PROJECT REPORT

Thesis Management System (TMS)

Course Instructor

Mr. Asif Muhammad

Submitted By

Abdullah (22i-2515)

Shoaib Mehmood(22i-2448)

Spring 2024

Department of Computer Science

FAST – National University of Computer & Emerging Sciences, ISB

Table of Contents

Project Description

This project delves into the domain of educational technology, specifically focusing on enhancing the master's thesis process at FAST-NUCES. The project aims to develop a web-based thesis management system (TMS) to streamline communication, document management, and feedback mechanisms for stakeholders involved in thesis research (Al-Samarrai et al., 2020). The current process often relies on manual paperwork and email communication, leading to inefficiencies and potential data inconsistencies (Wong & Cheung, 2021).

Functional Requirements for Each Module

User Management

- **Secure User Registration and Login:**

  - Users must create accounts with unique usernames and strong passwords meeting complexity requirements (e.g., minimum length, combination of uppercase and lowercase letters, numbers, and symbols).

  - Password hashing should be implemented to store passwords securely, never storing them in plain text.

  - Two-factor authentication (2FA) can be offered as an optional layer of security.

  - Email verification should be required to confirm user accounts and prevent unauthorized access.

- **Role-Based Access Control (RBAC):**

  - The system should define distinct user roles (e.g., student, supervisor, examiner, MSRC member, academic staff, administrator) with assigned permissions.

  - Permissions should clearly define what functionalities each user role can access within the system.

  - For example, students can only access functionalities related to submitting documents, while supervisors can access functionalities for reviewing documents and providing feedback.

- **User Profile Management:**

  - Users should be able to view and update their personal information within the system, such as name, email address, and affiliation.

  - The system should allow users to change their passwords periodically to maintain account security.

Document Management System

- **Secure Online Storage:**

- The system should provide secure storage for thesis documents uploaded by students.

- Encryption at rest and in transit should be implemented to protect data confidentiality.

- Access controls should be enforced to ensure only authorized users can access specific documents based on their roles.

- **Version Control:**

  - The system should track all changes made to documents over time, allowing users to revert to previous versions if necessary.

  - Version history should include timestamps, user information, and a brief description of the changes made.

  - Users should be able to easily compare different versions of a document.

- **User Permissions for Document Access and Modification:**

  - The system should define permissions for document access and modification based on user roles.

  - Students should be able to upload, edit, and download their own thesis documents.

  - Supervisors and examiners should be able to access and review documents assigned to them.

  - Permissions can be further granular, allowing control over specific actions like adding comments, deleting sections, or approving final versions.

Communication Tools

- **Secure Forum:**

  - The system should provide a secure platform for threaded discussions between students, supervisors, and committee members.

  - Access to forums should be controlled based on user roles and thesis assignments.

  - Features like message threading, quoting, and search functionalities should be implemented for efficient communication.

- **Real-Time Notification System:**

  - The system should deliver real-time notifications to users regarding important updates and deadlines.

  - Notifications can be sent via email or in-app messages depending on user preferences.

> ➢ Notifications can include information about document submissions, feedback received, upcoming deadlines, and committee meetings.

Feedback Tools

- **Online Tools for Providing and Receiving Feedback:**

  > ➢ The system should offer tools for supervisors and examiners to provide detailed feedback on submitted documents.

  > ➢ Annotation capabilities should allow highlighting specific sections of the document, adding comments, and suggesting edits.

  > ➢ Feedback should be timestamped and attributed to the respective user for clarity.

- **Option for Students to Respond to Feedback:**

  > ➢ Students should be able to view feedback provided on their documents and respond to comments or questions raised.

  > ➢ The system should facilitate discussions between students and supervisors/examiners for clarification and improvement.

Progress Tracking

- **Personalized Dashboard:**

  > ➢ Each user should have a personalized dashboard displaying their current thesis progress.

  > ➢ The dashboard should provide an overview of submitted documents, completed milestones, and upcoming deadlines.

  > ➢ Visual elements like progress bars or timelines can be used to effectively communicate progress.

- **Clear Display of Deadlines and Milestones:**

  > ➢ The system should clearly display all deadlines and upcoming milestones associated with the thesis process.

  > ➢ Deadlines can be categorized by type (e.g., submission deadlines, feedback deadlines, committee meeting deadlines).

  > ➢ Users should be able to set reminders or receive automated notifications for approaching deadlines.

