Flexible Virtual School

Table of Contents

Revision Control History

Version	Date	Description of change

Introduction

The Arabic Universal Academy is creating a virtual school to teach Arabic online. This new approach uses the latest internet tools to reach people all over the world, making it easier for both native and non-native Arabic speakers to learn. The goal is to be flexible with class times, locations, and instructors, and to broadcast classes to multiple places at the same time. This way, more people can access Arabic language education in a way that suits them best.

Methodology

We will implement our system using Agile methodology, it offers significant advantages tailored to its collaborative nature and evolving requirements. Agile's flexibility allows us to accommodate changes effectively, ensuring the system remains relevant and valuable.
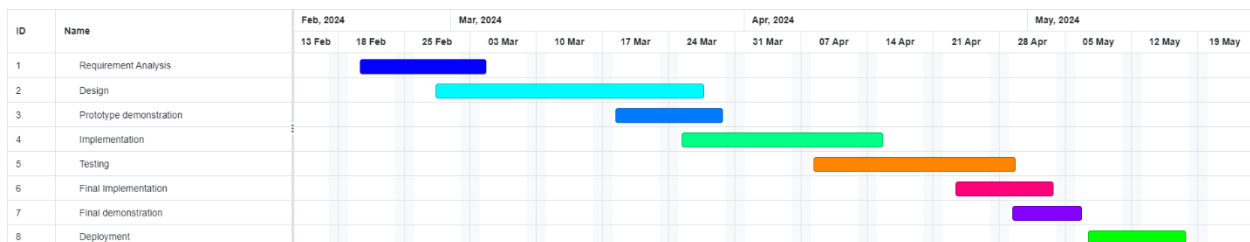
The system will be separated into 3 sub-systems:

1-Interface (held by Saleh Alghaith, Khaled Alharbi, Talal Alothman)

2-Process (held by Saud Alkatheeri, Talal Alkahtani, Mshari Alaeena)

3-Database (held by Talal Alothman, Talal Alkahtani)

How component should be tested:

For the interface, usability testing is crucial to evaluate ease of use and accessibility, also functional testing to verify that all elements work as needed.

For the process (business logic) we have unit testing to validate individual components and integration testing to ensure seamless interaction. Functional and regression testing confirm expected behavior and prevent regressions, also concurrency and performance testing check the efficiency under load.

For the database, data integrity testing ensures data reliability, with data manipulation testing for CRUD (Create, delete, update, read) operations. Performance testing measures efficiency, security testing ensures protection against breaches, and backup and recovery testing validates resilience of the database.

| ID	Name	Feb, 2024			Mar, 2024				Apr, 2024				May, 2024			
		13 Feb	18 Feb	25 Feb	03 Mar	10 Mar	17 Mar	24 Mar	31 Mar	07 Apr	14 Apr	21 Apr	28 Apr	05 May	12 May	19 May
1	Requirement Analysis															
2	Design															
3	Prototype demonstration															
4	Implementation															
5	Testing															
6	Final Implementation															
7	Final demonstration															
8	Deployment															

System Architecture

Design decisions.

In developing our Flexible Virtual School system, usability is paramount, given its diverse user base. Reliability takes precedence over performance, particularly considering the sensitive educational data stored. Additionally, scalability and stringent security measures must be integrated seamlessly into our system.

- The system shall use Oracle Database.
- Server-side shall be implemented using Python.
- Client-side shall be implemented using HTML, CSS, and JS.
- The system shall be able to work with Gmail API.
- The system shall be able to work with Collabinate API.
- The system shall be able to work with Moyasar payment gateway.

