research Design – John W. Creswell
(3) Practical Research Methods – Catherine Dawson
(4) Business Research Methods – Donald Cooper
(5) Research Methods for Computer Science – Greg Wilson
(6) Quantitative and Qualitative Research Approaches – John Creswell

Course Outcome:

After successful completion of this course, students will be able to:

- Understand advanced research methods and analytical techniques.
- Design and plan research studies in computing domains.
- Apply quantitative and qualitative research approaches.
- Analyze and interpret research data effectively.
- Use research tools and technical documentation practices.
- Identify emerging trends and perform applied computing research activities.

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

CS –47: Data Science using Python

Objectives:

- Understanding the fundamental concepts of Data Science and the data analytics lifecycle.
- Familiarize students with statistical methods, measures of central tendency, and dispersion.
- Enable students to explore, analyze, and visualize data effectively using Python libraries.
- Understand the principles of correlation, exploratory data analysis (EDA), and statistical quality control.

Prerequisites:

- Basic Computer Skills.
- Fundamental Programming Concepts (Python basics).
- Basic understanding of Mathematics and Statistics.

Unit No	Topic	Details
1	Introduction & Exploring Data	• Introduction to Data Science & Life Cycle • Definition of Data Science • Importance of Data Science • Applications of Data Science • Data Science Life Cycle • Roles of a Data Scientist • Data Science vs Data Analytics • Python ecosystem for Data Science (NumPy, Pandas overview). • Explore Data using Python • Understanding Characteristics of the dataset • Size, Shape, value counts • Data Types and change data types.
2	Descriptive Statistics	• (Merged Measures of Central Tendency & Dispersion) Arithmetic Mean, Median, Mode. • Relationship between mean, median and mode. Computation of the measures for grouped and ungrouped data. • Range, mean deviation and standard deviation • coefficient of variation and its use • Quartiles and Inter quartile range • Quintiles, Deciles and Percentiles • Skewness and Kurtosis and their uses.
3	Conducting EDA using Python	▪ Data Analysis using Python ▪ Handling missing data ▪ Computing metrics ▪ Analysis & Interpretation of the data and connected visualization.

4	Understanding Correlation	▪ Introduction to Covariance, ▪ Understanding correlation ▪ calculating correlation using Pandas ▪ Interpreting a correlation, Matrix.
5	Statistical Quality Control	▪ Nature of Control Limits ▪ Purpose of Control Charts ▪ Control Charts for Variables ▪ Control Charts for Attributes.

Reference Books

1. Hands-On Data Science and Python Machine Learning, Kane Frank, Packt Publishing Limited.
2. Data Science and Machine Learning Interview Questions Using Python a Complete Question Bank to Crack Your Interview, Vishwanathan Narayanan, BPB Publications.
3. Python for Data Science, Tony F Charles.
4. Data Structures using Python, Shriram K. Vasudevan, OUP India.
5. Python Data Science Handbook, Jake Vanderplas, O'Reilly Media.

Course Outcome:

- Understand the data science lifecycle and explore datasets using Python.
- Calculate and interpret measures of central tendency and dispersion.
- Perform Exploratory Data Analysis (EDA) and handle missing or messy data.
- Compute and analyze correlations between different data variables.
- Implement statistical quality control charts for data-driven decision-making.

CS –48: React JS & Express JS

Objectives:

- Articulate what React is and why it is useful.
- Explore various attributes of web development.
- Explore the basic architecture of a React application.
- Explore various web development techniques of this Java Script and would learn new techniques based on industry requirement.
- Gain a deep understanding of JSX and Hooks.

Prerequisites:

- JavaScript, HTML, CSS and OOPs

Unit No	Topic	Details
1	Express JS & Java Script	• Java Script • Java Script Overview & Basics • Variable, Conditional Statements, Loops in JS, • Functions, Arrays & Events in JS • ES6 Overview & Basics • ES6 Classes, functions & Promises • Express JS • Setting up an app with ExpressJS, Routing in ExpressJS, Connecting views with templates, configurations and error handling.
2	Introduction to JSX & REACT JS	• Introduction: What is ReactJS? Installation or Setup, Hello World Program, Create a first app, folder structure • Components: Creating components, Basic components, Nesting components, functional component, class component • Introduction to JSX: JSX Programs • Props: ReactJS Props, React State, Destructing Props and State, setState, methods as Props.
3	Form Handling, components and fragments	• Event Handling: Event Handling and Binding event handlers • Rendering: Conditional Rendering and List Rendering, List and keys, Index as Key Anti-pattern • Introduction: Basic form handling • Components: Components Life Cycle Methods, Components Mounting Lifecycle methods, Components Updating Lifecycle methods, Pure Components • Fragments