Reporting Tools
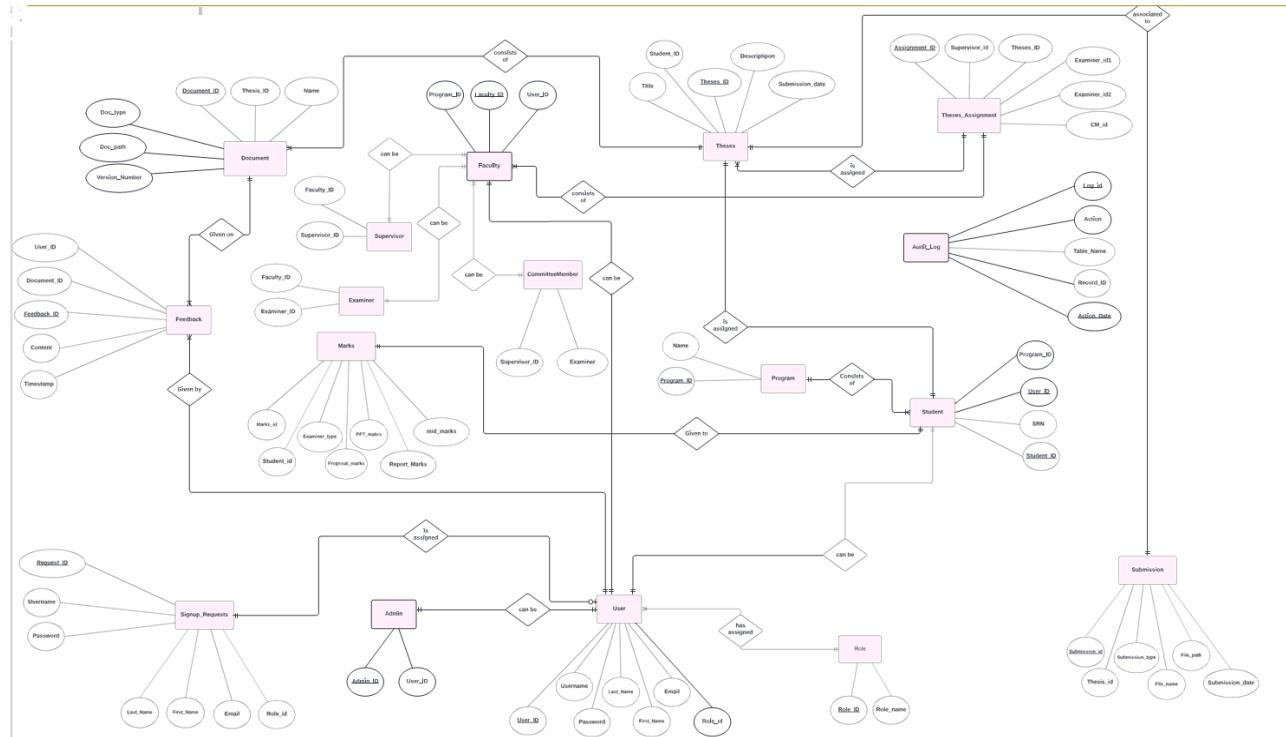
- **Ability to Generate Reports:**

  > ➢ The system should offer the ability to generate reports on various aspects of the thesis process, subject to further analysis and user needs.

- ➢ Reports can provide insights into student progress rates, committee activity levels, and overall thesis workload distribution.

- ➢ Different report types can be offered, such as individual student reports, committee reports, and program-wide reports.

- ➢ Filters and export options should be available to allow for detailed analysis and data sharing.
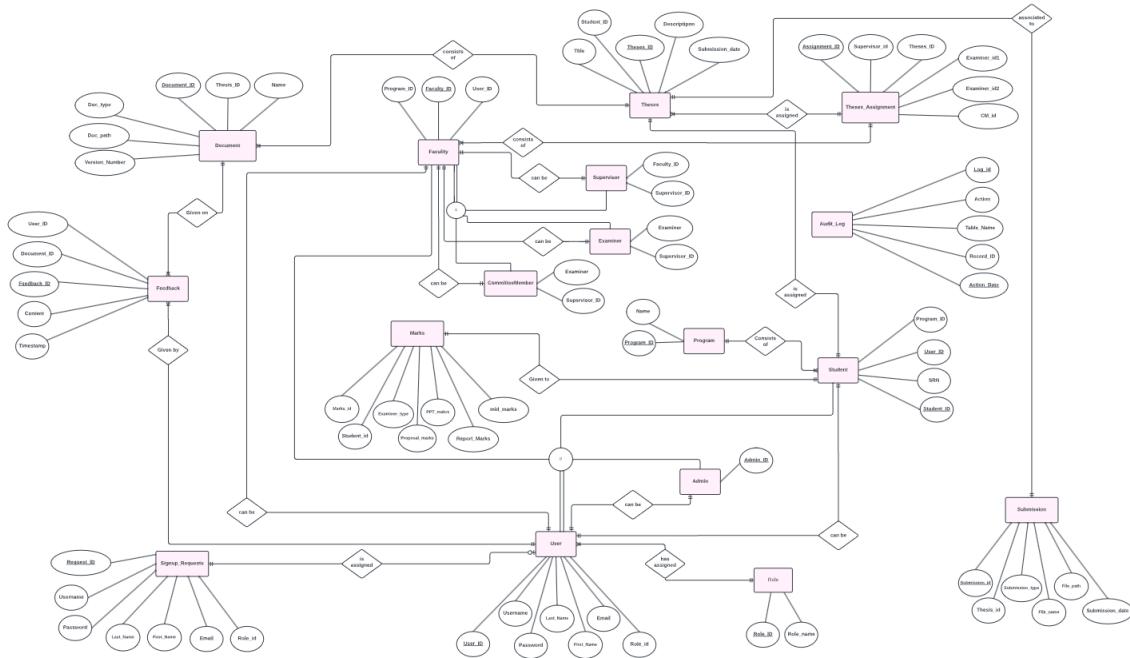
Committee Management

- **Scheduling and Communication:** The system should provide tools for scheduling committee meetings, including features for:

  - ➢ Setting dates and times for meetings.

