ACADEMIC TASK-1

CSE320

(SOFTWARE ENGINEERING)

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Name: Pankaj Bashera

Registration number: 12314197

Roll No.: RK23SBA04

Section: K23SB

Submitted to :

Dr. Nikita

LOVELY PROFESSIONAL UNIVERSITY

DECLARATION

I, Pankaj Bashera , a student of Bachelor of Technology under CSE discipline at Lovely Professional University, Punjab, hereby declare that all the information furnished in this Specific Requirements Specifications (SRS) is based on my own work and is genuine.

Pankaj Bashera

Registration Number: 12314197

INDEX

4. Other Non-Functional Requirements.
   4.1 Accessibility.
   4.2 Alternate Text Description.
   4.3 Keyboard Navigation.
   4.4 Screen Reader Compatibility.

5. System Design
   5.1 First Level DFDs.
   5.2 Second Level DFDs
   5.3 Zero Level DFDs.
   5.4 UML

1. Introduction

1.1 Purpose

The main objective of this document is to illustrate the requirements of the main project University Management System (UMS). This document describes the design decision, architectural design and the detailed design needed to implement the system. It provides the visibility in the design and provide the information needed for the software support. The document gives the detailed description of both functional and non-functional requirements proposed by the client.

1.2. Intended audience and Reading Suggestions

The document is intended for all the stakeholder's customer and the developer. The reader is assumed to have basic knowledge of all the algorithm used to reduce the complexity and also have knowledge of all the basics which are used in the development and maintainence of the project or an online system and also some basic knowledge of Data Flow Diagrams.

1.3 Definitions and Abbreviations.

The following are the list of conventions and acronyms used in the documentation.

- Admin : A login id representing a user with administration privileges to the software.
- User: A general login id assigned to users
- Client: Intended users for the software.
- SQL : Structured Query Language.

- ASP - Active Server Pages: A webpage formatted on the server and delivered to the browser
- User Interface Layer - The section of the assignment referring to what the user interacts with directly.
- Application layer - This section referring the web server where all computations are completed
- Data Storage Layer - where all the data is being stored
- Data Flow Diagram - DFDs - shows the dataflow between the entities
- Boolean - A true/false statement
- Interface - Something used to communicate across different mediums
- Unique Keys: Used to differentiate entities in a database
- Layers: Represents the section of the project

1.4 Scope of Project

The UMS is developing for university to replace the old paper work system. The software supports a computerized university management system network. The network enables Teachers, students to complete simple tasks via UMS. It identifies a USER by login id which is provided by the administrator and a password. It collects information through the database by following the login id (e.g., profile, attendance, examination, fees payment). The software must handle concurrent accesses to the same account correctly

2. Overall Description

2.1 Product Perspective

The proposed UMS will provide a view to submit online payment, uploading various documents and other resources. This view will differ from user different authorities. The staff (faculty) can add/remove/update the resource or an automatic removal of accessing features when the time limit completes. The system has an ADMIN who have full fledged rights with regards to managing resources across branches. While the student user can view, submit online payment, uploading documents and information about their account.

2.2 Product Feature

Every type of USER will have different interfaces.

- → University chancellor - who will be acting as the ADMIN.
- → Faculty Members - who are second level users accessing UMS
- → Students - who will be accessing the UMS online.

The features available for the ADMIN are:

- Creation/Deletion/Projection of accounts.
- Change the Password
- Can hide any kind of feature from both users
- Insert/delete/edit the information on the UMS.
- Access to all the accounts of faculty and students.

The features available for the Faculty are:-

- Mark the attendance of students online.
- Create continuous assessments for students.

- Can view the attendance of the students.
- Can submit the question papers online
- Can upload marks, assignments, reading material for the students.

The features available for Students are:-
- Can view reading materials, assignments, marks and their attendance.
- Can view or modify profile to a limited extent

2.3 User classes and characteristics

The user include -
- Chancellor who will be acting as the controller and will have all the priveleges of administrator
- Faculty members who will be using the above features by accessing the UMS online.
- Students who will access the UMS online.