4	Memo, Refs, Props and Context	• Memo • Introduction to Refs: Refs, Refs with Class Components, Forwarding Refs and Portals • Components: Higher Order Components • Props Again! Rendering Props and Context • HTTP: HTTP and React, GET and React, POST and React.
5	Introduction to Hooks and its implementation	• Introduction: React Hooks introduction, useState Hook, useState Previous state, useState with object, useState with array. • useEffect: useEffect Hook, useEffect after render, Condionally run effects, run effects only once, useEffect with cleanup, useEffect with incorrect dependency. • Fetching data: Fetching data with useEffect, useContext Hook • useReducer Hook: useReducer – simple state and action, complex state and action, multiple useReducers • useContext: useContext, useReducer, Fetching data with useReduer, useState vs useReducer

Reference Books

1. Learning React,Martin Bean,Kirupa Chinna thambi Pearson Addison Wesley
2. ReactJS Notes for Professional,Goal Kicker,Web site ebook,
3. The Road to React_Your journey to master plain yet pragmatic React,Lean Pub Book, Robin Wieruch -Independently Published (2020)

CS –49: Practical Based on CS-47, CS-48	
CCE- Continuous and comprehensive Evaluation as follow • The continuous Comprehensive Evaluation (CCE) for each subject will be conducted by the teacher of that subject. The teacher will decide how the evaluation will be done. Usually CCE includes things like class participation, case studies and presentation, assignments, tutorials, small test (announced or surprised), quizzes and attendance or a mix of these. • Students must submit their work for internal evaluation on time to time. • Another part of CCE is the mid-term exam, which is compulsory for all students. This exam will be conducted internally by the college.	50 Marks
SEE – Semester End Examination as per the following • Practical exams may be scheduled before or after the theory examinations. Exam is conducted by Saurashtra University using external examiner (3 hours duration) • Students must prepare a practical notebook/book for the final practical examination. (The practical book serves as a record of all practical work, observations, procedures and results performed during the semester in lab. It is essential for evaluation during the final practical examination)	50 Marks

CS –49: Practical Based on CS-47, CS-48	**Total Marks - 100**	
Topics	**CCE**	**SEE**
CS-47:Data Science using Python	25	25
CS– 48: REACTJS & EXPRESSJS	25	25

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

<table>
<tr><th colspan="4">CS –50: Internship / Industry Project / Research project</th></tr>
<tr><td colspan="4">Objectives:
• To apply theoretical concepts learned in previous semesters to practical software development.
• To gain hands-on experience in using programming languages, tools, and technologies.
• To understand and implement the complete Software Development Life Cycle (SDLC).
• To enhance problem-solving and analytical skills by designing solutions to real-life problems.
• To develop teamwork and collaboration skills through group-based project execution.
• To improve project planning and time management abilities.
• To practice documentation and reporting skills as per industry standards.
• To prepare students for major projects, internships, or professional roles in software development.</td></tr>
<tr><th>No.</th><th>Topic</th><th colspan="2">Guidelines for Internship</th></tr>
<tr><td>1</td><td>General</td><td colspan="2">• Students must get prior approval from the college before starting the internship / project topic.
• Choose an Internship/project topic
• Projects can be individual or group-based (2 Students)(as per college policy)
• Submit a project proposal with the title, objective, scope, tools/technologies to be used, and team members.
• Get the proposal approved by the project guide/faculty coordinator.
• Use any suitable programming language or platform
• Project must be developed in any Company / Organization /Institute or computer laboratory of college.
• Students must maintain a daily activity log during the internship.</td></tr>
<tr><td>2</td><td>Documentation</td><td colspan="2">Maintain a project report including the following:
• Title Page
• Certificate
• Acknowledgment
• Table of Contents
• Introduction & Objective
• System Analysis (Problem Definition, Feasibility Study)
• System Design (DFD, ER Diagram, etc.)
• Data Dictionary
• Implementation (Screenshots)
• Testing (Test cases and results)
• Conclusion & Future Scope</td></tr>
<tr><td>3</td><td>Submission</td><td colspan="2">• Submit a soft copy (CD/ Pen drive) and hard copy of the project report. Project report must be printed on both side of the page.
• Project must be submitted before two weeks of commencement of theory exam
• During the project viva examination project must be run.</td></tr>
<tr><td>4</td><td>Assessment Criteria</td><td>SEE –Exam is conducted by Saurashtra University using external examiner (examiner will evaluate the executable copy, hard copy of the project and take the viva voce)</td><td>150 Marks</td></tr>
</table>

