## Jaypee Institute of Information Technology, Noida

**Department of Computer Science and Engineering**



**Software Requirements Specification (SRS) Report on**

**TEACHER-STUDENT INTERACTION PORTAL**

**(Jaypee Insight)**

**Submitted to: Submitted by:**

**Mr. ANKUR KULHARI RADHIKA TRIVEDI-9912103531**

**ANUJ GARG-9912103554**

**DEEPIKA SAINI-9912103557**

**PRIYANSHA DWIVEDI-9912103559**

# CONTENTS

**1 INTRODUCTION ................................................................................................................................................ 4**

1.1 DOCUMENT PURPOSE ................................................................................................................................................ 4

1.2 PRODUCT SCOPE ................................................................................................................................................. 4

1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW ................................................................................................................................................. 4

1.4 DEFINITIONS, ACRONYMS AND ABBREVIATIONS ................................................................................................................................................. 5

1.5 DOCUMENT CONVENTIONS ................................................................................................................................................. 5

1.6 REFERENCES AND ACKNOWLEDGMENTS ................................................................................................... ……………………………..6

**2 OVERALL DESCRIPTION ..............................................................................................................................……………6**

2.1 PRODUCT PERSPECTIVE ............................................................................................................................ ………….....6

2.2 PRODUCT FUNCTIONALITY ......................................................................................................................... ………………6

2.3 USERS AND CHARACTERISTICS ....................................................................................................................................... …….9

2.4 OPERATING ENVIRONMENT ………........................................................................................................................ ……….10

2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS ............................................................................................................................................... ..10

2.7 ASSUMPTIONS AND DEPENDENCIES .......................................................................................................... ………………………..10

**3 SPECIFIC REQUIREMENTS .......................................................................................................................... …………….10**

3.1 EXTERNAL INTERFACE REQUIREMENTS ..................................................................................................... …………………………..10

3.2 FUNCTIONAL REQUIREMENTS .................................................................................................................... …………………12

3.3 BEHAVIOUR REQUIREMENTS ...................................................................................................................... ………………..15

**4 OTHER NON-FUNCTIONAL REQUIREMENTS ......................................................................................... …………………………………...18**

4.1 PERFORMANCE REQUIREMENTS ................................................................................................................ ……………………18

4.2 SAFETY AND SECURITY REQUIREMENTS ................................................................................................... …………………………….18

4.3 SOFTWARE QUALITY ATTRIBUTES ............................................................................................................. ………………...........18

**5 OTHER REQUIREMENTS .............................................................................................................................. …………...19**

**APPENDIX B - REFERENCE PAPER SUMMARY & GANTT CHART &RESEARCH PAPER................................................................................................................................... 20**

1. INTRODUCTION

1.1 DOCUMENT PURPOSE:

The purpose of this SRS document is to provide the detailed overview of our software product, its parameters and its goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client will see the product and its functionality.

This document also records the requirements for the design and the development of our website “JAYPEE INSIGHT” so that we can refer this document in future to see any functionality of the system. This document overall plays an important role in the development of our project.

# 1.2 Product Scope

The scope of the website is to provide a friendly set of web pages that are easy to navigate and at the same time provides the sufficient depth and information about the college events and the workshops. Also teachers can upload assignments for the students on the website and the students can submit the assignments before the deadline.

Two potential groups of viewers exist:

1. Teachers: To upload assignments and study material for the students and also view the college’s latest news.
2. Students: To view the college events, workshops information and to submit the assignments uploaded by the teachers before the deadline and to submit and respond to query.

The goals of “JAYPEE INSIGHT” is to provide easiness to students in submitting assignments and for teachers to upload it easily thus providing an easy and friendly interaction .It encourages more and more students to register on the website and submit their assignments online and get all the college events , workshops information online.

# Benefits for using this site should include:

1. Students can submit their assignments online and get all the latest news and updates about the college events.
2. Teachers upload assignments online and also set the deadline for the submission of the assignments.

# 1.3 Intended Audience and Document Overview

This document is meant for the developers and the faculty members. This document aims to explain in an easy manner, the basic idea behind the project and how the developers aim to achieve their goals.

