**Software Requirements**



**Specification**

**for**

**II-<ProjectNo.><ProjectName>**

**Version <X.X>**

**Prepared by**

**Group Number:** ​**II-<X>**

**<name of Team Lead first> <Roll #>**

**<name> <Roll #>**

**<name> <Roll #>**

**<name> <Roll #>**

**<name> <Roll #>**

**Project Owner:<*place your allotted TA’s name here>*** ​

**Course: CS4096D Software Engineering Laboratory**

**Date:<place the date of submission here>**

**This template is based on the one available**

**from GMU site by Dr. Rob Pettit.**

**Modifications specific to NITC are made**

**and will be used for academic purpose**

**only.**

**Contents**

1

[**1 Introduction** **3**](#_Toc18734)

[1.1 Document Purpose 3](#_Toc18735)

[1.2 Product Scope 3](#_Toc18736)

[1.3 Intended Audience and Document Overview 3](#_Toc18737)

[1.4 Definitions, Acronyms and Abbreviations 3](#_Toc18738)

[1.5 Document Conventions 3](#_Toc18739)

[1.6 References and Acknowledgments 4](#_Toc18740)

[**2 Overall Description** **5**](#_Toc18741)

[2.1 Product Overview 5](#_Toc18742)

[2.2 Product Functionality 5](#_Toc18743)

[2.3 Design and Implementation Constraints 5](#_Toc18744)

[2.4 Assumptions and Dependencies 5](#_Toc18745)

[**3 Specific Requirements** **6**](#_Toc18746)

[3.1 External Interface Requirements 6](#_Toc18747)

[3.1.1 User Interfaces 6](#_Toc18748)

[3.1.2 Hardware Interfaces 6](#_Toc18749)

[3.1.3 Software Interfaces 6](#_Toc18750)

[3.2 Functional Requirements 6](#_Toc18751)

[3.3 Use Case Model 7](#_Toc18752)

[3.3.1 Use Case #1 (use case name and unique identifier – e.g. U1) 7](#_Toc18753)

[3.3.2 Use Case #2 8](#_Toc18754)

[**4 Other Non-functional Requirements** **8**](#_Toc18755)

[4.1 Performance Requirements 8](#_Toc18756)

[4.2 Safety and Security Requirements 8](#_Toc18757)

[4.3 Software Quality Attributes 9](#_Toc18758)

[**5 Other Requirements** **9**](#_Toc18759)

[**Appendix A - Activity Log** **9**](#_Toc18760)

**Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft Type and  Number | Names of all team members | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# Introduction

*<TO DO: Please provide a brief introduction to your project and a brief overview of what the reader will find in this section.>*

## Document Purpose

*<*​*Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.*

*TO DO: Write 1-2 paragraphs describing the purpose of this document as explained above.>*

## Product Scope

*<*​*Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals.*

*TO DO: 1-2 paragraphs describing the scope of the product. Make sure to describe the benefits associated with the product.>*

## Intended Audience and Document Overview

*<*​*Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers*​*.* ​*Describe what the rest of this SRS contains and how it is organized.*​*>*

## Definitions, Acronyms and Abbreviations

*<*​*Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations.*

*TO DO: Please provide a list of all abbreviations and acronyms used in this document sorted in alphabetical order*​*.>*

## Document Conventions

*<In general this document follows the IEEE formatting requirements. Use Arial font size 11, or 12 throughout the document for text. Use italics for comments. Document text should be single spaced and maintain the 1” margins found in this template. For Section and Subsection titles please follow the template.*

*TO DO:* *Describe*​ *any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance.* *Sometimes,*​ *it is useful to divide this section into several sections, e.g., Formatting Conventions, Naming Conventions, etc.>*

## References and Acknowledgments

*<*​*List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document.*​ *>*

# Overall Description

## Product Overview

*<*​*Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface*​*. In this section it is crucial that you will be creative and provide as much information as possible.*

*TO DO: Provide at least one paragraph describing product perspective. Provide a general diagram that will illustrate how your product interacts with the rest of the environment and in what context it is being used. This is not a formal diagram, but rather something that is used to illustrate the product at a high level. You may draw this diagram using any online tool>*

## Product Functionality

*<*​*Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary is needed here. These can be at the level given in the project description.> TO DO:*

*Provide a bulleted list of all the major functions of the system. No need to explain them.*

