ITM UNIVERSITY

CENTRE FOR DISTANCE AND ONLINE EDUCATION

PROGRAMME PROJECT REPORT (PPR)

MASTER OF COMPUTER APPLICATIONS

MCA – ODL PROGRAMME

TWO-YEAR POSTGRADUATE DEGREE

PROPOSED SCHEME OF STUDY – 92 CREDITS

Effective from the academic year 2026–27 (Proposed)

ITM University, Gwalior
Centre for Distance and Online Education

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

PROGRAMME PROJECT REPORT (PPR)

I. Programme Profile

S. No.	Particulars	Proposed Details
1	Degree Awarding Body	ITM University, Gwalior
2	Teaching Institution	ITM University, Gwalior; Centre for Distance and Online Education (CDOE)
3	Name of the Programme	Master of Computer Applications (MCA)
4	Mode of Delivery	Open and Distance Learning (ODL)
5	Minimum Duration	2 Years / 4 Semesters
6	Maximum Duration	As per applicable University regulations
7	Minimum Eligibility	As per approved University admission regulations; detailed criteria subject to institutional confirmation
8	Programme Credits	92
9	Programme Level	Postgraduate / Master's
10	Nature of Target Group	Learners seeking flexible education in computer applications
11	Nature of Programme	Professional / Postgraduate
12	Board of Studies Approval	02 March 2026
13	Academic Council Approval	47th Academic Council, 01 June 2026

Implementation is subject to completion of the prescribed statutory process and applicable regulatory recognition. Programme-specific approval records and operating provisions remain subject to institutional confirmation.

II. Programme Mission and Objectives

To provide flexible, technology-enabled postgraduate education in computer applications that develops advanced computing knowledge, engineering and analytical competence, responsible use of contemporary technologies, professional practice and lifelong learning.

- Develop postgraduate competence in programming, algorithms, databases, application engineering, systems, networks, data/AI, cloud and security.
- Strengthen architectural, analytical and design thinking for solving non-trivial computing problems.
- Develop practical proficiency through laboratories, a structured mini project, elective specialization, industry/MOOC learning and a substantial major project/dissertation/industrial project.
- Build research literacy, technical writing, teamwork, software engineering, professional ethics and responsible-AI awareness.
- Enable self-directed upskilling through SLM, technical documentation, MOOCs, professional certifications and contemporary digital resources.

PEO	Expected Graduate Progression
PEO1	Contribute effectively in software/application engineering, data, AI, cloud/DevOps, cybersecurity or allied roles.
PEO2	Analyse organisational and user requirements and design reliable, secure, scalable and maintainable computing solutions.
PEO3	Demonstrate professional ethics, research literacy, communication, teamwork and project responsibility.
PEO4	Pursue lifelong learning, professional certification, entrepreneurship, research or higher studies.

III. Relevance of the Programme with the HEI's Mission and Goals

The programme supports access, employability, digital capability, innovation and continuing education by enabling eligible learners to undertake postgraduate computing study through a flexible ODL format.

DIRECTOR
Centre for Distance and Online Education (CDOE)
ITM University Gwalior

IV. Nature of Prospective Target Group of Learners

- Eligible graduates seeking postgraduate computer-application education.
- Working professionals seeking advancement in software, data, AI, cloud/DevOps or cybersecurity.
- Learners unable to attend regular programmes because of location, employment or family responsibilities.
- Career-transition learners seeking structured postgraduate computing competence.
- Learners preparing for technology entrepreneurship, certification, research or higher studies.

V. Appropriateness of the Programme to be Conducted in Online Mode

The programme is proposed in Open and Distance Learning (ODL) mode, supported by online teaching and learning resources. Online resources complement self-study, academic counselling and supervised practical assessment within the proposed ODL delivery arrangements.

Learning Need	ODL Design Response
Concept mastery	Unitised self-learning material with worked examples, diagrams, code traces, caselets, self-checks and answer guidance.
Programming/data practice	Locally installable/open-source tools, browser notebooks, datasets, starter code and scheduled practical counselling.
Network/cloud/security practice	Simulators, containers/VMs, provider sandboxes and strictly authorised lab environments.
Interaction	Scheduled academic counselling, discussion forums, doubt clearing, peer review and project clinics.
Research/project competence	Proposal, literature review, milestone reviews, repository/logbook evidence, testing and viva.
Authentication	Supervised skill checks, practical exams, oral defence, version evidence and learner-specific tasks.

VI. Programme Outcomes (PLOs)

PLO 1: Apply postgraduate computing, mathematical and programming knowledge.

PLO 2: Analyse complex computing problems and evaluate solution alternatives.

PLO 3: Design and implement reliable, secure and maintainable software/data/cloud solutions.

PLO 4: Use contemporary tools, platforms, automation and engineering practices.

PLO 5: Apply data science, AI, ML and generative-AI concepts responsibly.

PLO 6: Apply systems, network, cloud and cybersecurity principles.

PLO 7: Plan, implement, test, document and defend substantial computing projects.

PLO 8: Communicate technical/research ideas and work effectively in distributed teams.

PLO 9: Recognise ethical, legal, privacy, security, sustainability and societal responsibilities.

PLO 10: Engage in research-informed, self-directed and lifelong professional learning.

Centre for Distance and Online Education (CDOE) ITM University Gwalior

VII. Instructional Design

Course Curriculum

The MCA (ODL) is a two-year postgraduate programme designed to develop advanced competence in software and application engineering, algorithms and data systems, networks and security, data science and AI, cloud/DevOps, research-informed problem solving and major project work. The curriculum is progressive: Semester I establishes postgraduate computing foundations; Semester II focuses on application development and software engineering; Semester III develops advanced data, cloud and security competence; Semester IV emphasizes emerging technologies, specialization/certification and a substantial capstone.

L = Lecture, T = Tutorial and P = Practical/Project periods. Learning-hour allocations are subject to the University's approved ODL credit framework.

Learning Materials

Quadrant	Application
e-Tutorial	Short faculty concept videos, coding/data walkthroughs, problem-solving demonstrations and project clinics.
e-Content	SLM, lab manuals, notebooks, source code, datasets, readings, rubrics and downloadable resources.
Interaction	Counselling, discussion forums, peer critique, office hours/doubt clearing and mentor support.
Assessment	Self-checks, LMS quizzes, authentic assignments, labs, practicals, project milestones, viva and term-end exams.

- Pre-study: learner completes the prescribed SLM unit and embedded self-checks.
- Counselling: faculty addresses difficult concepts, examples, misconceptions and learner questions.
- Practice: learner performs coding/data/design/lab activity and submits evidence through LMS.
- Feedback: learner receives rubric-based feedback and remedial/enrichment resources.
- Mastery check: learner attempts a unit quiz/self-test before progression.
- Integration: learner completes an authentic course-level task, practical or project.

The course learning cycle applies to every course in the detailed syllabus. The Programme Coordinator should verify the availability, academic suitability and accessibility of supplementary MOOC, OER and video resources before each offering. Credit transfer is subject to approved University provisions.

Academic Review

The Programme Coordinator, course faculty and CIQA should review the curriculum, self-learning materials, practical support and assessment evidence periodically. Learner, alumni and industry feedback should inform documented revisions.

Outcome-Based Approach

Course outcomes should be mapped to programme outcomes and assessed through theory, practical and project evidence. Attainment analysis and learner feedback should inform corrective action in subsequent offerings.

Component	Requirement
Course identity	Code, title, L-T-P, credits, semester/category and approved assessment.
Prerequisites	Prior knowledge plus bridge/remedial support.
Objectives and COs	3-5 objectives and 4-6 measurable COs.
Unit plan	Five structured units or stages with detailed topics and applications.
ODL pedagogy	Pre-study, counselling, guided practice, feedback and mastery checks.
Assessment	CO-mapped authentic assignments, quizzes, practicals and examinations.
Practical/project	Lab manual, tools, evidence, safety/ethics, viva and authentication.
Resources	Textbooks, official documentation, SWAYAM/NPTEL and verified provider resources.
Integrity	Citation, plagiarism/code copying, data ethics, AI-use disclosure and authorship verification.

DIRECTOR
Centre for Distance and Online Education (CDOE)
ITM University Gwalior

VIII. Programme Structure: Master of Computer Applications (2 Years)

Semester I

Course Code	Course Title	Lecture	Tutorial	Practical	Credit
MCAOL 101	Mathematical Foundations for Computing	3	0	0	3
MCAOL 102	Programming and Problem Solving with Python	3	0	4	5
MCAOL 103	Database Systems	3	0	2	4
MCAOL 104	Data Structures and Algorithms	3	0	4	5
MCAOL 105	Computer Systems and Operating Systems	3	0	2	4
MCAOL 106	Professional Communication and Research Literacy	1	0	2	2
Semester Total		16	0	14	23

Semester II

Course Code	Course Title	Lecture	Tutorial	Practical	Credit
MCAOL 201	Object-Oriented Application Development with Java	3	0	4	5
MCAOL 202	Web and Full Stack Development	3	0	4	5
MCAOL 203	Computer Networks and Network Security	3	0	2	4
MCAOL 204	Artificial Intelligence and Machine Learning	3	0	2	4
MCAOL 205	Software Engineering and Agile Project Management	3	0	0	3
MCAOL 206	Mini Project-I	0	0	4	2
Semester Total		15	0	16	23

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

IX. Semester-wise Scheme of Study – Semesters III and IV

Semester III

Course Code	Course Title	Lecture	Tutorial	Practical	Credit
MCAOL 301	Data Science and Analytics using Python	3	0	4	5
MCAOL 302	Cloud Computing and DevOps	3	0	4	5
MCAOL 303	Data Warehousing and Big Data Analytics	3	0	2	4
MCAOL 304	Cyber Security and Digital Forensics	3	0	2	4
MCAOL 305	Elective-I	2	0	2	3
MCAOL 306	Research Methodology and Technical Writing	2	0	0	2
Semester Total		16	0	14	23

Semester IV

Course Code	Course Title	Lecture	Tutorial	Practical	Credit
MCAOL 401	Emerging Technologies and Generative AI	3	0	2	4
MCAOL 402	Elective-II	3	0	2	4
MCAOL 403	MOOC / Industry Certification	0	0	4	2
MCAOL 404	Major Project / Dissertation / Industrial Project	0	0	26	13
Semester Total		6	0	34	23

X. Semester-wise Scheme of Study – Semesters V and VI

Not applicable. The MCA programme comprises four semesters over two academic years.

