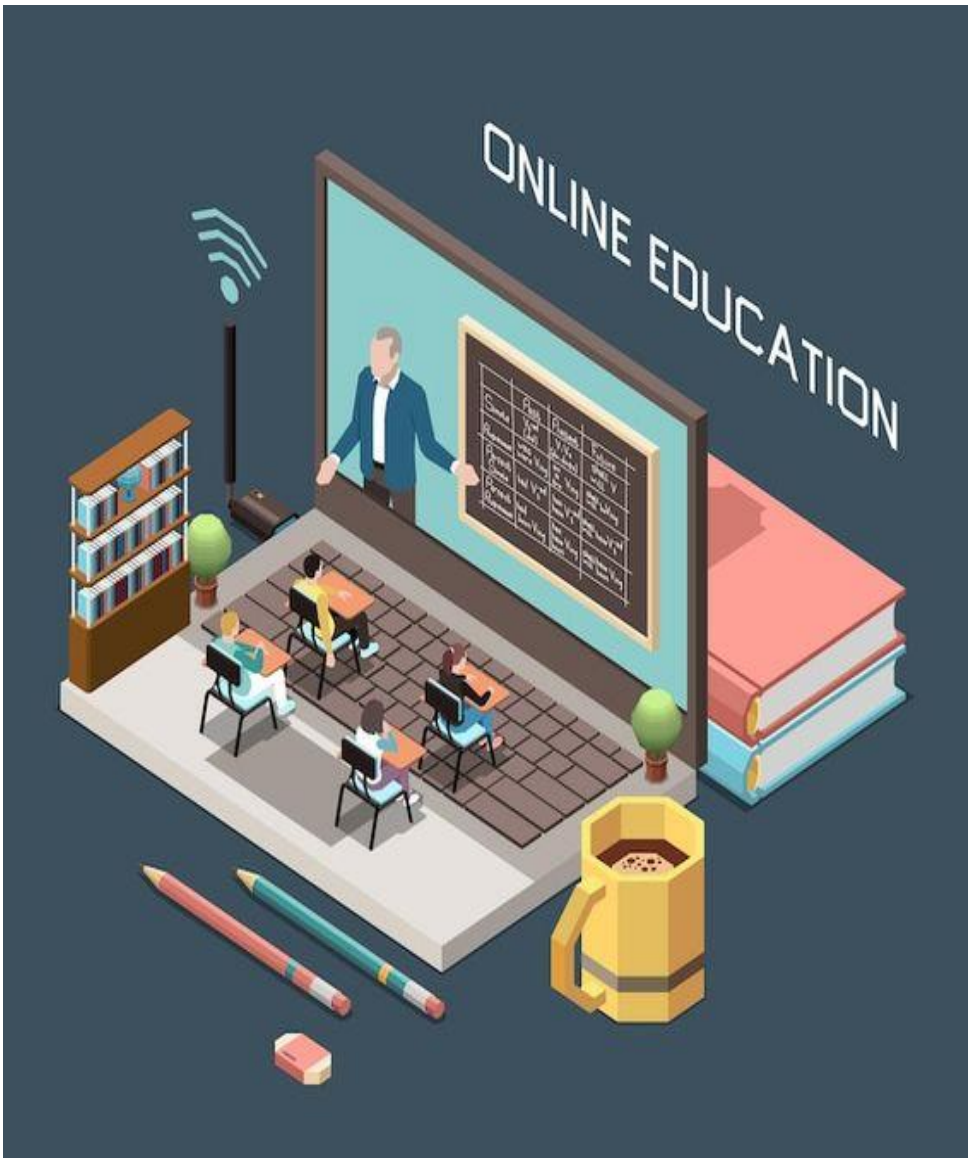


Online Learning Platform

ABSTRACT

A junior project is submitted in partial fulfillment of the requirements for the degree of graduation project from DEBI .

Project Team

1- عمرو سيد صادق عبدالرحمن
2- بلاول واءل بلال بدوي
3- عمر خالد عباده
4- محمد محمود حسنين

Under Supervision of

Eng. reham

Benha, October 2024

ABSTRACT

Online course platforms are digital platforms designed to offer a wide range of educational courses and training programs through the internet. These platforms provide learners with flexible, accessible, and often self-paced learning options. Users can access content such as video lectures, reading materials, assignments, and quizzes from anywhere with an internet connection. Popular platforms like Coursera, Udemy, and edX cater to diverse fields, including technology, business, arts, and personal development. Many offer certifications and are utilized by individuals, educational institutions, and corporations for skill development and lifelong learning.

