

# Flexible Virtual School

# Table of Contents

Introduction.....	2
Detailed design.....	3
a) Detailed system architecture .....	3
b) Detailed structural model.....	6
Detailed Object diagram .....	6
analysis diagram.....	6
view of participating classes (VOPC).....	8
c) Dynamic model.....	10
Sequence Diagram .....	10
State Diagram.....	12

## Revision Control History

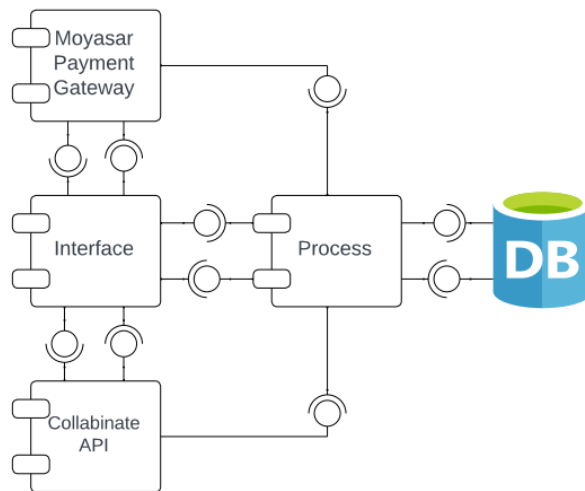
Version	Date	Description of change

## Introduction

This design document outlines the creation of a Flexible Virtual School by the Arabic Universal Academy. The aim is to use modern internet tools to offer Arabic language education globally. This project will develop a user-friendly platform with diverse programs, flexible scheduling, interactive tools, and quality instruction. The goal is to make Arabic language learning accessible and effective for learners worldwide through innovative online methods.

# Detailed design

## a) Detailed system architecture



Component Name	Process
Description	This component manages the logic and flow of operations within the system, including handling student requests, processing refunds, and coordinating interactions between the interface and database systems.
Properties/data	Course enrollment queue, Scheduling algorithm and Resource allocation rules.
Behavior/functionality	<ul style="list-style-type: none"><li>- It processes student enrollment requests, verifies eligibility, and assigns students to appropriate courses based on availability and preferences.</li><li>- It generates class schedules, assigns instructors, and notifies students of class times</li><li>- It records assessment scores and feedback.</li><li>- It allocates virtual classrooms, instructors, and learning materials according to demand and availability.</li></ul>
Connectors and Interfaces	<ul style="list-style-type: none"><li>- It is connected to the Interface component via remote method invocation to receive requests submitted by users.</li><li>- It is also connected to the Database component via remote method invocation to retrieve any needed data.</li></ul>
Dependencies	<ul style="list-style-type: none"><li>- Interface component to receive inputs from users.</li><li>- Database component to access and update data.</li></ul>
Resources	Interface component.

<b>Component Name</b>	Interface
<b>Description</b>	It provides user interfaces for students, instructors, and administrators to interact with the virtual school system.
<b>Properties/data</b>	Progress tracking dashboards, Virtual classroom interface, Course catalogs, User profiles and Enrollment forms.
<b>Behavior/functionality</b>	<ul style="list-style-type: none"> <li>- It allows users to log in to their profile and manage their profile.</li> <li>- It allows students to browse courses, select, and enroll, interacting with the Process component to submit enrollment requests.</li> <li>- It allows instructors to access virtual classrooms for conducting online classes, managing course materials, and interacting with students in real-time.</li> <li>- It enables instructors and students to track students' course progress.</li> <li>- It facilitates communication between users.</li> </ul>
<b>Connectors and Interfaces</b>	<ul style="list-style-type: none"> <li>- It is connected to the Process component via remote method invocation enabling users to access functionalities.</li> <li>- It is also connected to external APIs which are Collabinate API and Moyasar payment gateway.</li> </ul>
<b>Dependencies</b>	Process component to access functionalities.
<b>Resources</b>	Users.