2.4 Operating Environment

The product will be operating in Windows environment. Also it will be compatible with the IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. The only requirement to use this would be internet.

2.5 Design and Implementation Constraints.

The product is developed using ASP. The backend database for this SQL Server. The product is accomplished with login facility so that specific function is available to specific student.

3. Specific Requirements

3.1 Database Storage

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

- Profile of both users
- Student details
- Online Payment
- Staff information
- My account
- View Attendance / marks

It is also used for validation of user login in the UMS.

3.2 Interface Requirements

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

A. User Interfaces:-

- Web Interface - A user friendly website accessible via standard web browsers (e.g., Chrome, Safari, Safari, Firefox)
- Mobile application Interface - A mobile application carrying out certain functionalities of UMS (e.g., LPU Touch, Live)

B. Hardware Interfaces:-

- OS: Windows / Mac / UNIX
- Processor: Pentium 3.0 GHz or higher
- RAM: 256 MB or more
- Hard Drive: 10 GB or more.

C. Software Interfaces:-

- Database: SQL Server
- Application: ASP (Active Server Pages)

D. Communication Interfaces:-

- Network Protocols - Standard Protocols (HTTP or HTTPS) for communication between client and server.
- APIs - Facilitate Integration with 3rd party systems.

3.3 Functional Requirements

F. Requirement 1: User Authentication:

- Input : User provides username and password.
- Output : Grants access to user on successful authentication
- Processing: The system verifies the provided credentials against the stored user database.

FR2: Course Registration:

- Input : Student selects courses from the available offerings.
- Output : The system confirms course registration and updates the student's schedule.
- Processing: The system checks the student's eligibility for selected courses and updates enrollment records.

FR3: Fee Payment Integration:

- Input : Student selects the fee payment option.
- Output : The system generates the payment receipt and updates the student's financial records.
- Processing: The system securely processes the payment transaction through integrated payment gateways.

FR4: Grade Submission:

- Input : Faculty enter grades for enrolled student.

- Output: The system stores the submitted grades, updates student records and generates grade reports.
- Processing: The system validates grade entries and calculates final grades and GPAs.

FR5: Attendance Tracking:
- Input: Faculty records student attendance for each class session.
- Output: The system updates attendance records and notifies students of their attendance status.
- Processing: The system processes attendance data and calculates attendance percentage.

FR 6: Reporting and Analytics:
- Input: Admin requests specific reports or analytics
- Output: The system generates and present reports or analytics in a readable format for the ADMIN.
- Processing: The system retrieves relevant data from the database and performs analysis.

FR7: Email notifications:
- Input: System triggers events requiring notifications/deadlines.
- Output: User receives email notifications containing reminders.
- Processing: The system sends automated emails to relevant users.

FR8: Course Management:
- Input: Faculty updates course information (eg. syllabus, schedule)
- Output: The system reflects updates for students to access.
- Processing: The system validates and applies course updates.

FR9: User Profile Management:

- Input: User updates profile data.
- Output: The system confirms successful profile updates to the user.
- Processing: The system validates and updates user profile information.

FR10: System Configuration:

- Input: ADMIN configures system settings (eg. academic calendars)
- Output: The system confirms successful configuration changes and updates system settings accordingly.
- Processing: The system applies administrator-defined config.

FR11: RMS:

- Input: User inputs any grievances, enquiries, feedback etc.
- Output: The system validates the query and generates the RMS slip.
- Processing: The system validates the chosen options from the database.

3.4 Non-functional Requirements.

NFR1: User Authentication:

- Performance: The system shall authenticate users within 3 seconds under normal load conditions.
- Security: User credentials shall be encrypted during transmission and storage to ensure data confidentiality.
- Reliability: The authentication system shall have a uptime of atleast 99.9% to ensure continuous availability.

NFR2: Course Registration:

- Scalability: The system shall support simultaneous course

registration for atleast 1000 users without performance backlash.
- Usability: The course registration interface shall be intuitive and accessible, complying with WCAG standards for accessibility.