Domain model

These systems will interact with each other to perform certain procedures.
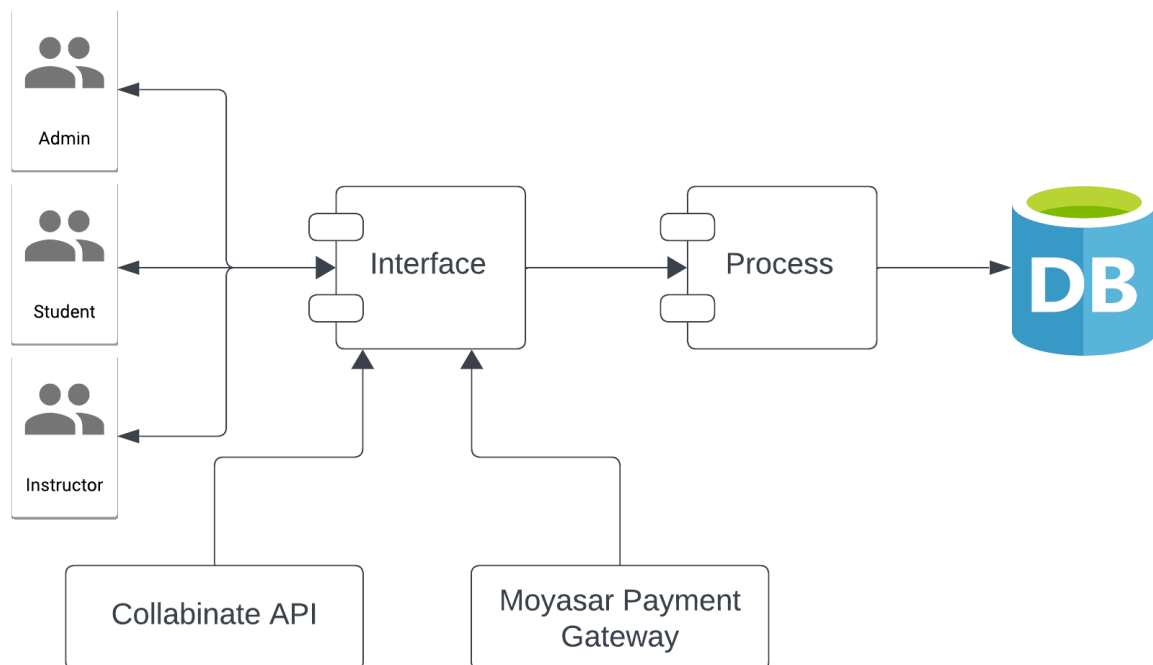
Architectural Style

Multi-Tier Architecture

The chosen architectural style for the Flexible Virtual School of the Arabic Universal Academy is Multi-Tier Architecture. This decision is based on the need for a scalable, modular, and flexible system that can handle various functionalities while maintaining separation of concerns.

Justification for Multi-Tier Architecture:

1. **Scalability:** Multi-tier architecture allows for the modularization of different components, such as presentation, application logic, and data storage. This modularity enables scalability by allowing individual tiers to scale independently. As the virtual school grows, each tier can be scaled as needed to accommodate increasing demand.

2. **Separation of Concerns:** This architectural style separates the user interface, application logic, and data storage into distinct layers. This separation makes the system more maintainable and easier to manage. Updates or changes to one tier can be performed without affecting the others, promoting flexibility and ease of development.

3. **Flexibility in Deployment:** Multi-tier architecture supports the deployment of different tiers on separate servers or even in different locations. This flexibility is crucial for a virtual school that aims to offer courses to students worldwide. It allows for efficient resource utilization and optimization.

4. **Enhanced Security:** With the separation of layers, security measures can be implemented at each tier, addressing specific security concerns. For example, the data tier can have robust encryption and access controls, while the presentation tier can focus on user authentication and authorization. This layered approach improves the overall security of the system.

## Considered and Rejected Alternative: Microservices

While Microservices architecture is a popular choice for building scalable and distributed systems, it might introduce unnecessary complexity for the specific needs of the Arabic Universal Academy's virtual school.

1- **Complexity Overhead:** Microservices come with increased complexity in terms of communication between services, service discovery, and management of distributed data. Considering the academy's primary focus on providing educational content, a simpler architecture might be more appropriate to minimize potential challenges in development and maintenance.

2- **Resource Optimization:** In the case of the virtual school, where courses may involve various multimedia elements and interactions, a monolithic approach might be more efficient in terms of resource usage. Microservices could lead to increased overhead due to inter-service communication and coordination.

In summary, the Multi-Tier Architecture is chosen for its scalability, separation of concerns, flexibility, and enhanced security, aligning well with the specific requirements of the Flexible Virtual School. The potential complexities and resource overhead associated with Microservices make it a less suitable choice for this scenario.

