



**NEERAJ®**

# B.C.S.-51

## INTRODUCTION TO SOFTWARE ENGINEERING

*By: Sakshi Jain, M.C.A.*

*Question Bank Cum Chapterwise Reference Book  
Including Many Solved Question Papers*



**NEERAJ  
PUBLICATIONS**

*(Publishers of Educational Books)*  
( An ISO 9001 : 2008 Certified Company )

Sales Office:  
1507, 1st Floor, Nai Sarak, Delhi - 6  
Ph.: 011-23260329, 45704411,  
23244362, 23285501  
E-mail: [info@neerajignoubooks.com](mailto:info@neerajignoubooks.com)  
Website: [www.neerajignoubooks.com](http://www.neerajignoubooks.com)

**MRP ₹ 280/-**

**Published by:**

**NEERAJ PUBLICATIONS**

Sales Office : 1507, 1st Floor, Nai Sarak, Delhi-110 006

E-mail: [info@neerajignoubooks.com](mailto:info@neerajignoubooks.com)

Website: [www.neerajignoubooks.com](http://www.neerajignoubooks.com)

**Reprint Edition with Updation of Sample Question Paper Only**

Typesetting by: *Competent Computers*

Printed at: *Novelty Printer*

**Notes:**

1. For the best & up-to-date study & results, please prefer the recommended textbooks/study material only.
2. This book is just a Guide Book/Reference Book published by NEERAJ PUBLICATIONS based on the suggested syllabus by a particular Board/University.
3. The information and data etc. given in this Book are from the best of the data arranged by the Author, but for the complete and up-to-date information and data etc. see the Govt. of India Publications/textbooks recommended by the Board/University.
4. Publisher is not responsible for any omission or error though every care has been taken while preparing, printing, composing and proof reading of the Book. As all the Composing, Printing, Publishing and Proof Reading, etc. are done by Human only and chances of Human Error could not be denied. If any reader is not satisfied, then he is requested not to buy this book.
5. In case of any dispute whatsoever the maximum anybody can claim against NEERAJ PUBLICATIONS is just for the price of the Book.
6. If anyone finds any mistake or error in this Book, he is requested to inform the Publisher, so that the same could be rectified and he would be provided the rectified Book free of cost.
7. The number of questions in NEERAJ study materials are indicative of general scope and design of the question paper.
8. Question Paper and their answers given in this Book provide you just the approximate pattern of the actual paper and is prepared based on the memory only. However, the actual Question Paper might somewhat vary in its contents, distribution of marks and their level of difficulty.
9. Any type of ONLINE Sale/Resale of "NEERAJ IGNOU BOOKS/NEERAJ BOOKS" published by "NEERAJ PUBLICATIONS" on Websites, Web Portals, Online Shopping Sites, like Amazon, Flipkart, Ebay, Snapdeal, etc. is strictly not permitted without prior written permission from NEERAJ PUBLICATIONS. Any such online sale activity by an Individual, Company, Dealer, Bookseller, Book Trader or Distributor will be termed as ILLEGAL SALE of NEERAJ IGNOU BOOKS/NEERAJ BOOKS and will invite legal action against the offenders.
10. Subject to Delhi Jurisdiction only.

**© Reserved with the Publishers only.**

**Spl. Note:** This book or part thereof cannot be translated or reproduced in any form (except for review or criticism) without the written permission of the publishers.

**Get Books by Post (Pay Cash on Delivery)**

If you want to Buy NEERAJ IGNOU BOOKS then please order your complete requirement at our Website [www.neerajignoubooks.com](http://www.neerajignoubooks.com). where you can select your Required NEERAJ IGNOU BOOKS after seeing the Details of the Course, Name of the Book, Printed Price & the Cover-pages (Title) of NEERAJ IGNOU BOOKS.

While placing your Order at our Website [www.neerajignoubooks.com](http://www.neerajignoubooks.com) You may also avail the Various "Special Discount Schemes" being offered by our Company at our Official website [www.neerajignoubooks.com](http://www.neerajignoubooks.com).

We also have "Cash on Delivery" facility where there is No Need To Pay In Advance, the Books Shall be Sent to you Through "Cash on Delivery" service (All The Payment including the Price of the Book & the Postal Charges etc.) are to be Paid to the Delivery Person at the time when You take the Delivery of the Books & they shall Pass the Value of the Goods to us. We usually dispatch the books Nearly within 3-4 days after we receive your order and it takes Nearly 4-5 days in the postal service to reach your Destination (In total it take nearly 8-9 days).



