

Software Architect

ROLE OVERVIEW

As a Software architect, you will research, design, evaluate, integrate, and maintain software applications, technical environments, operating systems, embedded software, information warehouses and telecommunications software.

You could be employed in information technology consulting firms, information technology research and development firms, or information technology units throughout the private and public sectors.

Software architects create the technical design and the functional model of a software system, based on functional specifications. They also design the architecture of the system or different modules and components related to the businesses or customer requirements, technical platform, computer language or development environment.

STRATA LEVEL: 3B – Technical Specialist

Also Known as:

- Embedded Software Engineer
- Software Designer
- Software Technical Architect

Education and Experience:

- A bachelor's degree, usually in computer science, computer systems engineering, software engineering or mathematics or completion of a college program in computer science is usually required.
- A master's or doctoral degree in a related discipline may be required.
- Licensed by a provincial or territorial association as a Professional Engineer (P.Eng.) may be required.

Associated NOC(s):

- **2173** – Software Engineers and Designers



TECHNICAL



Software and Systems Research

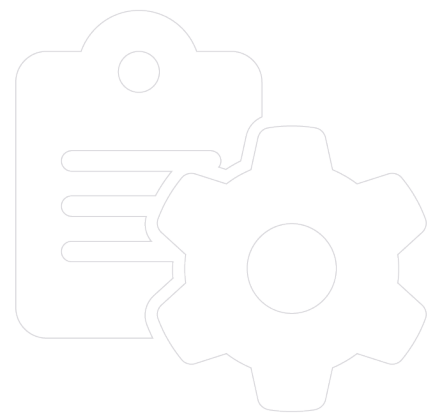
Research and updates knowledge and skills related to computer software and systems to keep up with technological advancements and recommend latest innovations or product improvements.

- Conducts comprehensive review of information and publications to ensure a complete understanding of a subject prior to development.
- Research emerging aspects in software and technological applications to recommend new applications and procedures when available.
- Assesses user requirements prior to undertaking projects to develop software applications or operating systems to meet user needs.
- Conducts market research using techniques and tools to elicit, define, or document user requirements for system or product.

Software Architecture

Creates and documents the structure of software products, systems, interfaces, or applications to ensure feasibility, functionality, and compatibility with existing platforms.

- Analyze and interprets client requirements and architecture models to inform software design activities.
- Structures software requirements into simplified classes to be used to blueprint objects to allow for improved data structures and reusability.
- Uses graphical notations to express software designs to produce a standard visualization of the system design to communicate with other staff, explore design potential, or validate the architectural design.
- Applies behaviour diagrams to show the dynamic behaviour of objects in a system to analyze how architecture interacts in dynamic settings to improve product, systems, interfaces, or applications.
- Applies structure diagrams to illustrate the static structure of the system and the different parts in different abstraction levels to demonstrate how each part is related to the greater system to represent meaningful concepts of a system.



Software Feasibility Assessment

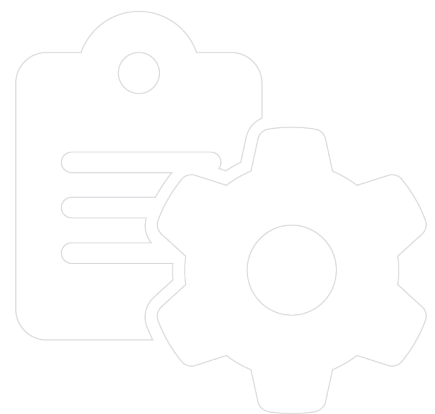
Assesses and evaluates software specifications to determine the feasibility of new or existing systems configurations to meet user requirements, and client goals (internal and external).

- Identifies the functional and non-functional requirements, constraints, and possible use cases of software to achieve positive interactions between users and software.
- Specifies the technical properties of goods, materials, methods, processes, systems, software, and functionalities required to achieve customer or organizational requirements.
- Assesses the financial or social costs and benefits of a software project or investment to determine the feasibility of design for a client or organization.
- Deliberates with other engineering staff to evaluate interface between hardware, software, operational, and performance requirements of system.

