IT314: Software Engineering

Req-View requirement tool

https://www.reqview.com/?utm_source=app&utm_medium=demo_project&utm_campaign=scope_links

ISO/IEC/IEEE 29148 Requirements Specification Templates

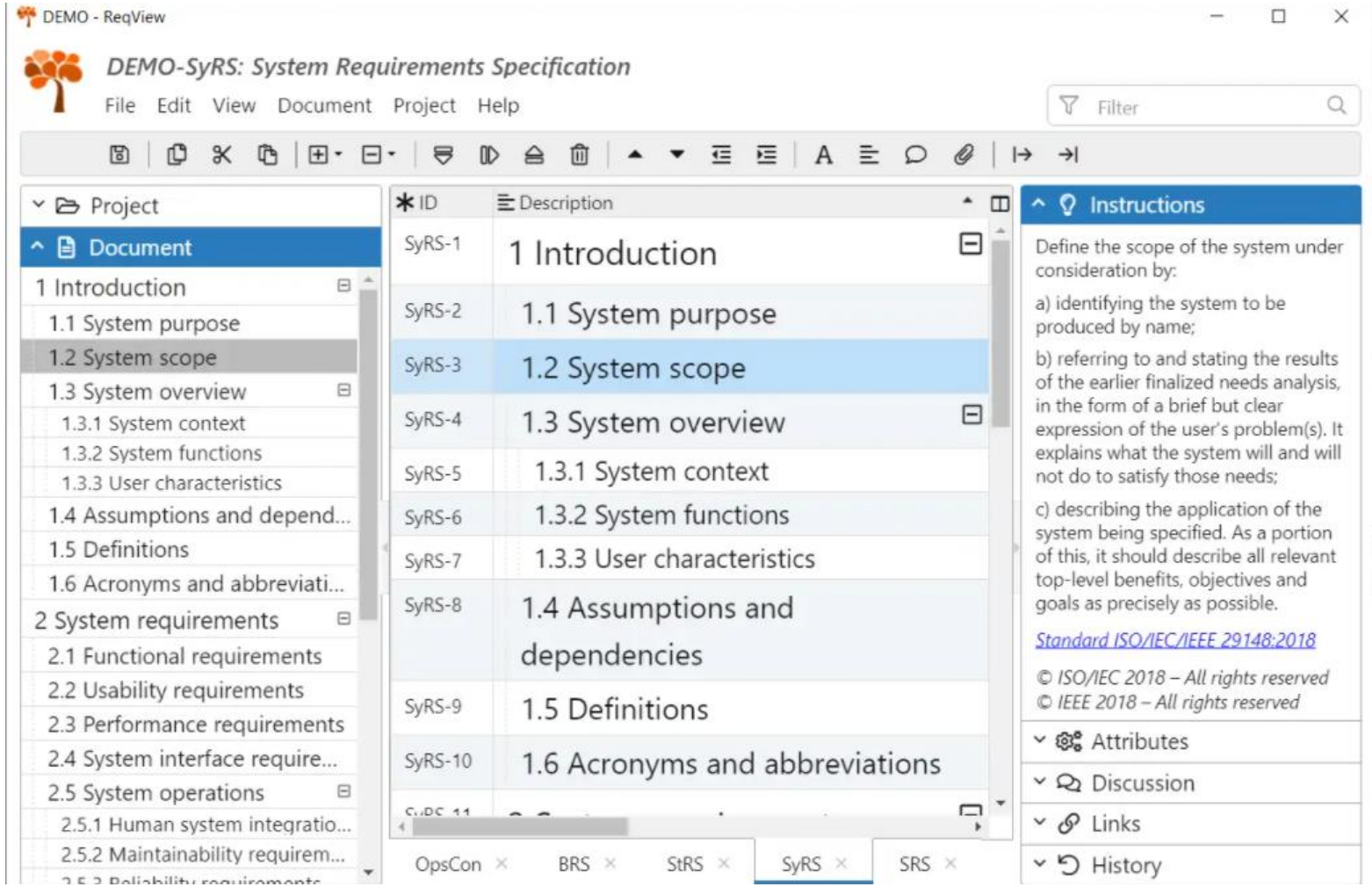
Table of Contents

Requirements specifications

- **Business Requirements Specification (BRS)** describing business or mission requirements,
- **System Operational Concept (OpsCon)** describing stakeholder needs,
- **Stakeholder Requirements Specification (StRS)** describing stakeholder requirements,
- **System Requirements Specification (SyRS)** describing system requirements,
- **Software Requirements Specification (SRS)** describing software requirements.

Template instructions

- The standard provides detailed information about all the requirements engineering process for software and system products



Attributes

- Attributes identify the content in your software.