The remaining section of this document provides the overall description which is discussed in section-2 and includes product perspective, product functionality, operating environment, design and implementation constraints, assumptions and dependencies. It also provides the specific requirements which is discussed in section-3 and includes external interface requirements, functional requirements and the behavior requirements. It also provides the non-functional requirements which is discussed in section-4 and includes performance requirements, safety and security requirements.

# 1.4 Definitions, Acronyms and Abbreviations

1. WWW- World Wide Web
2. HTML – Stands for the hypertext markup language; a standard for presenting and transporting mostly text-based information across the internet.
3. INTERNET –Global network of computers that can share all kinds of digital information.
4. PHP – Server scripting language.
5. MySQL –Database that will be used for the project.
6. Apache – A web service.
7. Web-based application – Software program that is executed using the World Wide Web.
8. Ajax-is a group of interrelated Web development techniques used on the client site to create asynchronous web applications.

# 1.5 Document Conventions

Throughout this documentation, the following conventions have been used:

1. Fonts: Times New Roman.
2. Size = 20 for the sections heading.
3. Size = 16 for the sub-sections heading.
4. Size = 14 for the rest of the document.

Words in bold are the important terms, and have been formatted to grab the attention of the reader.

# 1.6 References and Acknowledgments

1. Php, Mysql and Apache by Julie C. Meloni, Pearson Education.
2. Fundamentals of Database systems by Ramez Elmasri and Shamkant B.Navath.
3. JavaScript and JScript by Jaworski, bpb publication.

We also refer from some of the websites like:

1. http://[www.codrops.com](http://www.codrops.com)
2. http://www.formget.com
3. http://[www.menucool.com](http://www.menucool.com)
4. https://[www.w3schools.com](http://www.w3schools.com)
5. https://[www.apache.org](http://www.apache.org)
6. https://[www.php.net](http://www.php.net)
7. https://[www.mysql.com](http://www.mysql.com)
8. https://[www.codeproject.com](http://www.codeproject.com)
9. https://[www.youtube.com](http://www.youtube.com)
10. https://www.ieee.org

# 2. Overall Description

# 2.1 Product Perspective

This product is neither the update of any product nor the replacement of the existing product but it is the new self contained product. It must be flexible enough to run easily with a variety of different operating systems, computer architecture and be simple to install and use. The sole requirement for the user is the web browser (Google chrome, Mozilla Firefox) with an active internet connection.

The product requires the use of a keyboard and a pointing device to interface with the user. Its user interface uses the animations. The product is compatible with all the graphics display. The product should not require speakers. It also should not require special or the new hardware.

# 2.2 Product Functionality

# Major Functionalities of the system:

1. Teachers can upload the assignments, deadline to submit the assignments and the other related study material for the students.
2. Students can access the time table, academic calendar and all the information of the college events and the workshops organized in the college.
3. Teachers can also maintain the record of the students who have submitted the assignments and who have not.
4. Teachers and students also view the latest news and updates of all the college events.
5. Students can submit the assignments online and also access the reference material uploaded by the respective teacher.

|  |
| --- |
| **HOME PAGE** |
| **Teacher’s and student’s login in dialog box** |
| **News and updates panel** |
| **Timetable and academic calendar access and updates** |
| **Feedback –help to improve us** |

Page 6

ON LOGING IN

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FOR STUDENTS**   |  | | --- | | **Student name** | | **Year** | | **Batch** | | **DOB** | | **Enrolment no.** | |
| **Redirects to respective year page** |

|  |  |  |  |
| --- | --- | --- | --- |
| **FOR faculty**   |  | | --- | | **Teacher ID** | | **Password** | | **Campus** | |
| **web page of teacher** |
| **Uploading assignment and studymaterial batch list** |
| **Notices provided by teacher** |
| **Any query updates** |

|  |
| --- |
| **YEAR PAGE** |
| **Study material and**  **List of all faculty** |
| **list of all assignment along with the teacher, batch and last date of submission** |
| **Upload assignment** |