**NEERAJ PUBLICATIONS**

(Publishers of Educational Books)

(An ISO 9001 : 2008 Certified Company)

**1507, 1st Floor, NAI SARAK, DELHI - 110006**

**Ph. 011-23260329, 45704411, 23244362, 23285501**

E-mail: [info@neerajignoubooks.com](mailto:info@neerajignoubooks.com) Website: [www.neerajignoubooks.com](http://www.neerajignoubooks.com)

# CONTENTS

## INTRODUCTION TO SOFTWARE ENGINEERING

### *Question Bank – (Previous Year Solved Question Papers)*

<i>Question Paper—June, 2019 ( Solved )</i>	1-13
<i>Question Paper—June, 2018 ( Solved )</i>	1-12
<i>Question Paper—June, 2017 ( Solved )</i>	1-11
<i>Question Paper—June, 2016 ( Solved )</i>	1-11
<i>Question Paper—June, 2015 ( Solved )</i>	1-12

<i>S.No.</i>	<i>Chapterwise Reference Book</i>	<i>Page</i>
--------------	-----------------------------------	-------------

### **DEVELOPMENT OF SRS**

1. Characteristics of SRS	1
2. Function Oriented Modelling	9
3. Object Oriented Modelling	29

### **DESIGN AND TESTING**

4. Function Oriented Design	43
5. Object Oriented Design	59
6. Testing Techniques	75
7. Development and Execution of Test Cases	101

<i>S.No.</i>	<i>Chapter</i>	<i>Page</i>
<b><u>SOFTWARE ENGINEERING CONCEPTS</u></b>		
8.	Software Development Models	111
9.	Software Project Management Concepts	130
10.	Software Engineering Fundamentals	145
		■ ■

**Sample Preview  
of the  
Solved  
Sample Question  
Papers**

*Published by:*



**NEERAJ  
PUBLICATIONS**

[www.neerajbooks.com](http://www.neerajbooks.com)

# QUESTION PAPER

(June - 2019)

(Solved)

## INTRODUCTION TO SOFTWARE ENGINEERING

Time : 3 Hours ]

[ Maximum Marks : 100

Note: Question no. 1 is compulsory. Attempt any three questions from the rest.

**Q. 1 . (a) What is SRS? Explain the benefits of SRS. Develop SRS for Payroll Management System. Follow IEEE SRS format.**

**Ans.** Software Requirement Specification (SRS) is a document that completely describes what the proposed software should do without describing how software will do it. The basic goal of the requirement phase is to produce the SRS, Which describes the complete behaviour of the proposed software. SRS is also helping the clients to understand their own needs.

### Advantages

Software SRS establishes the basic for agreement between the client and the supplier on what the software product will do.

- A SRS provides a reference for validation of the final product.
- A high-quality SRS is a prerequisite to high-quality software.
- A high-quality SRS reduces the development cost.

### Introduction

Payroll system is the heart of any Human Resource System of an organization. The solution has to take care of the calculation of salary as per rules of the company, income tax calculation and various deductions to be done from the salary including statutory deductions like Income tax and provident fund deductions. It has to generate pay-slip, cheque summary and MIS reports.

It is understood that we are tired of managing thousand of odd papers, pay slip, payroll reports, and salary details and so on. Imagine that we have a payroll processing system which will generate our pay slips and payroll reports within seconds. We can help others automated your payroll system

by developing a customized payroll application that suits your specific requirements.

### Purpose

Main aim of developing Employee Payroll Management is to provide an easy way not only to automate all functionalities involved managing leaves and Payroll for the employees of company, but also to provide full functional reports to management of company with the details about usage of leave facility.

We are committed to bring the best way of management in the various forms of EPM. We understand that EPM in not a product to be sold, it is a tool to manage the inner operation of company related to employee leave and Payroll.

### Scope

This Application works in Multiple PC's installed on multiple computers but sharing same database by which users of different department can use it sitting at different locations simultaneously.

But in future we can make the application where the database will be hosted in order to manage the all departments which will be located in different places and by keeping domain of Application as Online.

### Benefits

- To improve the efficiency.
- Quickly find out information of an employee details.
- To provide easy and faster access information.
- To provide user friendly environment.

### Definitions, Acronyms, Abbreviations

**Visual Basic .NET (VB.NET)** is an object-oriented computer programming language that can be viewed as an evolution of the classic Visual Basic (VB) which is implemented on the .NET Framework.

**Microsoft SQL Server** is a relational model database server produced by Microsoft.

**SRS** - Software Requirements Specification.

**EPM** - Employee Payroll Management.

#### References

- SRS format from IEEE website ([www.ieee.org/format](http://www.ieee.org/format)).
- Software Engineering by "Roger Pressman".