Attributes

Name	Identifier	Type	Description
Id	*id*	`string`	Unique identifier within the document
Heading	*heading*	`string`	Short name of the document section or the requirement
Text	*text*	`xhtml`	Intent of the requirement
Owner	*owner*	`string`	The person or element of the organization that maintains the requirement
Priority	*priority*	`enum`	Requirements importance for the stakeholder relative to the whole project
Source	*source*	`string`	Source of the requirements, e.g., name of the person who raised the requirement or reference to a related standard
Rationale	*rationale*	`xhtml`	Explanation of the reason why the requirement is needed
Difficulty	*difficulty*	`enum`	Assumed difficulty of the requirement
Type	*type*	`enum`	Type of the document object — Section, Information, Functional Requirement, Non-functional Requirement
Status	*status*	`enum`	Requirements status supporting your workflow

Type section describes the attributes

ID	Type	Description
SRS-1	Section	1 Introduction Project name : College management system Group Id : 1 Student name : jash , nimmii
SRS-2	Information	1.1 Purpose The purpose of CMS is to provide necessary information on a single application.
SRS-3	Information	1.2 Scope Students can access the information through this application from anywhere.

Example : CMS(college management system)

ID	Type	Description	Source	Rationale
SRS-1	Section	1 Introduction Project name : College management system Group Id : 1 Student name : jash , nimmii		
SRS-2	Information	1.1 Purpose The purpose of CMS is to provide necessary information on a single application.		
SRS-3	Information	1.2 Scope Students can access the information through this application from anywhere.		
SRS-4	Section	1.3 Product perspective The system allows the student and teacher to create their accounts in the system and provides necessary features		
SRS-5	Section	1.3.1 System interfaces		
SRS-6	Section	1.3.2 User interfaces - Application homepage. - Website login page.		

References

- It includes the type of template is used in your software. i.e.
- ISO/IEC/IEEE 29148:2018 Standard
- Guide to the Systems Engineering Body of Knowledge (SEBoK)

How to write SRS document ?

1. Define your product’s purpose
2. Describe what you’re building
3. Detail the requirements

How system description helps to build SRS documents ?

- An SRS gives you a complete picture of your entire project.
- It provides a single source of truth that every team involved in development will follow.
- It is your plan of action and keeps all your teams , from development to maintenance on the same page
- This layout not only keeps your teams in sync but it also ensures that each requirement is hit.
- It allows for better understanding or your product, team, and the time it will take to complete.

System description

- In this system description, we present a **college management system (CMS)** where students and faculty from home or at any place can operate our application. This system will bring faculty and student together in one place. From a single application, they can access educational material from anywhere. This system is used to remove handwritten documents and make use of digital documentation.
- This application consists of two user : faculty and students.
- When it comes to the faculty side, they can operate the application in various ways where they can take attendance, upload documents, upload videos of lectures, report students based on the performance, can also view the feedback of students base on subject ranking, lecture taking, doubt solving skills of professor, extra labs, assignment views etc.
- Another part of this application is used by students who can operate this application from any place. They can manage to view attendance, download material, provide feedback to professor-related subjects, assignments, lectures, labs. Viewing of results for mid-sem end-sem, the viva is manageable from their end. Viewing of timetable, labs.

SRS : Table of content

Table of content

1. Introduction

1.1 Purpose

- The main purpose of this document is to provide a working example of a Software Requirements Specification (SRS) based on ISO/IEC/IEEE 29148:2018 standard.
- The purpose of this CMS application is to provide necessary information related to any educational activity on a single mobile application.
- This platform brings the transparency between their users.
- Communication between Users and Faculty can be done in much easier way.

1.2 Scope

- Student and faculty will be the end user of this software.
- The scope of this project will be up-to students and faculty.
- Students can access the important information from any place as well as faculty can provide documents.

1.3 Intended Audience

- Student
- Faculty

1.4 Product perspective

CMS database system stores the following information.

- Student Description :
- Faculty Description :
- i.e. name, Id, gender, E-mail, phone_no, address etc.

1.5 User class and characteristics

- Users of the system must be able to retrieve the necessary information
- i.e., Student and faculty must have access to their functionality.

Student can be able to

- Login
- View attendance
- Download material
- Give the feedback

1.6 Operating environment

- Distributed database
- client/server system
- Operating system: Windows.
- database: SQL + database
- platform: vb.net/Java/PHP

1.7 Memory Constraints

- In student profile, Image size should be less than 500 kB.

1.8 Limitations

- To use the application user must be enrolled in the college courses.

1.9 Assumptions and dependencies

- A1 - Students is enrolled in courses should have valid credentials to login.
- D1 - For a valid login student must be registered with valid email and phone number.

2. Requirements

2.1 System Interface

- It consists of User, Hardware, Communication, Software interfaces.

User Interface:-

- It opens the application homepage and established the login connection for further use.

Hardware Interface:-

- It consists of Ethernet, ATA/IDE,SCSI

System Interface

Software Interface:-

- CLI/ GPU

Communication Interface:-

- it will based on wireless connection.

