**Software Requirements Specification** 24th/07/2019

**For ANTIRAGGING.COM**

**Table of Contents**

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

a. References 3

15 Overall Description 3

a. Product Perspective 3

b. Product Features 3

c. User Classes and Characteristics 3

d. Operating Environment 4

e. Design and Implementation Constraints 4

f. User Documentation 4

g. Assumptions and Dependencies 4

16 System Features 4

17 External Interface Requirements 10

a. User Interfaces 10

b. Hardware Interfaces 10

c. Software Interfaces 10

d. Communications Interfaces 10

18 Other Non functional Requirements 10

a. Performance Requirements 10

b. Safety Requirements 10

c. Security Requirements 10

d. Software Quality Attributes 10

19 Other Requirements

1. **Introduction**

**1.1 Purpose**

The **AntiRagging.com** is a site that handles the ragging complains filed by the student using the internet as the sole medium. The purpose of this document is to present a detailed description of the AntiRagging.com System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

**1.2 Document Conventions**

The System name is highlighted all over the document with bolded letters as well as

underlined . Irrespective of that there is no specific convention provide. Every

requirement statement has its own priority.

**1.3 Project Scope**

This software system will be a AntiRagging.Com System. This system will be

designed to help students in filing their complains online direct to the anti ragging cell of the institution as well as for the anti ragging cell to react quickly to the complains filed by the students.

**a. Reference Websites:**

 www.google.com

 www.w3schools.com

 www.tutornation.com

 www.roseindia.com

**2 Overall Description**

The Overall Description section, of this document gives an overview of the

functionality of the product. It describes the informal requirements and is used to

establish a context for the technical requirements specification in the next heading.

The Product feature, Operating Environment, Design and Implementation

Constraints, of this document is written primarily for the developers and describes

the details of the functionality of the product. This system will be completely web-

based, linking to AntiRagging.Com and the remote web server from a standard

web browser. An Internet connection is necessary to access the system.

**a. Product Perspective**

AntiRagging.Com is a self contained system. This project envisages bridging the gap

between the student, and the management. The web portal should be user-friendly

and reliable for the above purpose. It is not a stand-alone

product and depends on the availability of Eclipse software. It should run

on Windows based platform.

**Product Features**

It is easy to use the product with all the information provided in the form of links.

The student, can access the portal with simple registration.

The product enables the student to file his complains online with minimum paper work.

Students can discuss their problems with the anti ragging cell via e-mail if necessary. There are

important contacts so that the students can refer if needed. There are FAQs also for students to read

if there is any doubt arise.

**Operating Environment**

**Technologies to be used**

**Programming languages**:

**JAVA EE:** Java Enterprise Edition is a programming platform— part of the Java

Platform-for developing and running distributed multi-tier architecture Java

applications, based largely on modular software components running on an

application server.

**HTML:** Hyper Text Markup is the predominant markup language for web

pages. It provides a means to describe the structure of text-based information

in a document and to supplement that text with interactive forms, embedded

images, and other objects.

**JavaScript:** A client side scripting language used to create dynamic web

content and user interface.

Tools &amp; Development Environment:

**Apache Tomcat 6.0.26 Server:** Apache Tomcat is a Servlet container

developed by the Apache Software Foundation (ASF). Tomcat implements the

Java Servlet and the Java Server Pages (JSP) specifications from Sun

Microsystems, and provides a pure Java HTTP web server environment

Java code to run.

**Eclipse IDE** , is a modular standards-based integrated development

environment (IDE), written in the Java programming language. The Eclipse

project consists of a full-featured open source IDE written in the Java

programming language and a rich client application platform, which can be

used as a generic framework to build any kind of application.

**Design and Implementation Constraints**

 The data can only be deleted or edit only by admin so it is secure.

 Limited to HTTP/HTTPS Protocols.

 No multilingual support( English only).

 User do not have any rights to edit any data in the system.

