

**PROJECT REPORT**

# Online Courses

|  |  |
| --- | --- |
| **Semester:** | Programming Fundamentals |
| **Class:** | PF1123 |
| **Group:** | Group 1 |
| **Instructor** | Tran Nguyen Hoang |
| **Team members:** | 114010123003 – Quan Van Vi  114010123008 – Ha Quang Anh |

# Index

Online Courses 1

Index 2

I. Project introduction 4

II. Analyze System Requirements……………………………………………………..5

1. Use Case 6
   1. Use Case 6
   2. Login Description 7
   3. Search courses description 8
   4. Register course description 9
   5. Create course description 11
   6. Rate a course description 13
   7. Update profile description 15
   8. Access course content description 16
   9. View course details description 17

2. Activity Diagram 18

2.1. Search courses 18

2.2. Create course 19

2.3. Evaluate a course 20

2.4. Access course content(Learing) 21

2.5. Payment course 22

III. Design Details 23

1. UI Design 23

1.1 Main access page 23

1.2 Login 23

1.3 Register 23

1.4 Home page 24

1.5 User menu 24

1.6 Payment history 24

1.7 User details 25

1.8 List of courses 25

1.9 Payment 25

1.10 Learning page 26

1.11 Lesson content 26

1.12 Instructor menu 26

1.13 Create a course 27

2. Code Design (Class Diagram) 28

3. Sequence Diagram 29

3.1 Search courses 29

3.2 Create course 30

3.3 Evaluate a course 31

4. Database Design 32

4.1 ERD 32

4.2 Database Design Details 33

IV. Task Assign (to each team member) 37

V. Installation Instructions 38

# Project introduction

"The Online Course Platform" provides an efficient and user-friendly management and learning system for instructors and students. Key features include course management, enrollment, search, and progress tracking, optimizing the user experience.

1. Proposed System

* Creating a program to Instructor create and manage courses.
* Creating a program to Student enrollment and course tracking.

1. The scope of the project to be applied

* Make for actor are Instructor, Student and Guest.

1. System Name

* Online courses management system

1. Deployment Environment

* Windows, Linux, Mac

1. Development Tools

* MySQL Workbench 8.0.34
* Draw.io
* Visual Studio Code

1. Customer Requirements (System features)

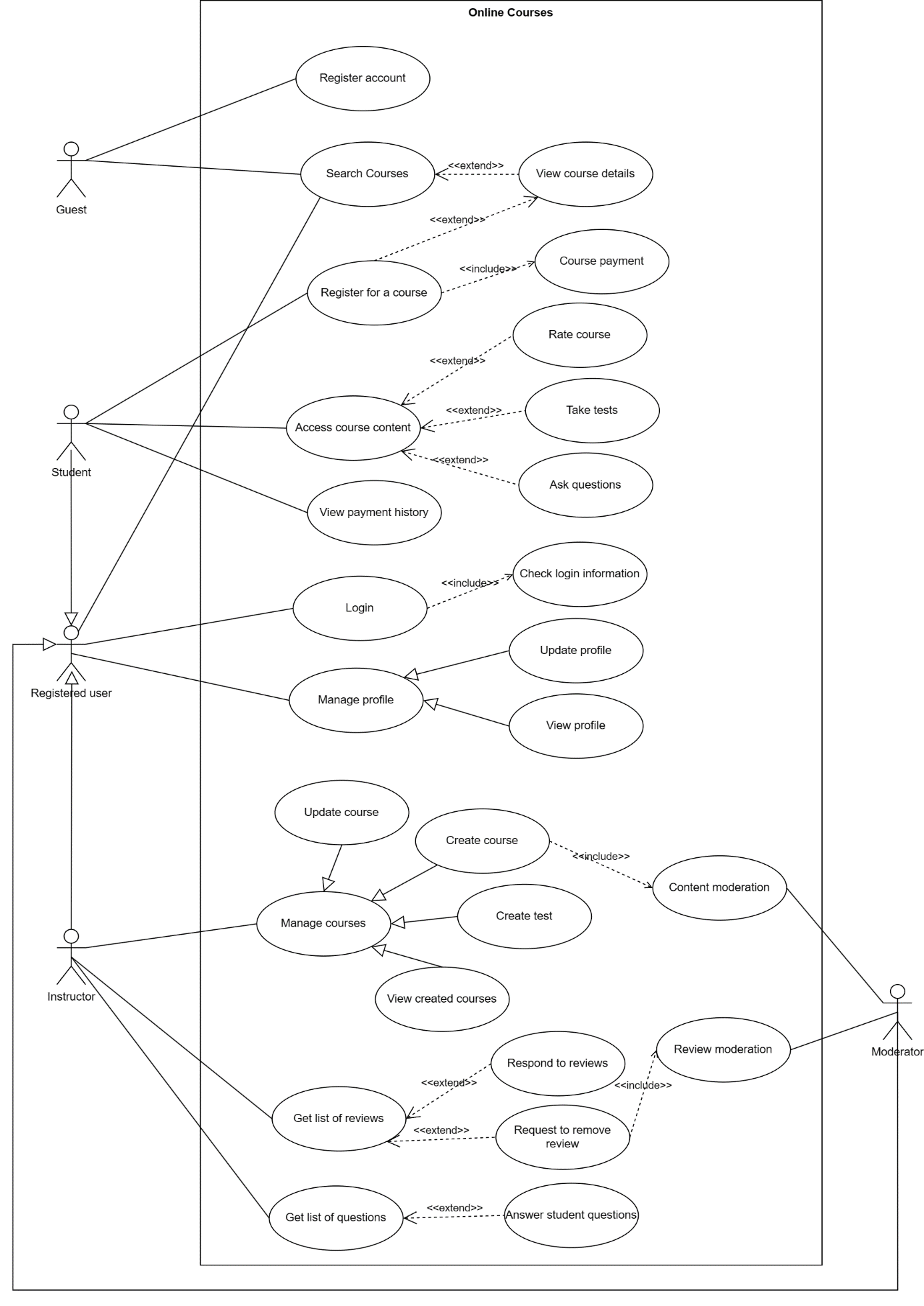
* User (Student, Instructor) login
* Show courses
* View course details
* Register the course
* Learning and learning progress
* Course evaluation
* Account management
* Course content management

# Analyze System Requirements

The online course system aims to provide a diverse range of courses for users, and users can register as instructors to create courses. The main features of the project include course management, online registration and payment, interaction between instructors and students, and student management. Students can search, register, and participate in courses according to their interests and needs, experiencing materials such as videos and exercises. Instructors can create and manage their own courses, as well as track the progress of students' learning.

1. Use Case

1.1 Use Case



1.2 Login description

|  |  |
| --- | --- |
| Use Case Name | Login |
| Use Case ID | UC\_001 |
| Description | Used to access the program and define the feture accessible depending on the type of account. |
| Actor | Student and Instructor |
| Organizational Benefits | Security: Ensures that only authorized users can access the system and personal data.  Customization: Allows for providing a customized experience based on the information and access rights of each user. |
| Triggers | The user selects the "Login" option on the console interface. |
| Preconditions | The user already has an account in the system.  The system is operational and can access the user database. |
| Postconditions | If successful, the user is directed to the main screen of the system with authenticated access.  If unsuccessful, the user receives an error message and may try again or recover the password. |
| Main Course | 1. The user enters the username and password. 2. The system authenticates the login information. 3. If the information is valid, the system logs the user in and directs them to the main screen. 4. The user now has access to functions and data based on their permissions. |
| Alternate Courses | AC1: Incorrect Login Information: User enters wrong username, password  1. Display an error message  2. Show the login screen again  3. Request re-entry |
| Exceptions | System Error: Connection error or a database issue prevents successful login.   1. Display “Connection error or a database” 2. Request to recheck internet connection |

1.3 Search courses description

|  |  |
| --- | --- |
| Use Case Name | Search courses |
| Use Case ID | UC\_002 |
| Description | The "Search Courses" feature enables users to search and find courses within the system based on various criteria such as keywords, average rating, price or category. This function aims to facilitate users in efficiently locating the courses that best match their interests or educational needs. |
| Actor | Guest and Registered User |
| Organizational Benefits | Enhances course discoverability, improves user experience, and potentially increases course enrollment. |
| Triggers | The user selects the "Search courses" option on the console interface. |
| Preconditions | The system must have a database of courses with searchable data (titles, descriptions, category, level etc.). |
| Postconditions | Success: Displays a list of courses matching the search criteria.  Fail: Informs the user no matching courses were found. |
| Main Course | 1. User accesses the search interface. 2. User enters search criteria (e.g., course name, category, level). 3. System processes the search and returns a list of matching courses. 4. User reviews the search results. |
| Alternate Courses | AC1: Advanced Search - User applies filters for more refined results. |
| Exceptions | EX1: Invalid Input - User enters unsupported characters or formats in the search field.   1. Display message error “Invalid input” 2. Request to re-enter search criteria   EX2: Disconnect to database   1. Display “Connection error or a database” 2. Request to recheck internet connection |

1.4 Register course description

|  |  |
| --- | --- |
| Use Case Name | Register course |
| Use Case ID | UC\_003 |
| Description | Allows students to enroll in a course by completing payment. The process includes selecting a course, proceeding to payment, and upon successful transaction, being officially registered and gaining access to the course materials. |
| Actor | Registered User |
| Organizational Benefits | Secures immediate funding for course operations.  Ensures only paid students are enrolled, maintaining financial viability. |
| Triggers | A user selects the "Enroll" option for a course within the system. |
| Preconditions | The course selected to register is still available.  User must be logged into their authenticated account. |
| Postconditions | Success: The student's payment is processed, they are enrolled in the course, and have access to the course content.  Fail: The student is notified of the failure, and no enrollment is processed. |
| Main Course | 1. Student navigates to the course they wish to enroll in. 2. Student selects the "Enroll" option. 3. System displays the course fee and payment options. 4. Student enters payment details and confirms the transaction. 5. Payment gateway processes the transaction. 6. Upon successful payment, the system registers the student for the course. 7. Student receives a confirmation email with access details to the course materials. |
| Alternate Courses | AC1: Payment Failure: If payment fails, the student is prompted to try a different payment method or to check their payment details.  AC2: Voucher: If the student has a voucher, they can apply this to offset the course fee before payment. |
| Exceptions | EX1:Technical Issue with Payment Gateway: The student is informed of technical issues preventing payment and is advised to retry after some time or contact support.  EX2: Insufficient Funds: If the student's payment method is declined due to insufficient funds, they are notified and asked to use a different method or ensure adequate funds are available.  EX3: Disconnect to database  1. Display “Connection error or a database”  2. Request to recheck internet connection |