  - ➢ Inviting relevant committee members.

  - ➢ Managing conflicts and finding suitable meeting times for all attendees.

  - ➢ Sending meeting agendas and materials electronically.

- The system should also facilitate communication between committee members before, during, and after meetings. This could include:

  - ➢ Secure chat functionality for real-time communication.

  - ➢ File sharing capabilities for exchanging documents and resources.

  - ➢ Option to record meeting minutes electronically.
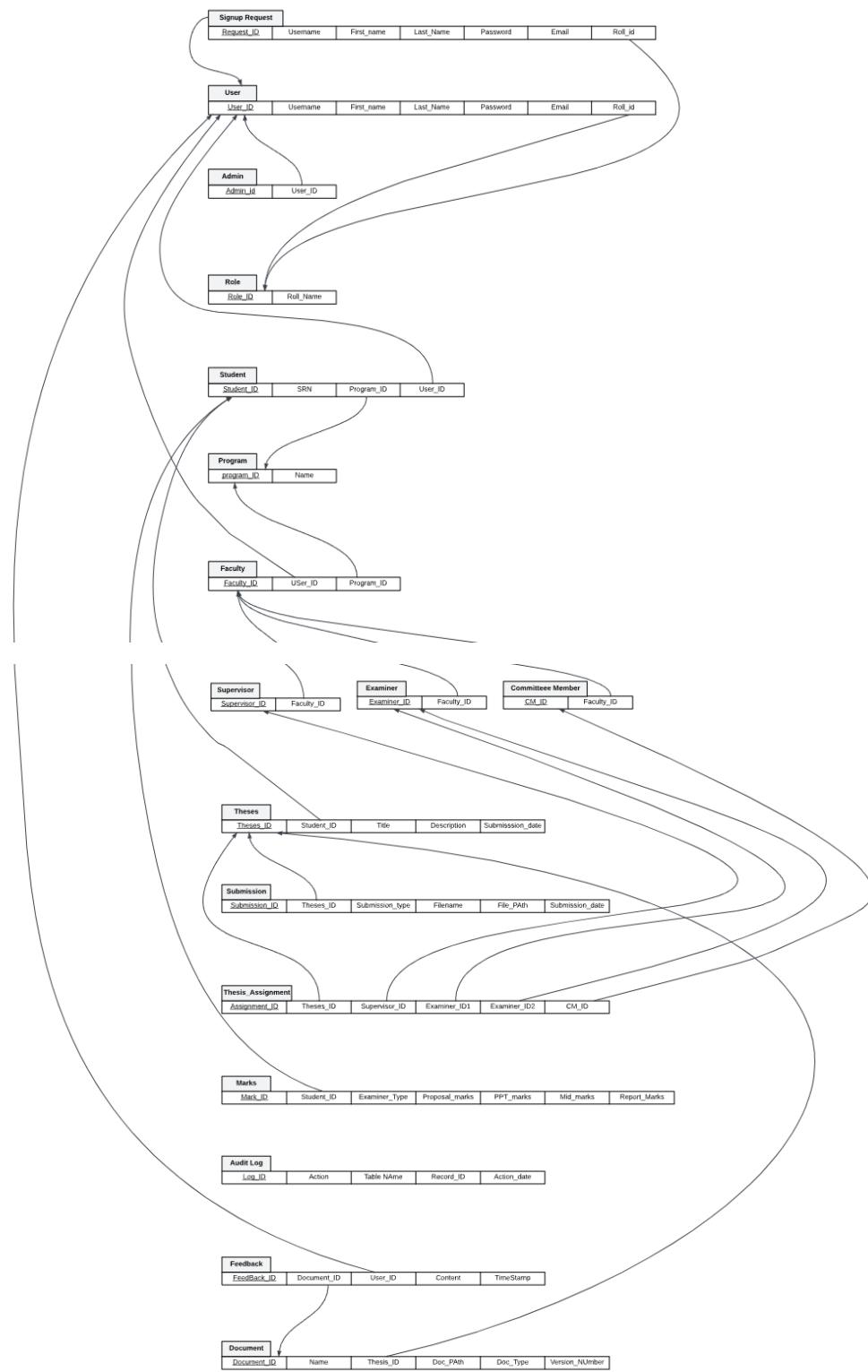
ERD

EERD

Relational Schema

Signup Request

Request_ID	Username	First_name	Last_Name	Password	Email	Roll_id

User

User_ID	Username	First_name	Last_Name	Password	Email	Roll_id

Admin

Admin_id	User_ID

Role

Role_ID	Roll_Name

Student

Student_ID	SRN	Program_ID	User_ID

Program

program_ID	Name

Faculty

Faculty_ID	USer_ID	Program_ID

Supervisor

Supervisor_ID	Faculty_ID

Examiner

Examiner_ID	Faculty_ID

Committeee Member

CM_ID	Faculty_ID

Theses

Theses_ID	Student_ID	Title	Description	Submisssion_date

Submission

Submission_ID	Theses_ID	Submission_type	Filename	File_PAth	Submission_date

Thesis_Assignment

Assignment_ID	Theses_ID	Supervisor_ID	Examiner_ID1	Examiner_ID2	CM_ID

Marks

Mark_ID	Student_ID	Examiner_Type	Proposal_marks	PPT_marks	Mid_marks	Report_Marks

Audit Log

Log_ID	Action	Table NAme	Record_ID	Action_date

Feedback

FeedBack_ID	Document_ID	User_ID	Content	TimeStamp

Document

Document_ID	Name	Thesis_ID	Doc_PAth	Doc_Type	Version_NUmber

Normalization

The schema provided is already in 3NF. Here is detailed proof of it:

1. **Each table has a primary key that uniquely identifies each row.** This is a fundamental requirement for any relational database table.