**User Documentation**

 The user should be familiar with the terminology like

Online form filling/Downloading/Using mail-account/Chatting etc.

 The user should be familiar with the Internet.

**Assumptions and Dependencies**

 Administrator is created in the system already

 Roles and tasks are predefined.

 Roles and responsibilities are already established.

**3 System Features**

**Registration interface:-**

**3.1.1 Description and Priority**

This feature will enable the new user whether he is a student or a faculty will enable him to enter his/her basic information to the data base.

**3.1.2 Stimulus/Response Sequences**

This form will consist of basic fields such as Name, Username, E-mail Id, Phone number, Branch etc for Students . while Faculty registration fields consist of fields like name department ,etc. There is a Register button in both the forms .If the user click the Register button will submit the data to the database at the server tier.

**3.1.3 Functional Requirements**

The most important requirement here is to input values in the database and store them there for future use. If any field is left to provide the data, the system will prompt the user by using the scripts and will not submit the data until corrections/data entries are made completely.

**Secure Login interface:-**

**3.2.1 Description and Priority**

This feature will enable the user to have a secure and simple login to

the system. To avoid handling a large number of errors and exceptions

this feature will enable the user to provide only a limited number of

inputs having constraints upon them and if there are any errors the

system will notify the user about them.

**3.2.2 Stimulus/Response Sequences**

It will consist of three basic fields Username and Password. There

are two buttons: Login and Forgot password. Login will submit the

entered data for approval followed by access, and Forgot password

button will change the details of the user.

**3.2.3 Functional Requirements**

The most important function is to only grant access to users that are

listed in the database. The customer will provide the information on

who will be allowed access. To implement the security, the web page

must check the database to see if the Username and Password are

valid. If they are not, the user will receive an “Enter correct username and

password ” as a response. If the user is not registered there is a link that will take user to the registration page

**4 External Interface Requirements**

**a. User Interfaces**

The user interface is screen shown on the browser. The Home screen of the

Web-Portal is where students and faculty can register and login.

The portal screen acts as an interface to provide services to the user which

are to be availed from the database.

**b. Hardware Interfaces**

A minimum of 40GB of HDD, with Pentium IV processor, a minimum of

256MB of RAM so that a suitable OS (Windows XP ) may be installed, and a

reliable internet connection is required for the client side/user side so that

may be accessed easily.

**c. Software Interface**

The system uses:

**JSP:** Java Server Pages. It is a technology that helps software developers

serve dynamically generated web pages based on HTML, XML and other

document types; uses java programming language.

**Servlet:** Java web-containers which holds actions to be performed; a Servlet

a java programming language class used to extend the capabilities of servers

that host applications access via a request response programming model.

.

**My Sql** : It is a object-Relational Database Management System. The

My Sql DBMS can store and execute stored procedures and functions within

itself.

**d. Communications Interface .**

Internet connection and Browser are required in order for several functions to be

executed such as downloading. The system uses the following browsers:-

 Mozilla Firefox

 Google Chrome

 Internet Explorer

**5. Other Non-functional Requirements**

a. Performance Requirements

Some Performance requirements identified is listed below

a. The database must be support more than 100 student, faculty records.

b. Can support many user at the same time.

c. Good internet speed .

d**. Software Quality Attributes**

There are a number of attributes of software that can serve as requirements. It is

important that required attributes should be specified so that their achievement can

be objectively verified. The following terms provide a partial list of examples

**Portability**

Some of the attributes of software that relate to the ease of porting the software to other host

machines and/or operating systems. This may include: Java is used to develop the product. So it

is easiest to port the software in any environment.

**Reliability**

Some of the attributes identified for the reliability is listed below:

1. All data storage for user variables will be committed to the database at the time of entry.

2. Data corruption is prevented by applying the possible backup procedures and techniques.

**Usability requirements**

Some of the usability requirements identified for this system are listed below:

1. A logical interface is essential to an easy to use system, speeding up

common tasks.