#### Project Overview

- The following subsections provide the complete overview of the software specifications requirements documentation for the product Employee Payroll Management. The entire SRS is documented in view of User and the following sub sections are arranged to give a complete outlook of the software, its perspective, features, system requirements and users know how it is.

#### Overall Description

##### Product Perspective

- This software is developed specifically to cater the company employees leave management, is totally self contained and works efficiently. It provides simple database rather than complex ones for high requirements and it provides good and easy graphical user interface to both new as well as experienced user of the computer.

#### Product Functions

##### Master Module

**Designation:** Contains the position or status of employee in departments.

**Department:** Contains the information about different departments in any company.

##### Employee Module

- **Employee Details:** This module contains the whole detail of employees of any system.

##### Attendance Module

**Leave:** This module is for keeping the records of leave taken by any employee.

**Attendance:** This module is for keeping the records of employee's presence.

##### Salary Module

**Allowance:** This module is for calculating the allowance given to employee by the institution.

**Deduction :** This module calculates the amount from number of days taken as leave and deduct these amount from salary.

**Pay Slip :** This module is for generating the final pay slip.

#### User Characteristics

##### End Users

- No specific knowledge or skills are required from the end user.
- End user should have basic idea about computer operations and database.

#### Administrator

- Administrator must be having good knowledge of database management system.
- Administrator to manage user rights.
- If the network connection does not work properly than our system should not work as intended.
- Also that is assumed that the product is installed properly at web server.
- This system will not take care of any virus problem, which might occur either on the client or the server system. Avoiding the use of pirated software and ensuring that floppies and other removable media are scanned for viruses before use could minimize the possibility of viral infection.
- Recovery of data after a system crash will be possible only if backups are taken at regular intervals.
- Manual interfaces cannot be fully avoided. Documented proofs like data entry of employees, etc. will have to be verified by the concerned management staff before entering it into the computerized system.

#### Assumptions and Dependencies

##### Assumptions

- The code should be free with compilation errors/syntax errors.
- The product must have an interface which is simple enough to understand.

##### Dependencies

- All necessary hardware and software are available for implementing and use of the tool.
- The proposed system would be designed, developed and implemented based on the

# Sample Preview of The Chapter

*Published by:*



**NEERAJ  
PUBLICATIONS**

[www.neerajbooks.com](http://www.neerajbooks.com)

---

# INTRODUCTION TO SOFTWARE ENGINEERING

---

DEVELOPMENT OF SRS



---

## Characteristics of SRS

---

### **INTRODUCTION**

Software Requirement Specification (SRS) is a requirement specification for the software that forms the basis for the development of a software product, program or set of programs that perform the desired functions. Depending on the writer of the SRS, various purposes could be served. It could be used by the following:

- Customer and users for understanding what is expected from them.
- Developer for writing the desired software product that is to be built.
- Testers for performing testing activities.
- Managers for estimating and planning the project to be delivered.
- Maintenance team for maintain the software in future.

SRS minimizes the time consumed by developers in developing as it becomes an ease in understanding the software project which in turn leads to less development cost. A good SRS will define all the software requirements correctly. No design and implementation details should be present in a SRS.

SRS should only define 'WHAT' a system does and not 'HOW' it is done.

### **CHAPTER AT A GLANCE**

#### **CHARACTERISTICS OF SRS**

The SRS should be correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable and traceable.

In this chapter, we will discuss the following characteristics:

- Completeness
- Unambiguity
- Inconsistency

#### **COMPLETENESS**

The SRS is complete if, and only if, it has the following properties:

1. Description of all significant requirements relating to functionality, performance, design, attributes or external interfaces.
2. Definition of the response of the software system for all the situations.
3. All the diagrams, tables and figures should be fully labelled and referenced.
4. All the abbreviations should be defined properly.

This is most difficult to achieve as it is difficult to specify all the system requirements at the beginning. Missing requirements are difficult to trap since they are not visible.

#### **UNAMBIGUITY**

The SRS is said to be unambiguous if every stated requirement has only one interpretation independent of the person or the time when the interpretation is done. The SRS needs to be unambiguous for all the users who will create and use the document that is to the authors, the users, the reviewers as well as developers and testers. A sentence in the SRS is said to be ambiguous if given to few people and asked for the interpretation and there is more than one interpretation. In cases, if a term that is

2 / NEERAJ : INTRODUCTION TO SOFTWARE ENGINEERING

to be used for particular context has multiple meanings, the term should be included in the glossary where all the meanings should be specified.