TABLE OF CONTENTS

1-intrudaction

1.1 Problem Definition:

In today's digital age, there is a growing demand for efficient and user-friendly platforms that enable educational institutions, instructors, and learners to manage online courses seamlessly. Traditional systems lack the flexibility and scalability required to handle large numbers of students, courses, and progress tracking efficiently. Additionally, manual enrollment and course management processes are time-consuming and prone to errors.

while instructors struggle to monitor student engagement and performance in a streamlined manner.

Inconsistent user experience: A need for a simple, intuitive interface where students can easily browse available courses, enroll, and navigate course materials without technical

ANALISES AND DESIGN

2.1 system users and using caces:

- **Administrators:**
  - **Manage the overall platform.**
  - **Create and manage user accounts, including assigning roles (instructors, students).**
  - **Oversee course creation, enrollments, and platform settings.**
  - **Generate reports on system usage, student progress, and course performance.**
- **Instructors:**
  - **Create, manage, and update course content (lectures, assignments, quizzes).**
  - **Enroll or manage students in their courses.**
  - **Track student progress, engagement, and performance.**
  - **Provide feedback, grade assignments, and generate reports.**
- **Students:**
  - **Browse and enroll in available courses.**
  - **Access course materials, participate in assignments, and complete quizzes.**
  - **Track their progress and performance in enrolled courses.**
  - **Interact with instructors and fellow students through discussions and feedback.**
- **Guests/Visitors:**
  - **Browse available courses without enrolling.**
  - **View basic course information such as descriptions, schedules, and requirements.**
  - **Register to become students and gain access to course content.**

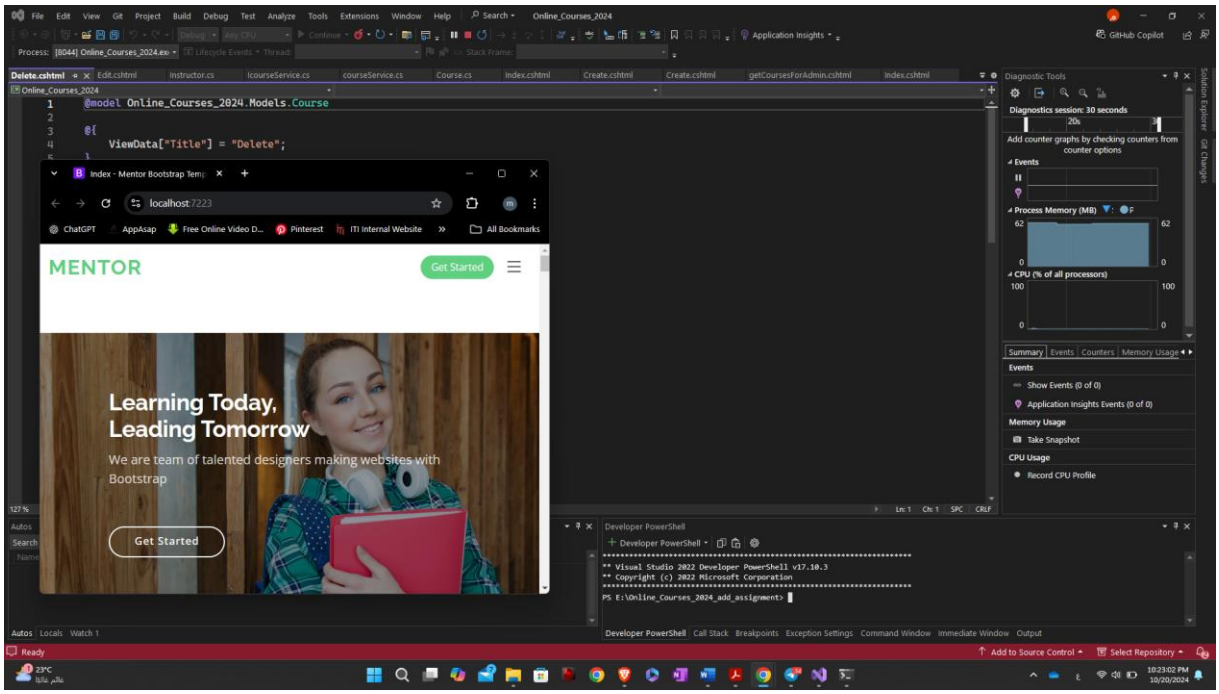
Using caces :