NFR 3: Fee Payment Integration:
- Security: Payment transactions shall be processed using PCI-DSS compliant system, Payment gateways to ensure secure handling of financial data.
- Reliability: The payment gateway integration shall have a uptime of at least 99.95% to prevent disruption in fee payment services

NFR 4: Grade Submission:
- Data Integrity: The system shall ensure the accuracy and integrity of grade submission through validation checks and audit trails.
- Performance: Grade submission processing time shall not exceed 5 seconds per student record to maintain efficient workflow.

NFR 5: Attendance Tracking:
- Scalability: The system shall efficiently handle attendance tracking for classes with upto 500 students without performance backlash.
- Accessibility: The attendance tracking interface shall be accessible and user-friendly accomodating users with disabilities.

NF 6: Reporting and Analytics:
- Performance: Report generation and analytics processing shall be completed within 10 seconds for standard queries.
- Compliance: The system shall adhere to data privacy regulations when generating reports containing sensitive information.

NFR 7: Email Notifications:

- Reliability: Email shall be delivered promptly with a delivery success rate of at least 99%.
- Usability: Email shall be clear and informative, with relevant subject lines and content to avoid confusion.

NFR 8: Course Management:

- Backup and Recovery: Course management data shall be regularly backed up, with a backup retention period of atleast 6 months to prevent data loss.
- Security: Only authorized faculty members shall have access to course management functionalities to prevent unauthorized modifications.

NFR 9: User Profile Management:

- Privacy: User profile information shall be stored securely and accessible only to authorized personnel to protect user privacy.
- Usability: The user profile management interface shall be intuitive and easy to navigate, allowing users to update their information effortlessly.

NFR 10: System configuration:

- Auditability: System configuration changes shall be logged and auditable, with logs retained for at least 1 year for compliance and accountability purposes.
- Reliability: System configurations shall be applied consistently across all system instances to ensure uniformity and stability.

4. Other Non-Functional Requirements.

4.1 Accessibility.

→ Ensures that UMS is designed and developed to be usable by individuals with disabilities. It involves compliance with WCAG (Web Content Accessibility Guidelines) to provide equal access to all users, regardless of disabilities.

→ Increases perceivability, operability, understandability and robustness.

4.2 Alter Text Description

→ AlterTD provide textual descriptions for non-text content such as images, graphics, and multimedia elements.

→ Essential for users who rely on screen readers or other assistive technologies to access and understand the content UMS

4.3 Keyboard Navigation.

→ Allows users to navigate through the UMS using keyboard input alone, without relying on a mouse. This feature is essential for users with mobility impairments who may have difficulty using a mouse or other pointing devices.