2.2 Functional requirements :-

User : Student

- It allows student to login.
- It allows students to View attendance
- It allows students to Download material
- It allows students to Provide feedbacks
- It allows students to View results, timetable

User : Faculty

- It allows faculty to login.
- It allows faculty to take attendance
- It allows faculty to upload material
- It allows students to check feedbacks.

2.3 Performance requirements :-

- After the successful login, students profile should be visible with name and id.

2.4 logical database:-

- Applications stores the user data.
- Fetch the response according to request.

2.5 SAFETY REQUIREMENTS

- If there is extensive damage to a wide portion of the database due to catastrophic failure
- Such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

2.6 SECURITY REQUIREMENTS

- Security systems need database storage just like many other applications.
- Student and faculty’s personal data must be kept secret.
- However, the special requirements of the security market mean that user must choose their database carefully.

3. Verification :

- **A SRS is verifiable** : if there exists a specific technique to quantifiably measure the extent to which every requirement is met by the system.

4. Supporting information

- **Supporting information contains** : Sample input/output formats, descriptions of cost analysis studies, or results of user surveys

5. References

- **References** : References is about listing any other documents or Web addresses to which this SRS refers

Req-View tool Demo

Step 1 : License activation

- Download the

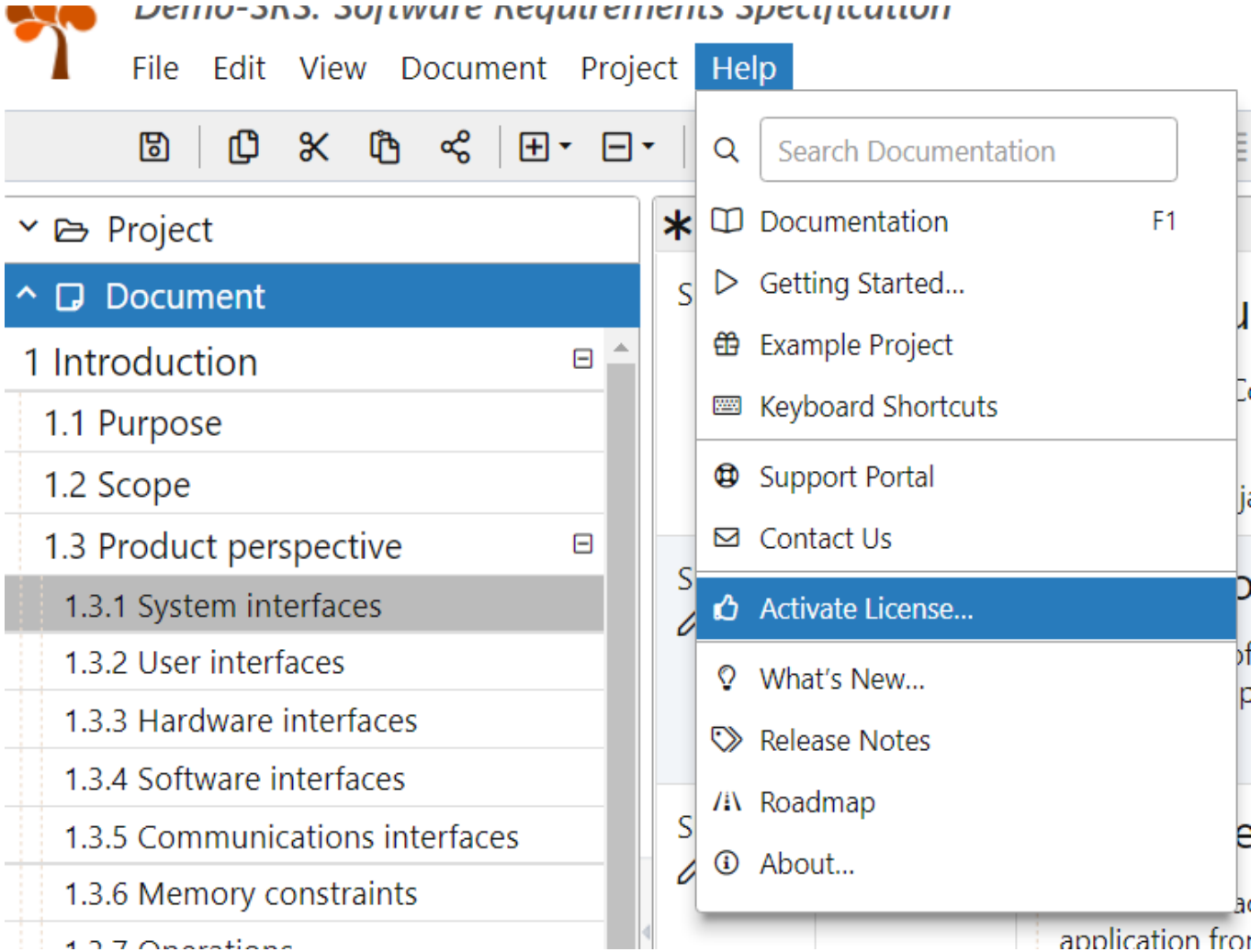
File from the

Req-View response .