Some of the issues related to ambiguity are as follows:

1. Requirements are often written in natural language. Natural language being imprecise and ambiguous in nature, so any SRS written in natural language should be reviewed by an independent party to identify the ambiguity in SRS and correct the language.
2. Ambiguity can be avoided by writing the SRS in a particular requirement specification language, which has a processor that automatically detects any lexical, syntactic and semantic errors. However, there is a disadvantage associated with it. It takes a lot of time to learn such language and sometimes it is not even understood by the users or customers.
3. The requirements could also be depicted using diagrams, tables or flow charts. These techniques improve the precision of the requirements, but it is a bit difficult to convert the requirements.

**INCONSISTENCY**

The SRS should not be inconsistent. The SRS is consistent if, and only if, no requirements described in it conflict. Any conflict between the described requirements should be identified and resolved. There are three types of conflicts that are likely to be seen between the requirements. They are as follows:

1. Characteristics of real world objects interacting with the system may conflict. For example:
  - (a) Format of an output report can be described in tabular form in one requirement whereas in other it can be stated in textual form.
  - (b) One requirement may state that all lights are red whereas other states that the colour of all lights is blue.
  - (c) In a time sheet system one requirement states that a worker cannot work more than 16 hours a day whereas in other requirement it may assume that a worker can work for 24 hours a day.
2. Two specific items can have logical or temporal conflict between them. For example:

- (a) One requirement states that two numbers should be added while the other says that they should be multiplied.
- (b) One requirement says that “A” should always follow “B” whereas the other says that “A and B” occur simultaneously.
- (c) In time sheet one requirement says that the work must be done by the end of the week (Sunday) whereas in other, it states that the work should be done by the last working day of the week (Friday).

3. The requirements describing same real world entity may have different terminologies. For example, a program’s request for a user input may be called a “prompt” in one requirement or “cue” in another. In one requirement start of week may be used as Monday whereas in another it may be used as Sunday.

The use of standard terminology and definitions promotes consistency.

**IEEE SRS**

The Institute of Electrical and Electronics Engineers (IEEE) has published the guidelines and standards to organize an SRS document. The SRS should be selected or defined at the beginning of all the activities. Different projects require differently organized requirements thus, the SRS should be selected accordingly.

The general organization of an SRS is as follows:

1. Introduction
  - 1.1 Purpose
  - 1.2 Scope
  - 1.3 Definitions, Acronyms and Abbreviations
  - 1.4 References
  - 1.5 Overview
2. The Overall Description
  - 2.1 Product Perspective
    - 2.1.1 System Interfaces
    - 2.1.2 Interfaces
    - 2.1.3 Hardware Interfaces
    - 2.1.4 Software Interfaces
    - 2.1.5 Communication Interfaces
    - 2.1.6 Memory Constraints
    - 2.1.7 Operations
    - 2.1.8 Site Adaption Requirements
  - 2.2 Product Functions
  - 2.3 User Characteristics
  - 2.4 Constraints

- 2.5 Assumptions and Dependencies
- 2.6 Apportioning of Requirements
- 3. Specific Requirements
  - 3.1 External Interfaces
  - 3.2 Functions
  - 3.3 Performance Requirements
  - 3.4 Logical Database Requirements
  - 3.5 Design Constraints
    - 3.5.1 Standards Compliance
  - 3.6 Software System Attributes
    - 3.6.1 Reliability
    - 3.6.2 Availability
    - 3.6.3 Security
    - 3.6.4 Maintainability
    - 3.6.5 Portability
  - 3.7 Organizing the Specific Requirements
    - 3.7.1 System Mode
    - 3.7.2 User Class
    - 3.7.3 Objects
    - 3.7.4 Feature
    - 3.7.5 Stimulus
    - 3.7.6 Response
    - 3.7.7 Functional Hierarchy
- 4. Change Management Process
- 5. Document Approvals
- 6. Supporting Information

Now we will study the explanation of the above SRS format.

**Introduction:** The sub-sections of this topic will provide an overview of the entire SRS.

**Purpose:** In this sub-section identify the purpose of the SRS and specify its intended audience.

**Scope:** In this sub-section we will:

- (i) Identify the names of the software product to be produced.
- (ii) Tell about what and what not a software product will do.
- (iii) Application of the software being specified is being described along with the relevant benefits, objectives and goals.
- (iv) Be consistent with similar statements in higher-level specifications, if they exist.