## Design and Implementation Constraints

*<*​*Describe any items or issues that will limit the options available to the developers. These might include: hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software). You can be creative here to some degree. As you know, we will be using Android Studio for development and will follow SDLC. You may mention them here.*​*>*

## Assumptions and Dependencies

*<*​*List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project. If you are using a location based project, for example, you may decide to depend on google maps. This should come here.*

*TO DO: Provide a short list of some major assumptions that might significantly affect your design.>*

# Specific Requirements

## External Interface Requirements

### User Interfaces

In our application, we provide a very interactive user interface for both users (who upload and see the papers) and administrator (Who verify the papers) so they can easily access the papers according to their requirements like which year paper he/she wants or from which subject or department the paper belongs. And also user can upload the paper by giving the specific year, paper name or type of papers(like T1, T2 or End-Semester) with the name of the teacher and department name.

### Hardware Interfaces

a) Serve-Side: The mobile application will be going to host on different App Stores available for IOS and Android.

b) Client-side: The application will be display on Mobile Screen and on the application, there are different widgets that will be present/available.

### Software Interfaces

We need a mobile phone in which our application can run. we need a hosted database from which application can get the data.

## Functional Requirements

1. See Question: In our application user is capable to see the past year papers according to their requirements like which year paper he/she wants or from which subject or department the paper belongs.

2. Upload-Question: user can upload the paper by giving the specific year, paper name or type of paper(like T1, T2 or End-Semester) with the name of the teacher and department name.

3. Upload – Solution: Here user (Student/Faculty) can upload solution of the particular question or paper which are present in our application and also he can review them.

4. See – Solution: We already mentioned that a user can upload a solution of questions so a user also can

see the solved papers or questions whose solution he/she wants to see.

5. Request to upload question paper: In this section, we can request users to upload question papers of a particular paper or particular year paper.

6. Request to upload Solution: In this section, the user can request to upload the solution of a particular question or particular paper.

7. Verification: If the user uploads either question or solution of a particular question it has to verify by the administrator.

8. Login for Administrator: Here Administrator means the person who is going to approve the question or solution so before he/she approve uploaded thing he/she has to authenticate there self.

9. Discussion: Our application has a very good functionality which is a discussion form in which users/students can discuss a particular question or any doubt they want to clear.

10. Status: When the user uploads a question paper after upload it goes to the administrator to approve so the user can check the status. The status may be approved, waiting or rejected.

## Use Case Model

### Use Case 1 (Login U1)

**Author –** Kapil Kumar Chhipa

**Purpose**​ - Administrator logs in to system using existing profile for verify uploaded questions and solutions. We are using it to maintain security.

**Requirements Traceability –** to verity uploaded questions and answers administrator has to log in to authenticate his/her identity.

**Priority**​ - High

**Preconditions**​ - User is not logged in to a profile, input profile exists in data base, user password matches profile.

**Post conditions**​ - Administrator will be redirected to page where all the latest questions and solutions uploaded so that he/she can verify/reject them.

**Actors**​ – Actors (Administrator) that trigger the use case to execute or provide input to the use case

**Extends –** Authenticate

**Flow of Events**

1. Basic Flow Administrator will see all the questions and solutions which are not verified till now.
2. Alternative Flow Invalid password, invalid username, or mismatched username and password redirect to error message and previous page.
3. Exceptions – Internet is not working of the user during login.

**Includes**​ (8)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved

### Use Case 2(upload Question)

**Author –** Kapil kumar chhipa

**Purpose**​ - Any User can upload question into the Application so that others can download for exams. He/she want to share questions.

**Requirements Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User need not to login in application to upload the questions.

**Post conditions**​ - Any other person to whom the content was made available is able to download it.

**Actors**​ – Any User(Student of nit Calicut, faculty)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- The user uploads a file to be shared using the upload box. The file then gets uploaded to the server and desired users are able to download.
2. Alternative Flow- file is corrupt, slow internet.
3. Exceptions – server is not responding.

**Includes**​ (other use case IDs)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved .

### Use Case 3(upload Solution)

**Author –** Tushar Gupta

**Purpose**​ - Any User can upload solution into the Application so that others can download for exams. He/she want to share solution.

**Requirements Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User need not to login in application to upload the questions.

**Post conditions**​ - Any other person to whom the content was made available is able to download it.

**Actors**​ – Any User(Student of nit Calicut, faculty)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- The user uploads a file to be shared using the upload box. The file then gets uploaded to the server and desired users are able to download.
2. Alternative Flow- file is corrupt, slow internet.
3. Exceptions – server is not responding.