2. **There are no partial dependencies.** A partial dependency exists when a non-key attribute is dependent on only a part of the primary key. In all the tables, any non-key attribute is either dependent on the entire primary key or another candidate key.

3. **There are no transitive dependencies.** A transitive dependency exists when a non-key attribute is dependent on another non-key attribute, which in turn is dependent on the primary key.

Table-wise Analysis

By analyzing each table:

- **Roles:** role_name uniquely identifies each role.

- **Users:** user_id uniquely identifies each user, and all other attributes are dependent on it. There are no transitive dependencies.

- **Programs:** program_id uniquely identifies each program.

- **Students:** student_id uniquely identifies each student. user_id (foreign key) ensures a user can only be one student, and program_id references the program table. There are no transitive dependencies.

- **Faculty:** faculty_id uniquely identifies each faculty member. user_id ensures one user per faculty member, and program_id (optional) is dependent on the faculty's expertise. There are no transitive dependencies.

- **Theses:** thesis_id uniquely identifies each thesis, and student_id (foreign key) references the student who submitted it. There are no transitive dependencies.

- **ThesisAssignments:** assignment_id uniquely identifies each assignment. thesis_id references the thesis, and supervisor_id, examiner_id1, examiner_id2, and committee_member_id (all foreign keys) reference faculty members. There are no transitive dependencies.

- **Documents:** document_id uniquely identifies each document, and thesis_id (foreign key) references the thesis it belongs to. There are no transitive dependencies.

- **Feedback:** feedback_id uniquely identifies each feedback entry. document_id references the document it's for, and user_id references the user who provided it. The ON DELETE CASCADE on the foreign key for user_id ensures feedback is deleted if the user is deleted, preventing orphaned data. There are no transitive dependencies.
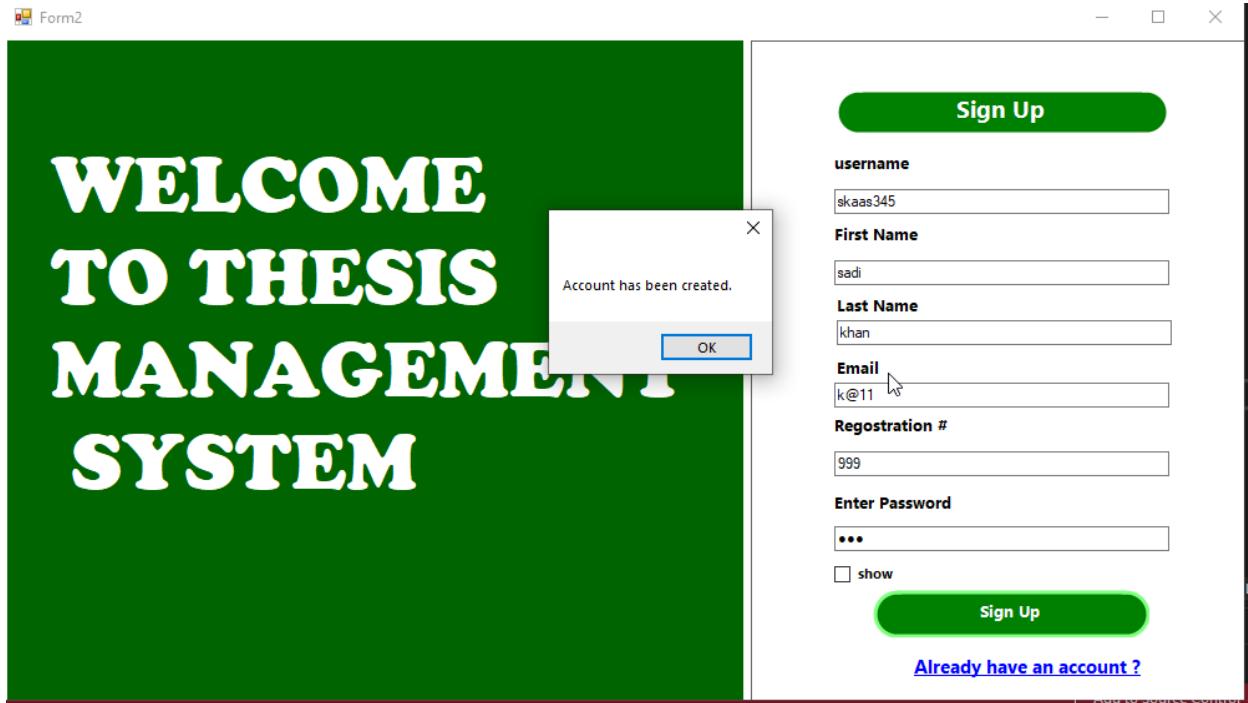
- **SignupRequests:** request_id uniquely identifies each signup request, and all other attributes define the request details. There are no transitive dependencies.