**Definitions, Acronyms and Abbreviations:** This section provides the definitions of all the terms, acronyms and abbreviations that are required to understand the SRS. This information may be provided by referring the document or an Appendix.

**References:** This sub-section will:

- (i) Provide a complete list of all documents that are referenced in the SRS.
- (ii) Identify the document by title, report number, date and publishing organization.
- (iii) Sources from which the references are obtained should be specified.

**Overview:** This sub-section will include:

- (i) Description of the rest of the SRS.
- (ii) Explanation of how SRS is organized.

**The Overall Description:** The factors that affect the product and its requirements are being described here. This section does not state specific requirements instead it makes easier to understand the requirements that are stated later as it provides background for those requirements.

**Product Perspective:** Product is being put into perspective with other related products. Independent and self-contained product should be stated here. If the product defined in the SRS is a component of a larger system, then this sub-section relates the requirements of the larger system to functionality of the software interfaces between the system and the software is being identified.

**System Interfaces:** All system interfaces are listed and the functionality of the software to accomplish the system requirement and the interface description to match the system is being identified.

**Interfaces:** In this sub-section, we will specify:

- (i) The logical characteristics of each interface between the software product and its users.
- (ii) All the aspects of optimizing the interface with the person who must use the system.

**Hardware Interfaces:** The logical characteristics of each interface between the product and the hardware component of the system are specified. Configuration characteristic is also included. It also tells about what and how devices are to be supported and protocols.

**Software Interfaces:** The required software products and interfaces with other application system are specified. The required software product includes the following:

- (i) Name
- (ii) Mnemonic
- (iii) Specification number
- (iv) Version number
- (v) Source

Each interface provides:

4 / NEERAJ : INTRODUCTION TO SOFTWARE ENGINEERING

- (i) The software relating to the software product is being discussed.
- (ii) The interface in terms of message content and format is defined.

**Communications Interfaces:** This sub-section specifies the various interfaces to communications.

**Memory Constraints:** The applicable characteristics and limits on primary and secondary memory are specified.

**Operations:** In this sub-section, we will specify the normal and special operations that are required by the user. Some of them are:

- (i) Modes of operation in user organization.
- (ii) Periods of interactive and unattended operations.
- (iii) Data processing support functions.
- (iv) Backup and recovery operations.

**Site Adaptation Requirements:** In this sub-section:

- (i) The requirements specific to a given site, mission, or operational mode for any data or initialization sequences is defined.
- (ii) The site that should be modified to adapt the software for particular installation is specified.

**Product Functions:** In this section, we will provide the major functionality that the software will perform. Sometimes the function summary is taken from the section of higher level specification that allocates the functions to a software product.

For clarity:

- (i) The functions should be organized in such a way that they are understandable and readable by the customer or anyone who is reading the information for the first time.
- (ii) Different functions and their relationships could be shown using textual or graphical methods.

**User Characteristics:** It describes the characteristics of the intended users of the product that is the educational level, experience and technical expertise. It also tells about why a specific requirement is discussed later in the SRS.

**Constraints:** Provide a general description of those items that could limit the options for developers. They can include regulatory policies, hardware limitations (for example, signal timing requirements), interface to other applications, parallel operation, audit functions, control functions, higher-order language requirements, signal

handshake protocols ( for example, XON-XOFF, ACK-NACK), reliability requirements, criticality of the application and safety and security considerations.

**Assumptions and Dependencies:** All the requirements that are affected in the SRS are listed. These factors are the changes to the design constraints that can affect the requirements in the SRS. For example, an assumption might be that a specific operating system would be available on the hardware designated for the software product but instead it is not present, so the SRS has to change accordingly.

**Apportioning of Requirements:** The requirements that may be delayed until the future versions of the system are identified.

**Specific Requirements:** This section contains all the detailed information of the software requirements required by the designers to design the system, for tester to test the system for those software requirements. Throughout the section the requirements stated are perceived by the user, operators or both. These should include at least a small description of every input, output and functions performed by the system.

The following principles apply:

- (i) Requirements should be stated in accordance with the characteristics of a good SRS, that is it should be, correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable and traceable.
- (ii) Requirements should be cross-referenced to earlier requirements that relate.
- (iii) For maximum readability of the SRS the requirements should be organized carefully.

**External Interfaces:** This sub-section contains all the information of all inputs and outputs of the system. It contains both content and format as follows:

- Name of item
- Description of purpose
- Source of input or destination of output
- Valid range, accuracy and/or tolerance
- Units of measure
- Timing
- Relationships to other inputs/outputs
- Screen formats/organization
- Window formats/organization
- Data formats
- Command formats
- End messages.