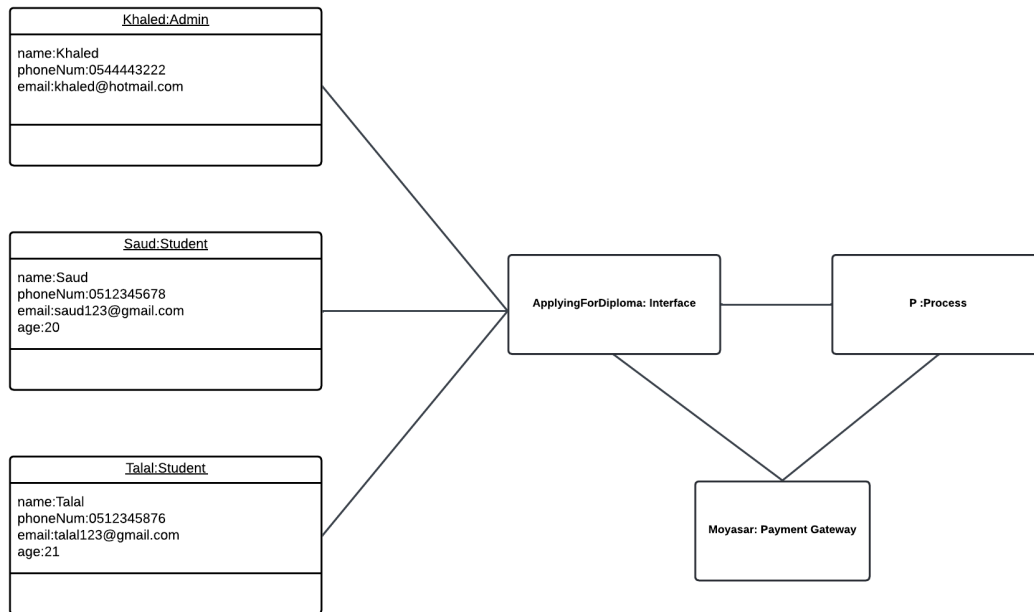
<b>Component Name</b>	Database
<b>Description</b>	It manages the storage, retrieval, organization, and manipulation of data related to the virtual school.
<b>Properties/data</b>	It contains structured data organized into tables, with each table representing a specific entity of the virtual school.
<b>Behavior/functionality</b>	It provides data insertion, retrieval of specific information from the database, modification of existing records and removal of records.
<b>Connectors and Interfaces</b>	It is connected to the Process component via remote method invocation to facilitate data retrieval, updates, and validation.
<b>Dependencies</b>	None
<b>Resources</b>	Process component

<b>Component Name</b>	Collabinate API
<b>Description</b>	It is designed to integrate real-time voice, video, and interactive streaming functionalities into the virtual school platform.
<b>Properties/data</b>	Virtual classrooms (Channels), Media controls, Chat messages and Audio and Video streams.
<b>Behavior/functionality</b>	<ul style="list-style-type: none"> <li>- It facilitates classes and lectures in real-time voice and video communication between instructors and students during live classes.</li> <li>- It allows students to ask questions, participate in discussions, and collaborate with classmates and instructors in real-time using voice, video, and chat messages.</li> <li>- It supports the sharing of multimedia content</li> </ul>
<b>Connectors and Interfaces</b>	It is connected to the Interface component via remote method invocation.
<b>Dependencies</b>	Interface component
<b>Resources</b>	Collabinate API system

<b>Component Name</b>	Moyasar Payment Gateway
<b>Description</b>	It is a service based in Saudi Arabia. It provides a platform to accept online payments securely and conveniently and it supports various payment methods.
<b>Properties/data</b>	Payment methods, transaction data, user card information and financial records.
<b>Behavior/functionality</b>	<ul style="list-style-type: none"> <li>- It facilitates secure payment processing through encryption and compliance.</li> <li>- It verifies payment details and authorizes transactions in real-time.</li> <li>- It generates payment confirmation messages or receipts.</li> <li>- It supports recurring payments and automated billing for subscription-based services.</li> </ul>
<b>Connectors and Interfaces</b>	It is connected to the Interface component via remote method invocation.
<b>Dependencies</b>	Interface component
<b>Resources</b>	Moyasar Payment Gateway system