AFTER SUBMISSION

|  |
| --- |
| **Submitted assignment will be stored in our database and directed to the respective batch teacher** |
| **Batch link which includes three list-**  **-List of student who submitted the assignment**  **-List of late submission in case of extended date criteria**  **-List of those students who have not submitted the assignment** |
| **Submission option will be valid till the last date or extended date provided by teacher** |

|  |
| --- |
| **NEWS AND UPDATE PANEL** |
| **Consists of three panel**   * **JYC BULLETIN** * **WORKSHOPS** * **REGULARS NEWS AND UPDATES(EXLUDING JYC AND WORKSHOPS)** * **A news will be deleted automatically after the event date expires** |

|  |
| --- |
| **GENERAL QUERY OR PUBLIC QUERY PANEL** |
| **Any registered member can ask query related subject or college to any other registered member** |
| **Before query submission panel will prompt student to log in to maintain database and to avoid any offense activities by students**  **In case of offense – student will be warned once and blocked then**  **Receiver will also get the details (name and enrol no.) of sender automatically fetched form students login details** |

|  |
| --- |
| **FEEDBACK PANEL** |
| **Any improvements ides for the websites by teachers and students**  **Only visible to admin** |

# 2.3 User classes and characteristics

1. Faculty members:Faculty members will use this system in order to upload reference material and assignments for the students. Faculty members can also view the latest events information and updates.
2. Students:Students cannot upload anything on the website. Students can refer to the study material uploaded by the respective faculty member and also submit the assignment to their faculty member before the deadline of submitting the assignment. Students can also view the latest events information uploaded on the website. Students can also view the information of the workshops organized in the college.

The most important users for this product are the faculty members and the students but there are some functions which cannot be performed by the students like uploading of assignments which means the product functions vary from user to user.

# 2.4 Operating Environment

Operating environment is the environment in which our software product will operate. Product will operate with a variety of different operating systems like windows, ubuntu. Product also requires the web browser (Google chrome, safari) with active internet connection, keyboard and mouse in order to operate on personal computers.

# 2.5 Design and Implementation Constraints

1. Cannot fix where the objects will appear on the viewers screen.
2. Different users will see a website on the different viewports which implies design the WebPages that will appear in the mobile phones and maybe even projected onto wall-sized displays.
3. Another constraint of website design is that unlike print designs, where the viewing area of any design is fixed, web users can zoom in or out as users interact with a web page, changing the size of text and images. And, by the way, different browsing environments handle zoom differently — some enlarge images as text is enlarged, and other times enlarging text doesn’t affect other page elements.

# 2.6 Assumptions and Dependencies

1. The product must be web-based.
2. There are no memory requirements.
3. The computers must equip with web browsers such as internet explorer.
4. The product must be stored in such a way that allows the client easy access to it.
5. The product must have a user friendly interface that is easily understandable.

# 3. Specific Requirement

## Interface Requirements

**Software Interface:**

-I- **Front End Client:** Html

-I- **Web Server:** xampp

-I- **Data Base Server:** mysql

-|- **Back End:**php

**Hardware Interface:**

- **Client Side: PC (Monitor)**

- **Server Side: PC**

**User Interfaces**

* The website should work and be tested against IE, Firefox, Google Chrome and Netscape.

In this project User Interface plays the most important role for communication between user and software product. The interface should present and acquire information in a consistent fashion. This implies that:

1. All visual information is organized according to a design standard that is maintained throughout all screen displays;
2. Input mechanisms are constrained to a limited set that are used consistently throughout the application.

Mechanisms for navigating from task to task are consistently defined and implemented

Following by, these considerations can be given GUI requirements.

**First, GUI requirements:**

1. register user interface has to contain:

* NAME(first name and last name)
* Enrollment number
* Batch
* Year
* Branch
* Phone number
* Email
* Date of birth

2. Login User Interface has to contain:

* Login dialog to input login and password
* Type of login(teacher/student)
* Password reset option

**GUI requirements for student’s part:**

User Interface for assignment submissions should be implemented as a web-based interface. It contains the following requirements:

1. General. Interface consist of:

* Student login to submit assignment
* List of uploaded assignments. Possibility to clear field of request
* Chat box to initiate or reply a query

**GUI requirements for teacher’s part:**

User Interface for uploading assignments by teachers should be implemented as a web-based interface. It contains the following requirements:

