SPECIFIC COMPETENCES (OUTCOMES) OF PROFILES ASSESSED BY CONAIC
DEFINED BY ANIEI AND CONAIC

conaic

Bachelor degree in Software Engineering – B

Specific competences. Profile B:

CURRICULUM PERIODS	COURSE	1 Performs software engineering requirements	EVIDENCE	2 Designs Software	EVIDENCE	3 Builds software	EVIDENCE	4 Carries out Software Testing	EVIDENCE	5 Carries out Software Maintenance	EVIDENCE	6 Manages software projects	EVIDENCE	7 Estimates parameters for the software project	EVIDENCE	8 Ensures Software Quality	EVIDENCE	9 Establishes security mechanisms	EVIDENCE	10 Uses life cycles	EVIDENCE	11 Checks software solutions quality	EVIDENCE	12 Uses software creation tools	EVIDENCE
COMPETENCE (OUTCOME)																									
Attribute		Recognizes the context and needs, and individuals involved in a system using techniques to identify, collect, analyze, prioritize, document, verify and validate the requirements in the context of life cycles and software development processes.		Designs and Evaluates the behavior, architecture and interface of software solutions based on requirements and using strategies, methods, techniques and modeling languages characteristic to software design.		Develops software for different types of applications, using programming methodologies and paradigms in the context of life cycles and software development processes, with the required quality attributes.		Plans, assigns and runs types, techniques, processes and controls inside test scenarios according to the required quality attributes.		Applies maintenance types, processes and techniques, in accordance with the required quality attributes.		Uses methods, strategies, processes, tools and techniques for software projects management.		Applies metrics for software estimation (size, cost, effort, personnel, time, productivity, quality and documentation) in accordance with system life cycle models.		Uses techniques, tools, and strategies for planning, ensuring and controlling a software product quality.		Creates, Evaluates or proposes methods and strategies to evaluate safety and selection of criteria to avoid security vulnerabilities in the software.		Uses elements and criteria in the use of life cycle models in accordance with the context of software development processes.		Uses various test models in order to ensure software product quality.		Used industrial methods and CASE tools for the different stages in the software process.	
		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE		A) START B) DEVELOP C) EVALUATE	
1																									
2																									
3																									
4																									