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7: CS–41:UI/UX Design and Interactive Systems

Note:

- The practical exercises of this subject shall be performed using standard computer laboratory facilities and modern UI/UX design tools.
- Students may use software/tools/platforms such as Figma, Canva, Adobe XD, HTML5, CSS3, Bootstrap, Tailwind CSS, JavaScript, and standard web browsers for practical implementation and frontend development activities.
- Open-source, free, or equivalent software/tools may be used wherever applicable.
- The practical work should focus on interface design, responsive frontend systems, prototyping, usability evaluation, and frontend implementation concepts.

Minimum Following Exercises Should be Performed by the Students During the Semester

Unit – I: UI / UX Design Principles

1. Study and analysis of modern website interfaces based on UI and UX principles.
2. Design of typography combinations for educational and business applications.
3. Preparation of color palettes and visual themes for digital interfaces.
4. Development of visual hierarchy layouts using spacing, typography, and alignment principles.
5. Preparation of responsive layout structures for desktop and mobile interfaces.
6. Design of user journey maps for online registration or e-commerce systems.
7. Comparative analysis of good and poor interface designs.
8. Design of responsive navigation structures and menu systems.
9. Accessibility evaluation of selected websites and applications.
10. User-centered interface planning for digital platforms.

Unit – II: User Research and Analysis

11. Creation of user personas for educational or business applications.
12. Preparation of online survey forms for collecting user feedback.
13. Conduct of user interviews and preparation of analysis reports.
14. Development of requirement analysis documents based on user feedback.
15. Preparation of sitemap and information architecture layouts for websites.
16. Content organization and navigation flow analysis for digital systems.
17. Observation and analysis of user interaction patterns with selected interfaces.
18. Preparation of usability and user behavior study reports.
19. Comparative analysis of frontend navigation systems.
20. Preparation of UX research documentation for selected applications.

Unit – III: UI / UX Design Tools and Prototyping

21. Development of low-fidelity wireframes using modern UI/UX tools.
22. Design of high-fidelity interface prototypes using Figma or similar tools.

23. Creation of responsive mobile application prototypes.
24. Development of interactive clickable prototypes for frontend systems.
25. Design of dashboard interface prototypes for analytical applications.
26. Creation of e-commerce frontend UI prototypes.
27. Development of reusable UI component libraries.
28. Collaborative interface design activity using shared platforms.
29. Responsive prototype development for multiple screen sizes.
30. Complete UI prototype preparation for real-world applications.

Unit – IV: Visual Design and Interactive UI Components

31. Design of responsive forms and input interface systems.
32. Development of modern navigation bars and interactive menus.
33. Creation of dashboard cards, panels, and frontend UI components.
34. Design of dark mode interface systems.
35. Development of animated buttons and interactive frontend elements.
36. Preparation of reusable frontend component libraries.
37. Accessibility improvement activity for selected interfaces.
38. Design of multi-platform interface layouts for desktop and mobile devices.
39. Implementation of motion design and transition effects in frontend systems.
40. Development of interactive frontend mini applications integrating advanced UI components.

Unit – V: UI / UX Evaluation and Iteration

41. Conduct of usability testing for selected websites or applications.
42. Preparation of heuristic evaluation reports for digital interfaces.
43. User feedback collection and interface improvement analysis.
44. Comparative A/B testing of frontend interface designs.
45. Preparation of UI redesign proposals based on usability findings.
46. Interface optimization activity based on accessibility standards.
47. Analysis of frontend responsiveness and performance.
48. Preparation of UI evaluation presentations and reports.
49. Iterative redesign of interfaces based on testing observations.
50. Development and presentation of a final mini project integrating wireframing, prototyping, responsive frontend design, usability evaluation, and interface optimization concepts covered in the course.