1.5 Create course description

|  |  |
| --- | --- |
| Use Case Name | Create course |
| Use Case ID | UC\_004 |
| Description | Allows instructors or administrative users to create a new course within the system. This includes setting up course details, schedules, content, and enrollment fees. Upon creation, the course becomes available for students to register and requires payment to enroll. |
| Actor | Instructor |
| Organizational Benefits | Expands the educational offerings of the institution or platform.  Facilitates revenue generation by adding new paid courses. |
| Triggers | An instructor selects the option to create a new course in the system's course management section. |
| Preconditions | The user must have instructor-level access to the system.  The user must have all necessary information, including course descriptions, pricing, and scheduling details. |
| Postconditions | Success: A new course is created in the system, is made available for student registration, and is listed in the course catalog with the appropriate fee.  Fail: The user is notified of the failure, and the course is not created. |
| Main Course | 1. Instructor navigate to the course management area. 2. Instructor select the option to create a new course. 3. The system presents a form to enter course details, including title, description, and fee. 4. The instructor fills out the form and submits it. 5. The system validates the information and confirms successful creation. 6. The new course is listed in the course catalog with an associated fee required for registration. |
| Alternate Courses | AC 1: Invalid category id or Level or Price   1. Display error 2. Request re-entry |
| Exceptions | EX1: Unauthorized Access: If a user without the appropriate permissions attempts to create a course, the system denies the action and logs the attempt.  EX2: Disconnect to database  1. Display “Connection error or a database”  2. Request to recheck internet connection |

1.6 Rate a course description

|  |  |
| --- | --- |
| Use Case Name | Rate a course |
| Use Case ID | UC\_005 |
| Description | Allows students to provide a rating for a course they have enrolled in, which can include a star-based rating system and an optional text review. This feedback is visible to other users and can be used for improving course quality. |
| Actor | Registered User |
| Organizational Benefits | Feedback for Improvement: Provides valuable feedback to instructors for potential improvements.  Informed Decisions: Helps other students make informed decisions based on previous attendees' experiences.  Course Reputation: Builds the reputation of courses, aiding in the marketing and promotion strategy. |
| Triggers | Student completes a course or reaches a point where leaving a rating is permitted (e.g., after a certain percentage of course completion). |
| Preconditions | The student must be enrolled in and reaches a point where leaving a rating. |
| Postconditions | The student's rating is recorded and reflected in the course's overall rating.  If the rating is publicly visible, the student's feedback is now viewable by the course community. |
| Main Course | 1. Student navigates to the course page. 2. Student selects the "Rate this Course" option. 3. System presents the rating interface with a star-based system and an optional text review field. 4. Student submits a star rating and optionally adds a text review. 5. System validates and saves the rating and review. 6. System updates the course's average rating and displays the new rating and review. |
| Alternate Courses | AC 1: Invalid star rating (must be from 1 to 5)   1. Display error 2. Request re-entry |
| Exceptions | EX1: Ineligible to Rate: Student attempts to rate a course they are not enrolled in or have not reached the rating threshold for.   1. Display message error “Ineligible to Rate”   EX2: Disconnect to database   1. Display “Connection error or a database” 2. Request to recheck internet connection |

1.7 Update profile description

|  |  |
| --- | --- |
| Use Case Name | Update profile |
| Use Case ID | UC\_006 |
| Description | Allows users to update their personal information and account settings within their profile on the system. |
| Actor | Registered User |
| Organizational Benefits | Data Accuracy: Ensures user data is current.  System Integrity: Maintains the integrity and relevance of the user database. |
| Triggers | The user chooses to update their profile information, in the user profile section. |
| Preconditions | The user must be logged into their account. |
| Postconditions | The user's profile is updated with the new information provided.  Any changes are reflected immediately across the system where user information is displayed or utilized. |
| Main Course | 1. The user navigates to their profile page. 2. The user selects the option to update profile information. 3. The system presents a form with the current profile information. 4. The user makes desired changes to fields such as full name, contact information, bio, etc. 5. The user submits the changes. 6. The system validates and saves the updates. 7. The system confirms the successful update to the user. |
| Alternate Courses | AC1: Additional Verification: For sensitive changes (like email or password), the system requires additional verification before applying the changes. |
| Exceptions | EX1: Validation Errors: The information provided is in an incorrect format (e.g.,email format, phone format), and the user is prompted to correct it.   1. Display message error “Invalid input” 2. Request to re-enter   EX2: Disconnect to database   1. Display “Connection error or a database” 2. Request to recheck internet connection |

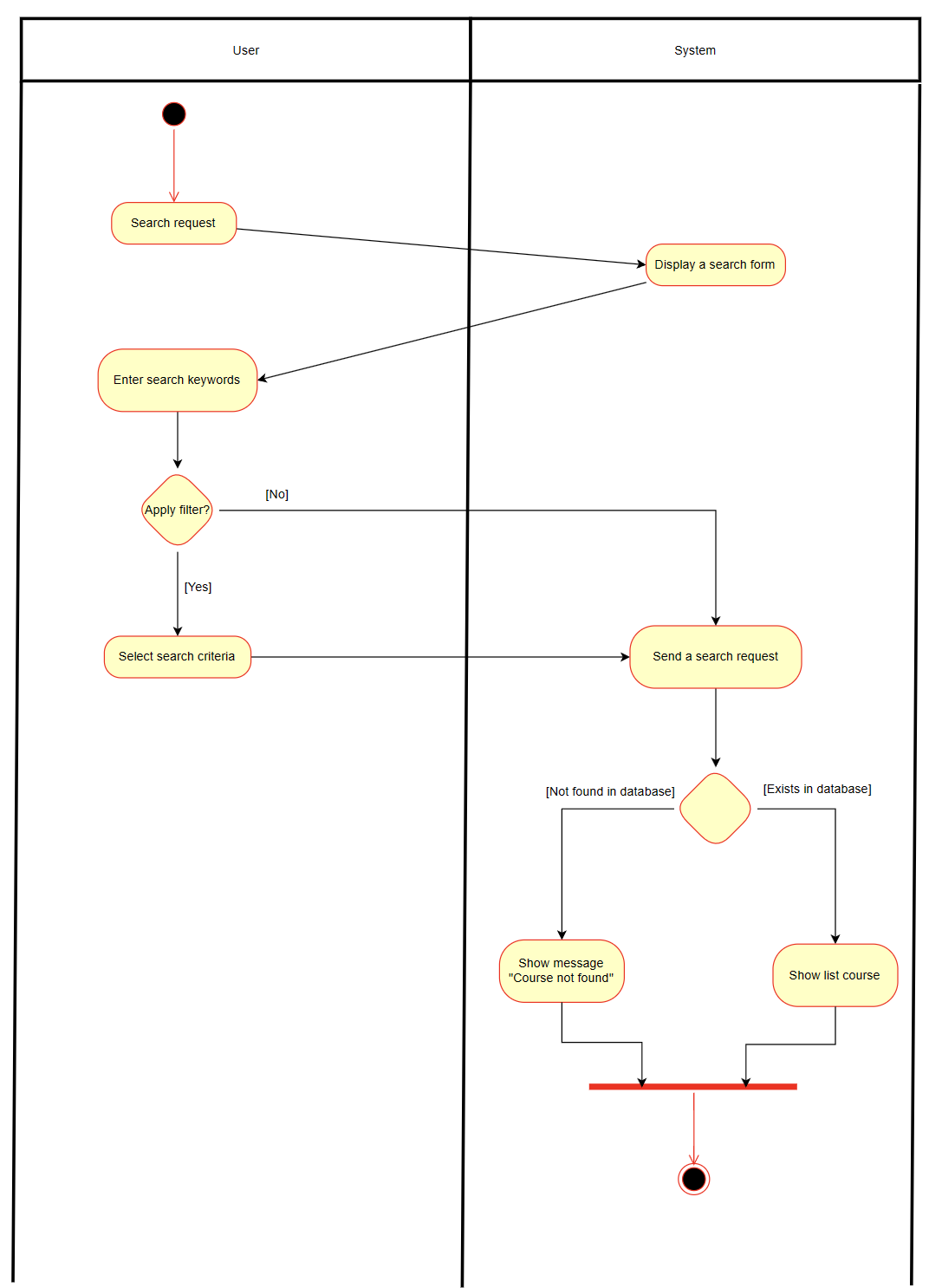
* 1. Access course content description

|  |  |
| --- | --- |
| Use Case Name | Access course content |
| Use Case ID | UC\_007 |
| Description | Allows enrolled students to access and navigate through the content of a course, including lectures, resources, assignments. |
| Actor | Registered User |
| Organizational Benefits | Increased Engagement: Easy navigation and availability of content can lead to higher engagement and course completion rates.  Content Utilization: Ensures that the educational content created is being utilized by the students. |
| Triggers | The student selects a course they are enrolled in from their dashboard or course list. |
| Preconditions | The student must be logged in and have an active enrollment in the course. |
| Postconditions | The student is able to view and interact with the course content.  The system records the student’s progress with the content. |
| Main Course | 1. Student logs into their account and navigates to the enrolled courses section (Go To My Learning). 2. Student selects a specific course to view. 3. Students choose a lesson to view, including video lectures and reading materials. 4. System provides access to the selected content. 5. Student consumes the content at their own pace, with the system tracking progress. |
| Alternate Courses | AC1: Content Restrictions: Some content is locked until the student completes prerequisite sections, requiring them to complete those before advancing. |
| Exceptions | EX1: Access Denied: Student tries to access content for a course they are not enrolled in.   1. Display message error “Access Denied” 2. The system navigates to the course registration page   EX2: Disconnect to database   1. Display “Connection error or a database” 2. Request to recheck internet connection |