Credit Summary

Semester	Credits
I	23
II	23
III	23
IV	23
Programme Total	92

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

XI. Discipline Specific Elective (DSE) Basket

DSE	Code	Course	Credit
Elective-I	MCAOL 305A	Natural Language Processing	3
Elective-I	MCAOL 305B	Internet of Things and Edge Computing	3
Elective-I	MCAOL 305C	Business Intelligence and Data Visualization	3
Elective-II	MCAOL 402A	Computer Vision	4
Elective-II	MCAOL 402B	Mobile and Cross-Platform Application Development	4
Elective-II	MCAOL 402C	Enterprise Application and CRM Platform Development	4

Learners select one option from each elective basket. Offerings are subject to academic approval and the availability of learning materials, faculty and practical resources.

Other Approved Learning Opportunities

MCAOL 403 provides a two-credit MOOC / Industry Certification component. Prior approval and authentication through completion evidence, a practical portfolio and University viva are proposed.

XII. Faculty and Support Staff

Programme delivery requires qualified faculty, academic counsellors, project supervisors, technical staff and administrative support. Individual appointments and staffing allocations are subject to the approved University plan.

Staff Category	Proposed Requirement
Programme Coordinator	Qualified full-time faculty
Course Coordinators	Subject experts in all revised course areas
Academic Counsellors	ODL learner-facing faculty
Lab/Technical Support	Python/Java/web/database/network/cloud/security/data environments
Project Supervisors	Mini and Major Project/Dissertation/Industrial Project
Instructional Design / Media	SLM, multimedia, accessibility and LMS publishing

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

XIII. Delivery Mechanism

Weeks	Indicative Activity
1	Orientation, diagnostic bridge activities and software/LMS setup.
2-4	Units 1-2, counselling, first assignment/lab.
5-7	Unit 3, applied practice and feedback.
8	Mid-semester evaluation/review.
9-11	Unit 4, practical/project milestone.
12-13	Unit 5/integration task, remedial and enrichment work.
14	Practical/project review and viva preparation.
15	Revision, self-assessment and doubt clearing.
16	Term-end/practical evaluation as scheduled.

- Programme/LMS orientation and diagnostic bridge support.
- Course coordinator and academic counsellor contact channels.
- Technical helpdesk for software, LMS and submission issues.
- Published counselling/practical/project schedules.
- Career, certification, research and higher-study guidance.
- Grievance and accessibility support in accordance with University policy.

Medium of Instruction: English.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

XIV. Procedure for Admissions, Curriculum Transaction and Evaluation

Admission Procedure

Admission, document verification, reservation and refund provisions shall follow the approved University admission policy and applicable regulatory requirements.

Duration of the Programme

The normal duration is 2 academic years divided into 4 semesters. The maximum completion period is governed by the approved University regulations.

Eligibility

Eligibility for graduates, including any required bridge learning is governed by the approved University admission regulations. The detailed qualifying criteria remain subject to institutional confirmation.

Fee Structure

Fees and refund provisions are governed by the approved University fee notification. Programme-specific fee amounts remain subject to institutional confirmation.

Credit System

The programme comprises 92 credits across 4 semesters. The semester schemes specify the course-wise lecture, tutorial and practical/project components. Learning-hour equivalence is subject to the approved University framework.

Evaluation

Where a theory slot exists, each 100-mark theory component may retain the University's existing 60 marks End-Term + 20 marks Mid-Term + 20 marks Activity-Based Continuous Evaluation pattern. Where a practical slot exists, each 100-mark practical component may retain 60 marks End-Term Practical + 20 marks Lab/Portfolio + 20 marks Viva/PBL/Sessional. For the ODL mode, the Activity-Based Continuous Evaluation should explicitly include authenticated LMS quizzes, case/problem assignments, coding/data tasks and oral/portfolio verification as appropriate. Final weightages and pass criteria must follow the approved examination ordinance.

The proposed course totals are 100 marks for a theory-only or practical/project-only course and 200 marks where both theory and practical components are prescribed. Course-wise totals are retained below.

Code	Course	Total Marks
MCAOL 101	Mathematical Foundations for Computing	100
MCAOL 102	Programming and Problem Solving with Python	200
MCAOL 103	Database Systems	200
MCAOL 104	Data Structures and Algorithms	200
MCAOL 105	Computer Systems and Operating Systems	200
MCAOL 106	Professional Communication and Research Literacy	200
MCAOL 201	Object-Oriented Application Development with Java	200
MCAOL 202	Web and Full Stack Development	200
MCAOL 203	Computer Networks and Network Security	200
MCAOL 204	Artificial Intelligence and Machine Learning	200
MCAOL 205	Software Engineering and Agile Project Management	100
MCAOL 206	Mini Project-I	100
MCAOL 301	Data Science and Analytics using Python	200
MCAOL 302	Cloud Computing and DevOps	200
MCAOL 303	Data Warehousing and Big Data Analytics	200
MCAOL 304	Cyber Security and Digital Forensics	200
MCAOL 305	Elective-I	200

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

Code	Course	Total Marks
MCAOL 306	Research Methodology and Technical Writing	100
MCAOL 401	Emerging Technologies and Generative AI	200
MCAOL 402	Elective-II	200
MCAOL 403	MOOC / Industry Certification	100
MCAOL 404	Major Project / Dissertation / Industrial Project	100

Proposed Assessment Components

Course Type	Recommended Evidence
Theory	LMS quiz + analytical/application assignment + mid-term + end-term.
Theory + Practical	Theory evidence + lab portfolio + authenticated practical + viva.
Mini Project	Proposal + design + progress review + artefact + testing + report + viva.
MOOC / Certification	Provider completion + practical portfolio + applied integration + University viva.
Major Project / Dissertation / Industrial Project	Proposal + literature/method review + supervisor reviews + substantial evidence + final report + presentation/demo + viva.

- SLM/e-content released through the LMS with a semester study plan.
- Counselling scheduled around difficult concepts and applied problem solving.
- Practical courses supported by standard lab manuals, setup guides and authenticated evaluation.
- Research/project courses supported through proposal, supervisor allocation, milestones and viva.
- MOOC/industry certification approved in advance and authenticated by the University through portfolio/reflection/viva.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

XV. Requirement of Laboratory Support and Library Resources

- Every practical course should have a lab manual with objectives, prerequisites, exact task, expected evidence, troubleshooting prompts and viva questions.
- Prefer open-source/local tools, browser notebooks, Docker/VMs, simulators and approved sandboxes to reduce location and hardware barriers.
- Cybersecurity/forensics work must be performed only on authorised prepared targets/evidence.
- Mini Project-I serves as the structured intermediate project; the final 13-credit Major Project/Dissertation/Industrial Project provides the main capstone.
- All project work should use supervisor records, version evidence/logbook, testing/analysis and oral authentication.
- SLM units should contain outcomes, prerequisite recall, conceptual explanations, examples, code/algorithms, activities, check-your-progress questions, summary, glossary, self-assessment and further resources.
- LMS shells should include orientation, SLM, short videos, discussions, quizzes, assignment/lab submission, feedback and progress indicators.
- Lab assets should be downloadable and include source code/data/configuration plus expected outputs or validation criteria.
- Library access should include current computing texts, journals, standards, official documentation and research databases as available.
- Accessibility and low-bandwidth alternatives should be built into course design.

XVI. Cost Estimate of the Programme and Provisions

The programme budget should provide for learning materials, academic delivery, technology, learner support, assessment and quality assurance. Amounts and total allocations remain subject to approval under the University's financial provisions.

Cost Head	Proposed Provision
SLM development/revision	Subject to approved institutional budget
Video/multimedia production	Subject to approved institutional budget
LMS/hosting/digital services	Subject to approved institutional budget
Lab/cloud/software/sandbox services	Subject to approved institutional budget
Faculty/counselling/project supervision	Subject to approved institutional budget
Library/e-resources	Subject to approved institutional budget
Examination/practical evaluation	Subject to approved institutional budget
Learner support/helpdesk	Subject to approved institutional budget
CIQA/quality audit	Subject to approved institutional budget

Centre for Distance and Online Education (CDOE) ITM University Gwalior

XVII. Quality Assurance Mechanism

- Pre-offering review of syllabus, COs, SLM, labs, media, LMS and assessments.
- Annual content currency review for AI, cloud, cybersecurity, web and emerging-technology courses.
- Monitoring of counselling, LMS engagement, practical participation, project supervision and feedback.
- Assessment moderation and practical/project sampling for authenticity and consistency.
- Course-exit/programme feedback with documented action taken.
- Periodic stakeholder/industry/alumni input into elective and certification baskets.

CO-PO Attainment and Continuous Improvement

- Map every CO to relevant POs using the University-approved method.
- Tag internal and term-end tasks to COs.
- Use direct theory/practical/project evidence for attainment.
- Use learner feedback only as supplementary indirect evidence.
- Document weak-CO corrective actions in SLM, counselling, labs or assessment.
- Map every assessment task to one or more COs.
- Prefer application, analysis, design and interpretation over reproduction of notes.
- For practical/project work, authenticate the learner through supervised task, viva, live demonstration and/or version evidence.
- Require attribution/disclosure of external code, datasets, AI assistance and sources according to the course brief.
- Verify all external MOOC/video links at the beginning of each semester and use official provider/institution channels only.
- Fast-changing courses should be reviewed before every academic cycle for tools, versions, standards and deprecated practices.
- Core conceptual outcomes should remain platform-neutral; provider-specific tools should appear mainly in lab manuals and supplementary learning.
- Elective and certification options should be offered only when SLM, qualified faculty/mentor support and practical resources are ready.
- CIQA should retain version history and action-taken records for substantive syllabus/SLM changes.

