



**UNIVERSIDAD
DE CUENCA**

**GENERAL RUBRIC
STUDENT OUTCOMES**

COMPUTER SCIENCE PROGRAM

2025

SO1: Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.

PI1.1: Clearly and thoroughly articulate the problem statement, identifying all the requirements and constraints.

PI1.2: Use relevant theories, methods, tools, and techniques to analyze and break down the problem into well-defined parts.

PI1.3: Critically assess multiple potential solutions, considering all relevant factors, and justify the chosen approach.

Criteria	Failed (0-59.99%):	Regular (60-69.99%):	Good (70-79.99%):	Very Good (80-89.99%):	Excellent (90-100%):
Problem statement, requirements and constraints	Problem statement is unclear or incomplete; requirements and constraints are poorly identified or missing.	Basic articulation of the problem statement; some requirements and constraints are identified but lack depth.	Problem statement is clear; most requirements and constraints are identified with minor omissions.	Problem statement is well-articulated; all major requirements and constraints are identified and detailed.	Exceptionally clear and comprehensive articulation of the problem, identifying all requirements and constraints thoroughly.
Theories, methods, tools and techniques for analyzing the problem	Minimal or no use of relevant theories, methods, or tools; problem breakdown is ineffective.	Limited use of relevant tools and methods; problem breakdown is partially effective but incomplete.	Employs appropriate theories and tools; problem breakdown is effective but lacks precision.	Thorough and accurate use of theories and tools; problem breakdown is well-defined and logical.	Masterful use of relevant methods and tools; problem breakdown is insightful, comprehensive, and precise.
Potential solutions, considering relevant factors and justification	Little or no assessment of solutions; justification is absent or irrelevant.	Basic assessment of solutions; justification lacks depth or misses key factors.	Adequate assessment of solutions with reasonable justification for the chosen approach.	Thorough assessment of multiple solutions; chosen approach is well-justified with sound reasoning.	Comprehensive and critical assessment of all relevant solutions; justification is highly persuasive and well-supported.

SO2: Design, implement, and evaluate a computing-based solution.

PI2.1: Design a comprehensive solution addressing all requirements and constraints.

PI2.2: Use suitable tools and methodologies to implement the solution.

PI2.3: Conduct thorough testing and validation to meet predefined requirements.

PI2.4: Document the design, implementation, and validation processes.

Criteria	Failed (0-59.99%):	Regular (60-69.99%):	Good (70-79.99%):	Very Good (80-89.99%):	Excellent (90-100%):
Comprehensive solutions addressing all requirements and constraints	Design fails to address key requirements or constraints; lacks coherence.	Partial design that addresses some requirements but misses important details.	Design addresses most requirements; minor gaps or inefficiencies present.	Comprehensive design that effectively addresses all requirements and constraints.	Exceptionally innovative and comprehensive design; exceeds expectations and demonstrates creativity.
Suitable tools and methodologies to implement solutions	Implementation does not align with design; inappropriate tools or methods used.	Basic implementation; some alignment with design but with significant errors or inefficiencies.	Adequate implementation using appropriate tools and methods; minor errors present.	Effective implementation; tools and methods are used proficiently and align with design	Highly efficient and precise implementation using advanced tools and methods.
Testing and validation to meet predefined requirements	Minimal or no testing conducted; significant issues remain unresolved.	Limited testing conducted; addresses some but not all issues.	Adequate testing identifies and resolves most issues; minor gaps remain.	Comprehensive testing ensures the solution meets nearly all requirements.	Rigorous testing with detailed validation ensures complete alignment with all requirements.
Design, implementation and validation	Documentation is incomplete, unclear, or missing critical details.	Documentation is basic and covers some processes but lacks detail or clarity.	Documentation is clear and detailed, covering most processes effectively.	Thorough and well-structured documentation covering all processes clearly.	Exceptionally detailed and professional documentation, suitable for diverse audiences.

S03: Communicate effectively in professional contexts.

PI3.1: Write detailed and well-structured reports.

PI3.2: Present technical concepts clearly to diverse audiences.

PI3.3: Participate in discussions and collaborations constructively.

Criteria	Failed (0-59.99%):	Regular (60-69.99%):	Good (70-79.99%):	Very Good (80-89.99%):	Excellent (90-100%):
Detailed and well-structured reports	Reports are poorly written, unstructured, or fail to convey key information.	Reports convey basic information but are unorganized or lack detail.	Reports are organized and detailed but may contain minor errors or inconsistencies.	Reports are clear, detailed, and well-structured with minimal errors.	Reports are exemplary, comprehensive, and professionally presented.
Technical concepts for diverse audiences	Presentations are unclear, unengaging, or inappropriate for the audience.	Basic presentation with limited clarity or audience engagement.	Clear and engaging presentation, with minor gaps in communication.	Effective and engaging presentation tailored to the audience.	Outstanding presentation that is engaging, clear, and highly tailored to the audience.
Discussions and constructive collaborations	Minimal or disruptive participation in discussions or collaborations.	Limited participation; contributions lack depth or focus.	Active participation with relevant contributions; minor lapses in engagement.	Consistently contributes valuable insights and engages constructively.	Exceptional participation with insightful contributions and strong collaboration.