Feasibility Analysis

Produces a feasibility analysis of the practicality of a proposed project[s] including the economic viability, associated costs and benefits, and technical and time constraints of the project to ensure projects are completed on time and on budget.

- Analyzes the efficiency of existing infrastructure and processes to determine where improvements could be implemented.
- Assesses client needs and expectations to a project or service to identify and resolve inconsistencies to ensure project or service is completed resolutely.
- Analyzes the costs and benefits related to the design, development, and maintenance of a project to ensure organizational feasibility for the purposes of maintaining profitable operations.



Software Application Design

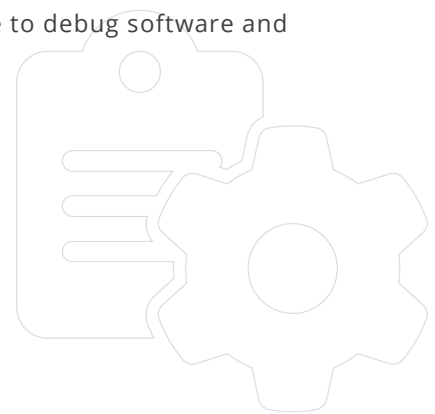
Devises a series of requirements into a clear and organized software design for the purposes of creating an application or system.

- Considers client requirements, target environment, user, and security requirements to develop software designs.
- Uses appropriate modelling techniques and design standards to design physical software components and modules.
- Develops a preliminary version of a software application or operating system to simulate aspects of the final product to present conceptual idea to client or organization.
- Composes flowchart diagrams using connecting lines and symbols to illustrate the systemic progress in a procedure or systems to communicate a visual process of the system or application.
- Uses reusable solutions and formulized best practices to solve common ICT tasks in software design and development.
- Identifies and manages changes to software designs to incorporate new business requirements to evolve software designs.

Software Modification/Debugging

Improves software or technical systems by amending codes, correcting faults, or upgrading programs to improve product functionality or maintain systems.

- Analyzes software testing results to identify defects in software to correct inputs, outputs, or unexpected results.
- Repairs defects identified in testing analysis to ensure accurate application outputs to ensure a functional product or system.
- Participates in peer code reviews by providing comments and feedback on software development to improve applications and team synergy.
- Identifies error types and applies appropriate method or technique to debug software and correct error to maintain usability of system.



DevOps

Applies appropriate software development methodologies and practices to manage the working relationship between the development and operations teams to encourage more efficient and effective products, systems, interfaces, or application development and releases.

- Applies an iterative process to software development seeking to automate and streamline the building, testing, and deployment process of new software products, systems, interfaces, or applications.
- Works with developers and other engineers to automate code integration and changes from multiple contributors into a single software project to enable developers to resolve issues more efficiently, improve quality, and reduce the time to update software.
- Uses processes to automate the software testing process to validate changes to code to accelerate the feedback loop between product initiation and customer delivery.
- Observes and detects compliance issues and security threats throughout each phase in the DevOps process to study metrics and ways to improve software in real-time.

Stakeholder Relations Management

Identifies the needs of relevant actors, working in partnerships with all necessary partners, to achieve a well-balanced solution to desired project, process, or program.

- Works towards establishing long terms relationships between organizations, clients, or other interested third parties to improve networks and develop potential future or returning clients.
- Seeks out new partnerships with potential clients to expand the organization's network and increase future potential relationships.
- Negotiates in the interest of the organization when working with clients to reach compromises and agreements so that all parties are satisfied.
- Illicit feedback from client(s) on software products, systems, interfaces, or applications to improve applications and deliver client satisfaction.
- Maintains working relationships, both internally and externally, to foster a network of influence and connectivity to improve future prospects.