Approval records, course and SLM reviews, faculty deployment, practical arrangements and CIQA action-taken reports should be maintained as programme evidence. Detailed eligibility, fees, staffing, budget, learning-hour allocations and examination rules remain subject to institutional confirmation.

XVIII. NEP 2020 / Multiple Entry-Exit Context

This PPR covers the 2-year, 4-semester MCA award. No Semesters V–VI or separate intermediate exit awards are specified in the proposed MCA curriculum.

Any multiple-entry, exit, re-entry or credit-transfer arrangement shall be governed by the applicable approved University provisions. No additional exit award is assumed under this proposal.

XIX. Annexure – Programme Credit Summary

Semester	Focus Area	Credits
I	Postgraduate computing foundations	23
II	Application development and software engineering	23
III	Advanced data, cloud and security	23
IV	Emerging technologies, electives, certification and major project	23
Total	2-Year MCA	92

END OF PROGRAMME PROJECT REPORT

Centre for Distance and Online Education (CDOE) ITM University Gwalior

MCA – SEMESTER I

Course Code	Course	L-T-P	Credits
MCAOL 101	Mathematical Foundations for Computing	3-0-0	3

Prerequisites: Undergraduate-level mathematics; bridge material should be provided for learners needing revision.

Course Objective:

Strengthen discrete, algebraic and probabilistic foundations used in postgraduate computing. Develop formal reasoning and mathematical modelling ability. Prepare learners for algorithms, data science, AI and advanced computing courses.

Course Outcomes:

CO1. Apply logic, sets, relations and functions to computational problems.

CO2. Use matrix and graph concepts in computing contexts.

CO3. Solve combinatorial and recurrence problems.

CO4. Apply basic probability and statistical reasoning to computing examples.

Course Content:

Module	Content
Module 1: Logic, Sets and Functions	Sets and operations; Cartesian products; relations, equivalence and partial order; functions and composition; propositions, predicates, truth tables, normal forms, quantifiers, inference, proof techniques and mathematical induction. Topics include: logical equivalence, tautology/contradiction, relation properties, bijection, direct/contradiction proof and induction in algorithmic reasoning.
Module 2: Linear Algebra for Computing	Matrices, determinants, inverse, rank, elementary transformations, systems of linear equations, vectors, vector spaces, basis/dimension awareness and linear transformations. Topics include: Gaussian elimination, matrix representation of transformations, introductory eigenvalue/eigenvector intuition and applications in graphics/data.
Module 3: Combinatorics and Recurrence	Counting principles, permutations, combinations, pigeonhole principle, inclusion-exclusion awareness, recurrence relations and elementary solution methods. Topics include: recursive algorithm models, growth patterns, substitution/iteration intuition and counting-based complexity examples.
Module 4: Graphs and Trees	Graph terminology, degree, paths/cycles, connectivity, directed/weighted graphs, trees, spanning trees and elementary traversal concepts. Topics include: adjacency matrix/list, BFS/DFS tracing, shortest path and minimum spanning-tree intuition, graph modelling of networks and dependencies.
Module 5: Probability for Computing	Sample spaces, events, conditional probability, independence, Bayes theorem, random variables, expectation and variance; binomial/Poisson/normal awareness. Topics include: probabilistic reasoning, simple reliability and classification examples, interpretation of uncertainty and common probability errors.

Assessment activities:

Unit-wise LMS quizzes with worked feedback. • Analytical problem set covering logic, matrices, recurrence and graphs. • Case task: model a computing/network problem using graph or matrix representation. • End-course integrative assignment with full reasoning and error reflection.

Practical and project work:

Guided problem sheets; optional Python/Spreadsheet verification of matrix, graph and probability calculations; learner submits method, intermediate steps and interpretation.

Recommended Books

Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill.

David C. Lay et al., Linear Algebra and Its Applications, Pearson.

Digital learning resources: NPTEL/SWAYAM: Discrete Mathematics; Mathematics for Computer Science; Probability for Computer Science. NPTEL-NOC IITM official lecture series on Discrete Mathematics / Probability; verify current playlist each semester.

DIRECTOR
Centre for Distance and Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER I

Course Code	Course	L-T-P	Credits
MCAOL 102	Programming and Problem Solving with Python	3-0-4	5

Prerequisites: Basic programming logic; a short bridge module should cover variables, flowcharts and algorithm tracing.

Course Objective:

Develop strong algorithmic problem solving using Python. Build modular, object-oriented and data-oriented Python competence. Introduce testing, debugging, files, APIs and software-quality practices.

Course Outcomes:

CO1. Design and implement Python solutions to non-trivial problems.

CO2. Use core collections, functions, files, exceptions and modules effectively.

CO3. Develop object-oriented and reusable Python components.

CO4. Test, debug and document a small application or data-processing solution.

Course Content:

Module	Content
Module 1: Python Foundations and Problem Solving	Python runtime, variables, types, operators, expressions, I/O, conditionals, loops and tracing. Topics include: dynamic typing, type conversion, formatted strings, loop patterns, input validation, debugging, algorithm decomposition and complexity awareness.
Module 2: Collections and Functional Decomposition	Strings, lists, tuples, sets, dictionaries, slicing, comprehensions, functions, parameters, return values, scope, recursion and lambda awareness. Topics include: mutability, nested structures, higher-order operations, iterator/generator concepts and reusable utility functions.
Module 3: Files, Exceptions and Modules	Text/CSV/JSON files, pathlib, exceptions, custom exceptions, modules/packages, standard library, virtual environments and package management. Topics include: robust file handling, logging, configuration files and dependency documentation.
Module 4: Object-Oriented and Applied Python	Classes, objects, constructors, encapsulation, inheritance, polymorphism and special methods; regular expressions, database/API awareness. Topics include: dataclasses awareness, composition, HTTP/JSON interaction, sqlite/database connection and reusable application structure.
Module 5: Testing, Quality and Integrated Application	Unit-testing concepts, assertions, debugging, logging, coding style, type hints awareness, documentation and integrated application. Topics include: test cases, edge conditions, version control awareness, security hygiene and final demonstration.

Assessment activities:

Weekly coding quizzes and short tracing/debugging tasks. • Applied assignment: file/data-processing program with validation and exceptions. • Practical portfolio containing selected source files, tests, observations and corrections. • Supervised coding skill-check and mini-application viva.

Practical and project work:

1. Environment setup and basic programs; 2. conditionals/loops; 3. collections; 4. functions/recursion; 5. files/CSV/JSON; 6. exceptions/logging; 7. classes/inheritance; 8. simple API/database task; 9. testing/debugging; 10. integrated mini-application.

Recommended Books

Allen B. Downey, Think Python, O'Reilly.

Mark Lutz, Learning Python, O'Reilly.

Digital learning resources: NPTEL/SWAYAM: The Joy of Computing Using Python; Programming, Data Structures and Algorithms Using Python. NPTEL-NOC IITM official Python programming videos; Python Software Foundation / PyCon educational material may supplement.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER I

Course Code	Course	L-T-P	Credits
MCAOL 103	Database Systems	3-0-2	4

Prerequisites: Programming fundamentals and basic data concepts.

Course Objective:

Develop sound database modelling, SQL and transaction competence. Introduce physical design, security and contemporary data-store choices. Enable learners to implement and evaluate database-backed solutions.

Course Outcomes:

CO1. Design normalized relational schemas from requirements.

CO2. Write and optimize SQL queries for realistic applications.

CO3. Explain transaction, concurrency, recovery and indexing concepts.

CO4. Build and secure a small database-backed solution.

Course Content:

Module	Content
Module 1: Database Architecture and Data Modelling	DBMS concepts, three-schema architecture, data independence, ER/EER modelling, keys, constraints and ER-to-relational mapping. Topics include: weak entities, specialization/generalization, cardinality/participation and modelling trade-offs.
Module 2: Relational Model and SQL	Relational algebra; SQL DDL/DML; joins, subqueries, set operations, grouping, views and constraints. Topics include: CTE awareness, window-function awareness, prepared statements, stored procedures/triggers overview and query readability.
Module 3: Database Design and Normalization	Functional dependencies, closure, candidate keys, normal forms up to BCNF and decomposition. Topics include: lossless join, dependency preservation, anomalies and design justification.
Module 4: Transactions, Concurrency and Recovery	ACID, schedules, serializability, locks/two-phase locking, deadlocks, logging, checkpoints and recovery. Topics include: isolation levels, optimistic/pessimistic concurrency awareness and failure scenarios.
Module 5: Physical Design, Security and Modern Data Stores	B+ tree/hash indexing, query plans, optimization awareness, authorization, roles, backup/recovery, NoSQL categories and CAP awareness. Topics include: relational vs document/key-value selection and secure database configuration.

Assessment activities:

ER-to-relational case assignment. • SQL challenge set with increasingly complex queries. • Normalization and transaction-analysis worksheet. • Mini database design review with security and performance justification.

Practical and project work:

Schema/constraint creation; CRUD; joins/subqueries; views; normalization case; indexes/query plan; transactions/rollback; roles/privileges; backup/restore demonstration; mini database application.

Recommended Books

Elmasri & Navathe, Fundamentals of Database Systems, Pearson.

Silberschatz, Korth & Sudarshan, Database System Concepts, McGraw-Hill.

Digital learning resources: NPTEL/SWAYAM: Introduction to Database Systems / Database Management System. NPTEL-NOC IITM official database systems lectures; PostgreSQL/MySQL official learning resources.

Centre for Online Education (CDOE) ITM University Gwalior

MCA – SEMESTER I

Course Code	Course	L-T-P	Credits
MCAOL 104	Data Structures and Algorithms	3-0-4	5

Prerequisites: Programming proficiency in at least one language.

Course Objective:

Develop rigorous data-structure and algorithmic thinking. Build implementation and complexity-analysis competence. Introduce major algorithm-design paradigms.

Course Outcomes:

CO1. Analyse algorithm time and space complexity.

CO2. Implement linear, tree, hashing and graph structures.

CO3. Apply searching, sorting and graph algorithms.

