For this, certain guidelines are followed while preparing SRS. These guidelines are listed below:

• Functionality should be separate from implementation.

• Analysis model should be developed according to the desired behavior of a system. This should include data and functional response of a system to various inputs given to it.

• Cognitive model (express a system as perceived by the users) should be develop instead of developing a design or implementation model.

• The content and structure of the specification should be flexible enough to accommodate changes.

• Specification should be robust. That is, it should be tolerant towards incompleteness and complexity.

This document has many sections, which are listed below:

• Introduction: Provides an overview of the entire information described in SRS. This involves purpose and the scope of SRS, which states the functions to be performed by the system. In addition, this section describes definitions, abbreviations, and the acronyms used. The references used in SRS provide a list of documents that are referenced in the document.

• **Overall description**: Determines factors, which affects the requirements of the system. It provides a brief description of the requirements to be defined in the next section called ‘specific requirement’. It comprises of various sub-sections listed below:

⍰ **Product perspective**:

Determines whether the product is an independent product or an integral part of the larger product. It determines the interface with hardware,software system, and communication. In addition, it also defines memory constraints and operations utilized by the user.

⍰ **Product functions:** Provide a summary of the functions to be performed by the software. The functions are organised in a list so that it is easily understandable to the user.

⍰ **User characteristics**: Determine general characteristics of the users.

Constraints: Provide the general description of the constraints, such as regulatory policies, audit functions, reliability requirements, and so on.

- **Assumption and dependency**: Provides a list of assumptions and factors that affect the requirements as stated in this document.

- **Apportioning of requirements**: Determine requirements that can be delayed until release of future versions of the system.

• **Specific requirements**: Determine all requirements in detail so that the designers can design the system according to the requirements. The requirements include description of every input and output of the system and functions performed in response to the input provided. It comprises of various sub-sections listed below:

- **External interface:** Determines the interface of software with other system, which can include interface with operating system and so on. External interface also specifies the interaction of the software with users, hardware, or other software. The characteristics of each user interface of the software product are specified in SRS. For the hardware interface, SRS specify the logical characteristics of each interface among the software and hardware components. If the software is to be executed on the existing hardware, then characteristics, such as memory restrictions are also specified.

- **Functions:** Determine the functional capabilities of the system. For each functional requirement, the accepting and processing of inputs in order to generate outputs are specified. This includes validity checks on inputs, exact sequence of operations, relationship of inputs to output, and so on.

- **Performance requirements:** Determine the performance constraints of the software system. Performance requirement is of two types: static requirements and dynamic requirements. Static requirements (also known as capacity requirements) do not impose constraint on the execution characteristics of the system. These include requirements like number of terminals and user to be supported. Dynamic requirements determine the constraints on the execution of the behaviour of the system, which includes response time (the time between the start and ending of an operation under specified conditions) and throughput (total amount of work done in a given time).

- **Logical database of requirement:** Determines logical requirements to be stored in the database. This includes type of information used, frequency of usage, data entities and relationship among them, and so on.

- **Design constraint:** Determines all design constraints that are imposed by standards, hardware limitations, and so on. Standard compliance determines requirements for the system, which are in compliance with the specified standards. These standards can include accounting procedures and report format. Hardware limitations implies when software can operate on existing hardware or some pre-determined hardware. This can impose restrictions while developing software design. Hardware limitations include hardware configuration of the machine and operating system to be used.

 Software system attributes: Provide attributes, such as, reliability, availability, maintainability, and portability. It is essential to describe all these attributes to verify that these attributes are achieved in the final system.

 Organizing specific requirements: Determine the requirements so that they can be properly organised for optimal understanding. The requirements can be organized on the basis of mode of operation, user classes, objects, feature, response, and functional hierarchy.

- **Change management process:** Determines the change management process in order to identify, evaluate and update SRS to reflect changes in the project scope and requirements.

• **Document approvals:** Provide information about the approvers of the SRS document with the details, such as approver’s name, signature, date and so on.

• **Supporting information:** Provide information, such as table of contents, index and so on. This is necessary especially when SRS is prepared for large and complex projects.