University Department Information System

Software Requirements Specification

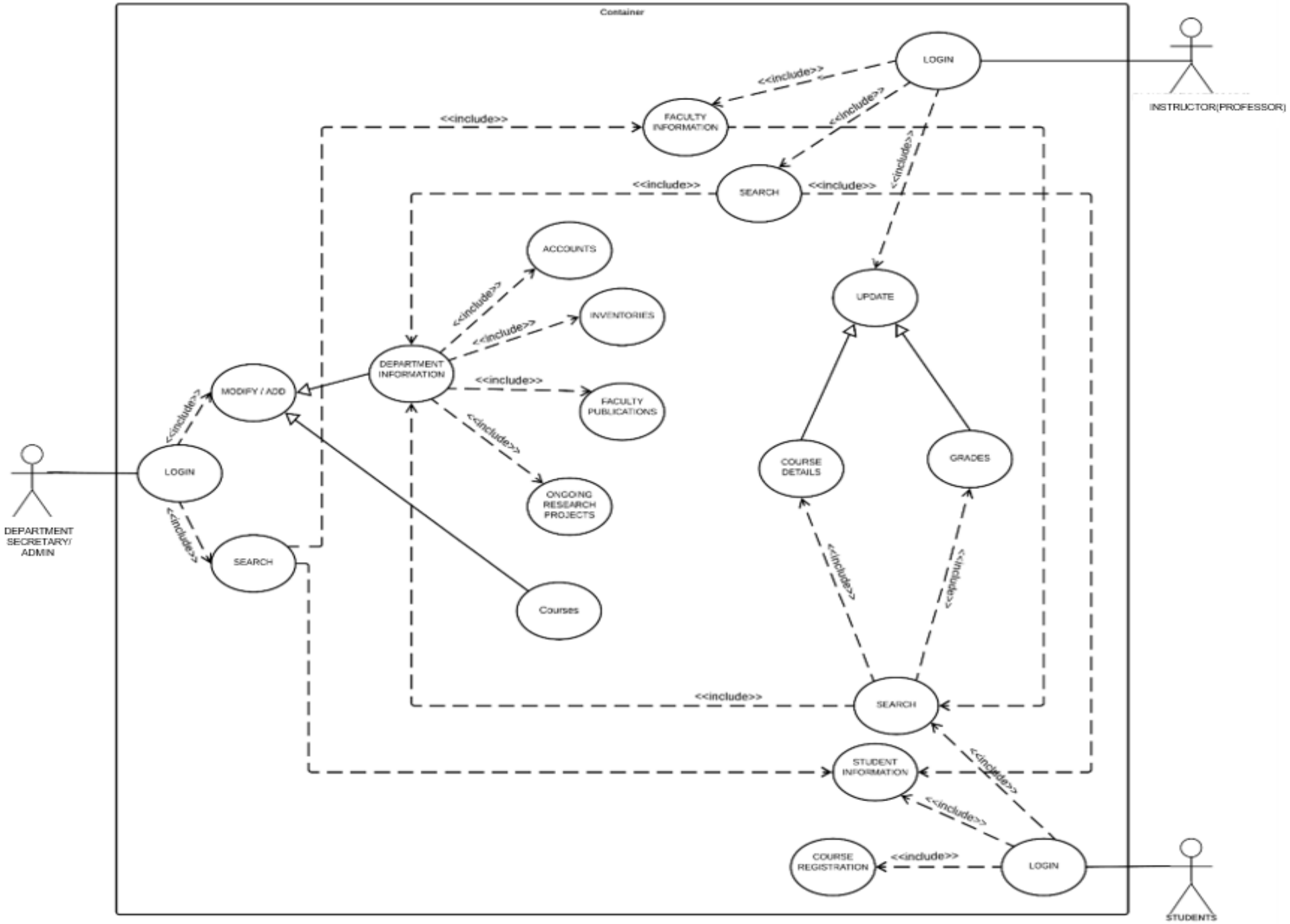
Prepared by

Anirudh M(21CS10004)
Aviral Pandey(21CS10089)
Mayukha Marla(21CS10041)

INDIAN INSITUTUTE OF TECHNOLOGY KHARAGPUR

Table of Contents

Use Case Diagram



1. SRS

Introduction

1.1. Purpose

The main objective of this document is to illustrate the requirements of the project **University Department Information system**. This document describes the design decisions, architectural design and the detailed design needed to implement the system. It provides the visibility in the design and provides information needed for software support. The document gives the detailed description of the both functional and nonfunctional requirements proposed by the client. The document is developed after considering the complete requirement specifications of the given Project. The final product of the team will be meeting the requirements of this document.