CO4. Select and justify algorithms for realistic constraints.

Course Content:

Module	Content
Module 1: Algorithm Analysis and Linear Structures	ADTs, asymptotic notation, arrays, linked lists, stacks, queues and recursion. Topics include: Big-O/Theta/Omega, amortized intuition, singly/doubly/circular lists and application patterns.
Module 2: Trees and Priority Structures	Binary trees, BST, traversals, heaps and priority queues. Topics include: AVL/balancing awareness, expression trees, heapify, priority scheduling and complexity.
Module 3: Hashing and Dictionaries	Hash functions, load factor, chaining, open addressing and dictionary/set applications. Topics include: collision strategies, resizing awareness and trade-offs between hashing and trees.
Module 4: Graph Algorithms	Graph representations, BFS, DFS, topological sorting, shortest path and MST concepts. Topics include: Dijkstra/Prim/Kruskal intuition, connected components and graph use cases.
Module 5: Searching, Sorting and Design Paradigms	Binary search; insertion/selection/bubble, merge, quick and heap sort; divide-and-conquer, greedy and dynamic-programming introduction. Topics include: stability, in-place algorithms, empirical benchmarking and problem-pattern recognition.

Assessment activities:

Complexity worksheet and algorithm-tracing quiz. • Implementation portfolio for core structures. • Benchmark two algorithms on multiple input sizes and interpret results. • Case assignment selecting an appropriate data structure/algorithm.

Practical and project work:

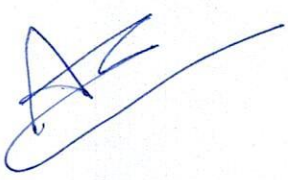
Linked lists; stack/queue; BST/traversals; heap; hash table; BFS/DFS; shortest path/MST; search/sort benchmarking; one mini algorithmic problem using greedy or dynamic programming.

Recommended Books

Mark Allen Weiss, Data Structures and Algorithm Analysis, Pearson.

Narasimha Karumanchi, Data Structures and Algorithms Made Easy.

Digital learning resources: NPTEL/SWAYAM: Data Structures and Algorithms; Programming, Data Structures and Algorithms. NPTEL-NOC IITM official DSA lectures; algorithm visualization may be used only as supplementary OER.



MCA – SEMESTER I

Course Code	Course	L-T-P	Credits
MCAOL 105	Computer Systems and Operating Systems	3-0-2	4

Prerequisites: Basic digital-computing concepts.

Course Objective:

Integrate computer organization and operating-system foundations at postgraduate-entry level. Explain how processor, memory, processes, files and I/O interact. Develop practical system-observation and troubleshooting skills.

Course Outcomes:

CO1. Explain processor organization, instruction execution and memory hierarchy.

CO2. Analyse process scheduling, synchronization and deadlocks.

CO3. Explain virtual memory, files, storage and I/O.

CO4. Use system tools to observe and interpret OS behaviour.

Course Content:

Module	Content
Module 1: Computer Organization Essentials	Data representation, CPU organization, registers, ALU, instruction cycle, addressing modes and RISC/CISC. Topics include: pipelining awareness, cache/locality, performance metrics and multicore overview.
Module 2: OS Structure, Processes and Scheduling	Kernel/user mode, system calls, processes, threads, PCB, context switching and scheduling. Topics include: FCFS, SJF, Priority, RR and multicore scheduling awareness.
Module 3: Concurrency and Deadlocks	Race conditions, critical section, mutex/semaphore/monitor concepts and deadlocks. Topics include: classical synchronization problems, resource-allocation graph, prevention/avoidance/detection.
Module 4: Memory and Virtualization	Contiguous allocation, paging, segmentation, TLB, virtual memory, page replacement and thrashing awareness. Topics include: working-set intuition, memory-mapped files and virtualization/container relationship.
Module 5: File Systems, I/O and Protection	Files/directories, allocation, journaling awareness, storage/disk scheduling, I/O buffering, protection/access control and system security. Topics include: interrupts, DMA, permissions, logging and Linux/Windows observation.

Assessment activities:

Scheduling and memory-management problem set. • System-observation assignment using Linux/Windows utilities. • Case analysis of concurrency/deadlock or resource-management failure. • Integrative quiz linking hardware organization with OS behaviour.

Practical and project work:

CPU/instruction simulation or trace; Linux/Windows process inspection; scheduling simulation; synchronization exercise; page replacement; file permissions; storage/filesystem observations; system-monitoring report.

Recommended Books

Abraham Silberschatz et al., Operating System Concepts, Wiley.

William Stallings, Computer Organization and Architecture, Pearson.

Digital learning resources: NPTEL/SWAYAM: Operating Systems; Computer Organization and Architecture. NPTEL-NOC IITM official Operating Systems and Computer Architecture lecture series.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER I

Course Code	Course	L-T-P	Credits
MCAOL 106	Professional Communication and Research Literacy	1-0-2	2

Prerequisites: Graduate-level English; additional language support may be provided through the LMS.

Course Objective:

Develop professional and technical communication for postgraduate computing work. Build information-search, source-evaluation and research-literacy skills. Prepare learners for project documentation, presentations and workplace communication.

Course Outcomes:

CO1. Write concise professional and technical documents.

CO2. Search, evaluate and cite credible technical/academic sources.

CO3. Deliver structured technical presentations and participate in professional interaction.

CO4. Prepare a project/research brief using ethical citation and responsible AI-use disclosure.

Course Content:

Module	Content
Module 1: Professional Communication	Audience, purpose, tone, professional e-mail, meeting agenda/minutes and status communication. Topics include: clarity, concise writing, action requests and remote-work etiquette.
Module 2: Technical Writing	Technical descriptions, user instructions, reports, executive summaries and documentation structure. Topics include: headings, visuals, tables, revision and reader-centred organization.
Module 3: Information and Research Literacy	Search strategies, scholarly databases awareness, official technical documentation, source credibility and literature-note taking. Topics include: keywords, citation chaining, primary vs secondary sources and misinformation awareness.
Module 4: Citation, Integrity and Responsible AI	Citation styles, paraphrasing, quotation, plagiarism, code/data attribution and generative-AI disclosure. Topics include: authorship, source verification, fabricated citations, privacy and acceptable-use statements.
Module 5: Presentation and Employability Communication	Technical presentation, visual hierarchy, viva, group discussion, interview and professional profile. Topics include: storytelling, Q&A, STAR response and reflective communication portfolio.

Assessment activities:

Professional email + meeting-document task. • Annotated source search: identify and justify five credible technical/academic sources. • Two-page technical brief with citations and AI-use declaration. • Recorded 5-minute technical presentation plus peer/self-review.

Practical and project work:

Listening/note-taking; technical rewrite; report formatting; source-evaluation worksheet; citation exercise; presentation recording; mock viva/interview.

Recommended Books

Meenakshi Raman & Sangeeta Sharma, Technical Communication, Oxford.

Gerald J. Alred et al., Handbook of Technical Writing.

Digital learning resources: NPTEL/SWAYAM: Technical Communication; Research Methodology; Developing Soft Skills and Personality. NPTEL official Technical Communication / Research Methodology videos; IEEE/ACM author education resources where available.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER II

Course Code	Course	L-T-P	Credits
MCAOL 201	Object-Oriented Application Development with Java	3-0-4	5

Prerequisites: Programming and basic data structures.

Course Objective:

Develop robust object-oriented application-development skills in Java. Use collections, exceptions, concurrency, databases and testing appropriately. Build maintainable modular applications.

Course Outcomes:

CO1. Design object models using OOP principles.

CO2. Use Java collections, exceptions, generics and file I/O.

CO3. Implement database-connected and concurrent application components.

CO4. Test and document a small Java application.

Course Content:

Module	Content
Module 1: Java Platform and Object Modelling	JDK/JVM/JRE, bytecode, classes, objects, constructors, encapsulation and packages. Topics include: this/static/final, access modifiers and object lifecycle.
Module 2: Inheritance, Interfaces and Robustness	Inheritance, overloading/overriding, polymorphism, abstract classes, interfaces and exceptions. Topics include: custom exceptions, assertions and design-for-change.
Module 3: Collections, Generics and Functional Features	List/Set/Map, iterators, comparators, generics, lambda expressions and streams awareness. Topics include: collection selection, immutability awareness and data-processing pipelines.
Module 4: I/O, Concurrency and Persistence	File/NIO, serialization awareness, threads, synchronization, JDBC and transactions. Topics include: prepared statements, connection management and safe concurrent access.
Module 5: Application Architecture and Testing	Layered structure, design principles, testing, logging and packaging. Topics include: JUnit awareness, dependency management and integrated application demonstration.

Assessment activities:

Class-model design assignment. • Collections/JDBC practical portfolio. • Supervised coding task on OOP + exceptions. • Mini application review with viva.

Practical and project work:

Classes/constructors; inheritance/interfaces; exceptions; collections/generics; streams/lambda awareness; files; threads; JDBC CRUD/transaction; testing; final layered app.

Recommended Books

Herbert Schildt, Java: The Complete Reference, McGraw-Hill.

Cay S. Horstmann, Core Java, Pearson.

Digital learning resources: NPTEL/SWAYAM: Programming in Java; Object-Oriented System Development. NPTEL-NOC IITM official Java programming lectures; Oracle Java official tutorials may supplement.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER II

Course Code	Course	L-T-P	Credits
MCAOL 202	Web and Full Stack Development	3-0-4	5

Prerequisites: Programming and basic web concepts.

Course Objective:

Develop end-to-end modern web-application competence. Integrate accessible front-end, APIs, authentication and persistence. Emphasize testing, security and deployment practices.

Course Outcomes:

CO1. Build responsive and accessible client interfaces.

CO2. Develop RESTful server-side APIs.

CO3. Integrate persistent storage and authentication/authorization.

CO4. Test, secure and deploy/package a full-stack application.