- From the Help bar and text file

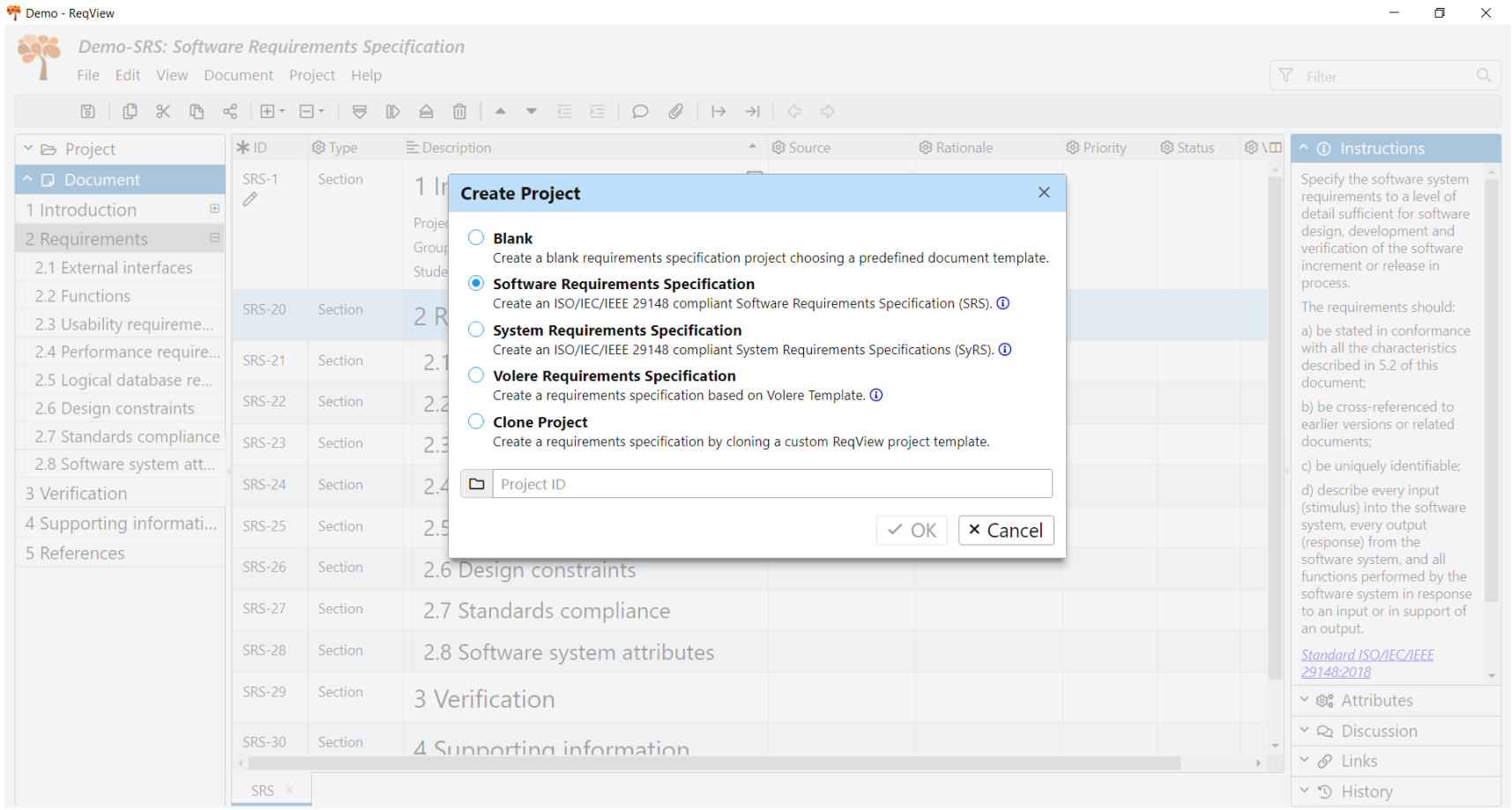
Of Email, activate the

License



ReqView

Step 2 : Create a new project

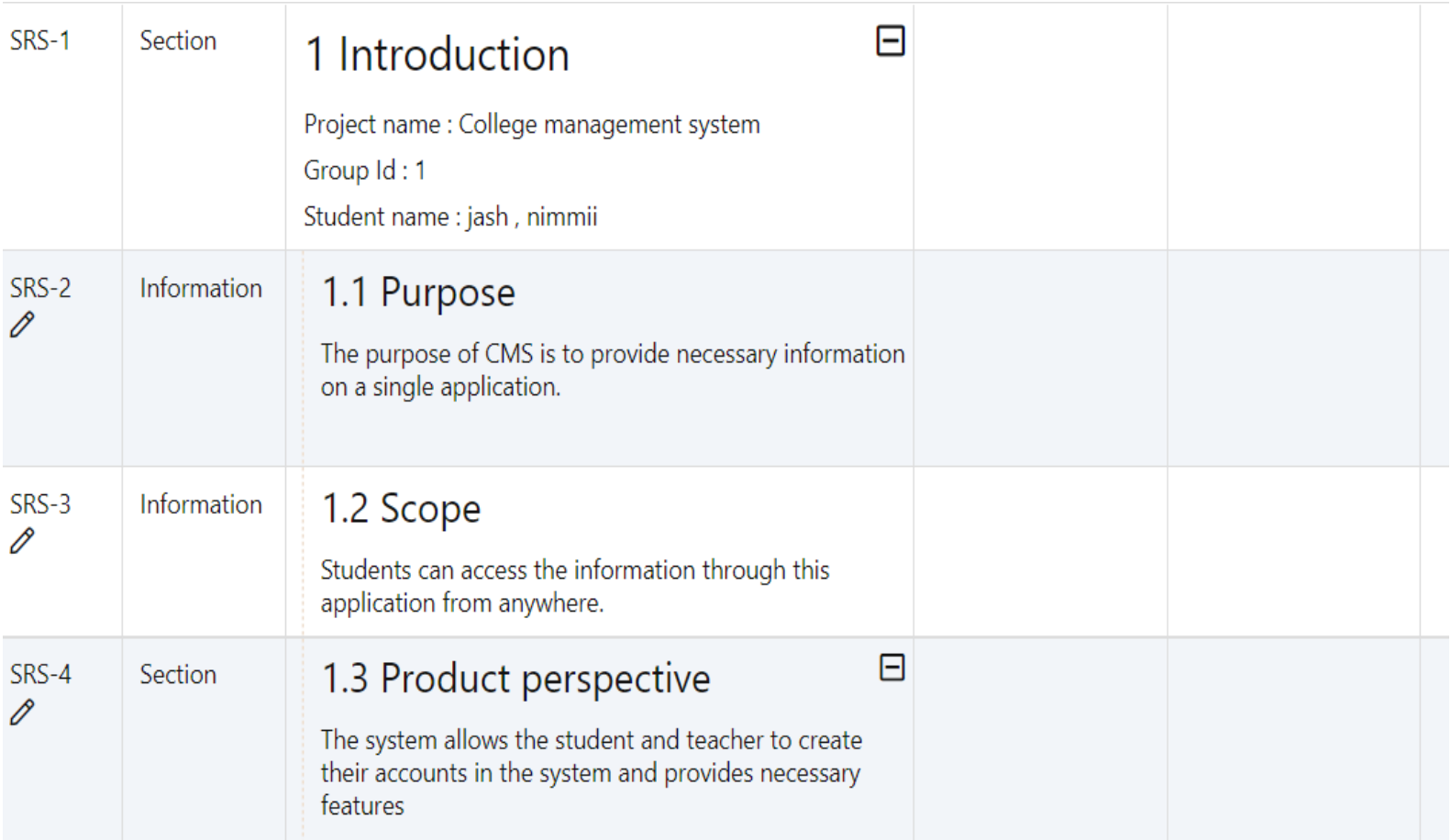


In ReqView tool model

Step 3 :

Define the terms ,
With respect to
the software

- Introduction
- Purpose and scope
- Product perspective



Step 4 : Define the Interfaces

- System interface : Interaction between the system
- it includes a reference (pointer) to the specific location in the definition document that defines the interface.

- User interface : The User Requirements Specification describes the business needs for what users require from the system.

- Software and hardware interface : It includes the environment area of your development tools and their specifications.

Memory constraints

- Every student can store the memory data
- (i.e.) Image size should be less than 500 kb

Communication interface

- Includes wireless / wired technology

Operations :

- It includes the routine operation or functionality in the system.

SRS-5	Section	1.3.1 System interfaces
SRS-6	Section	1.3.2 User interfaces - Application homepage. - Website login page.
SRS-7	Section	1.3.3 Hardware interfaces Ethernet, ATA/IDE, SCSI
SRS-8	Section	1.3.4 Software interfaces Cli , Gui ,
SRS-9	Section	1.3.5 Communications interfaces Wireless technology
SRS-10	Section	1.3.6 Memory constraints Image size should be less than 500 kb
SRS-11	Section	1.3.7 Operations Input - valid Credential output - successful response / error message

Step 5 : Define the limitations and assumptions

Limitations :

- It includes the some functionality not available in your software

Assumptions :

- List the assumptions that you have consider in your software

Dependencies:

- List all functions that are dependent on each other

Abbreviation :