Structural model

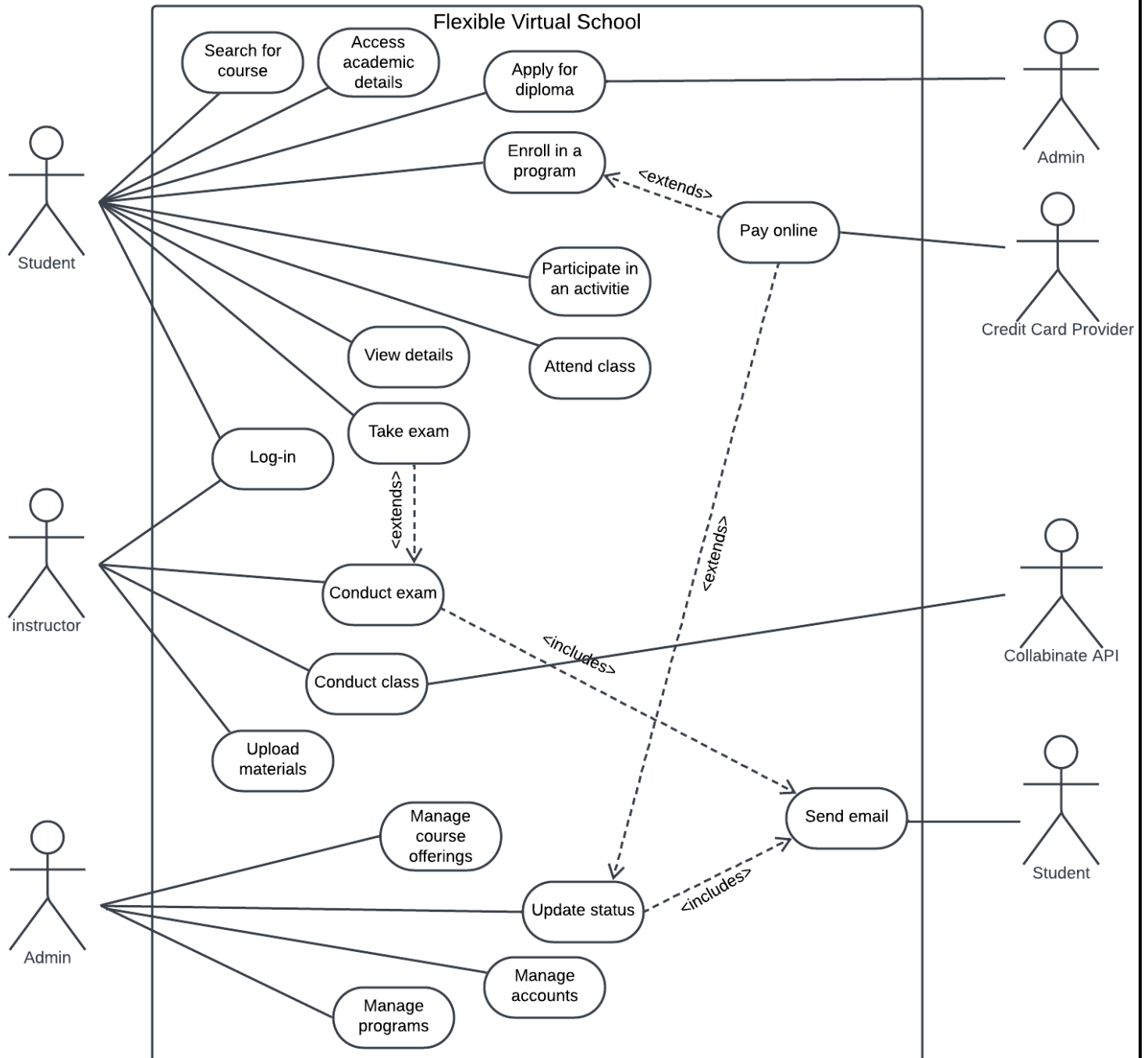i)   Component: process, database, and