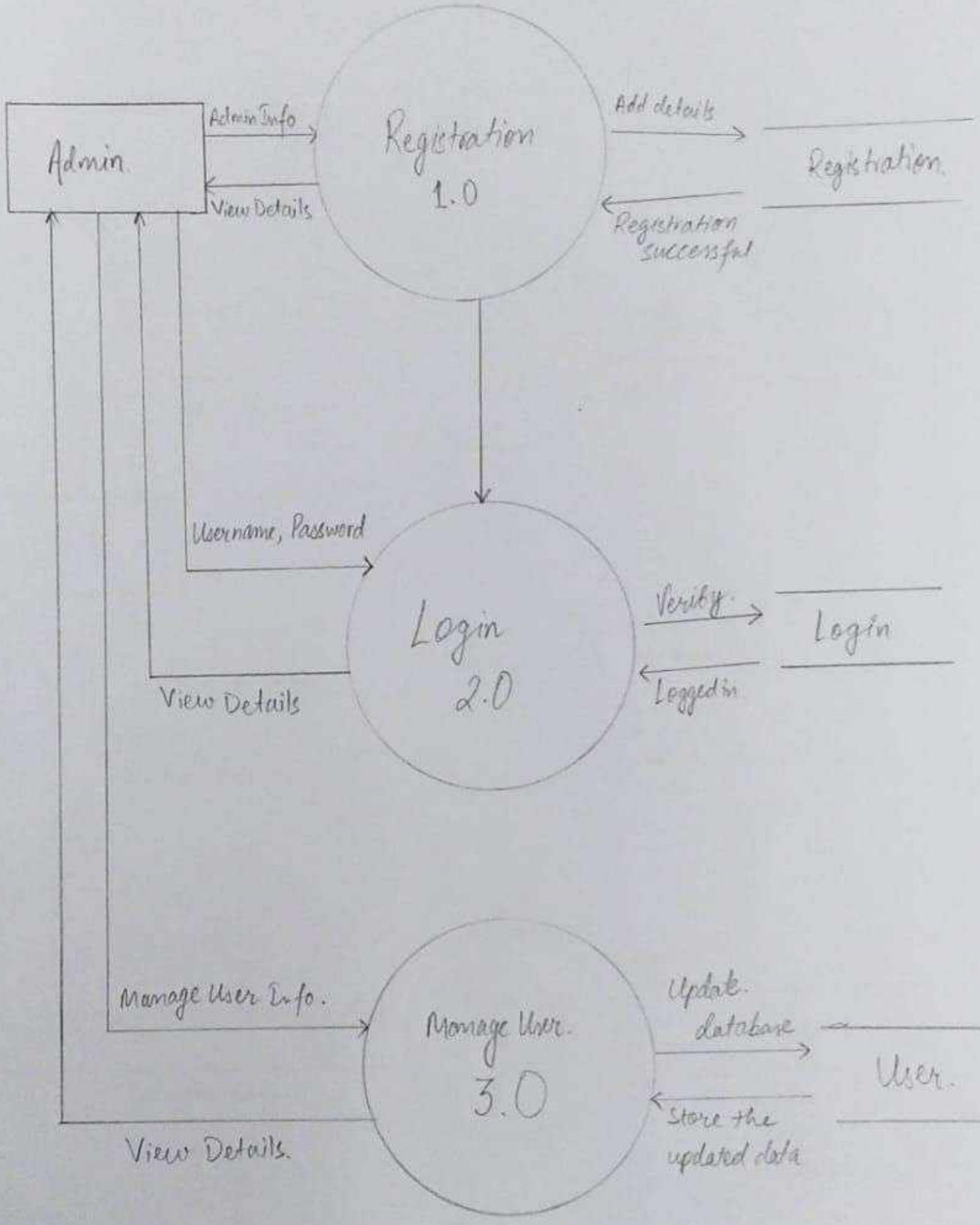
→ Keyboard focus indicators shall be clearly visible on focused element.