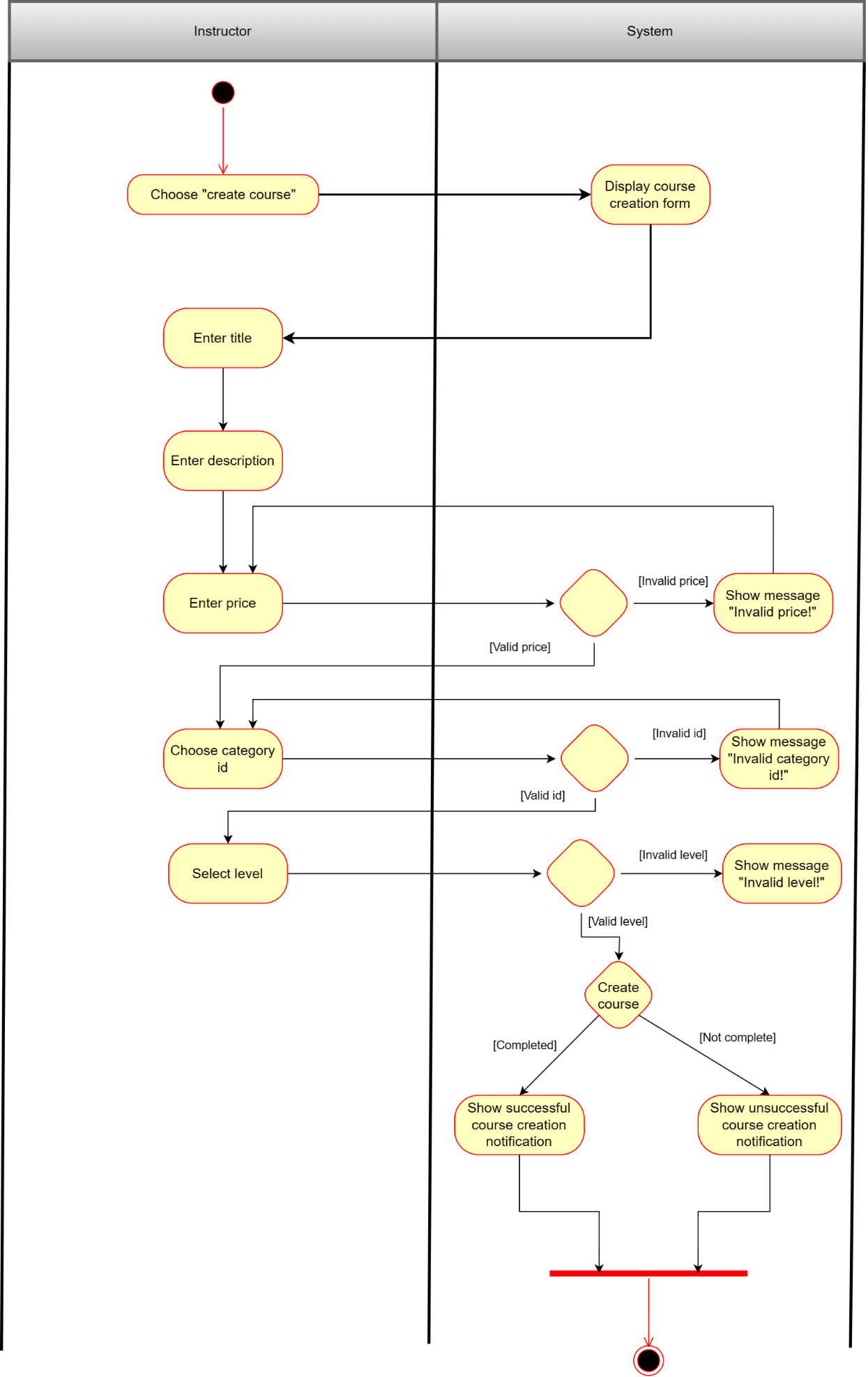
* 1. View course details description

|  |  |
| --- | --- |
| Use Case Name | View course details |
| Use Case ID | UC\_008 |
| Description | Provides a detailed view of the course content, including syllabus, price, and instructor information. |
| Actor | Guest and Registered User |
| Organizational Benefits | Increases transparency and informs decision-making for potential enrollees, which can improve the course enrollment rate. |
| Triggers | User navigates to the course and requests more information. |
| Preconditions | The course exists in the system and the user has access to view course listings. |
| Postconditions | User is presented with complete course details. |
| Main Course | 1. User selects a course from the listing. 2. System displays detailed information about the course. 3. User can read through the details and decide to register, if interested. |
| Alternate Courses | AC1: Content Restrictions: Some content is locked until the student completes prerequisite sections, requiring them to complete those before advancing. |
| Exceptions | EX1: Course details are unavailable or restricted due to system error or privacy settings; user is informed and provided with alternate contact options.  EX2: Disconnect to database   1. Display “Connection error or a database” 2. Request to recheck internet connection |

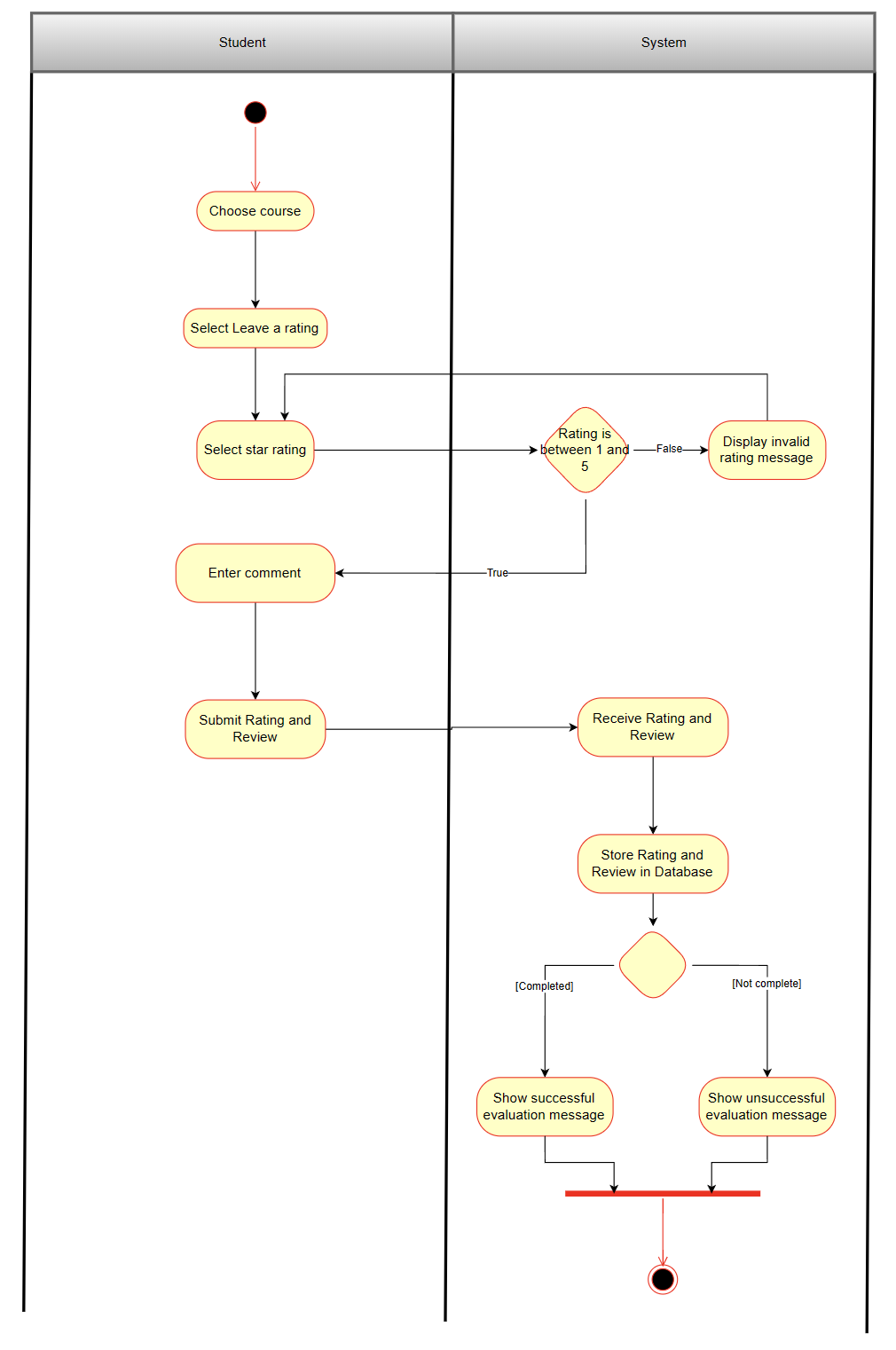
1. Activity Diagram:
   1. Search courses