interface.
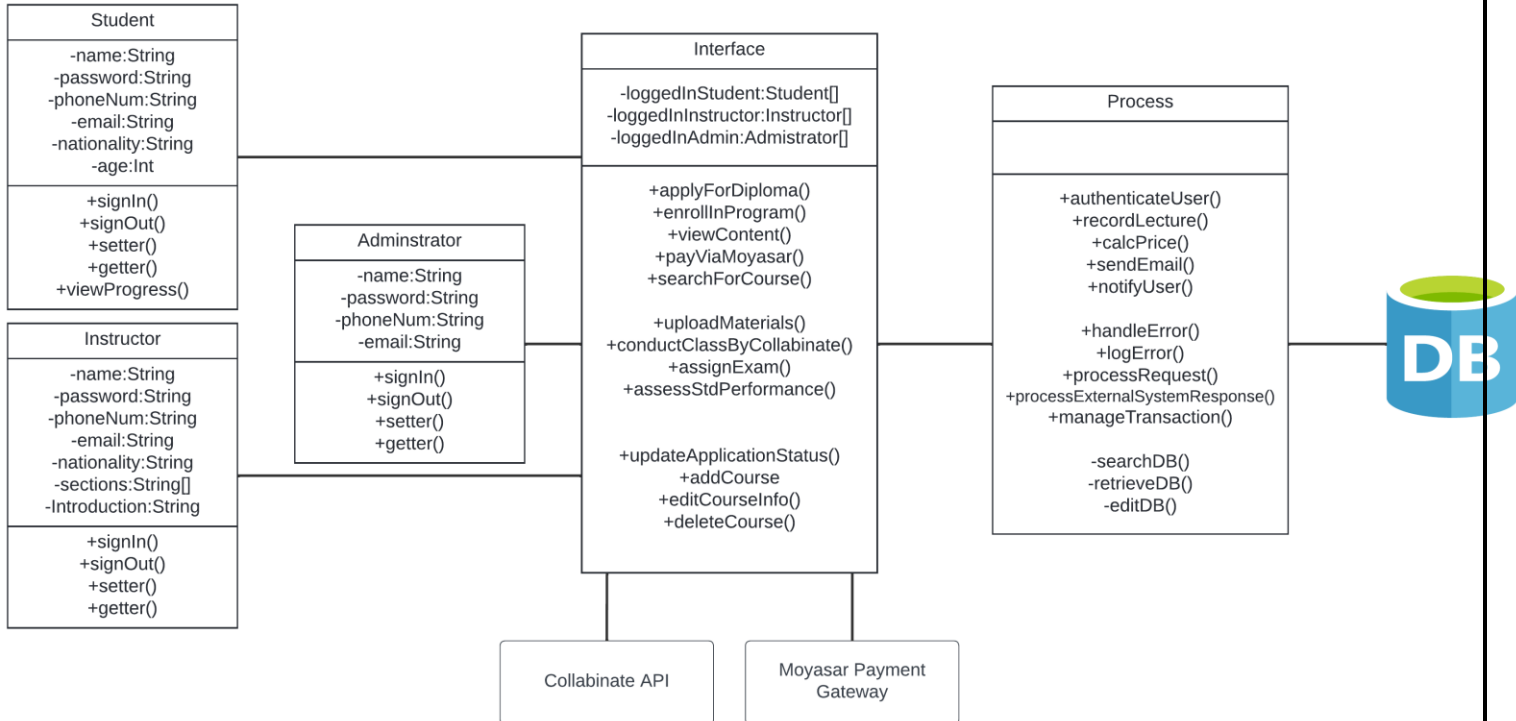
ii)   database and 3rd party:

We chose a cloud database because it will be more accessibility and flexible storage capacity, vital for accommodating the substantial volume of data we currently store and anticipate adding in the future. This includes storing lecture videos, course materials, and user information within the database.
3rd party:

1.Collabinate API
2.Moyasar payment gateway

iii)   Use Case diagram.

Flexible Virtual School

Student — Search for course, Access academic details, Apply for diploma, Enroll in a program, Participate in an activitie, View details, Attend class, Take exam, Log-in

instructor — Log-in, Conduct exam, Conduct class, Upload materials

Admin — Manage course offerings, Update status, Manage accounts, Manage programs

Pay online <extends> Enroll in a program

Take exam <extends> Conduct exam

Conduct exam <includes> Send email

Pay online <extends> Update status

Update status <includes> Send email

Admin
Credit Card Provider
Collabinate API
Student

Use Case Description	
System: *Flexible virtual school*	
Use Case name: *Manage programs*	
Primary actor: *Admin*	**Other actors:**
Stakeholders:	
Description: *The admin will be able to manage (delete, add, edit) programs.*	
Relationships **▪Includes:** **▪Extends:**	
Input:	
Pre-conditions:	
Steps:	

Actor	System
1. Chooses a program to manage. *3. Make the changes.* *4. Command: Update program.*	*2. Displays the program page.* *4. Verifies the information.* *6. Save and update the program.*

Alternative and exceptional flows:
4.1 IF there is missed information, THEN ask the admin to fill the missed part.
Post-conditions: Program updated.

Use Case Description	
System: *Flexible virtual school*	
Use Case name: *Manage account*	
Primary actor: *Admin*	**Other actors:** *Student*
Stakeholders:	
Description: *The admin will be able to manage (delete, edit) student account.*	
Relationships **▪Includes:** **▪Extends:**	
Input:	
Pre-conditions: *Student account exist.*	
Steps:	
Actor	**System**
1. Chooses a student account to manage. *3. Edit students account (or delete).* *4. Command: save updates.*	*2. Displays the student account page.* *5. Notifies the student.* *6. Saves updates.*
Alternative and exceptional flows:	
Post-conditions: Student account updated.	

Use Case Description	
System: *Flexible virtual school*	
Use Case name: *Update status*	
Primary actor: *Admin*	**Other actors:** *Student*
Stakeholders:	

Description: *The admin will update (accept or reject) the diploma application status.*

Relationships
▪Includes: Send e-mail
▪Extends: Pay online

Input:

Pre-conditions: *diploma application applied.*

Steps:

Actor	System
1. Chooses a diploma application. *3. Decide whether to accept/reject.* *4. Command: Update status.*	*2. Displays the application form.* *5. Saves the status.* *6. Notifies the student.*

Alternative and exceptional flows:
3.1 IF the admin has not made any decision, THEN put the application in under review status.

Post-conditions: Ask the applicant to pay.

Use Case Description	
System: *Flexible virtual school*	
Use Case name: *Manage course offerings*	
Primary actor: *Admin*	**Other actors:**
Stakeholders:	
Description: *The admin will be able to manage courses.*	
Relationships **▪Includes:** **▪Extends:**	
Input: Course information document.	
Pre-conditions:	
Steps:	
Actor	**System**
1. Command: offer a course. *3. Fills the form.*	*2. Displays the offering course form.* *4. Verifies the information.* *5. Adds course to the system.*
Alternative and exceptional flows: *4.1a IF there is missed information, THEN ask the admin to fill the missed part.* *4.1b IF the course already exists. THEN reject the request.*	
Post-conditions: New course added the system.	

Use Case Description	
System: *Flexible virtual school*	
Use Case name: *Take exam*	
Primary actor: *Student*	**Other actors:**
Stakeholders:	
Description: *The student takes an exam.*	
Relationships **▪Includes:** **▪Extends:** *Manage exam*	
Input: Answers.	
Pre-conditions: *Student logged in, Exam published by the Instructor.*	
Steps:	

Actor	System
1. Selects the exam page. *3.Command: Start exam.* *5. Fills the answers.* *6. Command: Submit.*	*2. Displays the exam page.* *4. Displays the questions.* *7. Saves answers.* *8. Saves updates.*

Alternative and exceptional flows:
Post-conditions: *Instructor notified.*

Use Case Description	
System: *Flexible Virtual School*	
Use Case name: *Enroll in a Program*	
Primary actor:	**Other actors:**
Stakeholders:	
Description: *The process by which a student selects and enrolls in a specific program.*	
Relationships **▪Includes:** **▪Extends:** *Pay online*	
Input:	
Pre-conditions: *Student logged in*	
Steps:	
Actor	**System**
1- Selects the desired program. *3- Command: Enroll Now*	*2- Displays the program details* *4- Displays the payment gateway* *5- Updates Enrollment Status*
Alternative and exceptional flows: **4.1-** *IF the payment rejected THEN cancel the request*	
Post-conditions: *Student notified*	