**Includes**​ (other use case IDs)

**Notes/Issues**​ - Any relevant notes or issues that need to be resolved.

### Use Case 4(Request Solution)

**Author –**  Tushar gupta

**Purpose**​ - Any User can request for a particular subject/year/exam question into the Application so that others can see the request and upload the questions(if they have) . **Requirements Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User need not to login in application to upload the questions.

**Post conditions**​ - All the user are able to see the request in the application.

**Actors**​ – Any User(Student of nit Calicut, faculty)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- The user will request particular questions/solutions.
2. Alternative Flow-user is not able to request .
3. Exceptions – Application is not responding.

**Includes**​ (other use case IDs)

### Use Case 5(status)

**Author –** Ramswarup Kulhary

**Purpose**​ - User can see the status of questions/solutions , he/she uploaded.

**Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User need not to login in application to upload the questions.

**Post conditions**​ - Any User can download questions/solutions after approved.

**Actors**​ – Any User(Student of nit Calicut, faculty)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- the user will see the status of questions/solutions uploaded by him/her.
2. Alternative Flow- slow internet.
3. Exceptions – server is not responding.

**Includes**​ (other use case IDs)

### Use Case 6(see and download paper)

**Author –** Ramswarup Kulhary

**Purpose**​ - User can see the question paper and download them.

**Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User must have application install in his/her mobile phone.

**Post conditions**​ - File should be download in User mobile Phone.

**Actors**​ – Any User(Student of nit Calicut, faculty)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- the user will see or download the question paper .
2. Alternative Flow- slow internet.
3. Exceptions – file is not downloading.

**Includes**​ (other use case IDs)

### Use Case 7(Discussion)

**Author –** Subhash kumar gupta

**Purpose**​ - User can discuss about a particular question and it’s solution.

**Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User must have application install in his/her mobile phone.

**Post conditions**​ - All the user can see the duscussion.

**Actors**​ – Any User(Student of nit Calicut, faculty)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- Users will discuss in discussion form .
2. Alternative Flow- discussion form is not working.
3. Exceptions – message is not sent.

**Includes**​ (other use case IDs) .

### Use Case 8(Authenticate)

**Author –** Subhash kumar gupta

**Purpose**​ - during administration login we have to authenticate whether he/she is valid user or not.

**Traceability –** Identify all requirements traced to this use case - the F​*n* ​numbers from Section 3.2 above

**Priority**​ - High

**Preconditions**​ – User must enter username and password for authentication.

**Post conditions**​ - Administration will be login after authentication.

**Actors**​ – Any User(Administration)

**Extends –** ​If this is an extension use case, identify which use case(s) it extends <Study what “extends” actually means before proceeding>

**Flow of Events**

1. Basic Flow- User will enter username and password and he/she is redirected to main page after authentication.
2. Alternative Flow- Authentication failed.
3. Exceptions – server is not responding.
4. **Includes**​ (other use case IDs)

…

# Other Non-functional Requirements

## Performance Requirements

## Our application is capable to handle a large amount of users simultaneously. We know that our application is for students and its capable to handle large amount of users so it provide reliability when application is under load and the possibility of under load occur when the exams are going on so in that condition our application should be reliable and provide a quick response to user.

## Safety and Security Requirements

<​Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied. ​Specify any requirements regarding security or privacy issues surrounding the use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Software Quality Attributes

The software consists of the following elements:

## Software Quality Attributes

**4.3.1 Reliability:** The Reliability of the overall program depends on the reliability components.

**4.3.2 Availability :** The system should be available at all times, meaning the user can access the details by using the application and our application is always available so it provide the availability until the mobile phone is out of order.

**4.3.3 Security:** We know that the user can access the app by its email or the administrator have to login before the use so it provide the security. And also if user want to upload question or solution of a particular paper of question it has to verify by the Administrator so we can say that our application is totally secure.

**4.3.4 Maintainability:** We use android studio for developing the application so with the help of that we can maintain our application.

**4.3.5 Portability:** Our application not only support by the Android devices it also be supported by then IOS or any other mobile devices so we can say that our application is portable.

# Other Requirements

*5.1. Mobile device.*

*5.2 Database*

*5.3 Internet*

# Appendix A - Activity Log

*<Provide details of group meetings - when you met and for how long - including the meeting details with the owner. You must also state what was the contribution (the sections mainly, then diagrams) of each of the team members. Team Lead will have complete responsibility and freedom to complete the Activity Log.>*