|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TSC Category** | Development and Implementation | | | | | |
| **TSC Title** | Self-Learning Systems | | | | | |
| **TSC Description** | Design and develop self-learning systems using reinforcement learning and evolutionary learning techniques | | | | | |
| **TSC Proficiency Description** | **Level 1** | **Level 2** | **Level 3** | **Level 4** | **Level 5** | **Level 6** |
|  |  | **ICT-DIT-3028-1.1** | **ICT-DIT-4028-1.1** | **ICT-DIT-5028-1.1** |  |
|  |  | Analyse, articulate and apply key artificial intelligence (AI) technologies in their work and that of the teams and organisation, in the area of business process automation and optimisation | Plan the end-to-end process to design, build and deploy adaptive software robots in hardware and devices, validating and optimising software robots in different application areas | Design and develop self-learning systems using reinforcement learning and evolutionary learning techniques |  |
| **Knowledge** |  |  | * Overview of robotic process automation (RPA), cognitive RPA, and artificial intelligence (AI) * Tooling tutorials * Data analysis and manipulation using virtual worker * Processing supplier invoice using virtual workers * Remote enterprise aplications * Data processing and management * Enterprise deployment methodology and tools * Quality assurance and system debugging * Intelligent Process Automation (IPA) project management * Open source IPA tools, installation and use cases * Automation using AI functionalities * Open-source conversational software robots fundamentals and installation | * Autonomy and agency * Automating repetitive tasks * Differences between Single agents and multiple agents * Sequences, flowchart, State Machine Workflow * User interface automation * System activities and user events * Recording and scraping websites * Testing and debugging * Multi-agent robots * Industry best practices in software robots * Application integration * Embedding and deploying robots * Validation and verification | * Types and applications of self-learning systems * Reinforcement learning process and approaches * Concepts and algorithms of Deep Reinforcement Learning (DRL) * Fundamentals of reinforcement learning and evolutionary learning techniques * Evolutionary learning * Evolutionary intelligent systems and its architecture and applications * Evolutionary computation techniques |  |
| **Abilities** |  |  | * Analyse business drivers and main application areas of IPA * Create automation applications using RPA as an integration tool between different enterprise applications * Create virtual workers for productivity without increasing actual workforce and/or hiring * Create automation applications with remote enterprise applications * Automate data processing and extraction from image documents * Deploy enterprise-scale automation applications * Manage IPA project lifecycle * Create business automation applications using cost-saving robotic automation tools and AI tools * Create digital ‘virtual assistants’ * Evaluate industrial IPA use cases | * Determine the best practices for building autonomous software robots * Plan the processes to design and build software robots * Define the technical requirements needed for the sofware robots * Construct fully functional software robots * Validate the design of the software robots * Introduce new and/or best practices in the design and planning of the software robots * Gather feedback on the performance of the software robots * Review the cording, testing and design criteria * Optimise the performance of the software robots during and after deployment | * Apply the concepts and algorithms of reinforcement learning * Identify the requirements for self-learning systems * Build deep reinforcement learning systems * Build model-based reinforcement learning systems * Build evolutionary learning systems using evolutionary computation techniques to solve optimisation problems * Assess the system performance and suggest possible improvements |  |
| **Range of Application** |  | | | | | |