* Teacher login to upload assignment
* list of students who have submitted the assignment
* reply to query portal(optional)

FUNCTIONAL REQUIREMENTS

Template for describing functional requirements.

|  |  |
| --- | --- |
| **Purpose** | a description of the functional requirement and its reason(s) |
| **Inputs** | what are the inputs; in what form will they arrive; from what sources can the inputs come; what are the legal domains of each input |
| **Processing** | describes the outcome rather than the implementation; includes any validity checks on the data, exact timing of each operation (if needed), how to handle unexpected or abnormal situations |
| **Outputs** | the form, shape, destination and volume of output; output timing; range of parameters in the output; unit measure of the output; process by which the output is stored or destroyed; process for handling error message produced as output |

**HOME PAGE**

|  |  |
| --- | --- |
| **Purpose** | This menu is the opening page for the system. |
| **Inputs** | This page automatically loads after providing url. |
| **Processing** | The product responds by d |
| **Outputs** | Display the html page of login dialog for students and teachers ,  Displays academic calendar and timetable  Displays latest news, updates and notifications of college events or jyc events  Feedback |

**YEAR PAGE**

|  |  |
| --- | --- |
| **Purpose** | Provides respective year’s assignment details   and query chat portal |
| **Inputs** | This page loads whenever a student registers or logged in |
| **Processing** | The product responds by assignment submitted by students  Ask for any query |
| **Outputs** | Displays Assignment upload dialog box  List of all assignment uploaded for respective year and batch  Students query chat portal |

**TEACHER’S PAGE**

|  |  |
| --- | --- |
| **Purpose** | To upload assignment and study material by any teacher to respective batches and to receive submitted assignments and keep the record of all the students who have submitted the assignments before deadline |
| **Inputs** | This page loads whenever a teacher logs in |
| **Processing** | The product responds by assignment uploaded by students |
| **Outputs** | List of students who have submitted the assignments(on time and late)  Teachers uploaded assignments and study material.  Any query responded by teachers. |

EVENT PAGE

|  |  |
| --- | --- |
| **Purpose** | This subsequent page of workshops and notifications |
| **Inputs** | This page loads automatically after clicking respective page |
| **Processing** | The product responds by assignment submitted by event query . |
| **Outputs** | Displays html pages of events.  Displays details of every events. |

General viewpoints of different people represent functional requirements and are lead as follows

|  |  |
| --- | --- |
| **Name1** | **Student** |
| **Focus** | Focus on making the software product easy to use. |
| **Concerns** | The portal should be simple to understand and implement |
| **Source** | Customer’s requirements document |
| **Requirements** | 1. Interface must be user friendly. 2. System should have an assignment details(to submit assignments to specific teacher on time) 3. System should have all the details of current college’s events(to keep updated about jyc ,workshops,) 4. System should have Interactive query submission portal. 5. System should have timetable and academic calendar. |

|  |  |
| --- | --- |
| **Name2** | **Teacher** |
| **Focus** | Focus on the simplicity of usage |
| **Concerns** | The portal should be simple to use and should have friendly interface. |
| **Source** | Customer’s requirements document |
| **Requirements** | 1. The system should have information of students and batch list. 2. User friendly interface of the portal 3. Easy and automated assignment uploads to specific batch. 4. The system should provide easy and genuine portal to submit any query’s answer. |

* 1. **BEHAVIOUR REQUIREMENTS**

**USE CASE DIGRAM**



**CLASS DIAGRAM**



## Other non-functional attributes

## Performance Requirements

The product can be web-based and run from a web server.  Initial load time depends on the media from which the product is run. Once the product is loaded, it runs completely within the computer's memory.

4.2 Safety and performance requirements

* **Security**

Priority: High

•Confidentiality – extent to which data and processes are protected from unauthorized disclosure

•Integrity – extent to which data and processes are protected from unauthorized modification

•Availability – extent to which data and processes are protected from denial of service to authorized users

**4.3 SOFTWARE QUALITY ATTRIBUTES:**

The product is made for students and teachers of the college. Both, the students and the teachers must be able to easily understand and use the product. The product must load quickly and work well on a variety of computer systems. It must also tolerate a wide variety of input possibilities from a user, such as any incorrect responses or any unforeseen keystrokes and selections from a pointing device.