- **Marks:** mark_id uniquely identifies each mark entry. student_id references the student, examiner_type specifies the examiner role (Supervisor, Examiner1, Examiner2), and each mark attribute is directly dependent on the combination of student_id and examiner_type. There are no transitive dependencies.

- **Submissions:** submission_id uniquely identifies each submission, thesis_id references the thesis it belongs to, and other attributes define the submission details. There are no transitive dependencies.

- **AuditLog:** LogId uniquely identifies each log entry, and other attributes define the action details. There are no transitive dependencies.

Therefore, based on the analysis of each table's structure and relationships, all tables appear to be in 3NF.

Implementation & User Guide
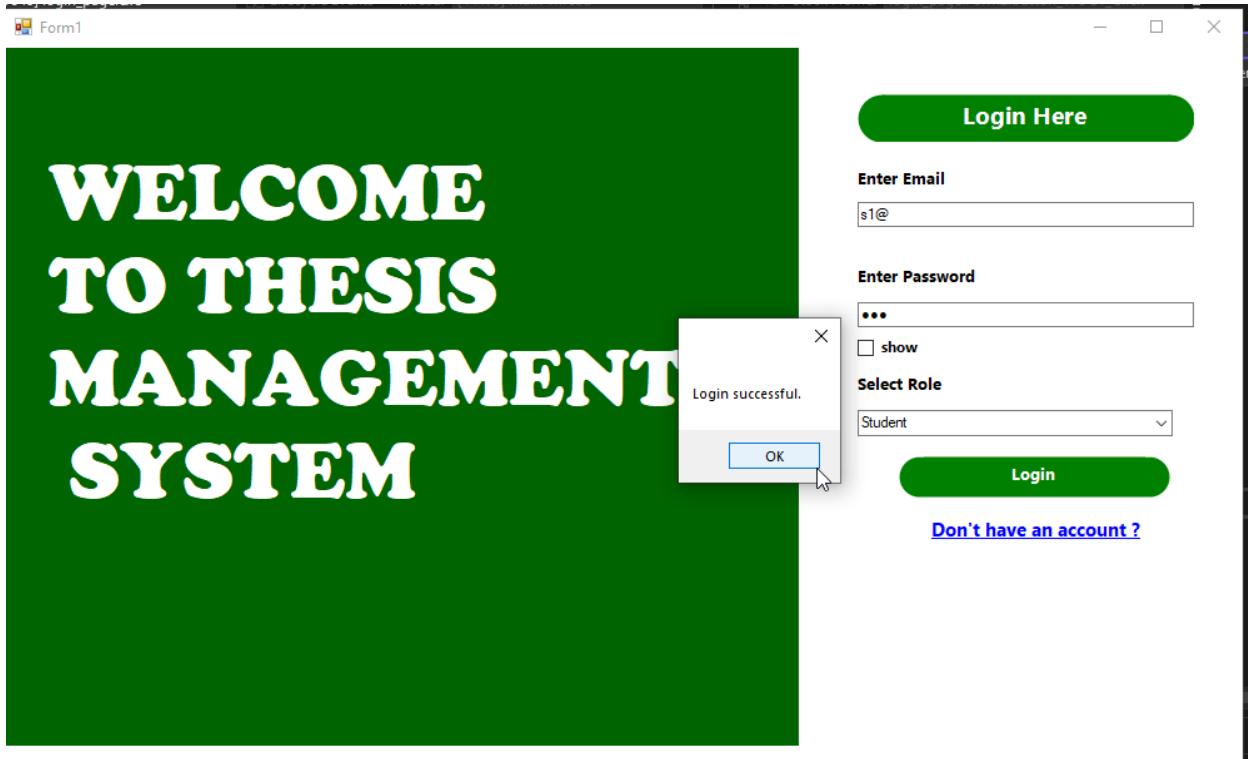
1. Student

1.1 Sign Up Page
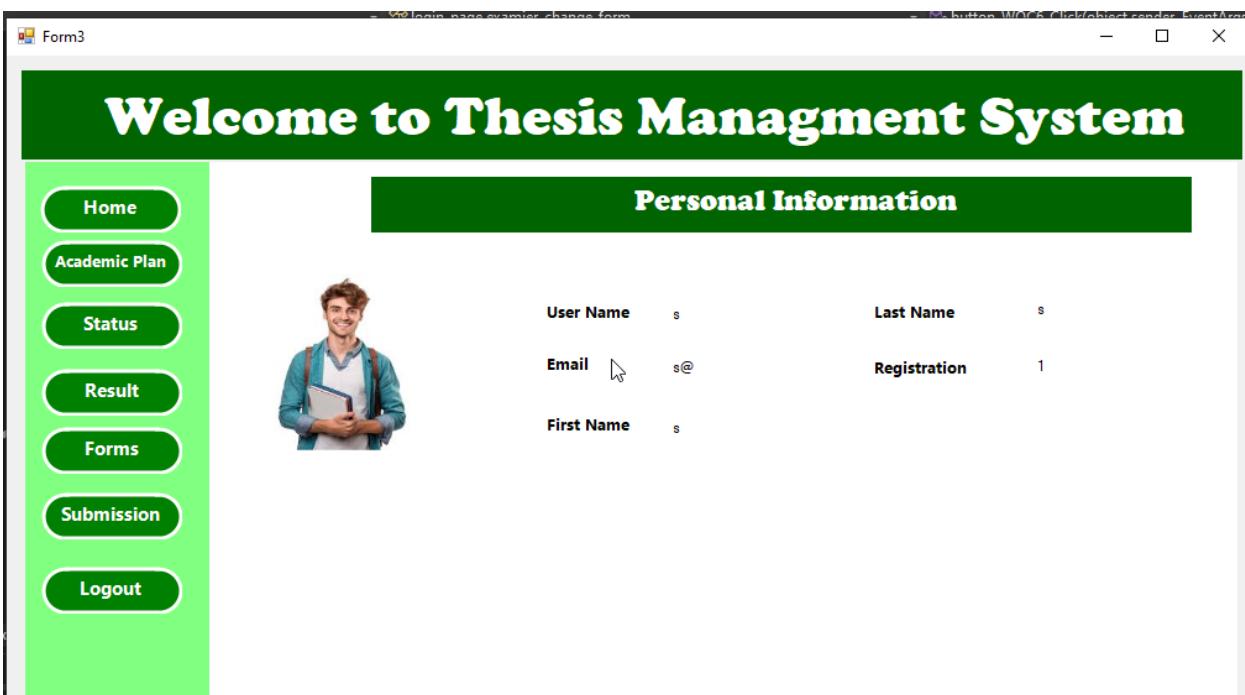
- First go to TMS portal

- Over here a signup page will be displayed to you

- You will have to add the required details to get you signed up

1.2 Sign In Page

- First go to the sign In page of TMS portal.

- Then enter your Email and password

- Select the type of account you are signing in for

1.3 Home Page

- After Logging in you will go to the home page

- Then you will have 6 options to choose from the side bar

- Your name will and your details will be displayed on the page

1.4 Calendar Page

- After clicking on the Academic Calendar you will be navigated to the calendar page