Saurashtra University
B.C.A. (Honours with research) (Sem - 7 and Sem - 8)
To be effective from June – 2026

BCA SEM 7: CS–43:Mobile Application Development With Flutter

Note:

- The practical exercises of this course shall be performed using standard computer laboratory facilities and modern mobile application development tools and platforms.
- Students may use software/tools/platforms such as Flutter SDK, Dart, Android Studio, VS Code, Android Emulator, Firebase, SQLite, REST APIs, and related mobile development tools for practical implementation and application development activities.
- Open-source, free, educational, or equivalent software/tools may be used wherever applicable.
- The practical work should focus on mobile user interface development, widget-based application design, responsive layouts, state management, database integration, Firebase services, REST API connectivity, application testing, and cross-platform mobile application development practices.

Unit – I: Introduction to Flutter and Dart Programming

1. Installation and configuration of Flutter SDK and development tools.
2. Creation and execution of basic Flutter applications.
3. Study and analysis of Flutter project architecture and file structure.
4. Demonstration of Hot Reload and Flutter themes.
5. Implementation of Dart programming examples using variables and operators.
6. Development of conditional and looping statement programs in Dart.
7. Implementation of function-based Dart applications.
8. Demonstration of object-oriented programming concepts in Dart.
9. Development of simple console-based Dart applications.
10. Emulator/device setup and testing activity.
11. Suggested Tools / Platforms:
12. Flutter SDK, Dart, Android Studio, VS Code, Android Emulator.

Unit – II: Flutter Widgets, Layouts and Navigation Systems

13. Design of mobile interfaces using stateless and stateful widgets.
14. Development of responsive layouts using rows, columns, and containers.
15. Creation of form-based mobile applications.
16. Implementation of button and gesture handling systems.
17. Development of multi-screen navigation systems.
18. Design of drawer and bottom navigation interfaces.
19. Implementation of form validation techniques.
20. Development of responsive mobile dashboards.
21. State management demonstration activity.
22. Development of mini UI-based Flutter applications.
23. Suggested Tools / Platforms:
24. Flutter SDK, Dart, Android Studio, VS Code.

Unit – III: Database Systems and Firebase Integration

25. SQLite database integration activity.
26. Development of CRUD operation applications.
27. Firebase project creation and configuration activity.
28. Integration of Firebase authentication systems.
29. Cloud Firestore database integration activity.
30. Data storage and retrieval using Firebase services.
31. Internationalization and localization activity using intl package.
32. External package integration demonstration.
33. Widget testing and debugging activity.
34. Development of Firebase-integrated mini applications.
35. Suggested Tools / Platforms:
36. Flutter SDK, Firebase Console, SQLite, Cloud Firestore, Android Studio.

Unit – IV: API Integration and Advanced Mobile Features

37. REST API integration in Flutter applications.
38. JSON data parsing and display activity.
39. Development of applications using HTTP requests and responses.
40. Asynchronous data loading activity using Future and async concepts.
41. Error handling implementation in Flutter applications.
42. Image and multimedia integration activity.
43. Camera and gallery access demonstration.
44. Location and map integration basics.
45. Push notification demonstration activity.
46. Development of API-based Flutter mini applications.
47. Suggested Tools / Platforms:
48. Flutter SDK, REST APIs, JSON Tools, Android Emulator.

Unit – V: Flutter Animation, Deployment and Real-World Applications

49. Animation and transition effect implementation in Flutter applications.
50. Package and plugin integration activity.
51. APK generation and deployment demonstration.
52. Development of educational or business mobile applications.
53. Creation of e-commerce mobile application interfaces.
54. Development of productivity or healthcare mobile systems.
55. Responsive mobile application optimization activity.
56. Case study analysis of modern Flutter applications.
57. Preparation of mobile application project documentation.
58. Development and presentation of a final Flutter-based mobile application integrating widgets, Firebase, APIs, animation, and deployment concepts covered in the course.