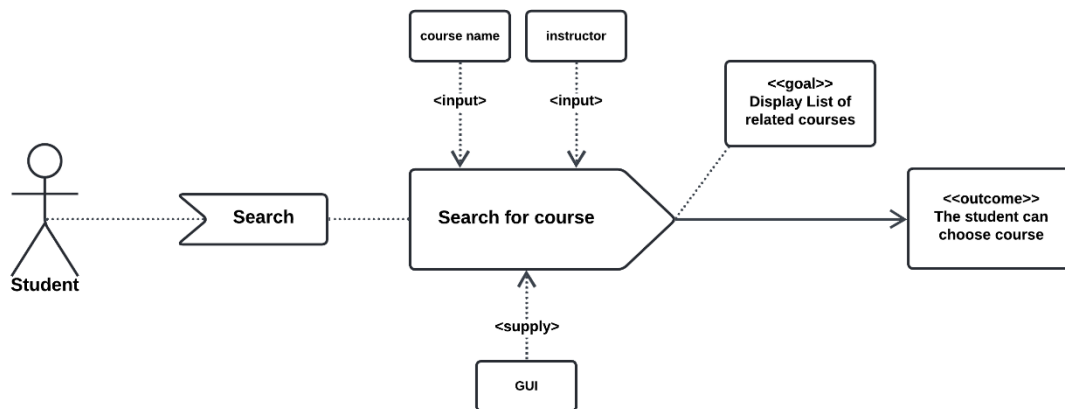
## b) Detailed structural model

### Detailed Object diagram

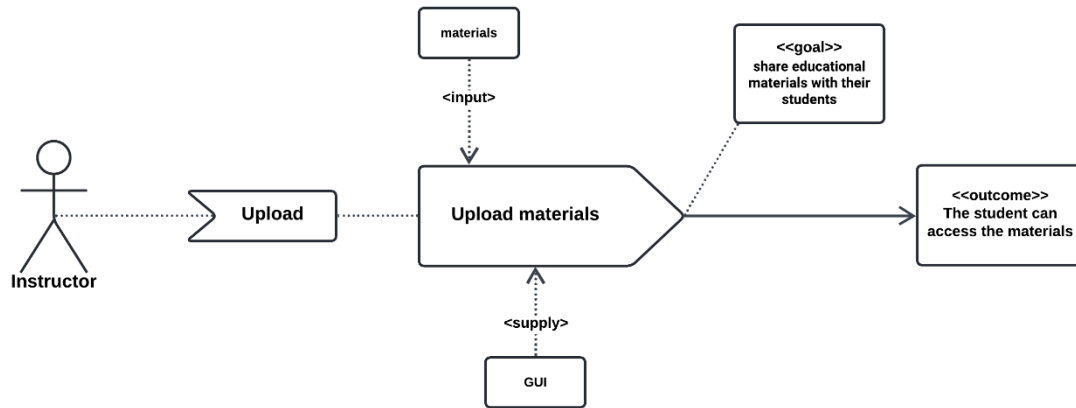


### analysis diagram

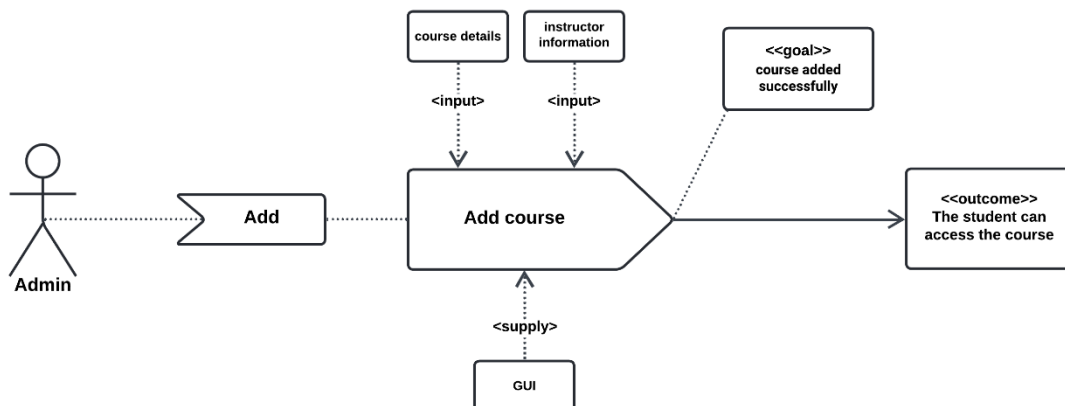
#### Search for course



## Upload materials

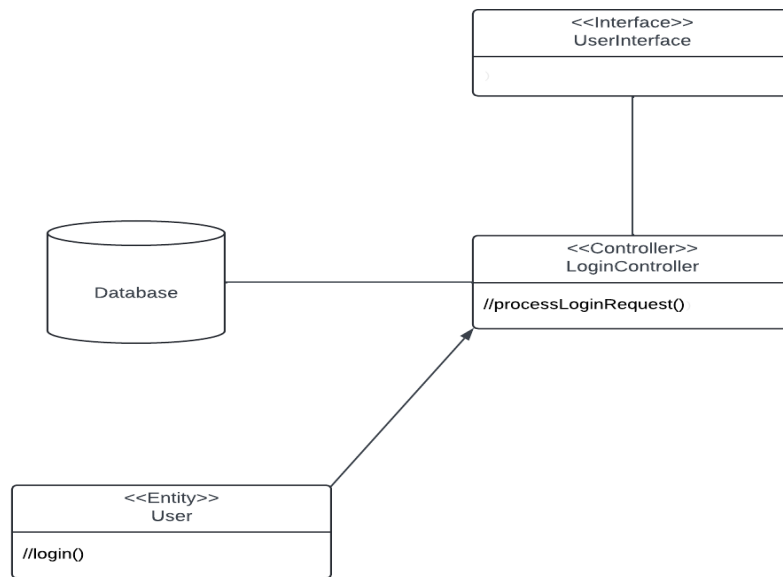