- Then you will have the Calendar displayed to you with all the events and the dates of them

1.5 Forms Page

15

- Click on the Forms tab

- Then you will have 4 different forms to choose from 1)Thesis registration form 2) Supervisor Change request form 3) Title change form 4) Examiner change form

- In these forms you will have to enter the specific details required for each of the forms according to use

1.6 Status Track Page

- Click the status tab

- Then you will see the status of different requests as that of thesis request form or supervisor change request form and title change request form
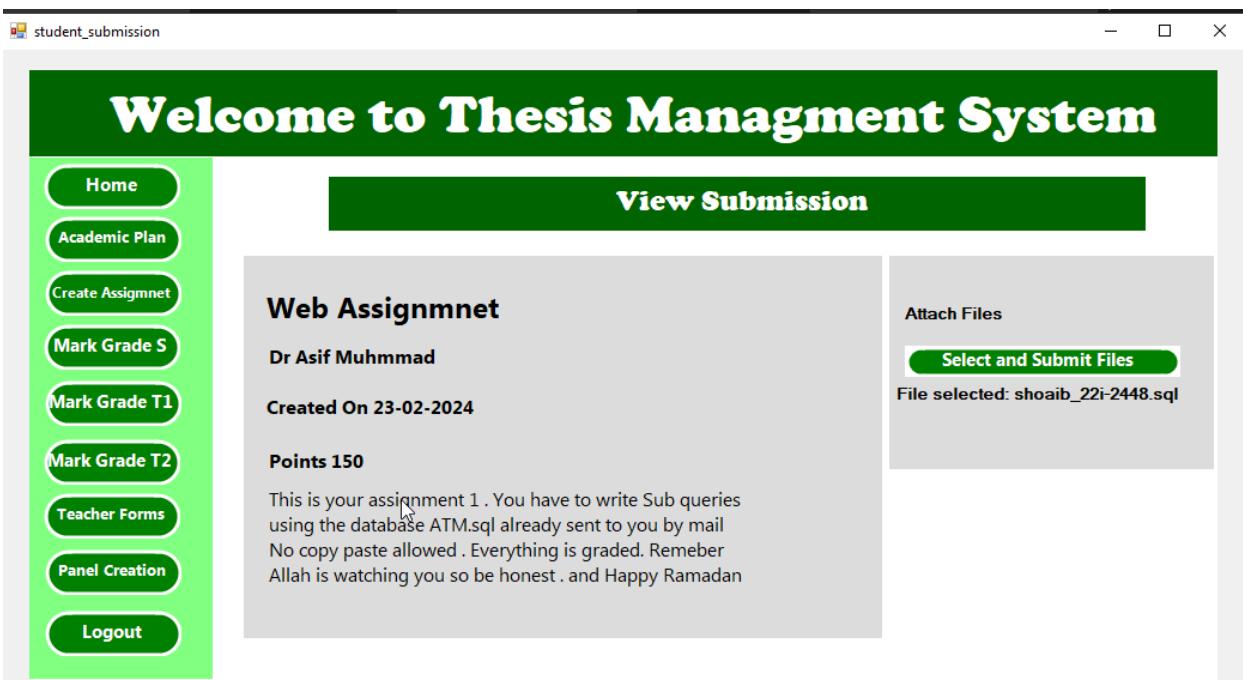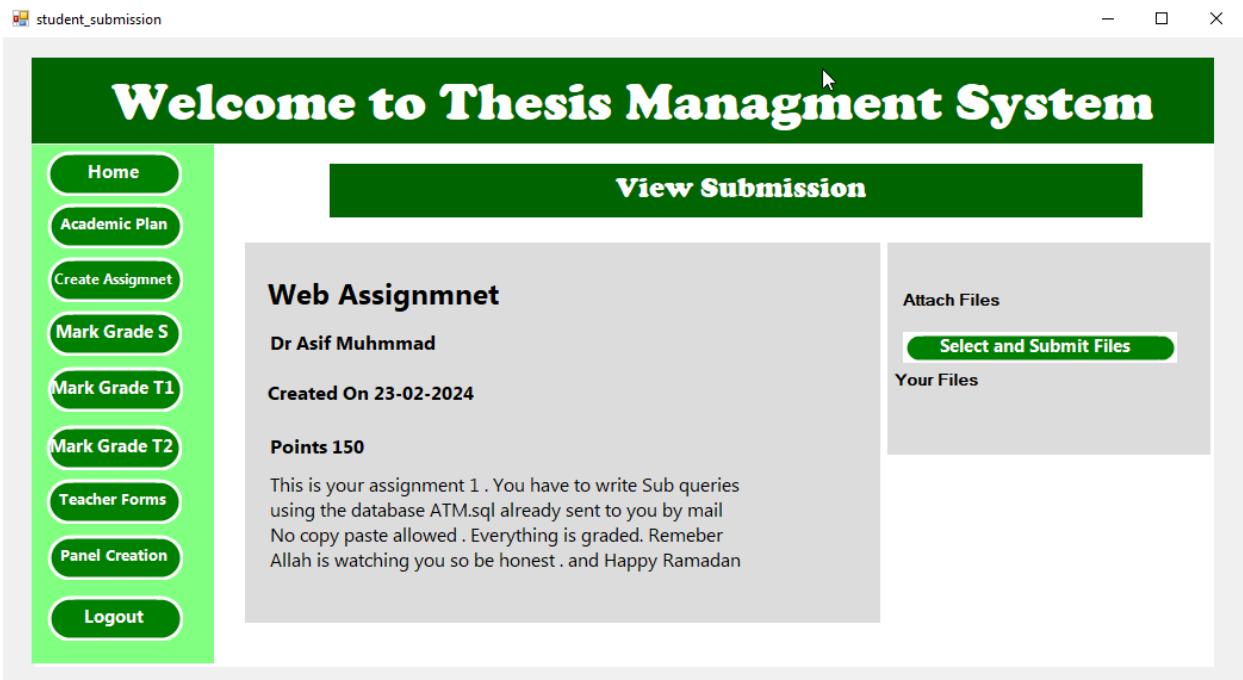
1.7 Results Page