- **Administrator Use Cases**
- **UC1: Manage Users:**
  - **The administrator can create, update, and delete user accounts (students, instructors, other admins).**
  - **Assign roles (admin, instructor, student) and permissions.**
- **UC2: Manage Courses:**
  - **The administrator can view and manage all courses on the platform.**
  - **Approve or delete courses submitted by instructors.**
- **UC3: Generate Reports:**
  - **The administrator can generate reports on system usage, course performance, and student progress.**
- **UC4: Manage Platform Settings:**
  - **Modify platform settings such as notifications, enrollment limits, and content visibility.**
- **2. Instructor Use Cases**
- **UC5: Create and Manage Courses:**
  - **Instructors can create, update, and delete their own courses.**
  - **Upload course content, set schedules, and add assignments or quizzes.**
- **UC6: Enroll and Manage Students:**
  - **Instructors can enroll students into courses or approve enrollment requests.**
  - **Remove students from courses or transfer them to different courses.**
- **UC7: Track Student Progress:**
  - **Instructors can monitor individual student progress, view grades, and track assignment completions.**
- **UC8: Grade and Provide Feedback:**
  - **Instructors can grade assignments, quizzes, and provide feedback to students directly through the platform.**
- **3. Student Use Cases**
- **UC9: Browse and Search Courses:**

  - **Students can search for and browse available courses by category, instructor, or difficulty level.**
- **UC10: Enroll in Courses:**
  - **Students can enroll in courses, either freely or through an approval system if required by the instructor.**
- **UC11: Access Course Materials:**
  - **Students can view course content, including video lectures, readings, and assignments.**
- **UC12: Track Progress:**
  - **Students can view their progress in enrolled courses, including completion percentages, grades, and feedback from instructors.**
- **UC13: Submit Assignments:**
  - **Students can submit assignments and take quizzes through the platform.**
- **UC14: Interact with Instructors:**
  - **Students can communicate with instructors through messaging or discussion forums.**
- **4. Guest/Visitor Use Cases**
- **UC15: Browse Available Courses:**
  - **Guests can browse available courses but will not have access to the course content.**
- **UC16: Register as a User:**
  - **Guests can register to become students by providing their personal details and creating an account.**

2.2 interface and reaction:

we have worked in reactive graphical user interface using asp.net as backend ,html,css.javascript as front end working in vs code as IDE

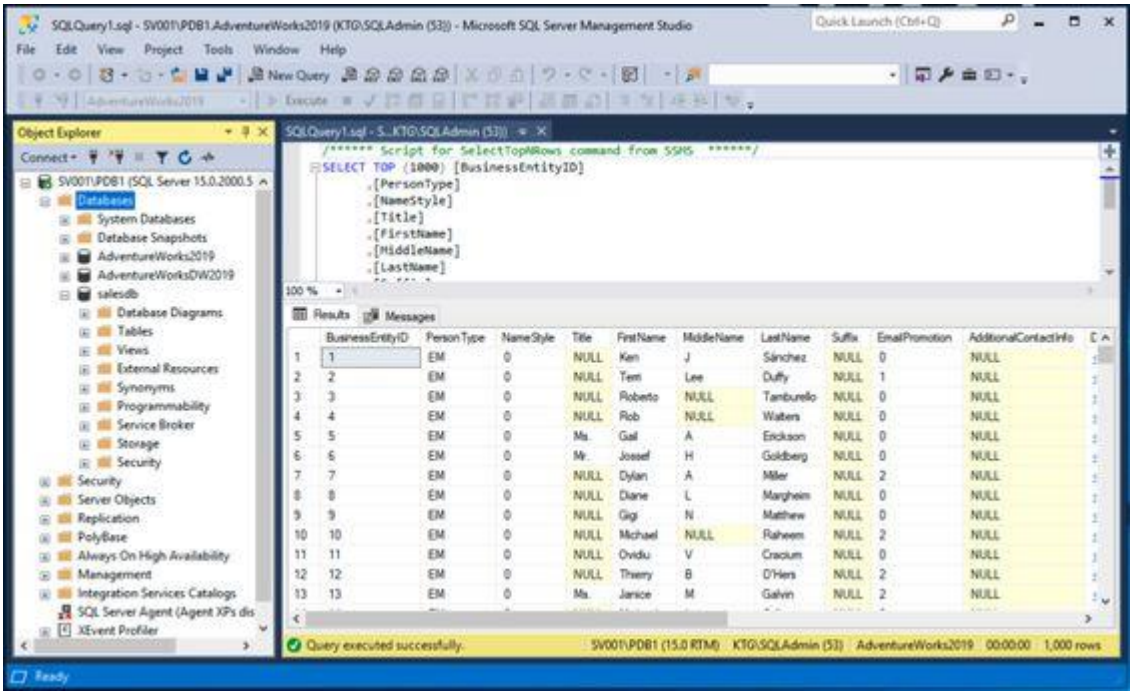


And we wonted to make user able to see and react with interface parts like buttons and compo boxes so we have worked to create the main pages of the system and we introduce them in next :