## Add Course

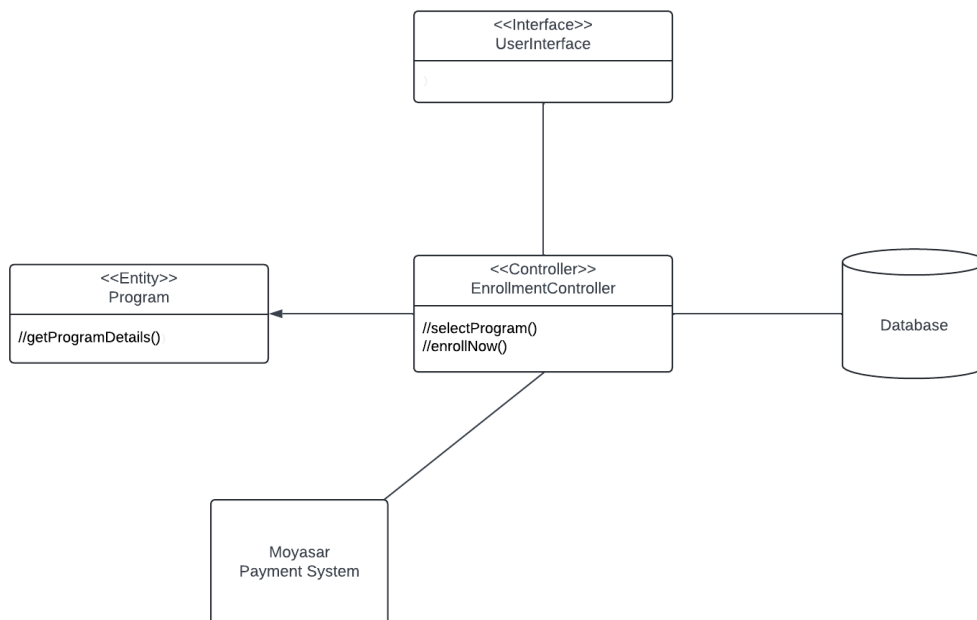


## view of participating classes (VOPC)

### User login VOPC

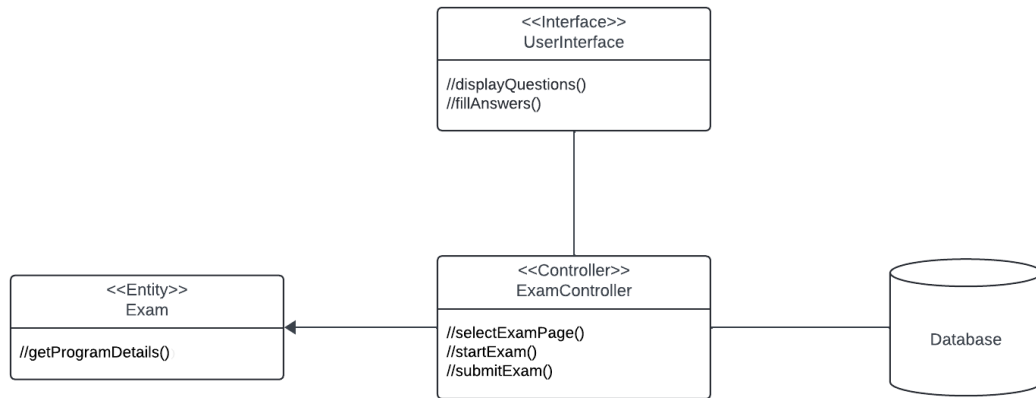


### Student enrollment in program VOPC





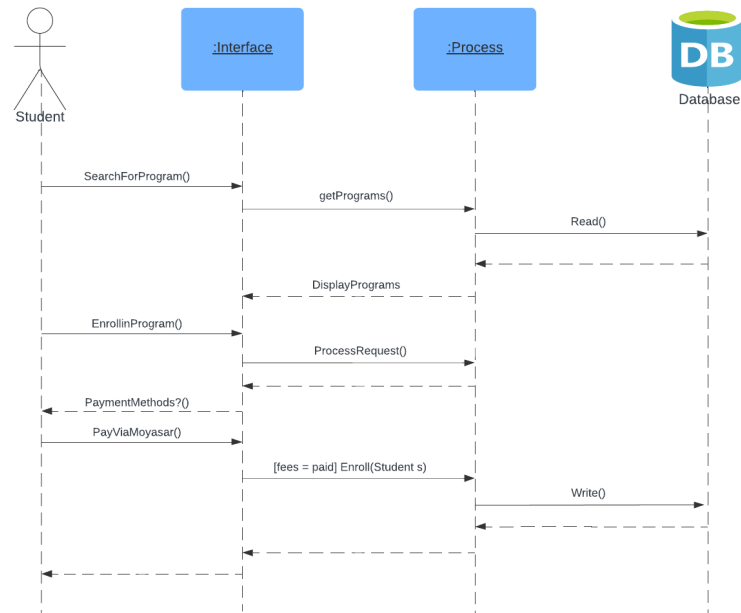