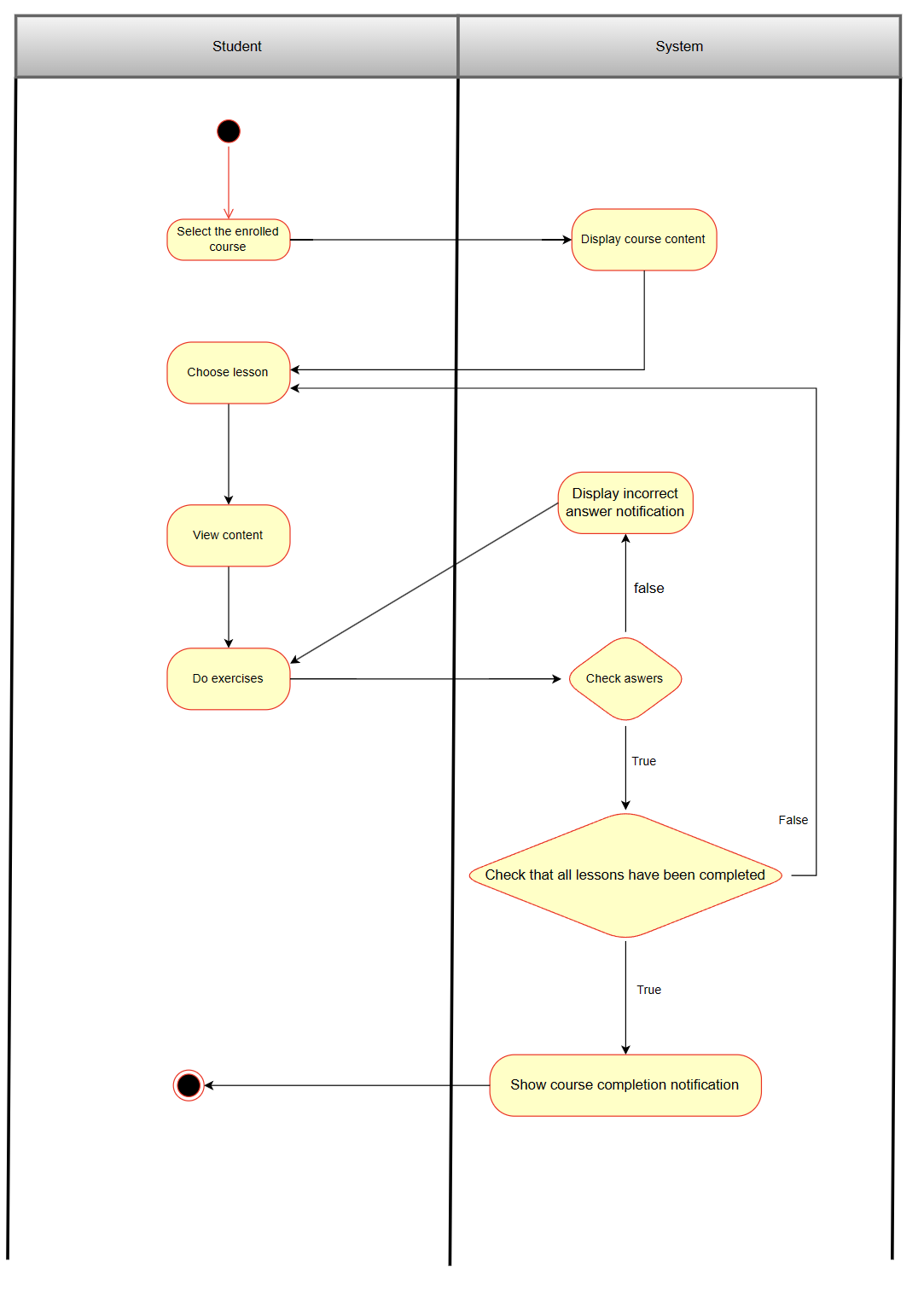
* 1. Create course



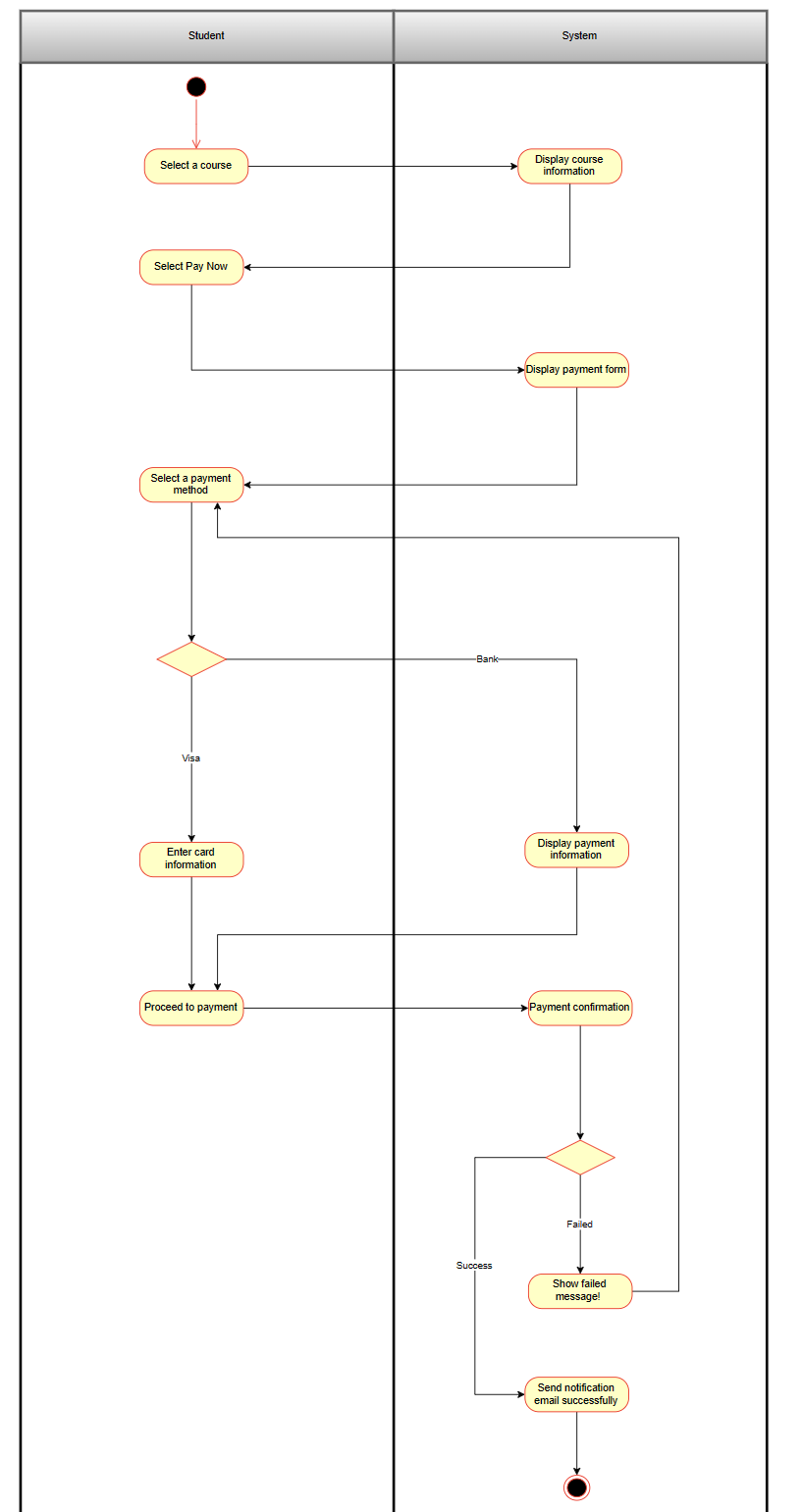
* 1. Evaluate a course



* 1. Learn



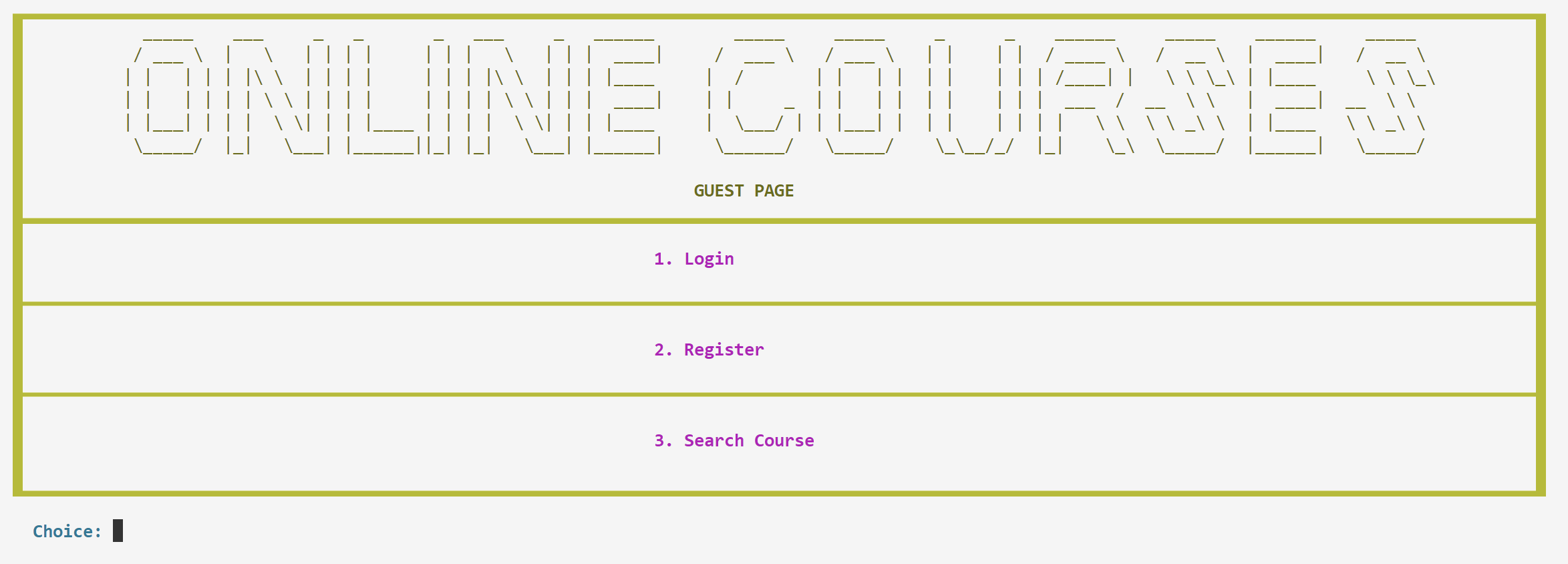
* 1. Payment course



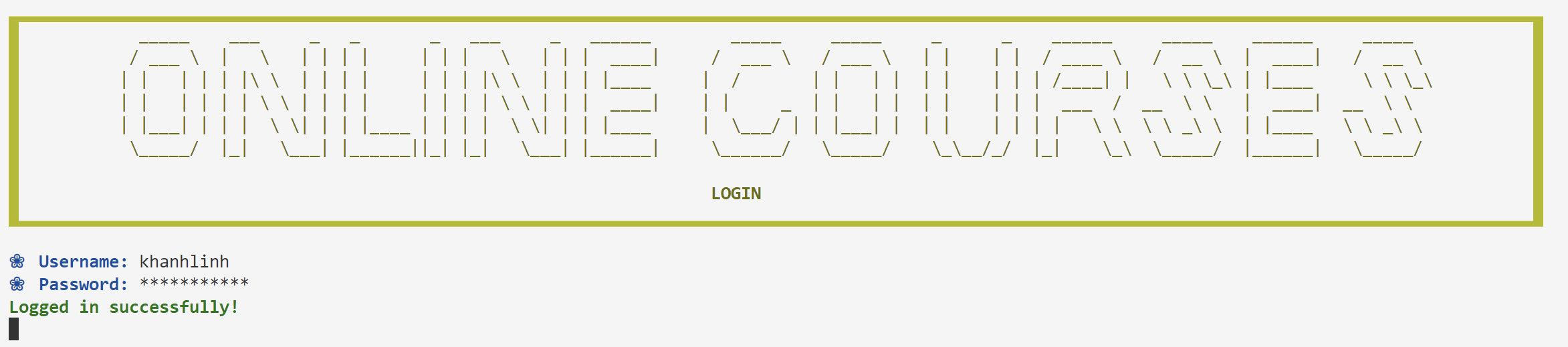