Course Content:

Module	Content
Module 1: Modern Front-End Foundations	Semantic HTML, responsive CSS, JavaScript ES concepts, DOM and accessibility. Topics include: Flexbox/Grid, forms, validation, modules, promises and fetch.
Module 2: Component-Based Client Development	Components, state, routing, reusable UI and client data flow using an approved framework. Topics include: lifecycle/hooks concepts, form state, error/loading states and accessibility.
Module 3: Server-Side and API Development	Server runtime/framework, routing, middleware, REST design, validation and error handling. Topics include: HTTP methods/status codes, API versioning awareness and logging.
Module 4: Persistence and Application Security	Relational/NoSQL integration, CRUD, ORM awareness, authentication/authorization and secure data handling. Topics include: sessions/tokens, password hashing concept, CORS/XSS/CSRF awareness and secrets.
Module 5: Testing, Deployment and Operations	Unit/API/UI testing awareness, environment configuration, containers, deployment and CI/CD concepts. Topics include: observability, performance, dependency security and final integrated application.

Assessment activities:

Responsive UI assignment. • REST API + database practical portfolio. • Security/design review of a small web application. • Full-stack integration task with demo and viva.

Practical and project work:

Semantic/responsive page; JS async/fetch; component UI; REST API; DB CRUD; auth demonstration; testing; Docker/environment config; final deployable/packageable app.

Recommended Books

David Flanagan, JavaScript: The Definitive Guide, O'Reilly.

Ethan Brown, Web Development with Node and Express, O'Reilly, or approved equivalent stack text.

Digital learning resources: NPTEL/SWAYAM: Web Technology / Modern Application Development. NPTEL official Web Technology material; MDN Web Docs and official framework/provider videos.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER II

Course Code	Course	L-T-P	Credits
MCAOL 203	Computer Networks and Network Security	3-0-2	4

Prerequisites: Basic computer systems and operating systems.

Course Objective:

Develop solid network-protocol and troubleshooting competence. Integrate network security into core networking rather than treating it as an afterthought. Prepare learners for cloud and cybersecurity study.

Course Outcomes:

CO1. Explain layered network operation and major protocols.

CO2. Perform IP addressing/subnetting and basic routing analysis.

CO3. Use network tools to diagnose communication.

CO4. Apply baseline network-security controls.

Course Content:

Module	Content
Module 1: Network Foundations and Data Link	OSI/TCP-IP, media, Ethernet, switching, MAC, framing and error detection. Topics include: VLAN awareness, MTU and packet encapsulation.
Module 2: Internet Layer and Routing	IPv4/IPv6, subnetting/VLSM, ARP, ICMP, NAT and routing concepts. Topics include: routing tables, distance-vector/link-state awareness and route selection.
Module 3: Transport and Applications	TCP/UDP, sockets/ports, flow/congestion control, DNS, HTTP/HTTPS, e-mail and DHCP. Topics include: TCP handshake, reliability and service troubleshooting.
Module 4: Wireless and Network Operations	Wi-Fi architecture, wireless security basics, network devices, monitoring and troubleshooting. Topics include: ping, traceroute, nslookup/dig, packet capture and latency/loss analysis.
Module 5: Network Security	Segmentation, ACL/firewall concepts, VPN/TLS, IDS/IPS awareness, secure protocols and network hardening. Topics include: least privilege, zero-trust awareness and incident-oriented packet/log analysis.

Assessment activities:

Subnetting/addressing assignment. • Packet-analysis lab portfolio. • Network-design and segmentation case. • Authenticated troubleshooting skill check.

Practical and project work:

IP/VLSM; simulator topology; switch/routing basics; ping/traceroute/DNS; Wireshark DNS/TCP/HTTP; ACL/firewall simulation; wireless-security case; troubleshooting practical.

Recommended Books

James Kurose & Keith Ross, Computer Networking: A Top-Down Approach, Pearson.

Behrouz Forouzan, Data Communications and Networking, McGraw-Hill.

Digital learning resources: NPTEL/SWAYAM: Computer Networks and Internet Protocol; Introduction to Cyber Security. NPTEL-NOC IITM official Computer Networks lectures; Wireshark official educational resources may supplement.

DIRECTOR
Centre for Distance and Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER II

Course Code	Course	L-T-P	Credits
MCAOL 204	Artificial Intelligence and Machine Learning	3-0-2	4

Prerequisites: Python programming, mathematical foundations and basic statistics.

Course Objective:

Connect classical AI problem solving with modern machine-learning practice. Develop principled model-building and evaluation skills. Embed responsible-AI thinking throughout.

Course Outcomes:

CO1. Apply search and knowledge-representation concepts.

CO2. Prepare data and train baseline ML models.

CO3. Evaluate classification/regression/clustering results appropriately.

CO4. Explain responsible-AI limitations, bias and interpretability.

Course Content:

Module	Content
Module 1: AI Problem Solving	Agents, state spaces, BFS/DFS, heuristic search, A* and game-playing overview. Topics include: problem formulation, completeness/optimality and heuristic quality.
Module 2: Knowledge and Uncertainty	Rules, logical representation, inference and uncertainty overview. Topics include: knowledge bases, probabilistic reasoning awareness and expert-system concepts.
Module 3: Supervised Machine Learning	ML workflow, preprocessing, regression, logistic regression, k-NN, trees and evaluation. Topics include: train/validation/test, leakage, confusion matrix, precision/recall/F1 and regression errors.
Module 4: Unsupervised and Model Improvement	Clustering, PCA awareness, ensembles, cross-validation and hyperparameter tuning. Topics include: k-means, hierarchical clustering, feature engineering and imbalance awareness.
Module 5: Responsible and Contemporary AI	Explainability, fairness, privacy, human oversight, generative-AI overview and model documentation. Topics include: model cards, hallucination awareness and deployment limitations.

Assessment activities:

Search/heuristic problem worksheet. • ML notebook comparing two supervised models. • Clustering/EDA task with interpretation. • Responsible-AI model card and viva.

Practical and project work:

Search algorithm trace; preprocessing; regression; classification; evaluation; decision tree/ensemble; clustering; cross-validation; model-card mini project.

Recommended Books

Stuart Russell & Peter Norvig, Artificial Intelligence: A Modern Approach, Pearson.

Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, O'Reilly.

Digital learning resources: NPTEL/SWAYAM: Artificial Intelligence; Introduction to Machine Learning. NPTEL-NOC IITM official AI and Machine Learning lecture series; Google for Developers ML educational resources may supplement.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER II

Course Code	Course	L-T-P	Credits
MCAOL 205	Software Engineering and Agile Project Management	3-0-0	3

Prerequisites: Programming and basic application-development experience.

Course Objective:

Develop disciplined software engineering and project-management thinking. Integrate requirements, design, quality, risk and agile practices. Prepare learners for project and industry work.

Course Outcomes:

CO1. Elicit and document software requirements.

CO2. Apply software design, testing and configuration-management concepts.

CO3. Plan scope, schedule and risk.

CO4. Use agile/Scrum practices and interpret project metrics.

Course Content:

Module	Content
Module 1: Software Process and Requirements	Software lifecycle models, feasibility, stakeholders, requirements elicitation, SRS, user stories and acceptance criteria. Topics include: functional/non-functional requirements, traceability and change control.
Module 2: Design and Architecture	Modularity, cohesion/coupling, architectural styles, UML awareness and UI/data design. Topics include: layered architecture, design principles and technical-debt awareness.
Module 3: Software Quality and Testing	Verification/validation, test levels, test design, defect management and quality models. Topics include: boundary/equivalence testing, regression, code review and quality metrics.
Module 4: Project Planning and Control	WBS, estimation, scheduling, risks, configuration management and status reporting. Topics include: Gantt, critical-path awareness, risk register and baseline/change control.
Module 5: Agile and DevOps-Oriented Delivery	Agile values, Scrum, Kanban, backlog, sprint planning, retrospectives and DevOps awareness. Topics include: CI/CD linkage, team collaboration, distributed-work practices and closure/lessons learned.

Assessment activities:

SRS/user-story assignment. • Test-plan and defect-analysis task. • Project plan with WBS, schedule and risk register. • Simulated sprint review/retrospective.

Practical and project work:

Case-based requirements workshop; UML/design exercise; test-case design; estimation/schedule exercise; agile board/backlog; sprint simulation.

Recommended Books

Roger Pressman & Bruce Maxim, Software Engineering: A Practitioner's Approach.

Ian Sommerville, Software Engineering, Pearson.

Digital learning resources: NPTEL/SWAYAM: Software Engineering; Software Project Management. NPTEL official Software Engineering / Project Management lectures.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER II

Course Code	Course	L-T-P	Credits
MCAOL 206	Mini Project-I	0-0-4	2

Prerequisites: Successful engagement with Semester-I/II technical courses.

Course Objective:

Integrate programming, database, web/Java or AI concepts in an applied project. Develop disciplined project evidence and learner authentication in ODL. Prepare learners for larger postgraduate project work.

Course Outcomes:

CO1. Define a feasible problem and measurable deliverables.

CO2. Prepare requirements, design and project plan.

CO3. Implement and test a working prototype.

CO4. Document and defend the work in a viva.

Course Content:

Module	Content
Stage 1: Proposal and Scope	Problem statement, users/stakeholders, objectives, scope, feasibility and expected deliverables.
Stage 2: Requirements and Design	SRS/user stories, architecture, data model, UI/API design, test strategy and ethics/security considerations.
Stage 3: Implementation	Incremental build, source control/version evidence, coding standards, mentor reviews and issue log.
Stage 4: Validation	Functional tests, defect log, usability/performance/security checks where relevant and corrective iteration.
Stage 5: Final Demonstration	Report, README/user guide, source/data artefacts, presentation, live demonstration, individual contribution statement and viva.

Assessment activities:

Proposal review 15%; requirements/design review 20%; midpoint prototype 20%; testing/quality review 15%; final artefact/report 20%; presentation/viva 10% (indicative; University policy prevails).

Practical and project work:

Supervised project milestones; repository or dated artefact history; test evidence; final demonstration and viva.

Recommended Books

Roger Pressman & Bruce Maxim, Software Engineering.

CDOE MCA Mini Project Handbook (to be approved).