4.4 Screen Reader Compatibility.

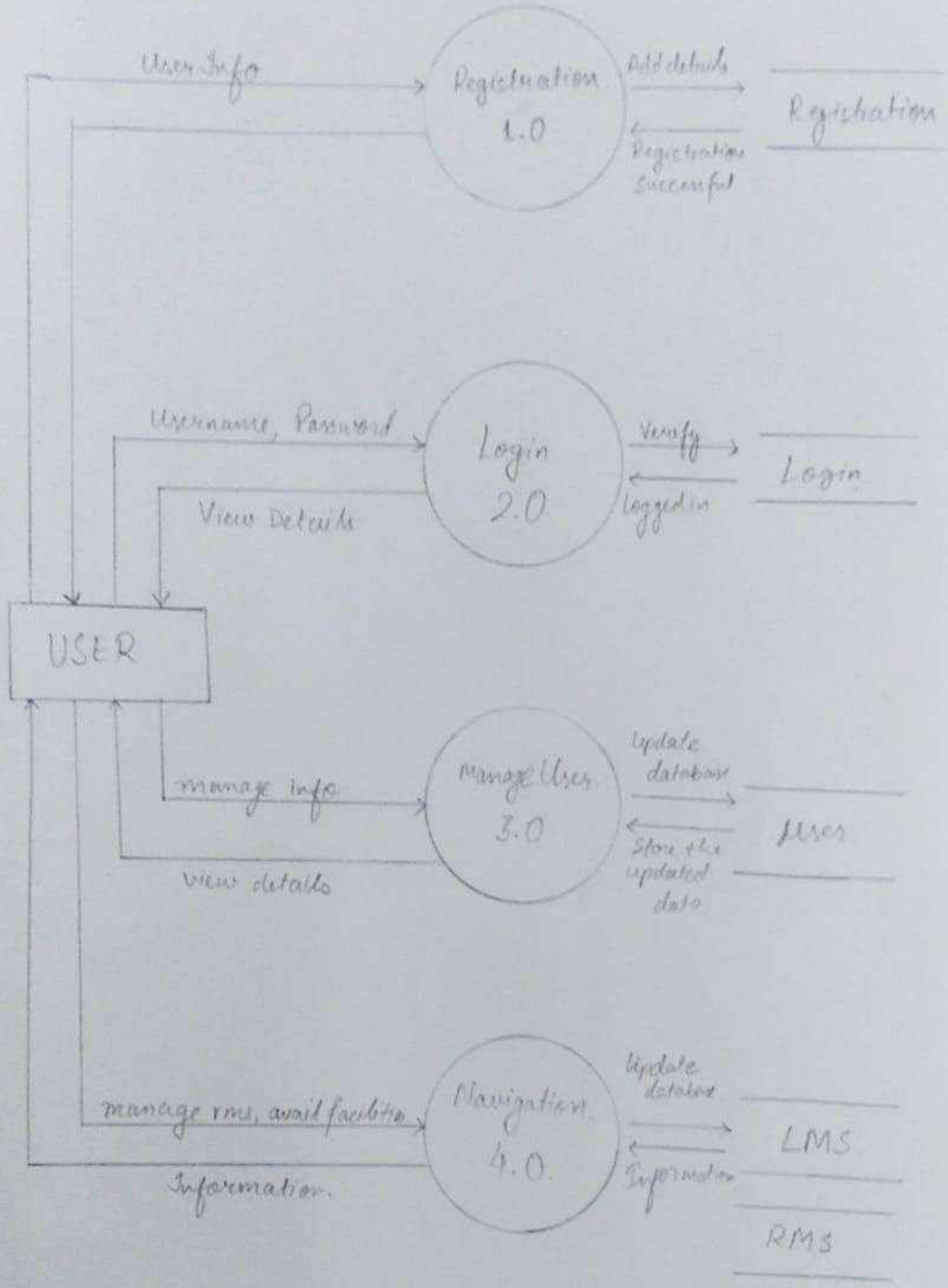
→ Screen Reader compatibility ensures that the UMS interface is compatible with screen reader softwares for users with visual impairments to access and interact with the system using auditory feedback provided by the screen reader.