- An abbreviation is a shortened form of a word used in place of the full word

SRS-16	Interface Requirem...	1.6 Limitations To use the application user must be enrolled in the college courses		
SRS-35	Interface Requirem...	Must have valid credentials to login in application.		
SRS-17	Non-func. Requirem...	1.7 Assumptions and dependencies A1 - Student is enrolled in courses had have valid credentials to login D1- For a valid login student must be registered with valid email and phone number		
SRS-18	Section	1.8 Definitions		
SRS-19	Section	1.9 Acronyms and abbreviations CMS - College management system		

Step 6 : Define the Requirements

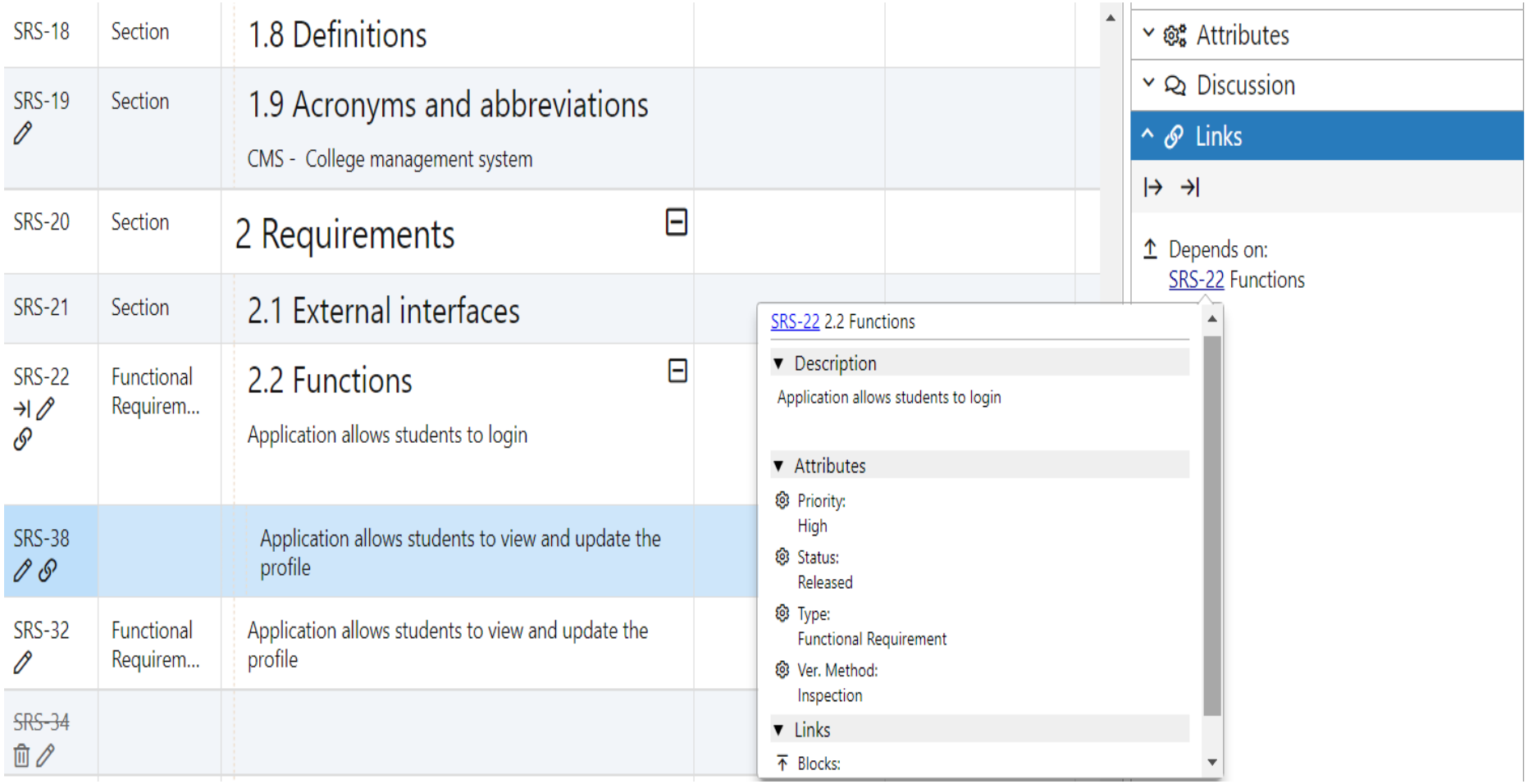
- External requirements. Requirements which arise from factors which are external to the system and its development process
- Functional requirements :It includes most of the functions of software
- Non Functional requirements: It includes requirements related to