Digital learning resources: Domain-specific SWAYAM/NPTEL MOOC may supplement the project but does not replace supervision. CDOE project orientation and official technology-provider videos matched to project stack.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 301	Data Science and Analytics using Python	3-0-4	5

Prerequisites: Python programming, probability/statistics and database basics.

Course Objective:

Develop end-to-end data-science and analytical workflow competence. Integrate data cleaning, EDA, statistics and predictive modelling. Emphasize reproducibility, interpretation and communication.

Course Outcomes:

CO1. Acquire, clean and transform structured data.

CO2. Perform exploratory and statistical analysis.

CO3. Build and evaluate baseline analytical models.

CO4. Communicate reproducible findings with limitations.

Course Content:

Module	Content
Module 1: Data Science Workflow and NumPy	Problem framing, data lifecycle, Jupyter, NumPy arrays and vectorization. Topics include: reproducibility, data provenance, broadcasting and computational efficiency.
Module 2: Data Wrangling with pandas	Import, indexing, missing data, duplicates, types, dates/categories, groupby, merge and reshape. Topics include: validation, tidy-data concepts and feature creation.
Module 3: EDA and Statistical Analysis	Descriptive statistics, distributions, outliers, correlation, sampling/inference awareness. Topics include: exploratory hypotheses, misleading associations and uncertainty.
Module 4: Visualization and Baseline Modelling	Matplotlib-based visualization, regression/classification foundations and evaluation. Topics include: chart selection/accessibility, train-test split, metrics and error analysis.
Module 5: Reproducible Analytics Project	Pipeline organization, notebooks, reporting, data ethics/privacy and analytical storytelling. Topics include: random seeds/environment capture, limitations and executive findings.

Assessment activities:

Data-cleaning assignment. • EDA notebook with visualizations and interpretation. • Supervised analytics skill check. • End-to-end analytical report + presentation.

Practical and project work:

NumPy; pandas import/clean; groupby/merge/pivot; statistics/outliers; visualization; regression/classification baseline; metrics; reproducible final notebook.

Recommended Books

Wes McKinney, Python for Data Analysis, O'Reilly.

Jake VanderPlas, Python Data Science Handbook, O'Reilly.

Digital learning resources: NPTEL/SWAYAM: Python for Data Science; Data Analytics with Python. NPTEL-NOC IITM official Python/Data Science lectures; pandas and Matplotlib official documentation/videos.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 302	Cloud Computing and DevOps	3-0-4	5

Prerequisites: Networks, operating systems and application-development basics.

Course Objective:

Build practical cloud and deployment competence. Integrate virtualization, containers, cloud services, security and DevOps. Develop cost, reliability and operational awareness.

Course Outcomes:

CO1. Explain cloud architectures and service models.

CO2. Use or simulate core compute, storage, network and IAM services.

CO3. Containerize and deploy a simple application.

CO4. Design a secure, observable and cost-aware cloud solution.

Course Content:

Module	Content
Module 1: Cloud Foundations and Virtualization	Cloud characteristics, IaaS/PaaS/SaaS, deployment models, regions/zones, virtualization and containers. Topics include: shared responsibility, images and elasticity.
Module 2: Core Cloud Infrastructure	Compute, object/block/file storage, virtual networks, subnets, load balancing, DNS and managed databases. Topics include: autoscaling, availability and service selection.
Module 3: Containers and Cloud-Native Services	Docker concepts, registries, orchestration overview, serverless, API gateways, queues and events. Topics include: microservices awareness and stateless/stateful trade-offs.
Module 4: DevOps and Infrastructure Automation	Version control, CI/CD, infrastructure-as-code awareness, configuration, artifact pipelines and environment promotion. Topics include: automated testing, secrets and release strategies.
Module 5: Cloud Security, Operations and Economics	IAM, least privilege, encryption, logging/metrics/alerts, backup/DR, cost and sustainability. Topics include: SRE awareness, RTO/RPO, budgets/tagging and FinOps.

Assessment activities:

Cloud architecture case. • Containerization + deployment practical portfolio. • IAM/security review task. • CI/CD or deployment pipeline mini integration with viva.

Practical and project work:

VM/sandbox compute; storage; VPC/subnet/security rules; IAM; Docker image/container; simple CI workflow; serverless demonstration; monitoring/logging; backup/DR architecture; cost estimation.

Recommended Books

Rajkumar Buyya et al., Mastering Cloud Computing, McGraw-Hill.

Thomas Erl et al., Cloud Computing: Concepts, Technology & Architecture.

Digital learning resources: NPTEL/SWAYAM: Cloud Computing; DevOps offerings where available. NPTEL official Cloud Computing lectures; AWS Skill Builder / Microsoft Learn / Google Cloud official learning videos as approved.

Director
Centre for Distance and Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 303	Data Warehousing and Big Data Analytics	3-0-2	4

Prerequisites: Database Systems and Data Science fundamentals.

Course Objective:

Develop modern analytical data-platform understanding. Integrate dimensional modelling, ETL/ELT, warehouse and big-data concepts. Enable applied analytics over structured/semi-structured data.

Course Outcomes:

CO1. Design dimensional warehouse models.

CO2. Build/describe ETL-ELT and data-quality workflows.

CO3. Explain distributed big-data storage/processing concepts.

CO4. Perform analytical queries and interpret results responsibly.

Course Content:

Module	Content
Module 1: Data Warehousing Architecture	Operational vs analytical systems, warehouse/lake/lakehouse awareness, staging, ETL/ELT, metadata, lineage and governance. Topics include: batch vs streaming awareness and data-quality controls.
Module 2: Dimensional Modelling and OLAP	Grain, facts, dimensions, star/snowflake, surrogate keys and slowly changing dimensions. Topics include: conformed dimensions, fact types, cubes, hierarchies and OLAP operations.
Module 3: Big Data Foundations	Volume/velocity/variety, distributed files, partitioning, replication, MapReduce/Spark concepts. Topics include: data locality, resilient distributed processing and cluster awareness.
Module 4: Big Data Analytics Workflow	Data ingestion, transformation, SQL-on-big-data, aggregation, basic mining/analytics and visualization. Topics include: batch pipelines, semi-structured data and performance considerations.
Module 5: Governance and Applied Analytics	Security, privacy, cataloging, data lifecycle, bias and responsible analytics. Topics include: warehouse-to-dashboard case, KPI design and architectural trade-offs.

Assessment activities:

Dimensional-model assignment. • ETL/data-quality task. • Big-data architecture comparison case. • Analytical mini practical using warehouse/Spark-like environment.

Practical and project work:

Star schema; ETL with SQL/Python; OLAP queries; semi-structured data; introductory Spark/pyspark or approved sandbox; aggregation; final warehouse/analytics case.

Recommended Books

Ralph Kimball & Margy Ross, The Data Warehouse Toolkit.

Jiawei Han et al., Data Mining: Concepts and Techniques; selected big-data supplementary references.

Digital learning resources: NPTEL/SWAYAM: Data Warehousing and Data Mining; Big Data Computing. NPTEL official Data Mining / Big Data lectures; Apache Spark official learning resources where suitable.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 304	Cyber Security and Digital Forensics	3-0-2	4

Prerequisites: Networks, operating systems and web/database fundamentals.

Course Objective:

Provide defensive cybersecurity competence with forensic readiness. Integrate risk, identity, system/network/application security and evidence handling. Ensure practical work remains lawful, controlled and remediation-oriented.

Course Outcomes:

CO1. Assess common security risks and controls.

CO2. Apply baseline security hardening in an authorised lab.

CO3. Interpret logs and forensic artefacts.

CO4. Prepare incident/forensic findings and remediation recommendations.

Course Content:

Module	Content
Module 1: Security Foundations and Risk	CIA, assets, threats, vulnerabilities, risk, policies, ethics and authorization. Topics include: threat modelling, defence-in-depth and security baselines.
Module 2: Identity, Cryptography and System Security	Authentication/MFA, access control, hashing/encryption/signatures, OS hardening, patching and endpoint controls. Topics include: PKI, secrets and least privilege.
Module 3: Network and Application Security	Segmentation, firewall/IDS/IPS/VPN/TLS, secure protocols and OWASP-style application risks. Topics include: secure input/session handling, vulnerability-management lifecycle and remediation.
Module 4: Digital Forensics and Evidence	Forensic lifecycle, chain of custody, imaging/hashing concepts, file-system/log/browser/network artefacts and timeline analysis. Topics include: integrity verification, metadata and attribution limitations.
Module 5: Detection and Incident Response	Logging/SIEM awareness, triage, containment, eradication, recovery, reporting and lessons learned. Topics include: incident playbooks, evidence preservation and post-incident improvement.

Assessment activities:

Threat-model/risk-register assignment. • Authorised hardening/vulnerability practical portfolio. • Sanitised log/forensic artefact analysis. • Incident/forensic report + viva.

Practical and project work:

VM baseline/hardening; access controls; packet/log analysis; safe vulnerability scan of isolated target; secure-coding check; hash/evidence handling; browser/log artefact analysis; timeline; response playbook.

Recommended Books

William Stallings & Lawrie Brown, Computer Security: Principles and Practice.

Bill Nelson et al., Guide to Computer Forensics and Investigations.

Digital learning resources: NPTEL/SWAYAM: Introduction to Cyber Security; Digital Forensics where available. NPTEL official Cyber Security content; OWASP official educational resources; all lab activity limited to authorised environments.

DIRECTOR
Centre for Distance and Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 305A	Natural Language Processing	2-0-2	3

Prerequisites: Python, AI/ML and basic probability.

Course Objective:

Develop classical and modern NLP foundations. Build text-processing and evaluation skills. Introduce transformers/LLMs responsibly.

Course Outcomes:

CO1. Prepare and represent text computationally.

CO2. Build baseline NLP models.

CO3. Explain sequence, attention and transformer concepts.

CO4. Evaluate NLP systems including bias and hallucination risks.