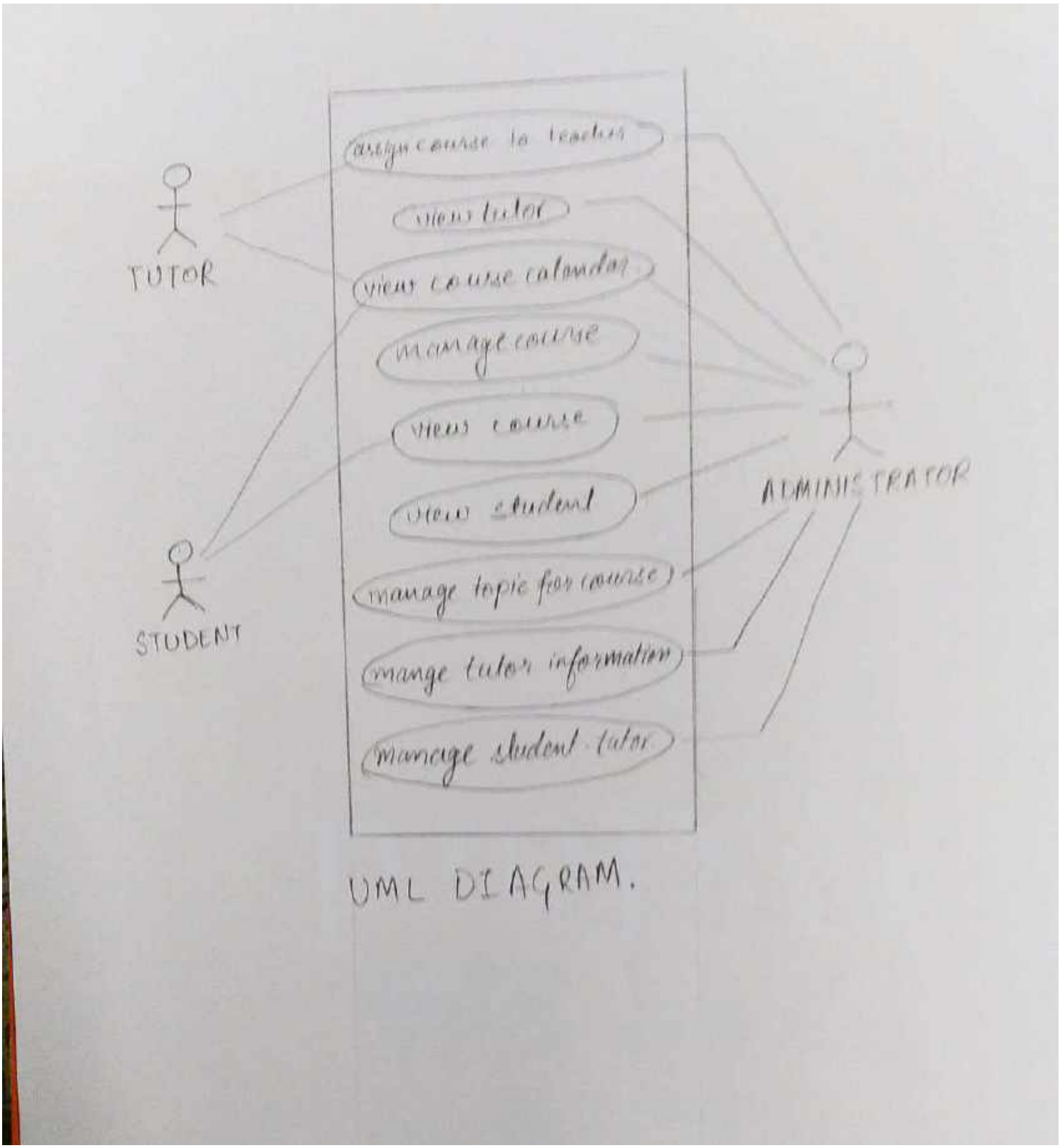
5. System Design.