# Design Details

1. UI Design

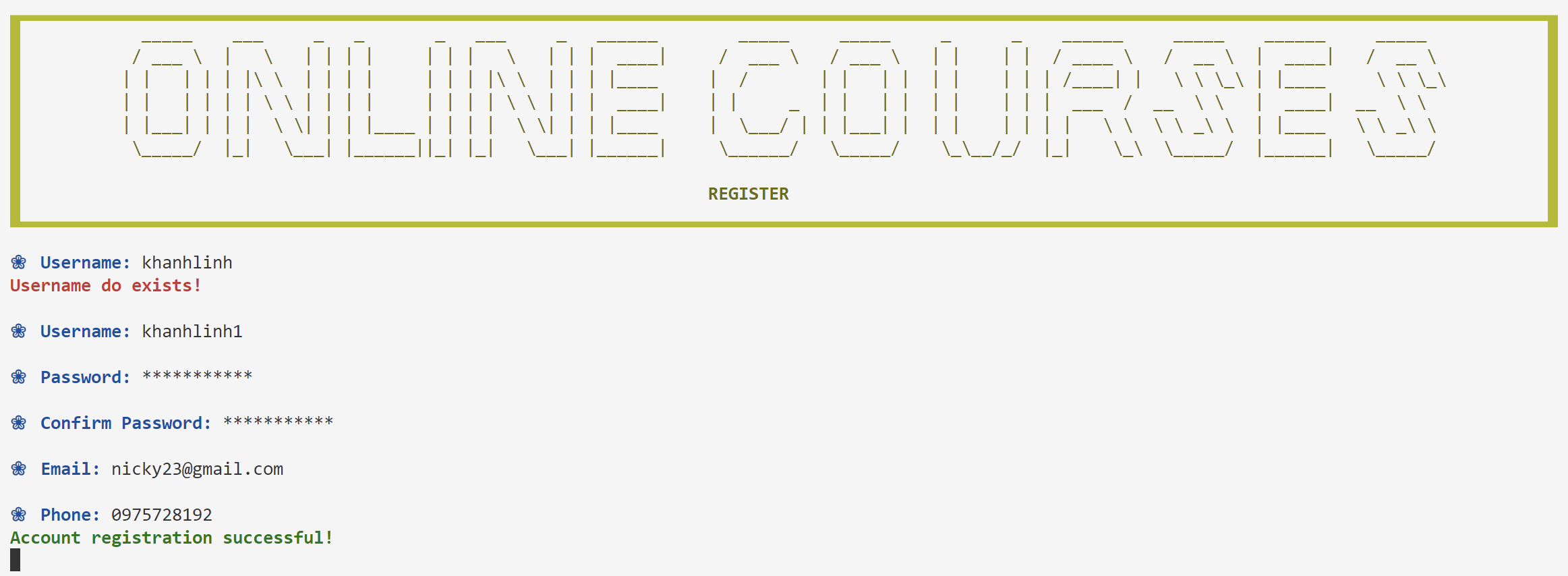
1.1 Main Access Page



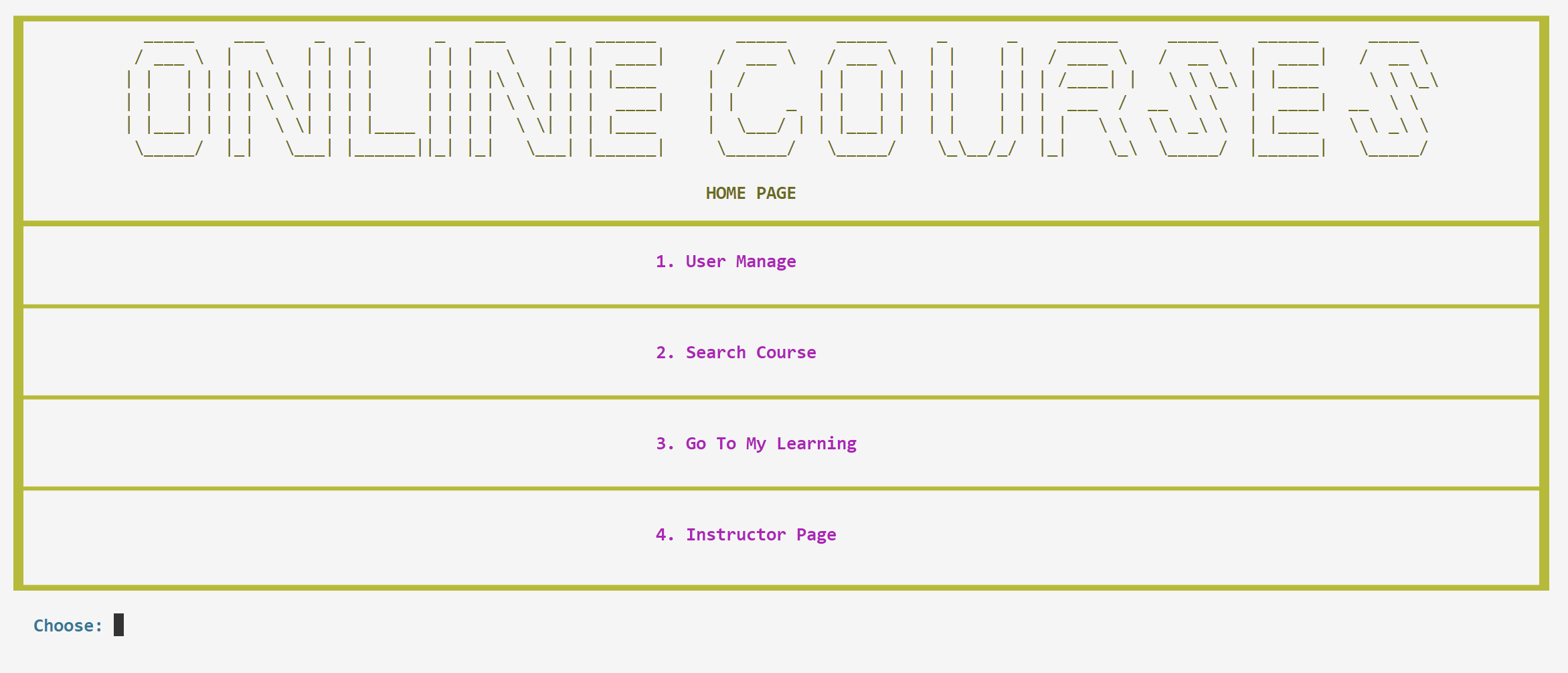
1.2 Login



1.3 Register



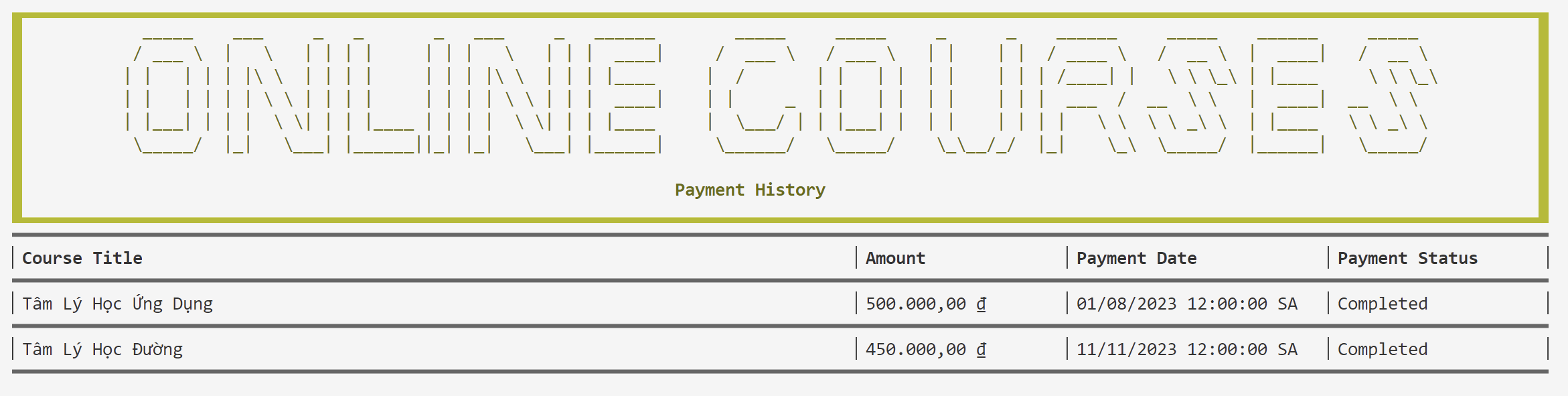