Course Content:

Module	Content
Module 1: Text and Linguistic Processing	Unicode, normalization, sentence splitting, tokenization, stemming/lemmatization and POS awareness. Topics include: ambiguity, corpus concepts and preprocessing trade-offs.
Module 2: Text Representation	Bag-of-words, n-grams, TF-IDF, similarity and embeddings. Topics include: sparse vectors, cosine similarity, word/contextual embedding concepts.
Module 3: Text Classification and Information Extraction	Classification, sentiment, NER and evaluation. Topics include: precision/recall/F1, imbalance and error analysis.
Module 4: Sequence Models and Transformers	Language modelling, RNN/LSTM overview, attention and transformers. Topics include: encoder/decoder, self-attention and pretrained-model inference.
Module 5: Generative NLP and Responsible Use	Prompting, summarization/QA, LLM overview, hallucination, bias, privacy and evaluation. Topics include: grounding/RAG awareness and human review.

Assessment activities:

Text preprocessing assignment. • Classification/NER notebook. • Transformer inference practical. • Responsible NLP evaluation report.

Practical and project work:

Tokenization; TF-IDF; similarity; text classification; NER; pretrained transformer inference; bias/error analysis.

Recommended Books

Daniel Jurafsky & James H. Martin, Speech and Language Processing.

Jacob Eisenstein, Introduction to Natural Language Processing.

Digital learning resources: NPTEL/SWAYAM: Natural Language Processing. NPTEL official NLP lectures; Hugging Face official course/video material may supplement.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 305B	Internet of Things and Edge Computing	2-0-2	3

Prerequisites: Networks and programming fundamentals.

Course Objective:

Develop end-to-end IoT and edge-computing understanding. Build secure sensor-to-cloud/edge prototypes. Address reliability, power and device-management constraints.

Course Outcomes:

CO1. Explain IoT/edge architectures and devices.

CO2. Use common IoT messaging protocols.

CO3. Prototype a sensor-data workflow.

CO4. Analyse IoT security and lifecycle risks.

Course Content:

Module	Content
Module 1: Embedded and IoT Foundations	Microcontrollers, sensors, actuators, GPIO/ADC/DAC and IoT reference architectures. Topics include: device constraints, sampling and gateway roles.
Module 2: Connectivity and Messaging	Wi-Fi/BLE/Zigbee/LPWAN awareness, MQTT, CoAP and HTTP. Topics include: QoS, publish/subscribe and protocol selection.
Module 3: Edge and Cloud Integration	Edge processing, gateways, ingestion, storage, dashboards and APIs. Topics include: time-series data, rules/events and latency trade-offs.
Module 4: Device Management and Security	Provisioning, identity, secure communication, firmware update and lifecycle. Topics include: secure boot awareness, credential rotation and threat modelling.
Module 5: IoT Solution Design	Smart home/industry/agriculture/health cases, power/reliability/scalability and prototype evaluation. Topics include: offline behaviour, privacy and maintainability.

Assessment activities:

Protocol-selection case. • MQTT/telemetry practical portfolio. • IoT threat model. • Mini sensor-to-dashboard prototype.

Practical and project work:

Sensor simulation; MQTT pub/sub; HTTP telemetry; edge filtering; dashboard; device-security checklist; integrated mini prototype.

Recommended Books

Arshdeep Bahga & Vijay Madisetti, Internet of Things: A Hands-On Approach.

Raj Kamal, Internet of Things, McGraw-Hill.

Digital learning resources: NPTEL/SWAYAM: Introduction to Internet of Things. NPTEL official IoT lectures; official Arduino/ESP32/Raspberry Pi provider resources where used.

Centre for Distance ...
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 305C	Business Intelligence and Data Visualization	2-0-2	3

Prerequisites: Database and data-analysis fundamentals.

Course Objective:

Develop business-intelligence and dashboard design competence. Connect KPIs, dimensional data and visualization. Emphasize interpretation, usability and ethical communication.

Course Outcomes:

CO1. Define KPIs and analytical questions.

CO2. Prepare data for dashboards.

CO3. Build interactive visualizations.

CO4. Communicate insights without misleading users.

Course Content:

Module	Content
Module 1: BI Foundations	Decision support, BI architecture, KPIs, metrics and analytical questions. Topics include: operational vs analytical reporting, stakeholder needs and KPI quality.
Module 2: Data Preparation for BI	Data cleaning, aggregation, joins/relationships, dimensions/hierarchies and calculated fields. Topics include: semantic models and refresh awareness.
Module 3: Visualization Principles	Data types, visual encoding, chart selection, scales, colour and annotation. Topics include: perception, misleading charts, accessibility and uncertainty.
Module 4: Interactive Dashboards	Filters, parameters, drill-down, maps, KPI cards and storytelling. Topics include: layout hierarchy, mobile views and dashboard performance.
Module 5: BI Governance and Storytelling	Sharing, security, governance, versioning and executive communication. Topics include: data ethics, context, limitations and action-oriented recommendations.

Assessment activities:

KPI definition task. • Dashboard build portfolio. • Peer usability/accessibility review. • Executive insight brief.

Practical and project work:

Data model; calculated fields; charts; filters/drill-down; dashboard; story; sharing/security; final BI case.

Recommended Books

Cole Nussbaumer Knaflic, Storytelling with Data.

Andy Kirk, Data Visualisation, Sage.

Digital learning resources: SWAYAM/NPTEL: Data Visualization / Business Analytics offerings. Microsoft Power BI / Tableau official learning videos if the approved tool uses those platforms.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER III

Course Code	Course	L-T-P	Credits
MCAOL 306	Research Methodology and Technical Writing	2-0-0	2

Prerequisites: Professional Communication and Research Literacy.

Course Objective:

Prepare learners for postgraduate dissertation/project inquiry. Develop research design, literature review and technical-writing competence. Embed ethics, reproducibility and academic integrity.

Course Outcomes:

CO1. Formulate research/problem questions and objectives.

CO2. Conduct a structured literature/technology review.

CO3. Select suitable methodology and evaluation criteria.

CO4. Prepare a proposal/report with ethical citation and reproducibility considerations.

Course Content:

Module	Content
Module 1: Research and Problem Formulation	Research vs development, problem statements, questions, objectives, scope and hypotheses where relevant. Topics include: feasibility, novelty and contribution.
Module 2: Literature and Evidence	Search strategy, databases, screening, note taking, synthesis and gap identification. Topics include: citation chaining, source quality and literature matrices.
Module 3: Research Design and Methods	Quantitative, qualitative, experimental, design-science and software/data project methods. Topics include: variables, sampling, baselines, metrics and validity.
Module 4: Data, Ethics and Reproducibility	Data collection, consent/privacy, bias, reproducibility, versioning and research ethics. Topics include: data management, AI-use disclosure and responsible reporting.
Module 5: Technical Writing and Proposal Development	Proposal/report structure, figures/tables, references, plagiarism, presentation and viva. Topics include: abstract, methodology, results, limitations and future work.

Assessment activities:

Research-question refinement exercise. • Literature matrix with 8-10 credible sources. • Methodology/evaluation design assignment. • Mini research/project proposal with references and integrity declaration.

Practical and project work:

Literature-search workshop; citation manager demonstration; proposal peer review; research-presentation practice.

Recommended Books

C.R. Kothari, Research Methodology: Methods and Techniques.

John W. Creswell, Research Design.

Digital learning resources: NPTEL/SWAYAM: Research Methodology; Technical Communication. NPTEL official Research Methodology lectures; IEEE/ACM author resources.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER IV

Course Code	Course	L-T-P	Credits
MCAOL 401	Emerging Technologies and Generative AI	3-0-2	4

Prerequisites: AI/ML, cloud and programming fundamentals.

Course Objective:

Expose learners to contemporary computing technologies with emphasis on generative AI. Develop practical literacy in foundation models, prompt design, grounding and responsible deployment. Build technology-evaluation skills rather than platform-specific dependence.

Course Outcomes:

CO1. Explain foundation-model and generative-AI concepts.

CO2. Use generative-AI tools/APIs responsibly for constrained tasks.

CO3. Describe RAG, agents and multimodal-system concepts.

CO4. Evaluate emerging technologies for feasibility, risk and governance.

Course Content:

Module	Content
Module 1: Generative AI Foundations	Generative models overview, embeddings, transformers, foundation models, tokens/context and inference. Topics include: pretraining/fine-tuning concepts, temperature/top-p awareness and model limitations.
Module 2: Prompting and Application Patterns	Prompt structure, few-shot examples, structured output, tool use and evaluation. Topics include: prompt injection awareness, hallucination reduction and reproducible prompt tests.
Module 3: Retrieval-Augmented and Agentic Systems	Embeddings/vector search, RAG architecture, tool/function calling and agent workflows. Topics include: chunking, retrieval quality, grounding, memory/state awareness and safeguards.
Module 4: Multimodal and Emerging Computing	Multimodal AI, edge AI, blockchain/Web3 awareness, digital twins, quantum-computing overview and autonomous systems. Topics include: use-case suitability and maturity assessment.
Module 5: Responsible Deployment and Governance	Privacy, security, bias, copyright/IP awareness, evaluation, monitoring, cost and sustainability. Topics include: red-teaming awareness, human-in-the-loop and organizational AI policy.

Assessment activities:

Prompt-evaluation workbook. • Small RAG/grounded-answer prototype or simulation. • Emerging-technology feasibility brief. • Responsible-AI risk assessment and presentation.

Practical and project work:

Prompt experiments; structured output; embeddings/similarity; simple RAG using approved environment; tool-calling demo; multimodal demo; safety/evaluation checklist; final use-case prototype.

Recommended Books

Selected current technical readings from reputable publishers/provider documentation; no single textbook should be relied on exclusively.

Russell & Norvig, Artificial Intelligence: A Modern Approach, selected relevant background.

Digital learning resources: NPTEL/SWAYAM AI/Deep Learning offerings; approved current GenAI short course from recognized platform if permitted. Official Google for Developers / Microsoft / AWS / OpenAI provider education may be used only after coordinator verification; links should be reviewed each semester.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER IV

Course Code	Course	L-T-P	Credits
MCAOL 402A	Computer Vision	3-0-2	4

Prerequisites: Python, machine learning and basic linear algebra.

Course Objective:

Develop image-processing and computer-vision foundations. Apply classical and deep-learning vision methods. Address evaluation and responsible use.

