CS232 - Project Report

# Project Title: Sorting Algorithm Visualizer

## Team Members:

|  |  |  |
| --- | --- | --- |
| Name | Roll Number | Contribution |
| Abdullah Ihsan | Ihsan | Solo |

## 1. Project Overview

This Sorting Algorithm Visualizer is an interactive visual learning tool that can be used by students, teachers and code learners to understand various sorting algorithms, thoer complexities and comparisons.

## 2. Technology Stack

Frontend: PyQt

Backend: Python

Database: PostgreSQL

Project Management: Github

Code Hosting: GitHub

Authentication: SHA-256

## 3. Entity-Relationship Diagram (ERD)

Entities: Student, Faculty, Course, Department, Enrollment, Assignment, Announcement, Attendance

Relationships:

- A student can enroll in many courses (Many-to-Many via Enrollment)

- A faculty member teaches many courses

- A department offers multiple courses

- Assignments are related to courses

- Announcements can be global or department-specific

## 4. Database Schema (Simplified)

Example - Student Table:

CREATE TABLE Student (

student\_id SERIAL PRIMARY KEY,

name VARCHAR(100),

email VARCHAR(100) UNIQUE,

department\_id INT,

password\_hash TEXT,

FOREIGN KEY (department\_id) REFERENCES Department(department\_id)

);

(Other tables follow similar structures)

## 5. Screenshots of Working System

- Student dashboard with enrolled courses

- Admin panel to add courses/faculty

- Assignment upload and announcement feed

(Screenshots should be inserted here)

## 6. Functionalities Implemented

- Secure login system (JWT)

- Course registration and faculty assignment

- Student enrollment system

- Upload/view assignments

- View announcements and attendance

## 7. GitHub & Jira Details

GitHub Repository: https://github.com/TeamCampusConnect/university-portal

GitHub Activity:

- 120+ commits across 4 contributors

- Separate branches for frontend and backend

Jira Board:

- Epics for DB design, frontend, backend, and testing

- Proper ticket lifecycle with comments and assignments

## 8. Challenges Faced

- Managing foreign key constraints during insertions

- Merging frontend-backend API responses

- Resolving Git merge conflicts

## 9. Future Enhancements

- Add support for online quizzes

- Mobile version of the portal

- Chat functionality between students and faculty

## 10. Conclusion

This project enhanced our understanding of relational database design, API development, and project lifecycle management using tools like Jira and GitHub.