2. Error prevention is integral to the system and is provided in a

number of formats from sanity checks to limiting free-text input.

**Appendix A: Glossary**

**Term Definition**

Database Collection of all the information monitored by this system.

**Administrator**

Is a person responsible for maintaining one or many websites. The duties of the

webmaster may include ensuring that the web servers, hardware and software are

operating correctly, designing the website, generating and revising web pages, replying

to user comments, and examining traffic through the site.

**HTML**

Hypertext Transfer Protocol is a transaction oriented client/server protocol between a

web browser and a Web Server

**HTTPS**

Secure Hypertext Transfer Protocol is a HTTP over SSL (secure socket layer).

**SRS(Software Requirements Specification)**

A document that completely describes all of the functions of a proposed system and the

constraints under which it must operate. For example, this document.

**Stakeholder**

Any person with an interest in the project who is not a developer. User Student or faculty

**IDE**

An integrated development environment (also termed integrated design

environment, integrated debugging environment or interactive development

environment) is a software application that provides comprehensive facilities to

computer programmers for software development

**Email**

Electronic mail, commonly known as email or e-mail, is a method of exchanging digital

messages from an author to one or more recipients.

**Appendix B: Analysis Models**

Under the analysis model, we analyse the system to check the following:

1. Whether it meets the requirements that guided its design and development;

2. Works as expected; and

3. Can be implemented with the same characteristics.

To perform these analyses of the model, the following testing is to be

implemented:-

**Unit testing:** Unit testing, also known as component testing, refers to tests that

verify the functionality of a specific section of code, usually at the function level.

In an object-oriented environment, this is usually at the class level, and the

minimal unit tests include the constructors and destructors.

**Integration testing:** Integration testing is any type of software testing that seeks

to verify the interfaces between components against a software design. Software

components may be integrated in an iterative way or all together).

Integration testing works to expose defects in the interfaces and interaction

between integrated components (modules). Progressively larger groups of tested

software components corresponding to elements of the architectural design are

integrated and tested until the software works as a system.

**System Testing:** system testing is done to ensure whether the system meet all

the requirements stated in the SRS.

System testing is performed on the entire system in the context of a Functional

Requirement Specification(s) (FRS) and/or a System Requirement Specification

(SRS). System testing tests not only the design, but also the behavior and even

the believed expectations of the customer. It is also intended to test up to and

beyond the bounds defined in the software/hardware requirements

specification(s).

The system testing is categorized into three:

**Alpha testing**

Alpha testing is simulated or actual operational testing by potential

users/customers or an independent test team at the developers site. Alpha

testing is often employed for off-the-shelf software as a form of internal

acceptance testing, before the software goes to beta testing.

**Beta testing**

Beta testing comes after alpha testing and can be considered a form of external

user acceptance testing . Versions of the software, known as beta versions , are

released to a limited audience outside of the programming team. The software is

released to groups of people so that further testing can ensure the product has

few faults or bugs . Sometimes, beta versions are made available to the open

public to increase the feedback field to a maximal number of future users.

**Acceptance testing**

Acceptance testing performed by the customer, often in their lab environment on

their own hardware, is known as user acceptance testing (UAT). Acceptance

testing may be performed as part of the hand-off process between any two

phases of development.

**Appendix C:**

**Issues List**

The problems that might occur with the software product are:

**o** The portal is dependent on web services. Though we make every effort to

ensure that services are provided on time but there is no guarantee The

problems that might occur are:

**o** Some web pages are lost . The user is looking for a specific Web page

but try as they might, they cant find it.

**o** Web pages load slow or incorrectly. The user found the Web page he

wanted but it took forever to load or things are jumping around on the

page while loading.

**JavaScript Errors.**

OForms are completely broken. After clicking submit button, an error

might occur.

**o** Broken Registration Process

**o** Site wont load. Websites are supposed to work fine whether you type

in the www or not. But an error might occur.