* **Portability:**

The product can be viewed from various web browsers from any location and any media provided that the internet connection at that place is working properly**.**

* **Reliability**

Priority: High

•Non-deficiency – degree to which SW does not contain undetected defects

•Fault tolerance – degree to which software will continue to work without system failure that would cause damage to users or property; also, the degree to which SW includes degraded operation and recovery functions

* **Maintainability**

Priority: High

–Corrective - finding and fixing faults

–Adaptive - modifying software for a changed environment

•rewrite documentation or comments

•renaming a variable or routine

* **Reusability**

Priority: Medium

•Generality – extent to which SW can be reused across applications

•Self-Descriptiveness – ease with which a SW component can be understood and used

* **Robustness**

**:**

The product must be able to take a lot of input possibilities from the user. It should check for any wrong or unacceptable entry by the user and validate it. The assignment of each student must be secured without any intermixing of data. Any unforeseen behaviour or response should be tactfully handed.

* **Maintainability:**

The product should be maintainable and upgradable.

1. **Other requirements:**

No other requirements as of now.

**Limitations**

1. “JIIT INTERACTION PORTAL” is an intra college website which cannot be accessed by any other college student. Only the students of JIIT will access this website.
2. Teachers cannot upload any video in the reference material for the students.
3. Students can submit their assignments online but cannot do the assignment work online.
4. Students cannot upload any upcoming events or workshops information

**GANTT CHART**

|  |  |  |
| --- | --- | --- |
| PHASE | CATEGORY | Estimated date of completion |
|  |  |  |
| Project discussion | communication |  |
| Requirement gathering |  | 20 OCT 2014 |
| Synopsis |  |  |
| Timeline of the project | planning | 28 OCT 2014 |
| Designing and architecture | modelling | 2 NOV 2014 |
| Coding and debugging | construction | 1 DEC 2014 |
| Testing and publishing | deployment | 3 DEC 2014 |

**Research Paper**

**Abstract**

Currently, the significance of a portal stems not only from being a handy way to access data but also from being the means of facilitating the integration with web applications. This paper proposes an interaction instance oriented approach for integrating web applications in portals. The key aspect is to enable a user to have the same interaction experiences from a portal that he/she accesses the web application directly. The approach is thus focused on the description of the presentation layer of the interaction instances of a web application and a portlet, which defines an interaction instance as consecutive web pages or fragments. Web application integration is then transformed to the problem that how to translate all web pages of an interaction instance of a web application to certain view equivalence regions, which form the interaction instance of a portlet. Experiments show that the approach is effective and efficient.

**Introduction**

Presentation level integration now becomes an important and fast growing trend in enterprise . The advent of portals and the standardization of the component model in portals bring a new and flexible mechanism for constructing portal-based composite applications that are the mainstream to realize the integration at presentation level. Portals enable the aggregation of interactive interfaces of different applications as components on the same web page . Portlet is the basic component of a portal, which represents an interactive web mini application and is deployed on a portal server . A portal typically decorates the HTML fragment returned by a portlet with a title and several buttons, such as minimize, maximize and edit etc., then aggregates all fragments together into a portal page. Over the last decade, enterprises have invested in web-enabling many of their existing systems. These enterprise because they are related to the enterprise business closely. Also, Internet is now a rich source of information and applications. Both aspects vindicate the effort of being able to tap into these resources, making them available from a portal. There are two types of integration of web applications in portals: shallow integration and deep integration. A shallow integration makes a web application accessible from a portal. However, the web application and the portal are actually two separate applications. They have no further relationships other than a link from the portal to the web application, thus the web application can not benefit from the information provided by the portal, such as user preferences, application context, etc. Furthermore, portlet interoperation is also hard to be achieved because the portal has no any information about the web application. By contrast, a deep integration makes the entire application available from the portal framework. That is to say, all interactions with the integrated web application are achieved without leaving the portal, thus the functionality of the web application can be supplemented with the customization or identification commodities offered by the portal framework. This implies to wrap the existing web applications as portlets. This paper addresses the generation of portlets, which allows a web application to be invoked from the context of a portal. To make the proposed approach a general solution, we consider the problem from a user interaction point of view. An interaction instance of a web application is defined as a sequence of web pages, while an interaction instance of a portlet is defined as a sequence of regions, each of which is a part of a web page. The goal of web application integration is to enable a user to have the same interaction experiences from a portal that he/she accesses the web application directly. We define such a same interaction experience as view equivalence. Thus, web application integration in portals is transformed to the problem that how to translate all web pages of an interaction instance of a web application to certain view equivalence regions, which form a portlet interaction instance.