1.2. Document Conventions

The following are the list of conventions and acronyms used in this document and the project as well:

- **Administrator**: A login id representing a user with user administration privileges.
- **User:** A general login id assigned to users
- **Client:** Intended users for the software
- **PostgreSQL:** Relational Database Management System; used to retrieve information from a database
- **Layer:** Represents a section of the project
- **User Interface Layer:** The section of the assignment referring to what the user interacts with directly.
- **Application Logic Layer:** The section of the assignment referring to the Web Server. This is where all computations are completed.
- **Data Storage Layer:** The section of the assignment referring to where all data is recorded
- **Data flow diagram:** It shows the dataflow between the entities.

- **Use Case:** A broad level diagram of the project showing a basic overview
- **Boolean:** A true/false notation
- **Interface:** Something used to communicate across different mediums.
- **Unique Key:** Used to differentiate entries in a database

1.3 Scope

The Department Office Management System is a software system that automates book-keeping activities for the department offices in different universities. The system is designed to handle various tasks such as managing student data, course registration, grading information, department inventories, accounting information, research projects, and faculty publications. The system will enable the department to maintain accurate records, eliminate paperwork, and improve efficiency.

1.1 Product Perspective

The proposed **University Management System** is an online University Management System. This System will provide a view, submit, online accounting, uploading various documents and other miscellaneous resources. This view will be based on the categories like grade sheet view and daily activities. Further the University management staff personnel(instructors) can add/update/remove resources.

The System will also have an ADMIN(Department Secretary) who has full-fledged rights with regards to managing resources across branches. The users can view, submit, upload various documents and information about their account etc. There are basic two types of users:- students and faculty members(instructors). Each user is facilitated with a different account number having a profile along with a password for private use. The two types of users differ from each other due to the accessing limits to online University Department management system.

1.2 Product Features

There are three different users who will be using this product:

- Department Secretary who will be acting as the administrator.
- Instructors who are second level users accessing UDIS.
- Student of the University who will be accessing the UDIS online.

The features that are available to the Administrator are:

- The administrator has the full-fledged rights over the UDIS.
- Can create/delete an account.
- Can view the accounts.
- Can hide any kind of features from the both of users.
- Insert/delete/edit the information of available on UDIS.
- Can access all the accounts of the instructors/students.

The features available to the instructors are:

- Can upload marks and grades for students.
- Can upload various research papers, publications etc of the department

The features available to the Students are:

- Can register for courses at the beginning of each semester.
- Can view their marks and grades.
- Can view the various publications, research papers and projects.
- Can view and modify their profile but can modify it to some limited extent only.

1.3 User Classes and Characteristics

There are various kinds of users for the product. Usually web products are visited by various users for different reasons.

The users include:

- Secretary who will be acting as the controller and he will have allthe privileges of administrator.
- Instructors who will be using the above features by accessing the UDIS online.
- Students who will be using the above features by accessing theUDIS online.

1.4 Operating Environment

The product can be operated in windows, macOS and most Linux distributions. Most of the features will be compatible with the Chrome, Mozilla Firefox and Opera 7.0 or higher version. The only requirement to use this online product would be the internet connection.

1.5 Design and Implementation Constraints

The Product is developed using Django(Python 3.10 or higher). The backend database for this is PostgreSQL. Frontend designing has been achieved using HTML, CSS and JS. The product is accomplished with login facility so that specific function is available to specific type of user.

1.6 User Documentation

The product will include user manual. The user manual will include product overview, complete configuration of the used software (such as PostgreSQL), technical details, backup procedure and contact information which will include email address. The product will be compatible with Google Chrome, Mozilla Firefox and Opera. The databases will be created in PostgreSQL version 15.2 or higher.

1.7 Assumptions and Dependencies

The product needs following third party product.