Level 0

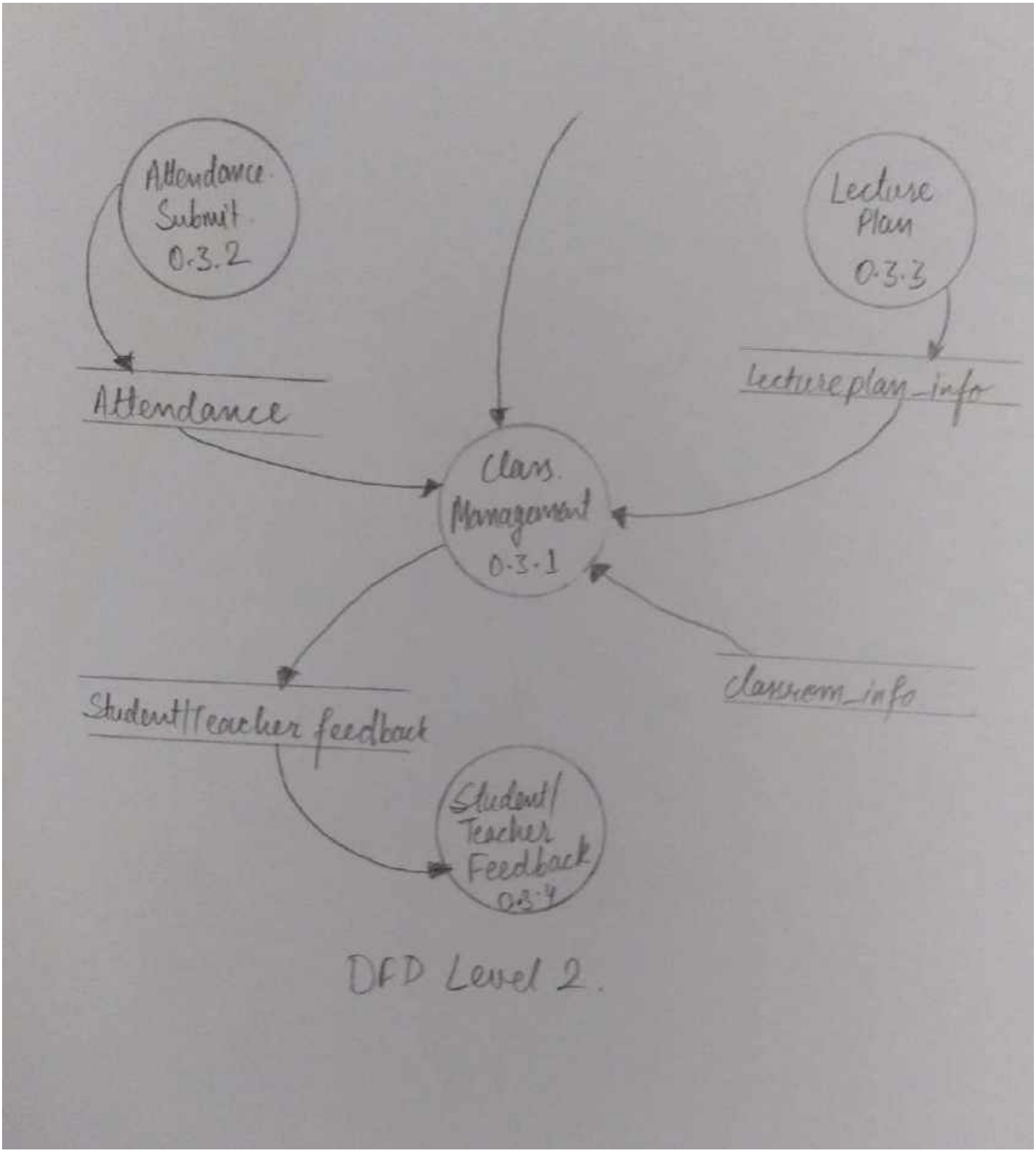


Level 0

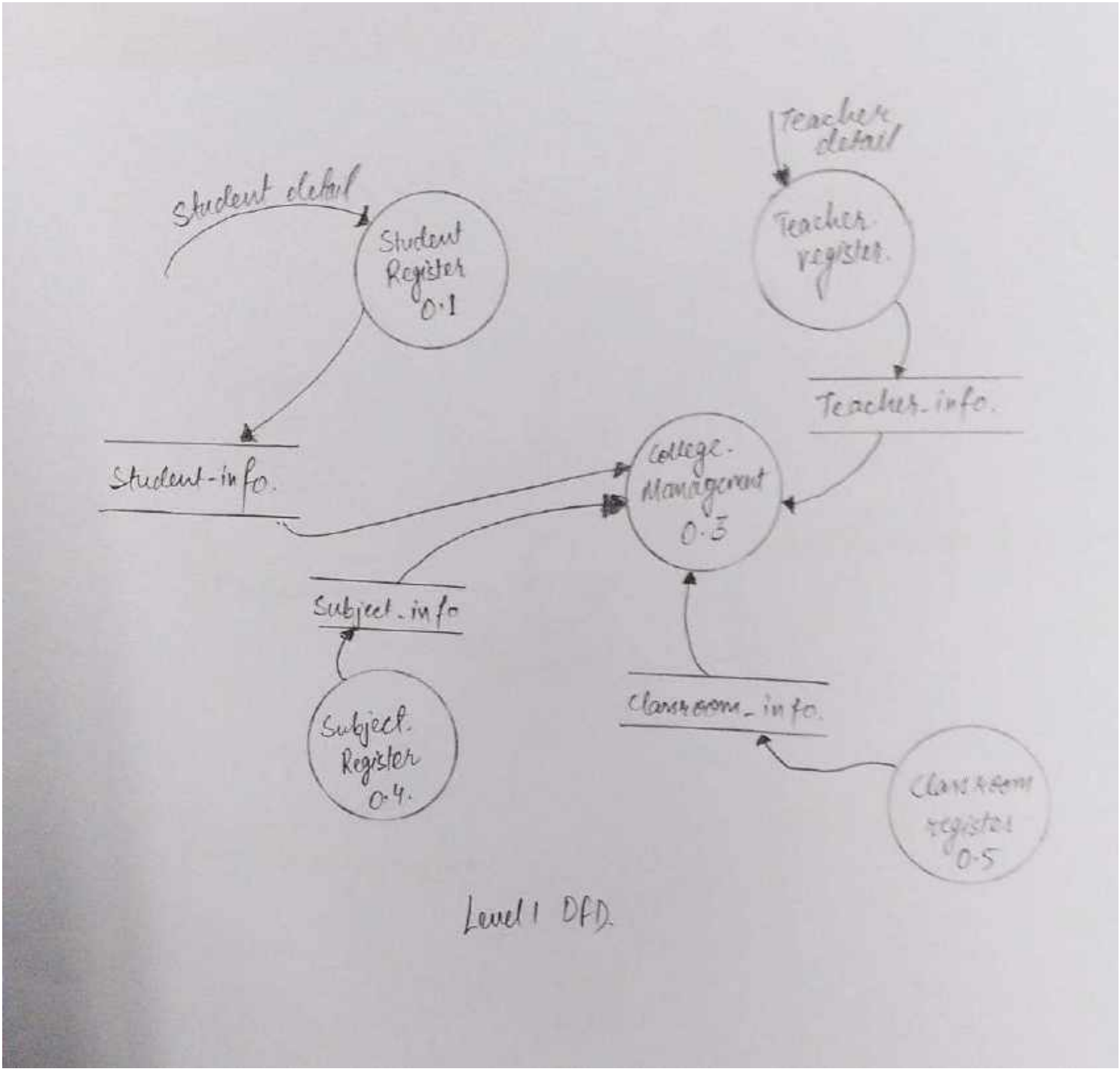

User Info
Registration 1.0
Add details
Registration
Registration Successful
Registration
Username, Password
Login 2.0
Verify
Login
logged in
View Details
USER
manage info
Manage User 3.0
Update database
User
Store the updated data
view details
manage rms, avail facilities
Navigation 4.0
Update database
LMS
Information
RMS


assign course to teacher
view tutor
view course calendar
manage course
view course
view student
manage topic for course
mange tutor information
manage student tutor
TUTOR
STUDENT
ADMINISTRATOR

UML DIAGRAM.


Attendance Submit 0.3.2
Lecture Plan 0.3.3
Attendance
lectureplan_info
Class Management 0.3.1
Student/Teacher feedback
classroom_info
Student/ Teacher Feedback 0.3.4

DFD Level 2.


Teacher detail
Student detail
Student Register 0·1
Teacher register
Teacher_info.
Student-info.
College Management 0·3
Subject_info
Classroom_info.
Subject Register 0·4.
Classroom register 0·5

Level 1 DFD

6. Testing

TID	Input	State	Output
1.	RMS	Active	Addition on Requests
2.	Log in details.	Active.	Open Website.
3.	Pay Fees.	Active.	Fee Slip
4.	Upload Assignment	Active.	Confirm dialog box
5.	Hostel leave - Submit	Active	Leave Slip generated
6.	View Seating Plan	Active	Seating plan opened
7.	Change Password	Active	Confirm updation
8.	Pay Fees	Inactive	Un-authorized access
9.	View Syllabus	Active	Open Syllabus PDF
10.	Hostel Booking	Inactive	Un-authorized acc
11.	View Attendance	Active	Attendance Sheet
12.	LPU - Live.	Active	Open LPU-Live