Use Case Description	
System: *Flexible virtual school*	
Use Case name: *Conduct class*	
Primary actor: *Instructor*	**Other actors:**
Stakeholders:	
Description: *The instructor will conduct a class.*	
Relationships **▪Includes:** **▪Extends:**	
Input: *class time and materials.*	
Pre-conditions: *Instructor logged in.*	
Steps:	

Actor	System
1. Initiates the class session. *3.Command: Start class.*	*2. System retrieves the materials.* *4. Displays the teaching materials on the screen.*

Alternative and exceptional flows:	
Post-conditions: *Students notified.*	

User interface

Start page

10:45

UNIVERSAL
Training Academy

Welcome To Arabic
Universal Academy

Swip-up to start

Sign in

10:45

UNIVERSAL
Training Academy

Sign-In

Username

Password

Do you forgot your password ?

Login   Sign-up

Sign-up

UNIVERSAL
Training Academy

Sign-Up

Your E-mail

Username

Password

Repeat-password

Sign-In   Register

Sign-up

10:45

Search . . . .

Diplomas

Programs

Contact Us

📞 0544030333

✉ Email@address.com

UNIVERSAL
Training Academy

Diplomas page

10:45

Search . . . .

Diploma in Arabic Language:

This 1-year programme in Arabic Language is designed to consolidate your previous knowledge at an upper intermediate to advanced level. You will be exposed to Arabic language, as spoken by native speakers, both in professional and day to day life situations.

40hrs

Introductory Certificate in Arabic Language:

This 1-year beginners programme in Arabic Language is designed to introduce the language to complete beginners or those with a basic knowledge. Develop your listening, reading, writing and speaking skills in this fascinating language.

60hrs

Arabic Language Taster Sessions:

This 1-year programme in Arabic Language is designed for intermediate learners, building on your previous language skills whilst also introducing the cultural aspects of the language.

60hrs

Contact Us

📞 0544030333

✉ Email@address.com

UNIVERSAL
Training Academy

Instructor page

10:45

Diploma in Arabic Language:: | 40hrs

Instructor: Talal Alothman

📍 Delivery method:
Online, real-time classrooms,
24/7 access to curriculum

📅 Start date:
September and throughout
the year

ⓘ Exam details:
8 exams

$ Tuition fees:
5000$

Back | Apply Now

Contact Us

📞 0544030333

✉ Email@address.com

UNIVERSAL
Training Academy

Course page

UNIVERSAL
Training Academy

Home   About   Arabic

Instructor: Talal Alothman
Rating: 3.9 ★★★⯪★

Courses:

Diploma in Arabic Language.

Introductory Certificate in Arabic
Language.

Arabic Language Taster Sessions.

Conduct virtual classes.

Upload course materials.

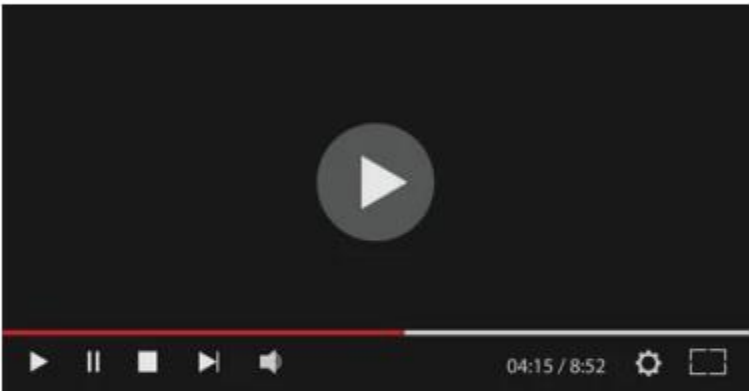Assess student performance.

Assign exams.

Manage exams.