- PostgreSQL to store the database.
- Django to develop the Product

2. System Features

3.1. Database – Storage

3.1.1. Description and Priority

Proposed Database is intended to store, retrieve, update, and manipulate information related to university which include

- Profile of both users
- Staff information
- Student details

- My account
- View marks/uploading of marks.
- View Cash book account/upload transactions

3.1.2. Stimulus / Response Sequences

Responses for Administrator: The administrator can Login and Logout. When the Administrator Logs into the University information system, the system will check for validity of login. If the Login and password are valid, the response to this action is the administrator will be able to modify, view, add, deleting and all other functions that can be performed on the database.

3.2. Functional Requirements

This section gives the list of Functional and non-functional requirements which are applicable to the University Management System.

3.2.1 Interface Requirements

This section describes how the software interfaces with other software products or users for input or output.

3.2.1.1 User Interfaces

This section briefly describes how this product interfaces with the user.

GUI

Describes the graphical user interface if present.

1. **Description**

   The user interface must be customizable by the administrator

2. **Criticality**

   This issue is essential to the overall system. All the modules provided with the software must fit into this graphical user interface and accomplish to the standard defined.

3. **Technical issues**

   In order to satisfy this requirement, the design should be simple and all

the different interfaces should follow a standard template. There will be the possibility of changing colors and images, plus switching between interfaces with the minimum impact for the users.

4. **Risks**

   To reduce the circumstances under which this requirement might not able to be satisfied, all the designers must have had developed web sites previously and they must be aware of html restriction and cross browsers implementations before starting the designing. In order to reduce the probability of this occurrence the entire design team will be trained in basic html development and macromedia fireworks since this tool might be needed instead of Photoshop.

5. **Dependencies with other requirements**

   All user interfaces should be able to interact with the user management module and a part of the interface must be dedicated to the login/logout module.

Input Requirements

User access

Each instructor and student are assigned a unique id upon admission to the university. Both must know this. This identifying key maps to all his/her registration record information in the main registration system. Admitted and current students have their online registration accounts also enabled. Such account maybe disabled during his/her stay as a matriculated student and/or after graduation or separation from the university.

Uploading of data

Each faculty member should facilitate with uploading of data such as marks, publications and other kind of reading material. Similarly,such options must be there for students to view their grades.

Accounts

The secretary should have the facility to keeps track of the Department accounts and add transactions such as yearly grant from the University, funds from different consultancy service the department provides to different organizations etc.

3. Non-Functional Requirements

4.1. User Interfaces

4.2. Hardware Interfaces

Server Side:

- Operating System: Windows 10/11, MAC or UNIX
- Processor: Pentium 3.0 GHz or higher
- RAM: 256 Mb or more
- Hard Drive: 10 GB or more

Client side:

- Operating System: Windows 10/11, MAC or UNIX.
- Processor: Pentium 3.0 GHz or higher.
- RAM: 256 Mb or more

4.3. Software Interfaces

- **Database:** PostgreSQL 15.2 or higher.
- **Application:** Django 4.1 or higher (Python 3.10 or higher).
- **Web Server:** IIS (Internet Information Services (IIS) is a powerful Web server that provides a highly reliable, manageable, and scalable Web application infrastructure)

4.4. Communications Interfaces

The Customer must connect to the Internet to access the Website:

- Dialup Modem of 52 kbps
- Broadband Internet

- Dialup or Broadband Connection with a Internet Provider.

6.Other Nonfunctional Requirements

5.1. Performance Requirements

The proposed system that we are going to develop will be used as the Chief performance system within the different campuses of the university which interact with the university staff and students. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the university.

5.2. Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

5.3. Security Requirements

We are going to develop a secured database for the university .There are different categories of users namely Secretary, Instructors and students. Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, delete, append etc. All other users other than University Staff only have the rights to retrieve the information about database.

5.4. Software Quality Attributes

The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database.

5.5 Hardware Constraints

The system requires a database in order to store persistent data. The database should have backup capabilities.

5.6 Software Constraints

The development of the system will be constrained by the availability of required software such as web servers, database and development tools.

The availability of these tools will be governed by the Lovely Professional University.

5.7 Design Constraints

The system must be designed to allow web usability. That is, the system must be designed in such a way that will be easy to use and visible on most of the browsers.