## Student taking an exam VOPC



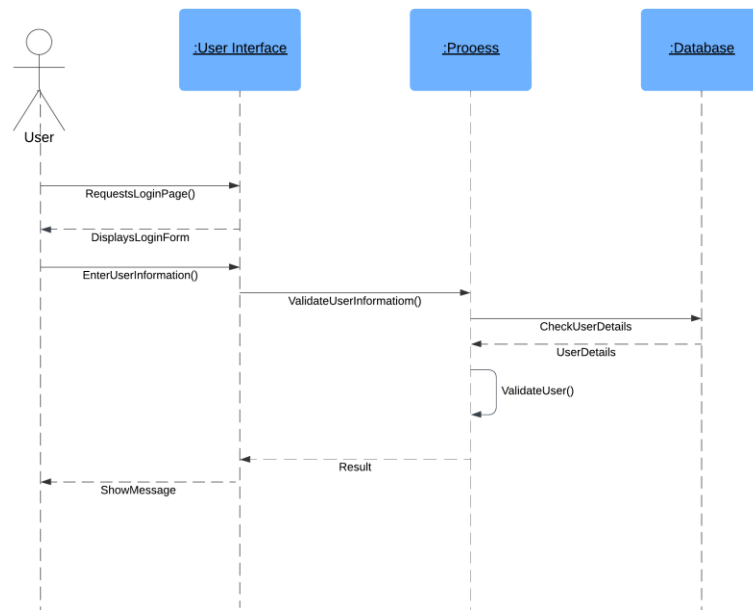
## c) Dynamic model

### Sequence Diagram

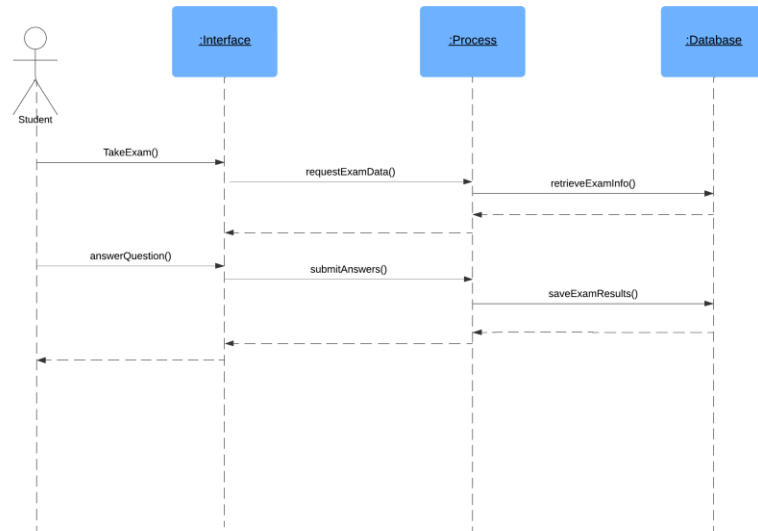
Sequence Diagram for Student enrollment in a program



Sequence Diagram for User log in