1.4 Home page



1.5 User menu



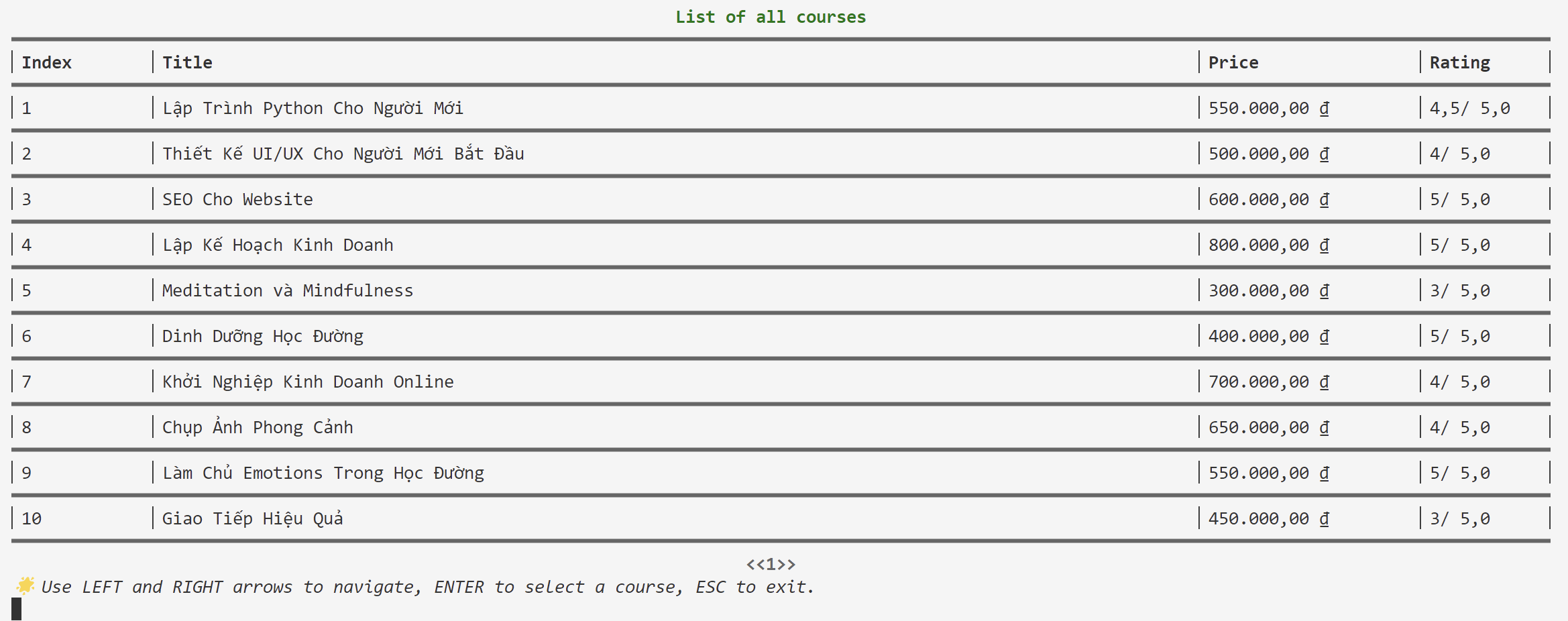
1.6 Payment history



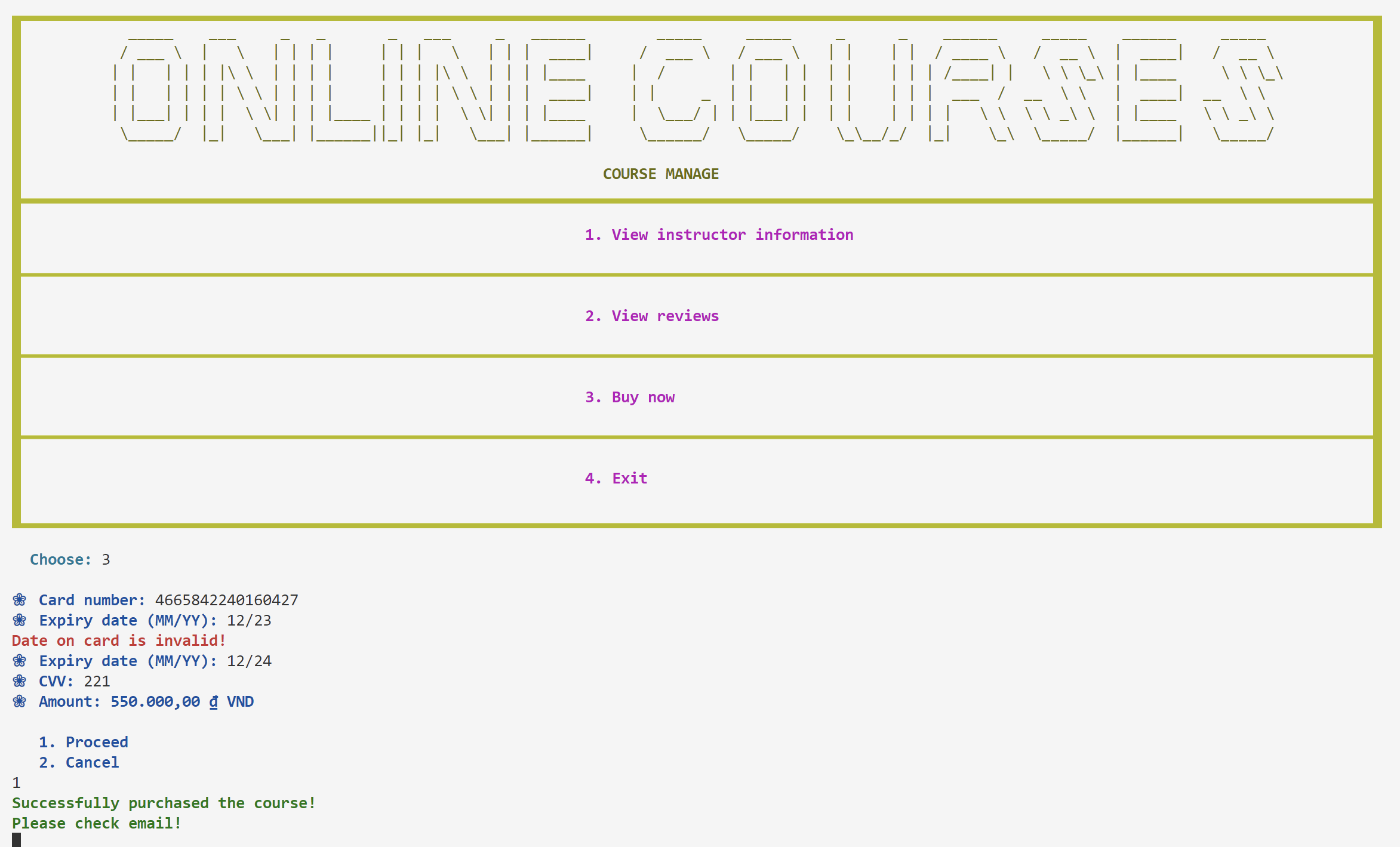
1.7 User details



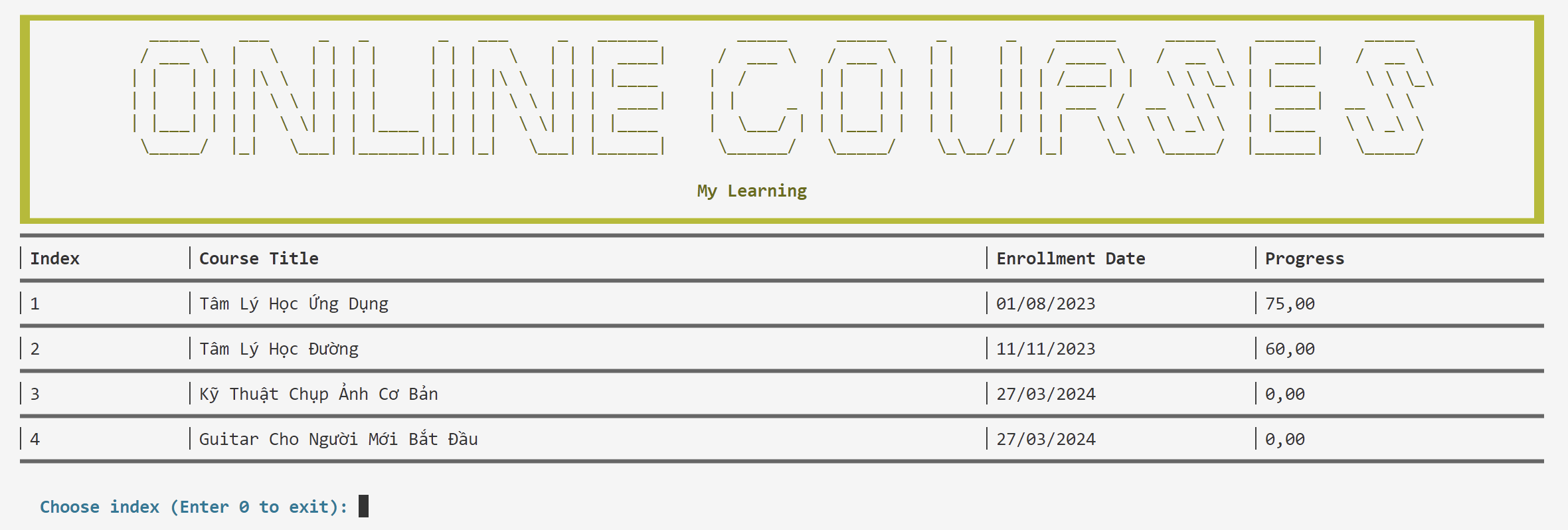
1.8 List of courses