Data base

3.1 date base design:

We have worked in flexible data base that allow to read an input from the user in the first main page and review it to show the all student data so we have used SQL data base with java that allowed us to



The database for the web application will include key tables to manage users, courses, enrollments, and course content. Here's a summary:

1. **Users Table**: Stores user details like ID, username, password, role (Admin, Instructor, Student), and contact info.
2. **Courses Table**: Holds course details such as course ID, name, description, instructor, and start/end dates.
3. **Enrollments Table**: Tracks which students are enrolled in which courses, along with their progress.
4. **Course Content Table**: Stores the course materials (lectures, assignments, quizzes) for each course.
5. **Assignments Table**: Manages assignments, including their details and due dates.

6. **Submissions Table**: Tracks students' assignment submissions, along with their grades and feedback from instructors.

The tables are interlinked with relationships like:

- Users (instructors) manage multiple courses.
- Students enroll in multiple courses.
- Courses contain multiple content pieces and assignments.

3.2 connection and process:

The main chain of user processes in a web application for managing online courses and student enrollments involves the interaction between the different user roles (Administrators, Instructors, Students). Here's the breakdown of the process flow for each type of user:

- **1. Administrator Process Chain**
- **User Registration/Management**: Admins create, update, or delete user accounts (students, instructors).
- **Course Management**: Admins approve or reject new course submissions by instructors.
- **Platform Settings**: Admins manage system-wide settings, notifications, and access controls.
- **Monitoring and Reporting**: Admins generate reports on user activity, course performance, and overall system health.
- **2. Instructor Process Chain**
- **Login and Authentication**: Instructors log into the system using their credentials.
- **Course Creation**:

- Instructors create new courses, adding course details such as title, description, and start/end dates.
- Upload course materials (lectures, assignments, quizzes) and organize the course structure.

- **Student Enrollment Management**:
  - Approve student enrollments if required.
  - Assign or remove students from the course as needed.
- **Content Delivery**: Instructors upload content that students can access, including video lectures, reading materials, and assignments.
- **Track Student Progress**: Monitor student performance, grades, and participation throughout the course.
- **Provide Feedback**: Review student submissions, grade them, and offer feedback.
- **3. Student Process Chain**
- **Account Registration**: Students sign up and create an account or log in if they already have one.
- **Course Browsing**:
  - Students browse through the list of available courses.

- Filter and search for courses based on interests, difficulty level, or instructor.

- **Enrollment**:
  - Enroll in a course (either self-enroll or await instructor approval).
- **Access Course Content**:
  - Once enrolled, students access course materials such as video lectures, assignments, and quizzes.
  - Participate in discussions, complete quizzes, and assignments.
- **Track Progress**:
  - Students can view their progress, grades, and feedback from instructors.
- **Submit Assignments**: Students complete and submit assignments before deadlines.
- **4. Guest/Visitor Process Chain**
- **Browsing**: Guests can browse available courses without enrolling.
- **Account Creation**: Guests can register and create an account to gain full access to the system.

- **Overall Process Flow**

1. **Admin**: Sets up users, approves courses, manages the platform.
2. **Instructor**: Creates courses, manages content and students, grades assignments.
3. **Student**: Enrolls in courses, completes content and assignments, tracks progress

3.3 ERD AND ENTITES

Based on the contents of the uploaded files, we can infer the following key classes and relationships for an ERD (Entity-Relationship Diagram):

- **1. Assignment:**
- **AssignmentId**: Primary key.
- **AssignmentTitle**: Title of the assignment.
- **Description**: Assignment details.
- **LessonId**: Foreign key referencing a Lesson.
- **2. Course:**
- **CourseId**: Primary key.

- **CourseName**: Name of the course.
- **Description**: Description of the course.
- **Price**: Cost of the course.
- **DurationAttribute**: Duration of the course.
- **InstructorId**: Foreign key referencing an Instructor.
- **3. ErrorViewModel:**
- **RequestId**: Used for logging errors (unrelated to the course/assignment relationships).