**Problem motivation and analysis**

For fragment writing, a user uses a browser to access a web application. In a traditional web page, a user would typically select a link on the page presented in the browser, which would lead to another HTTP request returning a completely new page of content. By contrast, a portlet markup occupies just a region of the whole portal page. Navigation within a portlet, should give the impression that the contents of only that region are changing. This implies that URLs embedded in a markup fragment often should not be direct links to the provider. Rather, portlet URL should point to the portal. They must be rewritten so that the portal intercepts and re-routes them to the correct portlet, adding the proper context. Moreover, each portlet returns a fragment, which is aggregated with a general page frame and the fragments returned from the other portlets to form the final portal page. If such a portlet is generated through integrating an existing web application, the original fragment usually contains some tags that should not appear in the fragment of a portlet, such as <HTML>, <BODY>, etc. Moreover, the action and view behavior of an element may be affected by other elements on the same web page, such as Script, CSS, etc. If such parts are not extracted together, the elements may not behave correctly. The basic reason that leads to the above three issues is that the application is moved from an exclusive runtime context to a portal context, which is shared with other portlets. To propose a general approach for such a problem that is introduced by context transfer, we need a solution that can clip a web application, and then can wrap the clipped part of the web application to a portlet, so that it can be used in a portal context.

**Web application integration model**

The ultimate goal of web application integration in a portal is that a user can obtain the same interaction experience in the portal as he/she can by accessing the application directly. That is to say the user can get all information interested when such an application is integrated into the portal. Noted that in a black box based approach, an application only accepts HTTP request as its input, and outputs a whole web page, we hope to find out a solution only based on such information so that a web application can be integrated without modifications. Hence, we consider the problem from an interaction instance point of view. The key aspects of our proposed web application integration model can be summarized as follows: a web page is defined as a sequence of HTML nodes, an interaction instance of a web application is defined as a sequence of web pages. A region is defined as a set of consecutive HTML nodes in a web page. Portlet interaction instance is then defined as a sequence of regions. As all information is displayed for a user via several interaction interfaces, we call such the same interaction experience as view equivalence. Web application integration in portals is thus transformed to the actions that translate all interaction instances of a web application to their view equivalence portlet interaction instances.

**Definition 1**. HTML Node is a triple N=(tn, A, value), where tn is the tag name of the node, tn∈HTN, which is the HTML tag name set defined in HTML specification; A is the set of node attributes and valueis the content of the node.

**Definition 2**. Node Attribute is a tuple ATT= (aname, avalue), where aname∈HAN, HAN=RAN∪AAN∪GAN. avalue is the value of a node. HAN is a set of HTML attribute name, which is composed of three subsets. RAN is the set of resource attribute name, which is used to depict a resource attribute. For example, in the tag <img src="./images/banner.jpg">, src is a resource attribute, which is used to denote the location of the image resource. AAN is the set of action attribute, which is used to define the attributes related to navigation and user interaction. Typical action attributes include href, action, etc. GAN is the set of general attributes, which are usually used to define the view behavior of an HTML element. Whatever a web application is developed, it is depicted as several hypertext documents that are written in HTML, which are received and displayed by a browser to fulfill the interaction with an end user.

**Definition 3**. Web page is a tree that uses HTML node as its basic unit. Usually, the corresponding HTML node of the root of a web page is <HTML>. Because the specification also employs a tree structure to describe an HTML document, building an HTML node tree from a web page is thus natural. Region is a part of a web page. A region can be aggregated with other region or HTML fragments to form a web page.