Performance, safety , Security and Quality

- Performance requirements : Requirements related to your system's performance

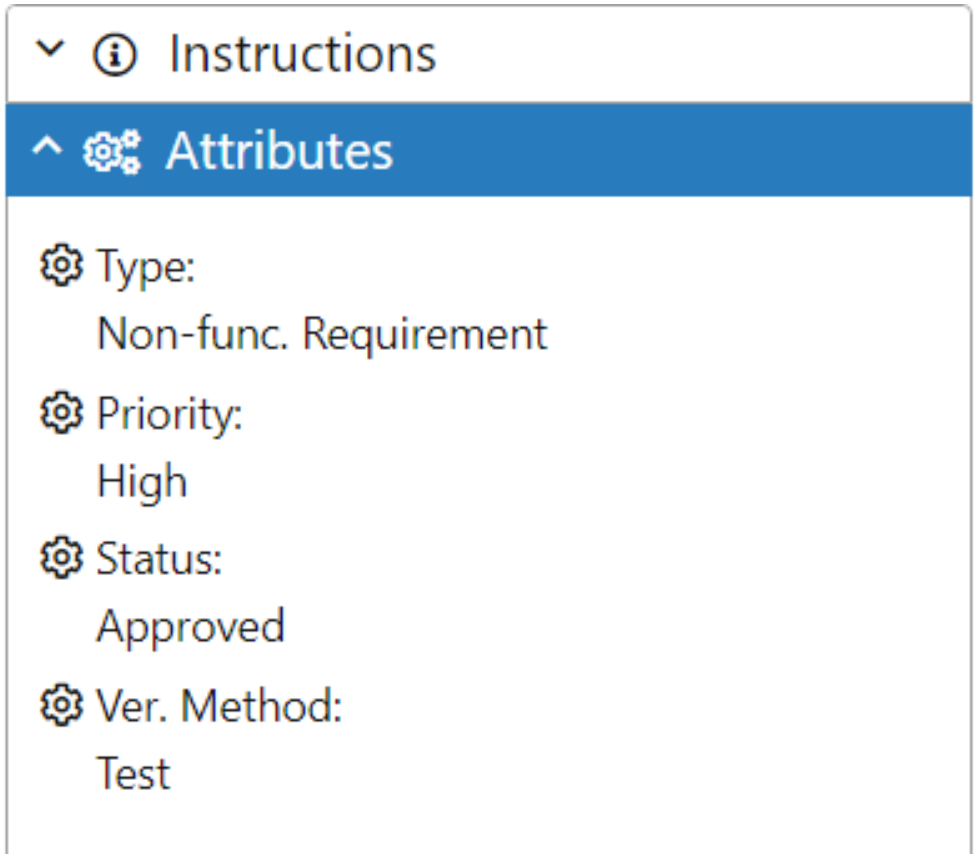
SRS-20	Section	2 Requirements		
SRS-21	Section	2.1 External interfaces		
SRS-22	Functional Requirem...	2.2 Functions Application allows students to login		
SRS-32	Functional Requirem...	Application allows students to view and update the profile		
~~SRS-34~~				
SRS-33	Functional Requirem...	Application allows Students to view their attendance		
SRS-23	Section	2.3 Usability requirements		
SRS-24	Non-func. Requirem...	2.4 Performance requirements After successful login , Student profile should be visible with name and details.		
SRS-25	Non-func. Requirem...	2.5 Logical database requirements Application must store the users data		