Welcome to Thesis Managment System

Student Results

Type	Supervisor Marks	Examiner # 1 Marks	Examiner # 2 Marks
Proposal Marks	1	2	1
Mid Marks	2	2	1
Presentation Marks	2	2	1
Report Marks	2	2	2
Total Marks	7	8	5

- Select the results tab
- Over here you will see the results of your thesis and other submissions

1.8 Submissions Page

- Select the submissions tab

- Then you will see the assignment or the project which is due for submission

- After clicking on it you will be navigated to its submission tab

- Then click add or create to upload your work

- After that click on the turn in button to submit your work

2. Academic Officer
2.2 Sign Up Page
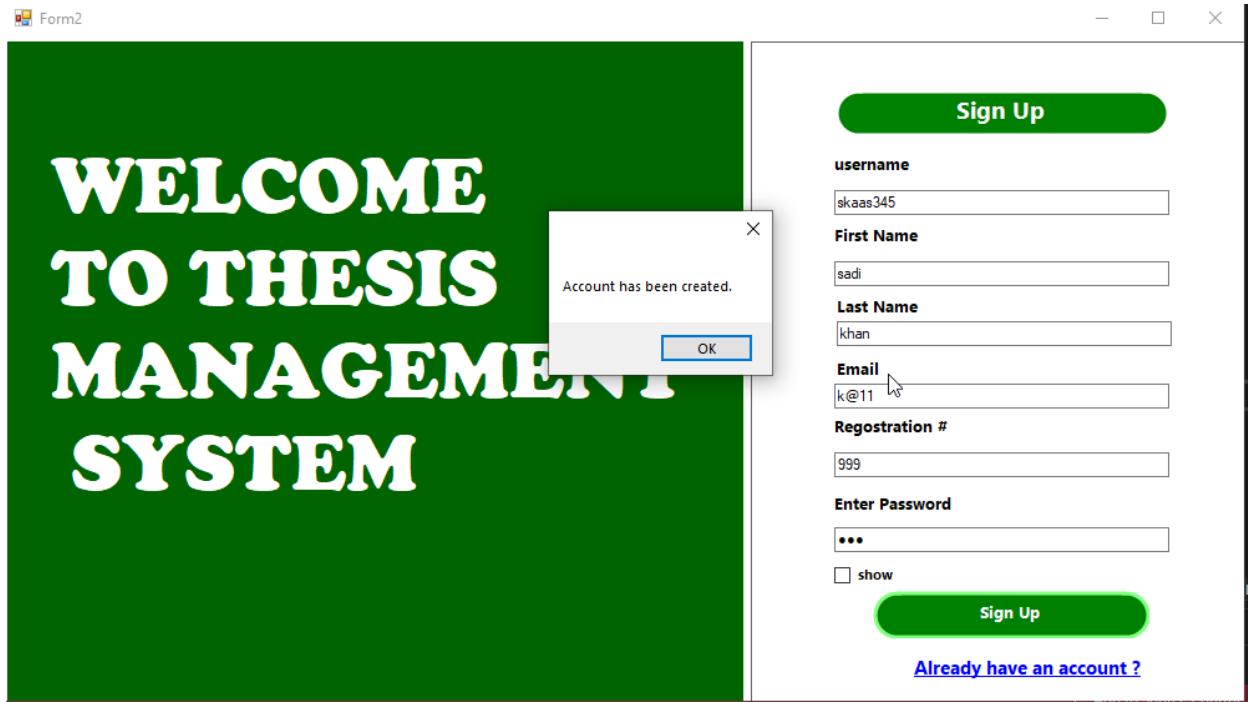
- First go to the TMS portal

- Here you will have to sign up for the academic officer

- You will have to fill in the required information for the academic officer
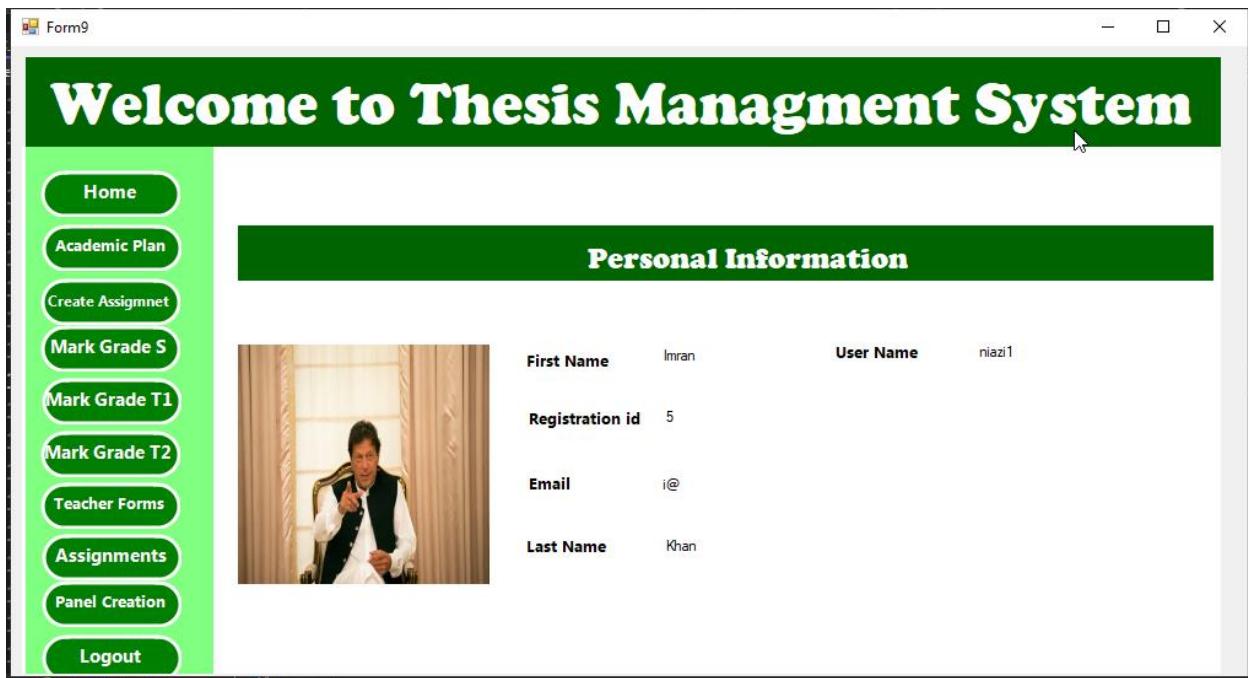
3.Teacher
3.1 Sign Up Page

- First go to the TMS portal

- Here you will have to sign Up for the Teacher

- You will have to fill in the required information for the teacher

3.2 Home Page

- After Logging in you will go to the home page

- Then you will have 7 options to choose from the side bar
- Your name along with your picture and your details will be displayed on the page

3.3 Teacher Forms Page

- Click on the Forms tab

- Then you will have 4 different forms to choose from 1)Thesis registration request forms 2) Supervisor Change request forms 3) Title change request forms 4) Examiner change request forms

- In these forms you will have to have to decide what request to accept or decline on the basis of your role

- If you have no form pending in the requests then a popup will show that no form pending

- If you want to do specific work on them then a popup will show that you are not a HOD, or a supervisor or an examiner depending on your role

- Click on any of the form pending it will then get you a detailed view containing the information in it that was sent by the student

3.4 Panel Creation Page

- Click on the Panel Creation tab

- Then you will have a form to choose from and enter the details in it

- Then after submitting the panels will be created

3.6 Assignments Page

- Click on the assignments tab

- Then you will have the option to create assignments as you like

- You can set the marks of the assignment, its due date, title and its description