Project Team Management

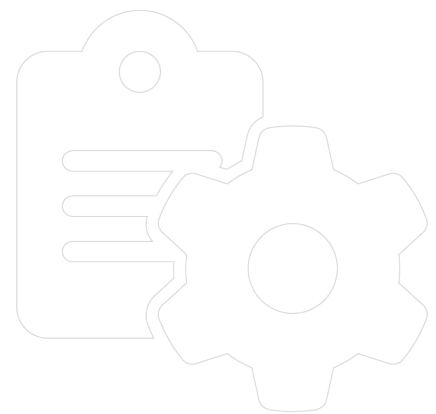
Oversees a team of professionals to effectively and efficiently produce the required output to ensure project[s] are completed on time and budget.

- Provides direction and supervision to engineers, technicians, and technologists in the design and development process to ensure clear and effective channels of communication across all departments.
- Monitors and controls the allocation of resources and reassigns staff as needed to support project deliverables.
- Sets clear accountability targets for supervised personnel to achieve project deliverables.
- Manage tasks and projects according to approved scopes of work to deliver quality reports on schedule and within budget
- Applies appropriate DevOps methods to remove barriers to communication and collaboration to improve the development process and product releases.

Computer Programming

Uses specialized software to develop code to instruct a computer, application, or software program how to perform under specific conditions so that applications or software function appropriately.

- *Insert.*



PERSONAL AND PROFESSIONAL



Communication

Positively directs outcomes by delivering communication that results in a better understanding of goals and objectives and that capture interest and gain support for immediate action.

- Shares valuable ideas in group meetings or discussions to improve system requirements or design functions of applications or systems.
- Communicates the project or organization mission and directive to ensure all team members share a cohesive vision for the progress of the organization.
- Prepares documentation for existing and upcoming products to describe functionality and composition to communicate technical specifications to a wide audience in plain language.
- Maintains a clear line of communication between the development and operations teams to maintain continuously information feedback loops on project deliverables.

Attention to Detail

Reviews completed work by monitoring and checking information, organizing tasks and resources efficiently, or all areas involved towards the completion of an objective.

- Provides accurate, consistent information on all pieces of work to ensure reliable results.
- Provides accurate and reliable feedback when reviewing others work to ensure consistent deliverables.
- Establishes procedures or processes to validate information to minimize disruptions to ensure project meets deliverables.
- Routinely checks in with client to take into consideration changing priorities or expectations to produce results that improve relationships and business objectives.
- Catches and corrects own errors or omissions, where applicable, to reduce future performance issues so that software products, systems, interfaces, or applications operate as expected.



Problem Solving

Identifies problems and uses logic, judgement, and evidence to evaluate alternative scenarios and recommend solutions to achieve a desired goal.

- Simplifies complex ideas and technical concepts into accessible information to communicate with stakeholders, senior management, and team members.
- Takes an unbiased stance to interpreting new information to solve a problem in an object manner.
- Identifies relationships between components and systems to come up with solutions to resolve problems.
- Applies logical and iterative analysis to identify the strengths and weaknesses of solutions to problems to evaluate potential outcomes.
- Maintains a working knowledge of computer programming techniques and languages to keep up to speed with developers to ensure harmonious project results.

Mentorship

Mentors junior staff by advising, supervising, and challenging them to facilitate the development and application of new knowledge in their role for the purposes of the delivery of sustainable environmental practices and professional growth.

- Engages with junior staff members to mentor and support growth to professional development.
- Shares knowledge with junior technical staff on institutional and organizational goals and values to promote a strong team environment.
- Provides constructive feedback for junior technical staff in meeting objectives to support professional development.
- Identifies training or developmental needs both for individuals and the organization and establishes new methods or programs to meet the need.
- Fosters a climate of openness and trust among team members to bolster creative thinking and improve decision making for product development.



LEGAL, REGULATORY, AND POLICY



Insert

Insert.

- *Insert*

Insert

Insert.

- *Insert*



ENVIRONMENTAL



Insert

Insert.

- *Insert.*

Insert

Insert.

- *Insert.*

