

**Name**: Kevin Taday

**Date**: 18-05-2018

**SOFTWARE REQUIREMENTS SPECIFICATION**

A software requirements specification (SRS) is a description of a [software system](https://en.wikipedia.org/wiki/Software_system) to be developed. It is modeled after [business requirements specification](https://en.wikipedia.org/wiki/Business_requirements" \o "Business requirements) [(CONOPS)](https://en.wikipedia.org/wiki/Concept_of_operations), also known as a [stakeholder requirements specification (StRS)](https://en.wikipedia.org/w/index.php?title=Stakeholder_requirements_specification_(StRS)&action=edit&redlink=1" \o "Stakeholder requirements specification (StRS) (page does not exist)).The software requirements specification lays out [functional](https://en.wikipedia.org/wiki/Functional_requirement" \o "Functional requirement) and [non-functional requirements](https://en.wikipedia.org/wiki/Non-functional_requirements), and it may include a set of [use cases](https://en.wikipedia.org/wiki/Use_case) that describe user interactions that the software must provide.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on how the software product should function (in a market-driven project, these roles may be played by the marketing and development divisions). Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure.

**Structure**

* [Purpose](https://en.wikipedia.org/wiki/Software_requirements_specification#Goals)
  + [Definitions](https://en.wikipedia.org/wiki/Definition)
  + Background
  + System overview
  + [References](https://en.wikipedia.org/wiki/Reference)
* [Overall description](https://en.wikipedia.org/wiki/High-_and_low-level)
  + [Product perspective](https://en.wikipedia.org/wiki/Product_requirements_document)
    - [System Interfaces](https://en.wikipedia.org/wiki/Interface_(computing))
    - [User interfaces](https://en.wikipedia.org/wiki/User_interface)
    - [Hardware interfaces](https://en.wikipedia.org/wiki/Hardware_interfaces)
    - [Software interfaces](https://en.wikipedia.org/wiki/Interface_(computing))
    - Communication Interfaces
    - [Memory Constraints](https://en.wikipedia.org/wiki/Computer_memory)
* Design constraints
* [Operations](https://en.wikipedia.org/wiki/Operations_support_system)
* Site Adaptation Requirements
* Product functions
* User characteristics
* Constraints, assumptions and dependencies
* Specific requirements
  + External interface requirements
  + [Functional requirements](https://en.wikipedia.org/wiki/Functional_requirement)
  + [Performance requirements](https://en.wikipedia.org/wiki/Performance_engineering)
  + Logical database requirement
  + Software System attributes
    - [Reliability](https://en.wikipedia.org/wiki/Reliability_engineering)
    - [Availability](https://en.wikipedia.org/wiki/High_availability)
    - [Security](https://en.wikipedia.org/wiki/Security_engineering)
    - [Maintainability](https://en.wikipedia.org/wiki/Serviceability_(computer))
    - Portability.
* [functional requirements](https://en.wikipedia.org/wiki/Functional_requirements)
* [functional partitioning](https://en.wikipedia.org/w/index.php?title=Functional_partitioning&action=edit&redlink=1)
* [functional description](https://en.wikipedia.org/w/index.php?title=Functional_description&action=edit&redlink=1)
* Control Description
* [Environment characteristics](https://en.wikipedia.org/w/index.php?title=Environment_characteristics&action=edit&redlink=1)
* Hardware
* peripherals
* people
* others..

**Goals**

* The Software Requirements Specification (SRS) is a communication tool between stakeholders and software designers. The specific goals of the SRS are:
* Facilitating reviews
* Describing the scope of work
* Providing a reference to software designers (i.e. navigation aids, document structure)
* Providing a framework for testing primary and secondary use cases
* Including [features](https://en.wikipedia.org/wiki/Software_feature" \o "Software feature) to customer requirements
* Providing a platform for ongoing refinement (via incomplete specs or questions)