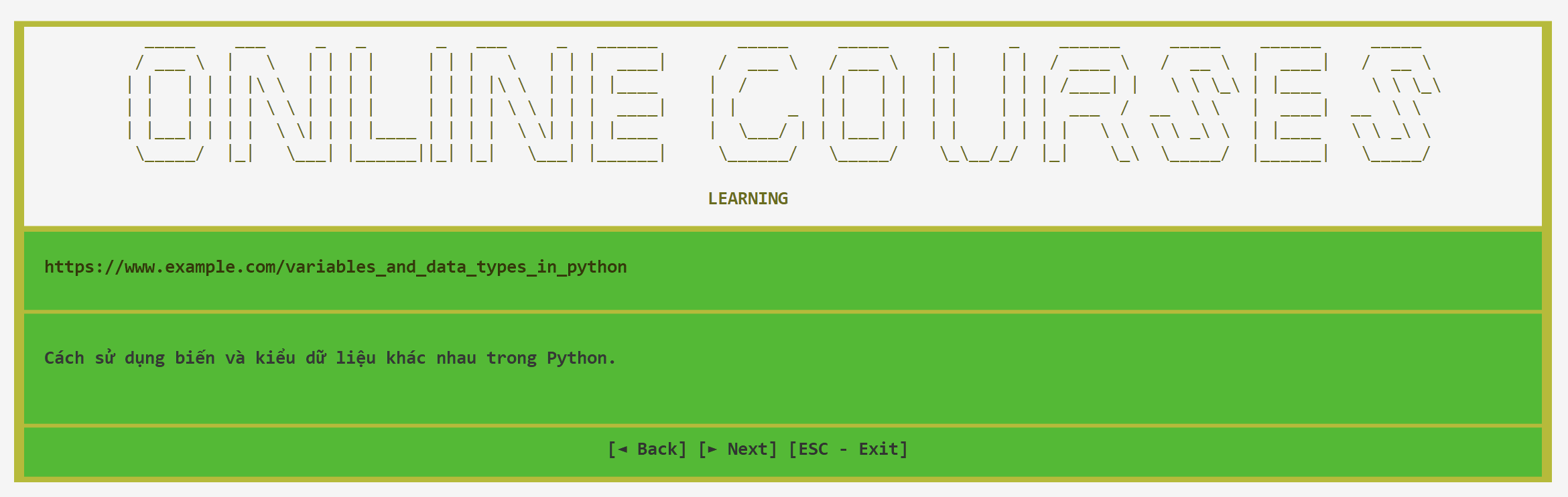
* 1. Payment



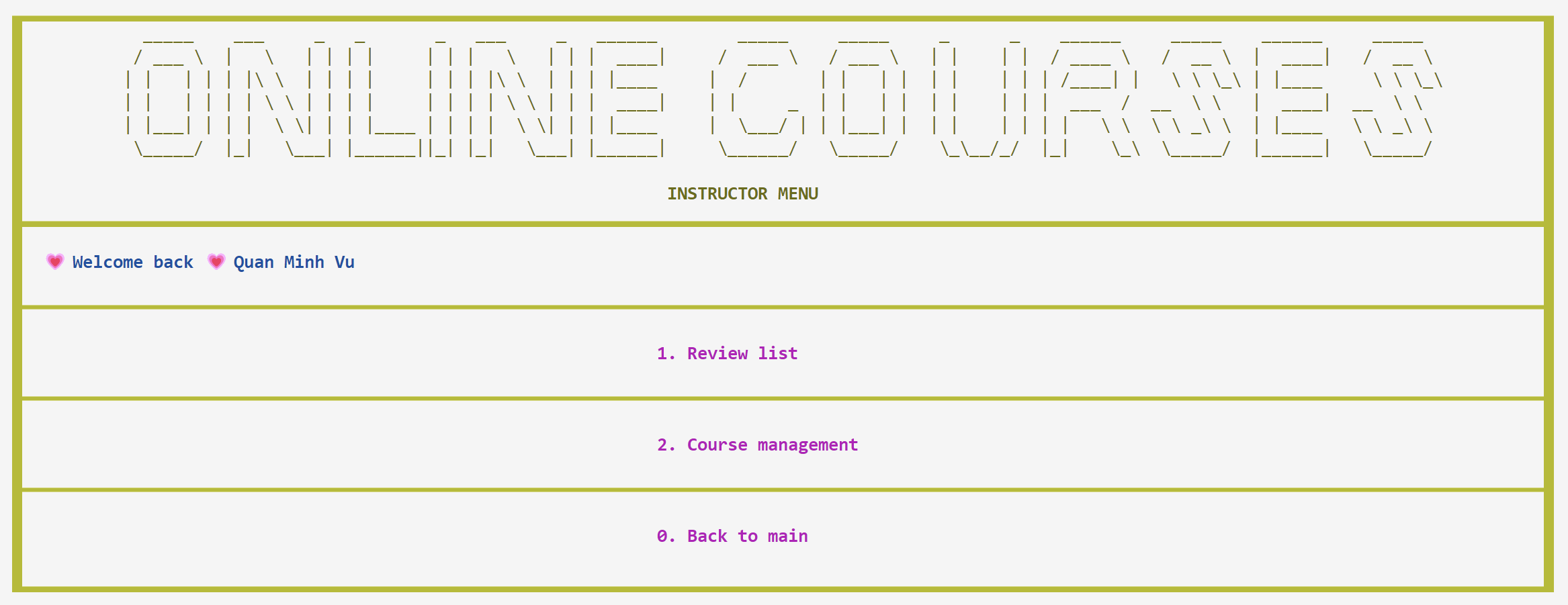
* 1. Learning page



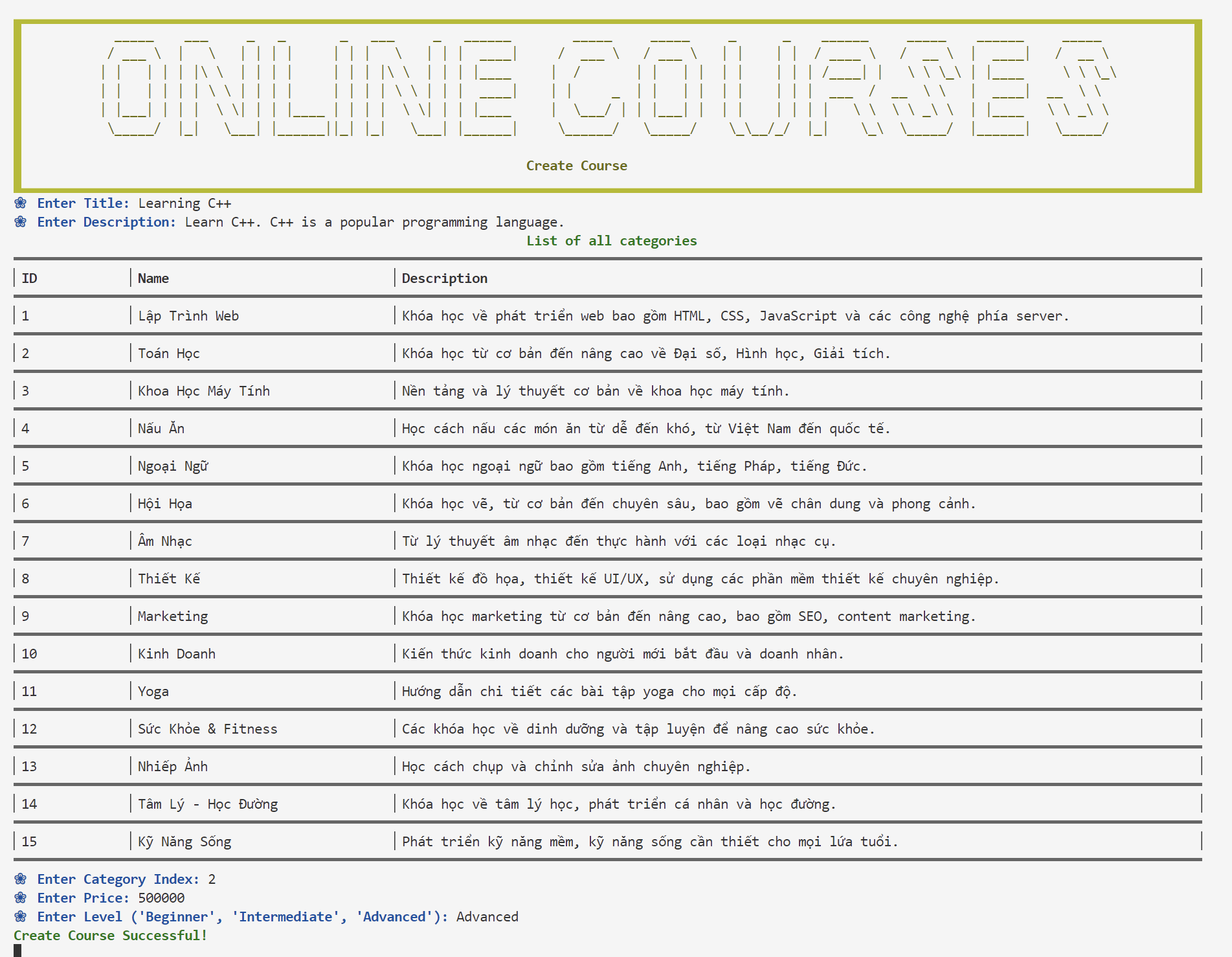
* 1. Lesson content

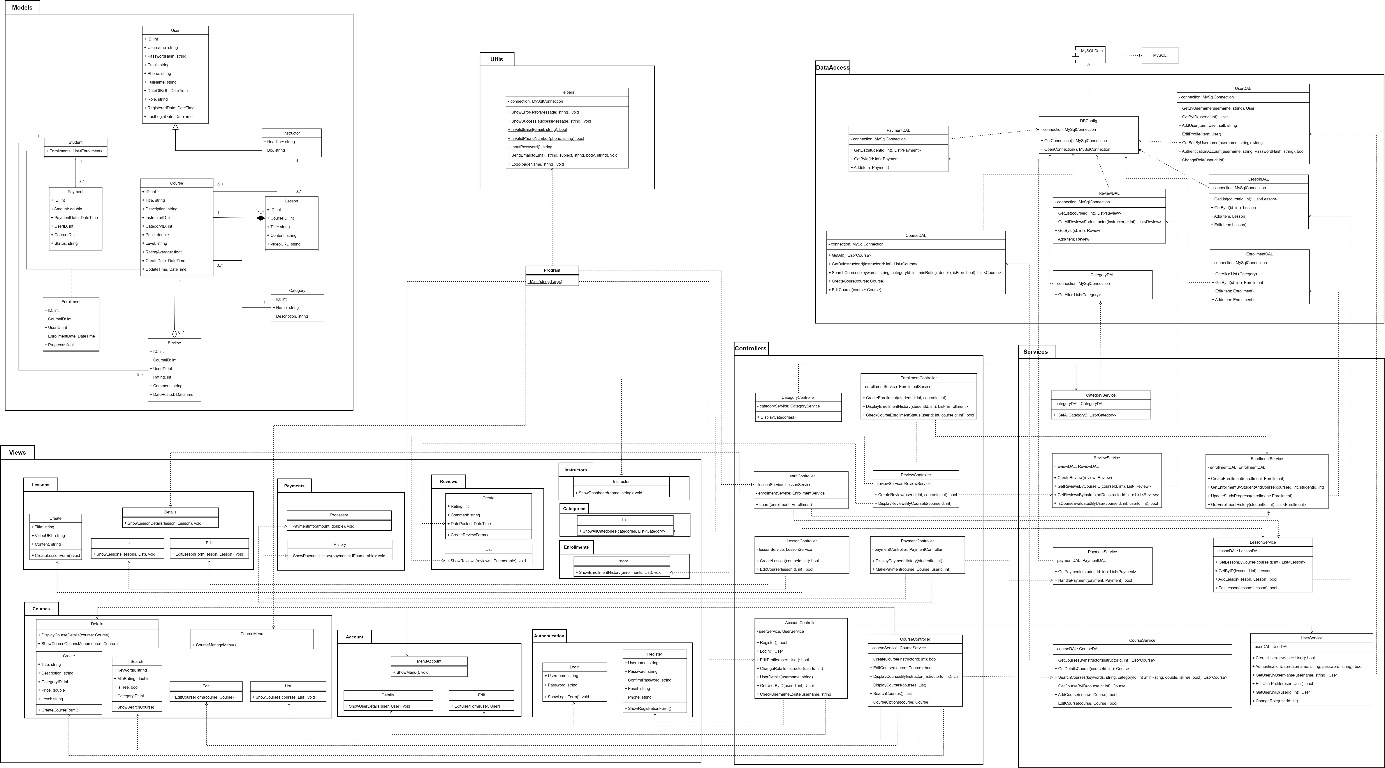