SO4: Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.

PI4.1: Recognize and articulate ethical dilemmas in computing scenarios, considering the impact on society.

PI4.2: Make informed decisions based on established ethical guidelines and professional standards.

PI4.3: Assess the social, legal, and economic implications of computing solutions.

Criteria	Failed (0-59.99%):	Regular (60-69.99%):	Good (70-79.99%):	Very Good (80-89.99%):	Excellent (90-100%):
Ethical dilemmas in computing scenarios	Fails to recognize ethical dilemmas or articulate their societal impact.	Identifies basic ethical issues but provides limited or superficial societal impact analysis.	Identifies ethical dilemmas and articulates societal impact with reasonable depth.	Thoroughly identifies ethical dilemmas and provides a well-rounded analysis of societal impact.	Demonstrates exceptional insight into ethical dilemmas, with deep and nuanced societal impact analysis.
Informed decision based on established ethical guidelines and professional standards	Decisions do not align with ethical guidelines; reasoning is unclear or flawed.	Makes basic decisions referencing ethical guidelines but with limited justification.	Decisions align with ethical guidelines and are supported by reasonable justification	Decisions demonstrate strong adherence to ethical guidelines, with thorough justification	Consistently makes highly informed and well-justified decisions adhering to ethical standards.
Social, legal and economic implications of computer solutions	Demonstrates little or no understanding of social, legal, or economic implications	Assesses some implications but lacks depth or misses key aspects.	Provides a balanced assessment of social, legal, and economic implications with minor gaps.	Thorough assessment of implications, addressing all relevant aspects effectively.	Exceptional assessment that deeply examines all implications, with insightful conclusions.

SO5: Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.

PI5.1: Actively participate in team activities, taking on appropriate roles and responsibilities.

PI5.2: Share information, ideas, and resources openly and respectfully.

PI5.3: Address conflicts promptly and professionally.

Criteria	Failed (0-59.99%):	Regular (60-69.99%):	Good (70-79.99%):	Very Good (80-89.99%):	Excellent (90-100%):
Team activities, roles and responsibilities	Minimal or no participation; roles and responsibilities are ignored or poorly handled.	Basic participation; roles and responsibilities are accepted but not fully carried out.	Actively participates and fulfills roles/responsibilities with some gaps.	Consistently participates and fulfills roles/responsibilities effectively.	Exemplary participation, consistently exceeding role expectations and contributing significantly to the team.
Respectful information exchange	Rarely shares information or does so in an unhelpful or disrespectful manner.	Shares some information but with limited openness or respect for others.	Shares relevant information and ideas respectfully with occasional lapses.	Consistently shares information and ideas openly and respectfully, fostering collaboration.	Proactively shares valuable information, fostering a highly collaborative and respectful environment.
Conflict addressing	Avoids or exacerbates conflicts; responses are unprofessional.	Addresses conflicts minimally or inconsistently; professionalism is limited.	Resolves conflicts professionally, with minor gaps in timeliness or effectiveness.	Effectively and professionally resolves conflicts in a timely manner.	Exemplary conflict resolution skills, fostering positive outcomes promptly and professionally.

SO6: Apply computer science theory and software development fundamentals to produce computing-based solutions.

PI6.1: Utilize core concepts such as algorithms, data structures, and computational theory to develop effective solutions.

PI6.2: Implement efficient algorithms and data structures tailored to specific problem requirements.

PI6.3: Conduct comprehensive testing to verify correctness and reliability of software.

Criteria	Failed (0-59.99%):	Regular (60-69.99%):	Good (70-79.99%):	Very Good (80-89.99%):	Excellent (90-100%):
Development of effective solutions by using core concepts	Demonstrates little or no understanding of core concepts; solutions are ineffective or incorrect.	Applies core concepts minimally; solutions are basic and lack depth.	Adequately applies core concepts to develop effective solutions with minor issues.	Effectively applies core concepts to produce strong solutions with minimal gaps.	Demonstrates mastery of core concepts, producing innovative and highly effective solutions.
Efficient algorithms and data structures tailored to specific problem requirements	Algorithms and data structures are inefficient or fail to meet requirements .	Implements basic algorithms and data structures with limited optimization.	Implements appropriate and moderately efficient algorithms and data structures.	Implements highly efficient algorithms and data structures that align well with requirements .	Consistently implements optimal algorithms and data structures tailored perfectly to problem requirements.
Verification of correctness and reliability of the software	Little or no testing conducted; software has significant unresolved issues.	Basic testing identifies some issues but leaves gaps in verification.	Adequate testing identifies and resolves most issues effectively.	Thorough testing ensures software correctness and reliability with minor gaps.	Rigorous and comprehensive testing ensures highly reliable and robust software.