© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 07 Issue: 04 | July – September 2016
Paper is available at www.jrps.in | Email : info@jrps.in

Study of Software requirements specification (SRS) and its components and Goals

Kamlesh

Introduction : A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers (in market-driven projects, these roles may be played by the marketing and development divisions) on what the software product is to do as well as what it is not expected to do. Software requirements specification permits a rigorous assessment of requirements before design can begin and reduces later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules.

An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, response time, availability, portability, maintainability, footprint, security and speed of recovery from adverse events are evaluated.

Components of SRS

Completeness of specifications is difficult to achieve and even more difficult to verify. Having guidelines about what different things an SRS should specify will help in completely specifying the requirements. The basic issues an SRS must address are:

1. Functionality
2. Performance
3. Design constraints imposed on an implementation
4. External interfaces

Functional requirements specify the expected behavior of the system—which outputs should be produced from the given inputs. They describe the relationship between the input and output of

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 07 Issue: 04 | July – September 2016
Paper is available at www.jrps.in | Email : info@jrps.in

the system. For each functional requirement, a detailed description of all the data inputs and their source, the units of measure, and the range of valid inputs must be specified.

All the operations to be performed on the input data to obtain the output should be specified. This includes specifying the validity checks on the input and output data, parameters affected by the operation, and equations or other logical operations that must be used to transform the inputs into corresponding outputs. For example, if there is a formula for computing the output, it shouldbe specified.

An important part of the specification is the system behavior in abnormal situations, like invalid input (which can occur in many ways) or error during computation. The functional requirement must clearly state what the system should do if such situations occur. Specifically, it should specify the behavior of the system for invalid inputs and invalid outputs. Furthermore, behavior for situations where the input is valid but the normal operation cannot be performed should also be specified. An example of this situation is a reservation system, where a reservation cannot be made even for a valid request if there is no availability. In short, the system behavior for all foreseen inputs and all foreseen system states should be specified.

The performance requirements part of an SRS specifies the performance constraints on the software system. All the requirements relating to the performance characteristics of the system must be clearly specified. There are two types of performance requirements: static and dynamic. Static requirements are those that do not impose constraint on the execution characteristics of the system. These include requirements like the number of terminals to be supported, the number of simultaneous users to be supported, and the number of files that the system has to process and their sizes. These are also called capacity requirements of the system.

Dynamic requirements specify constraints on the execution behavior of the system. These typically include response time and throughput constraints on the system. Response time is the expected time for the completion of an operation under specified circumstances. Throughput is the expected number of operations that can be performed in a unit time. For example, the SRS may specify the number of transactions that must be processed per unit time, or what the

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 07 Issue: 04 | July – September 2016
Paper is available at www.jrps.in | Email : info@jrps.in

response time for a particular command should be. Acceptable ranges of the different performance parameters should be specified, as well as acceptable performance for both normal and peak workload conditions.

All of these requirements should be stated in measurable terms. Requirements such as "response time should be good" or the system must be able to "process all the transactions quickly" are not desirable because they are imprecise and not verifiable. Instead, statements like "the response time of command x should be less than one second 90% of the times" or "a transaction should be processed in less than one second 98% of the times" should be used to declare performance specifications.

There are a number of factors in the client's environment that may restrict the choices of a designer leading to design constraints. Such factors include standards that must be followed, resource limits, operating environment, reliability and security requirements, and policies that may have an impact on the design of the system.

An SRS should identify and specify all such constraints. Some examples of these are:

- **Standards Compliance**: This specifies the requirements for the standards the system must follow. The standards may include the report format and accounting procedures. There may be audit requirements which may require logging of operations.
- **Hardware Limitations**: The software may have to operate on some existing or predetermined hardware, thus imposing restrictions on the design. Hardware limitations can include the type of machines to be used, operating system available on the system, languages supported, and limits on primary and secondary storage. Reliability and Fault Tolerance: Fault tolerance requirements can place a major constraint on how the system is to be designed, as they make the system more complex and expensive. Recovery requirements are often an integral part here, detailing what the system should do if some failure occurs to ensure certain properties.
- **Security:** Security requirements are becoming increasingly important. These requirements place restrictions on the use of certain commands, control access to data, provide different kinds of access requirements for different people, require the use of

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 07 Issue: 04 | July – September 2016
Paper is available at www.jrps.in | Email : info@jrps.in

passwords and cryptography techniques, and maintain a log of activities in the system. They may also require proper assessment of security threats, proper programming techniques, and use of tools to detect flaws like buffer overflow.

- **External interface specification** part, all the interactions of the software with people, hardware, and other software should be clearly specified. For the user interface, the characteristics of each user interface of the software product should be specified. User interface is becoming increasingly important and must be given proper attention. A preliminary user manual should be created with all user commands, screen formats, an explanation of how the system will appear to the user, and feedback and error messages. Like other specifications, these requirements should be precise and verifiable. So, a statement like "the system should be user friendly" should be avoided and statements like "commands should be no longer than six characters" or "command names should reflect the function they perform" used.
- **For hardware interface requirements**, the SRS should specify the logical characteristics of each interface between the software product and the hardware components. If the software is to execute on existing hardware or on predetermined hardware, all the characteristics of the hardware, including memory restrictions, should be specified. In addition, the current use and load characteristics of the hardware should be given.
- **The interface requirement** should specify the interface with other software the system will use or that will use the system. This includes the interface with the operating system and other applications. The message content and format of each interface should be specified.

Goals of SRS :

The Software Requirements Specification (SRS) is a communication tool between stakeholders and software designers. The specific goals of the SRS are:

1. Facilitating reviews
2. Describing the scope of work
3. Providing a reference to software designers (i.e. navigation aids, document structure)
4. Providing a framework for testing primary and secondary use cases

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 07 Issue: 04 | July – September 2016
Paper is available at www.jrps.in | Email : info@jrps.in

5. Linking features to customer requirements
6. Providing a platform for ongoing refinement (via incomplete

References :

1. https://en.wikipedia.org/wiki/Software_requirements_specification
2. http://rpl-blog.blogspot.in/2010/03/332-components-of-srs.html
3. http://www.techtud.com/doubt/what-are-component-good-srs-software-requirement-s
4. http://searchsoftwarequality.techtarget.com/definition/software-requirements-specification
5. http://www.cavdar.net/2011/10/10/characteristics-of-a-good-software-requirements-specification-srs/
6. http://www.slideshare.net/DanyalAkhlaq/software-requirements-specification-22497944