Contact Us

✉ Email@address.com   ☎ 0544030333   📍 Visit Riyadh/Arqh

🐦 📷

Materials page

Universal Training Academy

Home   About   Arabic

content

Chapter 1   ⌄
Chapter 2   ⌄
Chapter 3   ⌄
Chapter 4   ⌄
Chapter 5   ⌄
Chapter 6   ⌄
Chapter 7   ⌄
Chapter 8   ⌄
Chapter 9   ⌄
Chapter 10   ⌄
Chapter 11   ⌄

04:15 / 8:52

Quizzes   Homework   Previous lecture   Next lecture

Contact Us

Email@address.com   0544030333   Visit Riyadh/Arqh

Non-functional Properties:

Reliability: It is crucial for the virtual school as students and instructors rely on the platform for uninterrupted access to educational resources and live sessions. To address these needs, the architecture should provide redundancy and fault-tolerance mechanisms at every tier.

Scalability: It is essential to accommodate the growing number of students and resources within the virtual school platform. The architecture should be able to scale and handle increased traffic, data volume, and processing load. The 3-tier architecture supports scalability by providing multi-threading and reducing network traffic.

Usability: It is key in ensuring that users, including students and instructors, can easily navigate and utilize the virtual school platform. The architecture should prioritize intuitive user interfaces, responsive designs, and accessible features. The system aims to enhance user satisfaction and engagement.

Security: It is important in the virtual school platform because the platform handles sensitive user data and facilitates online interactions. To address these needs, the architecture should implement authentication mechanisms, data encryption, and secure communication protocols to protect user privacy and prevent unauthorized access or data breaches.

Performance: It is crucial for the virtual school platform as it directly impacts the user experience. Students and instructors rely on the platform for real-time interactions, content delivery, and seamless access to course materials. The architecture should optimize code, implement caching mechanisms, leverage content delivery networks.

Quality Assurance

To ensure quality, the development process will involve:

a. Reviews
- Regular code reviews such as peer reviews and walkthrough to ensure adherence to coding standards and design specifications.

- Design reviews at key milestones to evaluate architectural design conformance with requirements.

b. Verification
- This will be achieved by Unit testing, Integration testing, Component testing and regression testing and the use of code quality tools and version control systems to enforce standards and facilitate code review processes.

c. Validation

Client confidence in the system will be established by ensuring:

- **Requirements validation:** Regular communication and getting feedback from the client will ensure that requirements are accurately captured and met throughout the development process.

- **Security validation:** Penetration testing and vulnerability assessments will be conducted to identify and address security vulnerabilities, ensuring that the software is secure.

- **Reusability and expandability validation:** Abstraction, and well-defined interfaces will enhance code reusability and expandability. The architecture will be designed to accommodate future enhancements and extensions.

- **Performance validation:** Performance testing will be conducted to evaluate system responsiveness, scalability, and resource utilization.

d. Acceptance Criteria

i)   **Procedures:** Installation procedures will be provided in user-friendly documentation, including step-by-step instructions.
ii) **Testing:** An acceptance test will be conducted in addition to all the other tests to ensure the system meets quality standards and client requirements.
iii) **Training:** User training sessions and materials, such as manuals and tutorials, will be provided.
iv) **Documentation:** Technical documentation that will include system architecture diagrams and troubleshooting guides will be provided to help users and admins in understanding and maintaining the system.

Future Considerations

Based on our choice of design decisions and the architectural styles our system will be able to accommodate future changes.

The possible changes for our system can be:

1- **Integration of Additional Languages**: While the initial focus is on teaching Arabic, there may be a demand to expand language offerings to include other languages.

2- **Enhanced Virtual Classroom Features**: The system may evolve to include more advanced virtual classroom features, such as interactive whiteboards, or virtual breakout rooms for group activities and discussions.

3- User Interface Enhancements: Improving the user interface (UI) of the platform to make it more intuitive and user-friendly.

We made sure to choose the best architectural style so No changes will be made to the architecture.

The key parts of our system are the Interface, the processing unit, and the database.

The riskiest parts are the processing unit and the Database:

Processing Unit:

This part manages how the system works and responds to user actions.

Risks include errors that could crash the system or make it slow, and vulnerability to hackers.

Mitigation involves thorough testing, monitoring, and implementing security measures.

Database:

This part stores important information the system needs to function.

Risks include losing data, unauthorized access, and slow performance.

Mitigation involves securing data, ensuring it's accurate, and having backups.