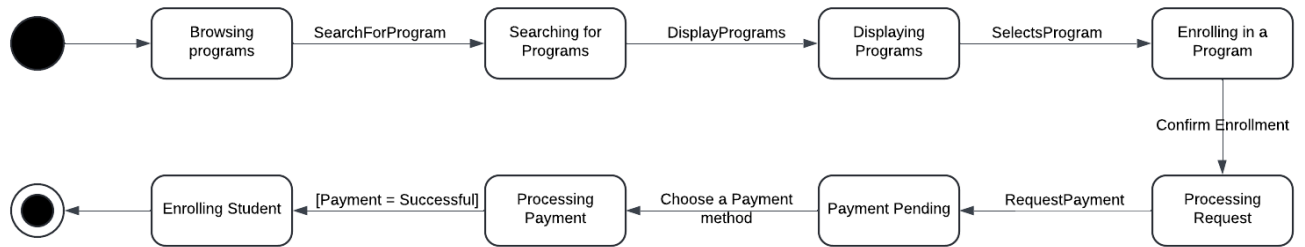


## Sequence Diagram for Student taking an exam

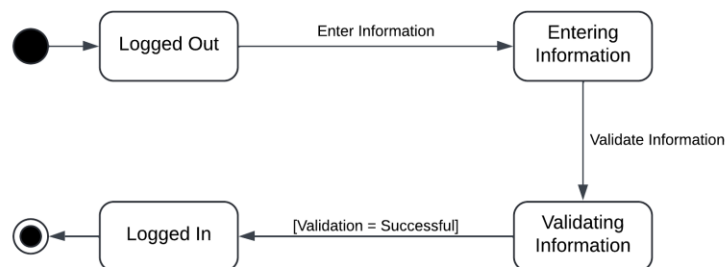


# State Diagram

## State Diagram for Student enrollment in a program



## State Diagram for User log in



## State Diagram for Student taking an exam

