Software Requirements Specification

for

Faculty Advisory System of NITC

Version 1.0 approved

Prepared by:

Vallu Manjunath	B150504CS
Raavi Prathap Reddy	B150681CS
Vishnumolakala Rohith	B150541CS
Padavala Bhavani Venkata Krishna	B150775CS

National Institute of Technology Calicut

October 8, 2017

Table of Contents

Table of Contents ..... **ii**
Revision History ..... **ii**
1. Introduction **1**
1.1 Purpose 1
1.2 Document Conventions 1
1.3 Overview of the document 1
1.4 Product Scope 1
1.5 References 2
2 Overall Description **2**
2.1 Product Perspective 2
2.2 Product Functions 3
2.3 User Classes and Characteristics 3
2.4 Operating Environment 3
2.5 Design and Implementation Constraints 3
2.6 User Documentation 4
2.7 Assumptions and Dependencies 4
3 External Interface Requirements **4**
3.1 User Interfaces 4
3.2 Hardware Interfaces 4
3.3 Software Interfaces 4
3.4 Communications Interfaces 5
4 System Features (Functional Requirements) **5**
4.1 Student 5
4.1.1 Enter Details: 5
4.1.2 Update Marks: 5
4.1.3 Register: 5
4.1.4 Read Messages: 5
4.2 Faculty Advisor 6
4.2.1 Access Details 6
4.2.2 Edit Data 6
4.2.3 Get List 6
4.2.4 Send Messages 6
4.3 HOD 7
5 Other Nonfunctional Requirements **7**
5.1 Basic Requirements 7
5.2 Performance Requirements 7
5.3 Safety Requirements 7
5.4 Security Requirements 7
5.5 Software Quality Attributes 7

Revision History

Name	Date	Reason For Changes	Version
-	-	-	1.0

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of Faculty Advisory System of NITC. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which the system will operate. This document is intended for Students, Faculty Advisors and HODs of National Institute of Technology Calicut and will be proposed to Dr. Abdul Nazeer KA, Associate Professor, CSED, NIT Calicut for approval.

1.2 Document Conventions

Here are some of the typographical conventions used while writing this document :

- NITC – National Institute of Technology Calicut
- CSED – Computer Science and Engineering Department
- <*italics*> between angular brackets indicate comments

1.3 Overview of the document

The next chapter, Overall Description, of this document gives an overview of the functionality of the product. It describes the informal requirements and also design and implementation constraints.

Third chapter, External interface requirements, describes all the interfaces like software interface, hardware interface and some communication interfaces required for full functionality of this product.

The fourth chapter, System features, describes all the use cases required for the developers. The fifth chapter describes other non-functional requirements like security and safety requirements.

1.4 Product Scope

This System is a web based software for Faculty Advisory system of NITC. This System is designed to computerize the current existing paper based system. It

provides tools for retrieving the personal data and data regarding the grades of individual students and also to get list of students with some filters.

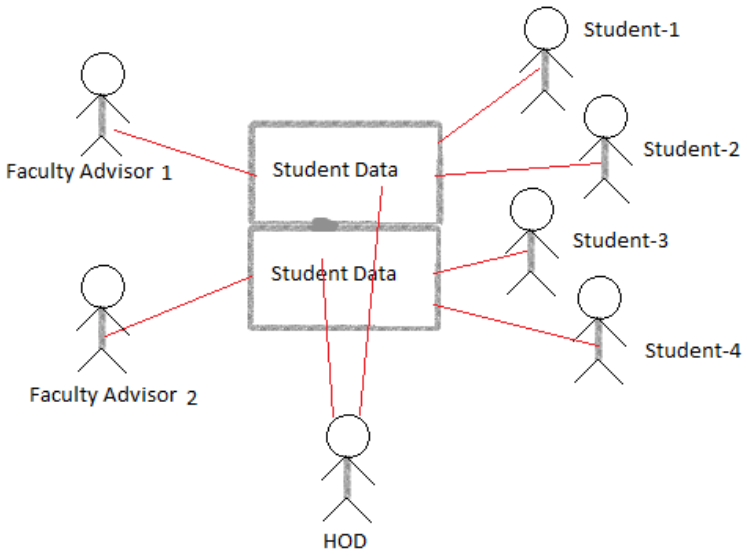
More specifically the system is designed to allow a Faculty Advisor to manage the data of all students under his advisoryship. The system contains relational database containing list of Students, Faculty and HODs.

1.5 References

- IEEE. *IEEE Std. 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998

2 Overall Description

2.1 Product Perspective



This System is a proposed replacement for the current existing manual Faculty Advisory system in which each student is assigned to a Faculty member of the same department as Faculty Advisor of that student and one of the faculty members will act as Head of the department, where each Faculty needs access to students' data who are

under his advisoryship where HOD can access data of all students in his department and student has access to his own data.

2.2 Product Functions

Major functions of this project include:

- Login to the individual portals through a username and a password.
- Enter and access all the data (personal and academic) of students
- Getting list of students satisfying a particular constraint
- Calculating CGPA and total number of credits earned by a student at a particular point of time.