Course Outcomes:

CO1. Process and transform images.

CO2. Apply segmentation and feature techniques.

CO3. Use CNN/transfer-learning concepts.

CO4. Evaluate vision systems and failure cases.

Course Content:

Module	Content
Module 1: Image Fundamentals	Sampling/quantization, colour models, geometric transforms and histograms. Topics include: resolution, interpolation and image normalization.
Module 2: Filtering and Enhancement	Convolution, smoothing, sharpening, edge detection and morphology. Topics include: noise models and parameter trade-offs.
Module 3: Segmentation and Features	Thresholding, contours, regions, keypoints/descriptors and matching. Topics include: connected components and feature robustness.
Module 4: Deep Vision	CNN concepts, training, augmentation, transfer learning, object detection/segmentation overview. Topics include: feature maps and pretrained models.
Module 5: Evaluation and Responsible Vision	Accuracy/IoU awareness, error analysis, privacy, bias and deployment constraints. Topics include: surveillance concerns and dataset shift.

Assessment activities:

Image-processing notebook. • Segmentation/feature task. • Transfer-learning experiment. • Vision model error/bias report.

Practical and project work:

Image I/O/transform; filtering; edges; morphology; segmentation; feature matching; transfer-learning classifier; evaluation/failure analysis.

Recommended Books

Rafael C. Gonzalez & Richard Woods, Digital Image Processing, Pearson.

Richard Szeliski, Computer Vision: Algorithms and Applications.

Digital learning resources: NPTEL/SWAYAM: Computer Vision / Digital Image Processing. NPTEL official Computer Vision lectures; OpenCV official resources.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER IV

Course Code	Course	L-T-P	Credits
MCAOL 402B	Mobile and Cross-Platform Application Development	3-0-2	4

Prerequisites: Object-oriented programming and web/API fundamentals.

Course Objective:

Develop practical mobile-app design and development competence. Integrate UI, local/network data and device services. Emphasize privacy, testing and deployment quality.

Course Outcomes:

CO1. Design accessible mobile interfaces.

CO2. Manage application state/navigation.

CO3. Integrate local persistence and APIs.

CO4. Test and package a mobile application.

Course Content:

Module	Content
Module 1: Mobile Architecture and UI	Mobile ecosystems, native/cross-platform choices, project structure, layouts/components and navigation. Topics include: adaptive design and accessibility.
Module 2: State, Lifecycle and Forms	Lifecycle/state management, events, validation and reusable components. Topics include: asynchronous UI and error states.
Module 3: Data and Services	Local storage/database, JSON, REST APIs and offline concepts. Topics include: caching, synchronization and network resilience.
Module 4: Device Capabilities and Security	Permissions, notifications, camera/location awareness, authentication, privacy and secure storage. Topics include: least privilege and platform policy.
Module 5: Quality and Release	Testing, debugging, performance/battery, packaging, signing and release/deployment overview. Topics include: crash/error reporting and final app documentation.

Assessment activities:

UI prototype. • API/data integration assignment. • Device/security checklist. • Final mobile app demo + viva.

Practical and project work:

UI/navigation; forms/state; local storage; REST API; permissions; notifications/device-service demo; testing; package/release checklist.

Recommended Books

Official Android/Flutter/React Native documentation for approved stack.

Reto Meier, Professional Android, selected relevant concepts if Android is used.

Digital learning resources: SWAYAM/NPTEL Mobile Computing / Application Development offerings where available. Official Android Developers / Flutter / React Native provider videos based on approved stack.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University Gwalior

MCA – SEMESTER IV

Course Code	Course	L-T-P	Credits
MCAOL 402C	Enterprise Application and CRM Platform Development	3-0-2	4

Prerequisites: Database, web and object-oriented application-development fundamentals.

Course Objective:

Introduce enterprise platform architecture, configuration and low-code/developer workflows. Develop data, security, automation and integration competence. Provide industry-relevant CRM/platform skills without locking the academic syllabus to one vendor.

Course Outcomes:

CO1. Model enterprise CRM data and security.

CO2. Configure workflows, reports and automation.

CO3. Explain extension/developer and API concepts.

CO4. Build a small enterprise application in an approved sandbox/platform.

Course Content:

Module	Content
Module 1: Enterprise CRM and Platform Architecture	CRM concepts, metadata-driven platforms, objects/entities, relationships and environments. Topics include: standard/custom data models and multi-tenant awareness.
Module 2: Identity, Security and Data Governance	Users, roles, profiles/permission sets, sharing, field security and audit concepts. Topics include: least privilege and data-quality rules.
Module 3: Low-Code Automation and Analytics	Forms/pages, validation, workflow/flow automation, reports and dashboards. Topics include: approvals, notifications and KPI reporting.
Module 4: Developer Extensions and Integration	Platform language/component concepts, APIs, queries and event/integration awareness. Topics include: governor/resource limits and secure integrations.
Module 5: Testing, Deployment and Lifecycle	Sandbox strategy, testing, data import/export, packaging/change promotion and release management. Topics include: version control, documentation and final enterprise solution.

Assessment activities:

Data model/security configuration assignment. • Automation/reporting practical. • API/developer-extension mini task. • Enterprise sandbox solution demo + viva.

Practical and project work:

Custom objects/entities; access control; validation; report/dashboard; workflow/flow; query/API; extension code awareness; data import/export; final CRM solution.

Recommended Books

Official platform documentation for the University-approved CRM/enterprise platform.

Selected enterprise application architecture reference approved by faculty.

Digital learning resources: Approved Salesforce Trailhead / Microsoft Power Platform / equivalent enterprise learning path, subject to University approval. Salesforce Developers / Microsoft Power Platform official videos depending on the approved platform.

MCA – SEMESTER IV

Course Code	Course	L-T-P	Credits
MCAOL 403	MOOC / Industry Certification	0-0-4	2

Prerequisites: Prerequisites depend on the approved certification track.

Course Objective:

Integrate verified external learning with MCA programme outcomes. Provide flexible industry-aligned upskilling suitable for ODL. Authenticate external learning through University assessment.

Course Outcomes:

CO1. Complete an approved MOOC/certification pathway.

CO2. Demonstrate practical competence through portfolio evidence.

CO3. Relate external learning to programme outcomes.

CO4. Reflect on limitations, ethics and future skill development.

Course Content:

Module	Content
Stage 1: Approval and Learning Contract	Selection from University-approved list; prerequisites, outcomes, provider, hours, assessment and mapping.
Stage 2: Guided Progress	Weekly/milestone progress log, quizzes and practice evidence; mentor check-ins.
Stage 3: Practical Evidence	Provider labs/notebooks/configurations or equivalent artefacts; detailed subtopics depend on approved pathway.
Stage 4: Applied Integration	Small application/case demonstrating how certification learning connects to MCA knowledge.
Stage 5: University Authentication	Certificate/completion evidence, portfolio, reflective report and viva/practical authentication.

Assessment activities:

Provider progress/completion evidence. • Practical portfolio. • Applied integration task. • University viva/reflection; indicative 20/30/30/20 distribution subject to policy.

Practical and project work:

Approved provider labs; University does not accept certificate alone as sufficient evidence of academic attainment.

Recommended Books

Provider materials plus faculty-selected supporting references.

Digital learning resources: University-approved SWAYAM/NPTEL/industry certifications in areas such as cloud, cybersecurity, data/AI, DevOps or enterprise platforms. Only official provider/institution channels; exact certification/video list to be approved and refreshed each semester.

DIRECTOR
Centre for Distance and
Online Education (CDOE)
ITM University

MCA – SEMESTER IV

Course Code	Course	L-T-P	Credits
MCAOL 404	Major Project / Dissertation / Industrial Project	0-0-26	13

Prerequisites: Successful completion of earlier semesters and Research Methodology; project area should match available supervision/resources.

Course Objective:

Provide a substantial culminating postgraduate computing experience. Integrate analysis, design/research, implementation, evaluation and professional documentation. Authenticate independent contribution and postgraduate-level judgement.

Course Outcomes:

CO1. Formulate a substantial computing problem or research question.

CO2. Conduct literature/technology review and select a defensible methodology.

CO3. Implement/evaluate a substantial solution, study or industry project.

CO4. Prepare a rigorous report/dissertation with reproducible evidence.

CO5. Defend decisions, results, limitations and individual contribution in viva.

Course Content:

Module	Content
Stage 1: Proposal and Review	Problem/research question, context, literature/technology review, gap, objectives, scope, contribution and ethics.
Stage 2: Design and Planning	Requirements/research design, architecture, data/method, evaluation metrics, project plan, risks, privacy/security and reproducibility plan.
Stage 3: Implementation / Investigation	Iterative development, data collection/experiments or industrial work; repository/logbook and supervisor review evidence.
Stage 4: Validation and Analysis	Testing, benchmarking, experiments, user evaluation or professional evidence; comparison with baseline; error/failure analysis and limitations.
Stage 5: Dissertation / Final Defence	Structured report, references, source/data/artefacts, user/deployment guidance where relevant, presentation, live demo and individual viva. Industrial-project variant also includes host mentor evaluation and competency mapping.

Assessment activities:

Proposal review 10%; literature/method/design review 15%; progress demonstration 20%; testing/evaluation review 15%; final artefact/dissertation 25%; presentation/viva 15% (indicative only; University policy prevails).

Practical and project work:

Continuous supervised project/dissertation/industrial evidence; repository/version history; weekly/fortnightly log; experiments/tests; supervisor feedback; pre-submission integrity check; final demonstration and viva.

Recommended Books

Roger Pressman & Bruce Maxim, Software Engineering: A Practitioner's Approach.

John W. Creswell, Research Design, where research-oriented.

University MCA Major Project/Dissertation/Industrial Project Handbook (to be approved).

Digital learning resources: Domain-specific SWAYAM/NPTEL learning may supplement but cannot replace supervision, project work or viva. CDOE project/research clinics plus official technology-provider videos relevant to the approved project domain.

Centre for Distance and Online Education (CDOE)
ITM University Gwalior