Step 7 : Define Link / Dependency

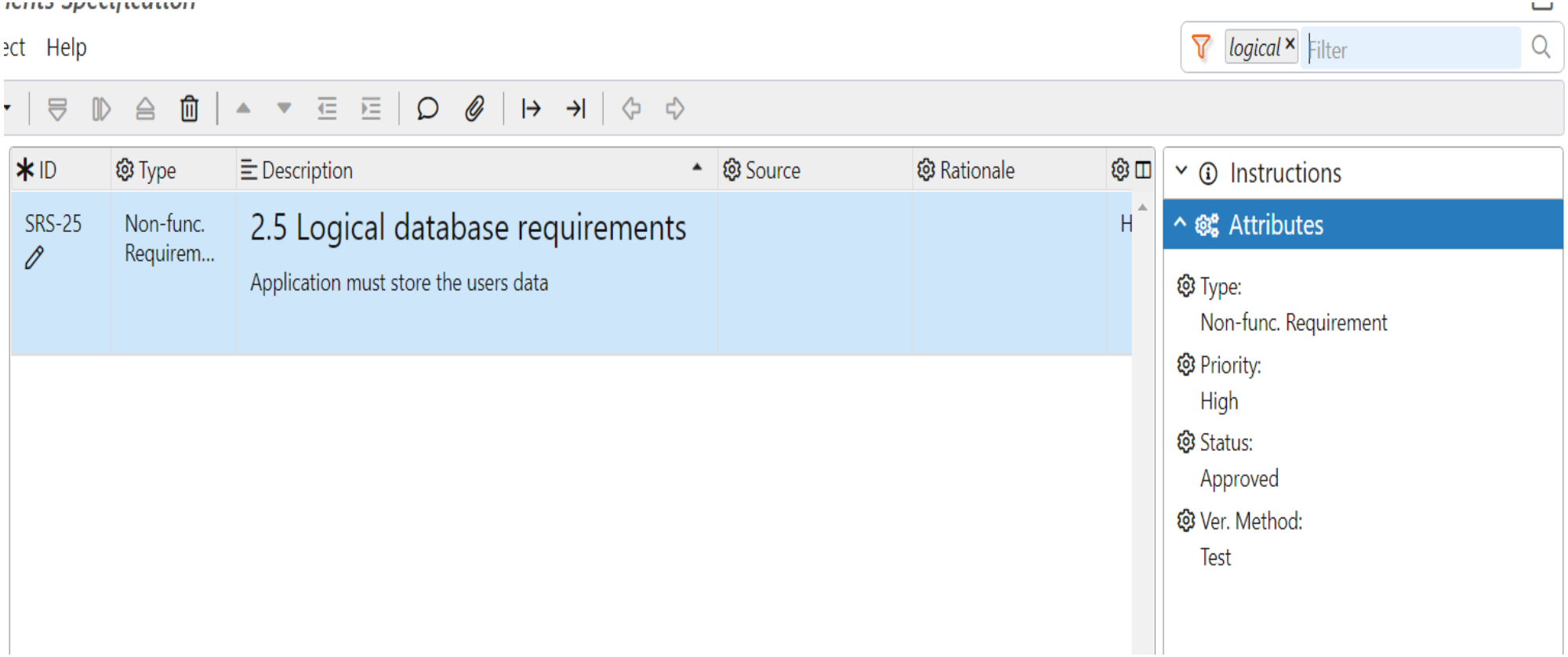


- With the help of attributes you can

Define the current state of art.



With the help of filter you can find the relevant information



Step 8 : Define the Design constraints :

- It includes the budget, amount of materials, type of materials, and time allowed to complete the project.

2.5 Logical database requirements Application must store the users data			High	Approved
2.6 Design constraints Login functionality should be ready in 1 week			Medium	Ready
Attendance must be filled in the database on a regular basis			High	Approved

Step 9 :

Define the relevant informations :

SRS-29	Section	3 Verification
SRS-30	Section	4 Supporting information
SRS-31	Section	5 References

- A SRS is verifiable : if there exists a specific technique to quantifiably measure the extent to which every requirement is met by the system.

- Supporting information contains : Sample input/output formats, descriptions of cost analysis studies, or results of user surveys

- Refrences : References is about listing any other documents or Web addresses to which this SRS refers

Comparison between different requirement tools

ReqView	Jama Connect	Matrix Requirements	codebeamer
Change Management **Collaboration Tools** **Prioritization** **Reporting/Analytics** **Stakeholder Defined Attributes** **Status Tracking** **Task Management**	**Change Management** **Collaboration Tools** **Prioritization** **Reporting/Analytics** **Stakeholder Defined Attributes** **Status Tracking** **Task Management**	Collaboration Tools Prioritization Reporting/Analytics Stakeholder Defined Attributes Status Tracking Task Management	Change Management Collaboration Tools Reporting/Analytics Stakeholder Defined Attributes Task Management
Pros:- Support via email and portal is very efficient , aslo effective in solving our issues. Cons:- Realtime Collaboration ist not yet implemented. Working two or more people on same documents sometimes is difficult	Pros:- Not a heavy application , easy to run Cons:- W.R.T jira , some basic configuration are hard to set up.	Pros:- Improved communication, resource sharing, employee development Cons:- heavy workloads, Additional expenses	Pros:- *Very flexible and configurable. Stable with new featuers comming fast.* Cons:- *Features leave the product and information about that is missing.*

ReqView	Aha!	Helix RM	Process Street
Change Management Collaboration Tools Prioritization Reporting/Analytics Stakeholder Defined Attributes Status Tracking Task Management	Change Management Collaboration Tools Prioritization Reporting/Analytics Stakeholder Defined Attributes Status Tracking Task Management	Change Management Collaboration Tools Prioritization Reporting/Analytics Stakeholder Defined Attributes Status Tracking Task Management	Change Management Collaboration Tools Prioritization Reporting/Analytics Stakeholder Defined Attributes Status Tracking Task Management
Pros: - Pros:- Support via email and portal is very efficient , aslo effective in solving our issues. Cons:- Realtime Collaboration ist no t yet implemented. Working two or more people on same documents sometimes is difficult	Pros: - *good response on a couple of key feature* *Cons: - it need to completely refresh the page before I see my changes in the UI so it makes things a bit slower to use.*	Pros:- Workflow customization, integration with source provider. Dashboards looks nice feature and allows to plan future releases. Cons:- Lack of HTML support in emails sent by Helix ALM. It still doesn't support inline images RP	Pros:- *tasks can have videos, links, text, whatever I need to customize it.* *Cons: - Trying to find archived lists can be a little difficult*