2.3 User Classes and Characteristics

The system is mainly used by three classes of users namely

- Student can access only his/her data through the system
- Faculty Advisor can access/edit the data of all the students under his advisoryship.
- HOD can access data regarding all the students in his department.

2.4 Operating Environment

This system is being developed in Ubuntu flavor of Linux running on Intel core i5 machine and is being tested on Firefox. It must be compatible with all browsers that support PHP.

2.5 Design and Implementation Constraints

We basically have two Constraints with us as part of development

- Time is our first constraint because of which we are limiting our model to be basic and usable.
- Second constraint is the efficiency of languages like CSS for developing a web based application because of which all the web pages might not be well organized and attractive.

2.6 User Documentation

A document named User Manual will be available along with this product which can be used by all the user classes which this product is intended for.

2.7 Assumptions and Dependencies

Since we are not taking security part as part of this project as of now, we assume that all the logins and passwords will be encrypted by some third party software without which our databases can be altered by unauthorized persons which may lead to severe consequences. We also assume that this product can be linked to DSS of NITC at a later point of time failing of which may lead to inconsistencies in the data.

3 External Interface Requirements

3.1 User Interfaces

- Whenever user tries to access the information a login page is displayed.
- If the user tries to access/edit the information of his/her profile without his/her login details an error is displayed.
- Whenever the user gives correct login details when asked he/she can access/edit the information.
- The student can read the published data by the Faculty Advisor.
- The faculty advisor can get the students according to his/her requirements for example students below 5 CGPA etc.

3.2 Hardware Interfaces

- Hardware requirement for this project will be a system with Ethernet access.

3.3 Software Interfaces

- Server side: the server side system should hold the database supporting mysql
- Client side: the client side system should have a browser which supports html and PHP.

3.4 Communications Interfaces

- This database should be securely accessible through internet communication channels.

4 System Features (Functional Requirements)

4.1 Student

All the use cases of Student have been described here

4.1.1 Enter Details:

- **Use Case Name:** Update Details
- **Trigger:** Student adds/updates all his personal details
- He also uploads his photograph.
- Student works on a personal details web page to enter his details

4.1.2 Update Marks:

- **Use Case Name:** Update Marks
- **Trigger:** Updates his marks/grades corresponding to each course in each semester.
- Student works on a page displaying grades and marks from his personal login.

4.1.3 Register:

- **Use Case Name:** Register
- **Trigger:** Register for some courses under a faculty member at the beginning of each semester.
- Student will be working on the same page where he added grades to enter the course details he has enrolled to in each semester.

4.1.4 Read Messages:

- **Use Case Name:** Read Messages
- **Trigger:** Read all messages/announcements sent by the Faculty Advisor.

4.2 Faculty Advisor

All the functional requirements of Faculty Advisor are described here

4.2.1 Access Details

- **Use Case Name:** Access Details
- **Trigger:** Output the details of students given a Roll Number
- Faculty Advisor works on a page in which he enters a Roll Number and gets personal and grades details of the student.

4.2.2 Edit Data

- **Use Case Name:** Edit Data
- **Trigger:** Edit the Students' data.
- Faculty Advisor works on the same page where he got the details of the students and updates them as and where necessary.
- Information regarding Disciplinary Actions, Probations etc. can also be added here.

4.2.3 Get List

- **Use Case Name:** Get List
- **Trigger:** Displays the list of students satisfying the given filters.
- Faculty Advisor works on a different page where he sets some required filters and Gets the list of all students satisfying them.
- Filters may include
    - CGPA range (min-max)
    - Students having backlogs
    - Students who applied for condonation
    - Students having R grades
    - Students under probation etc.

4.2.4 Send Messages

- **Use Case Name:** Send Messages
- **Trigger:** Send a message or an announcement to all students
- Faculty Advisor sends messages through a separate page displaying messages.

4.3 HOD

HOD can have same use cases that a Faculty Advisor has but the Domain being the whole Department.

5 Other Nonfunctional Requirements

5.1 Basic Requirements

Faculty Advisors should be able to verify the information provided and updated by Students in the Product.

5.2 Performance Requirements

The system must be interactive and the delays involved must be less .So in every action-response of the system, there are no immediate delays. In case of opening windows forms, of popping error messages and saving the settings or sessions the delays should be minimized.

5.3 Safety Requirements

Information transmission should be securely transmitted to server without any changes in information.

5.4 Security Requirements

The main security concern is for users account hence proper login mechanism should be used to avoid hacking.

5.5 Software Quality Attributes

The additional characteristics for the Product are:

- **Portability:** The Product is portable from one system to other systems.
- **Reliability:** As the system provide the right tools for the accessing and updating the information, problem solving it must be made sure that the system is reliable in its operations and for securing the sensitive details.

- **Availability:** If the internet service gets disrupted while sending information to the server, the information needs to be send again.
- **Usability:** As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.