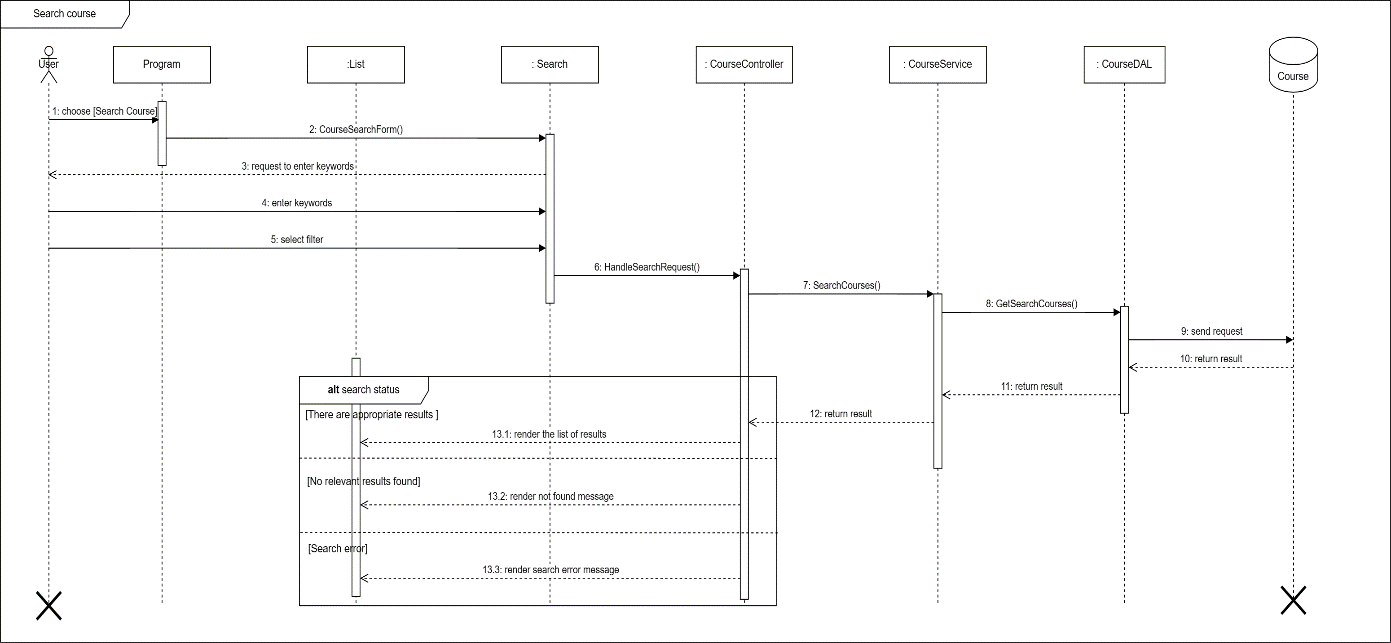
* 1. Instructor menu



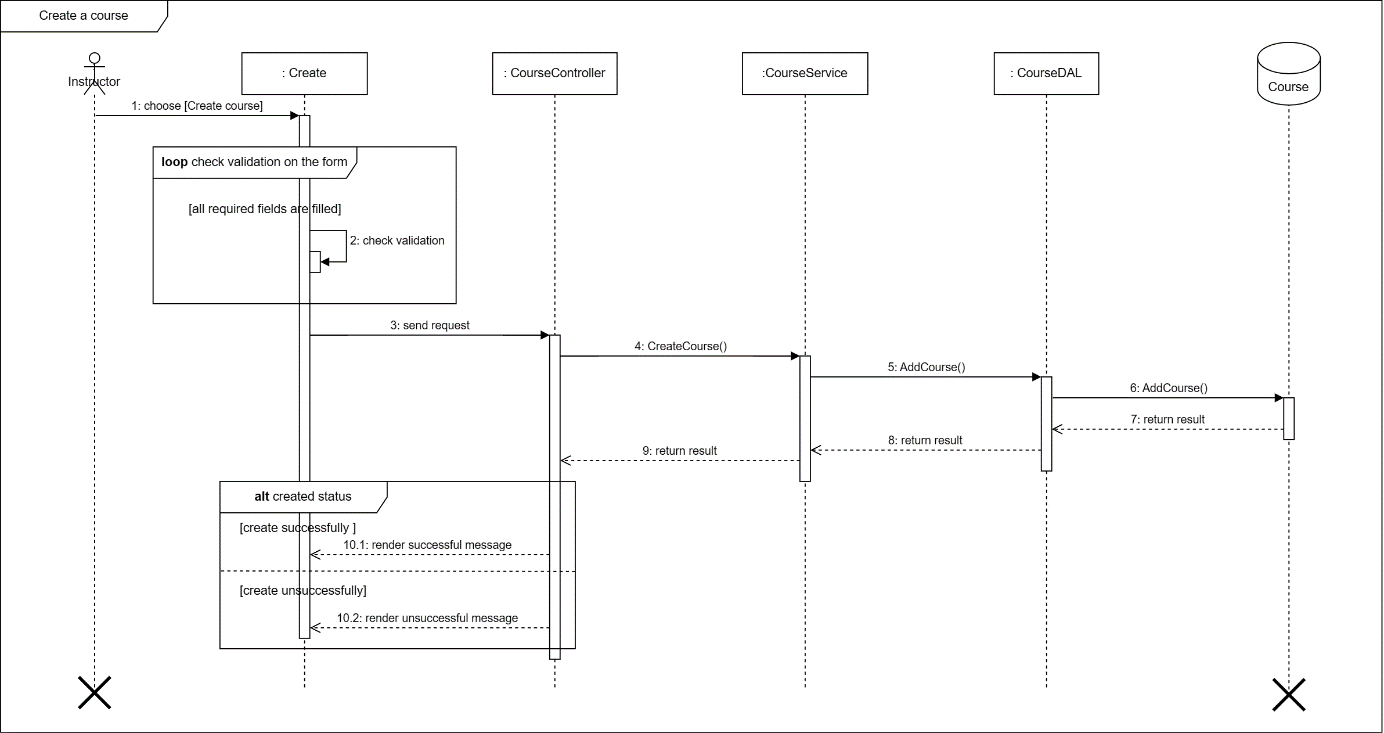
* 1. Create a course



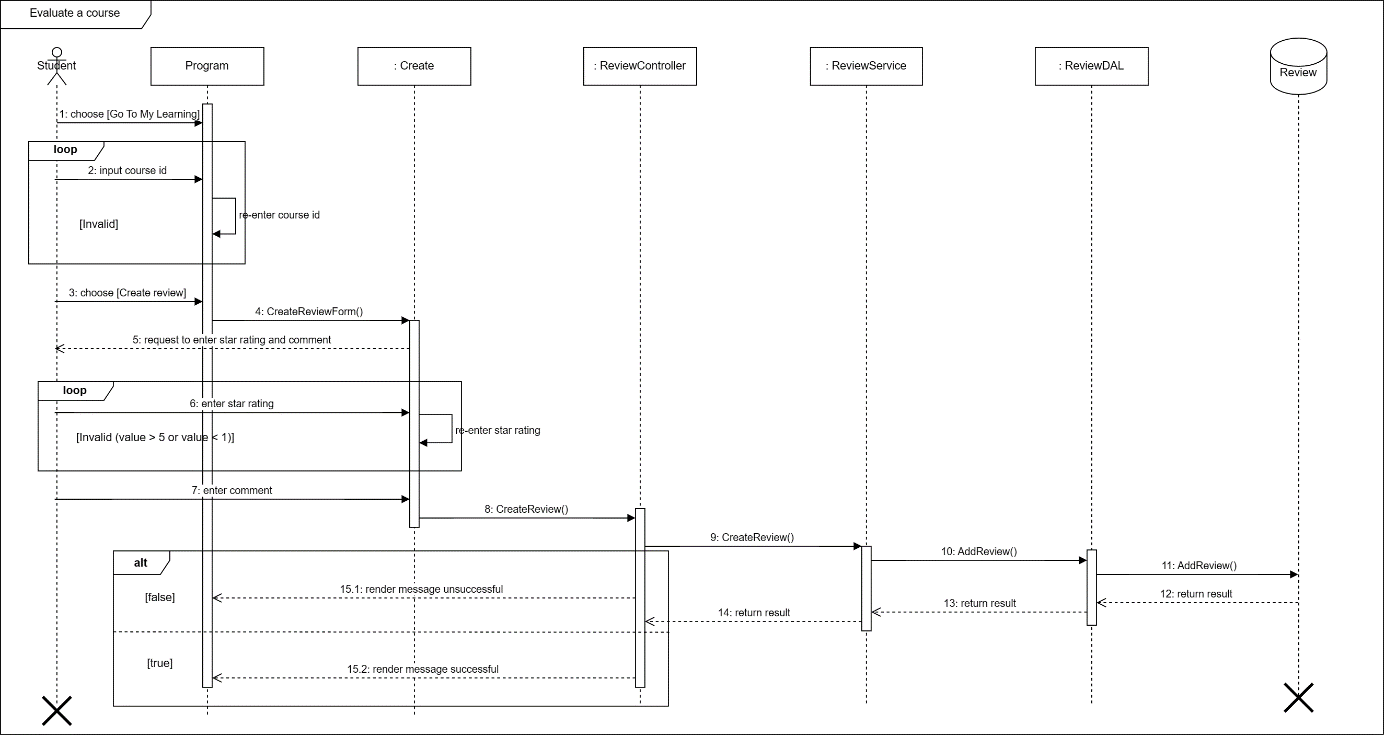
1. Code Design (Class Diagram)
2. 
3. Sequence Diagram
   1. Search courses



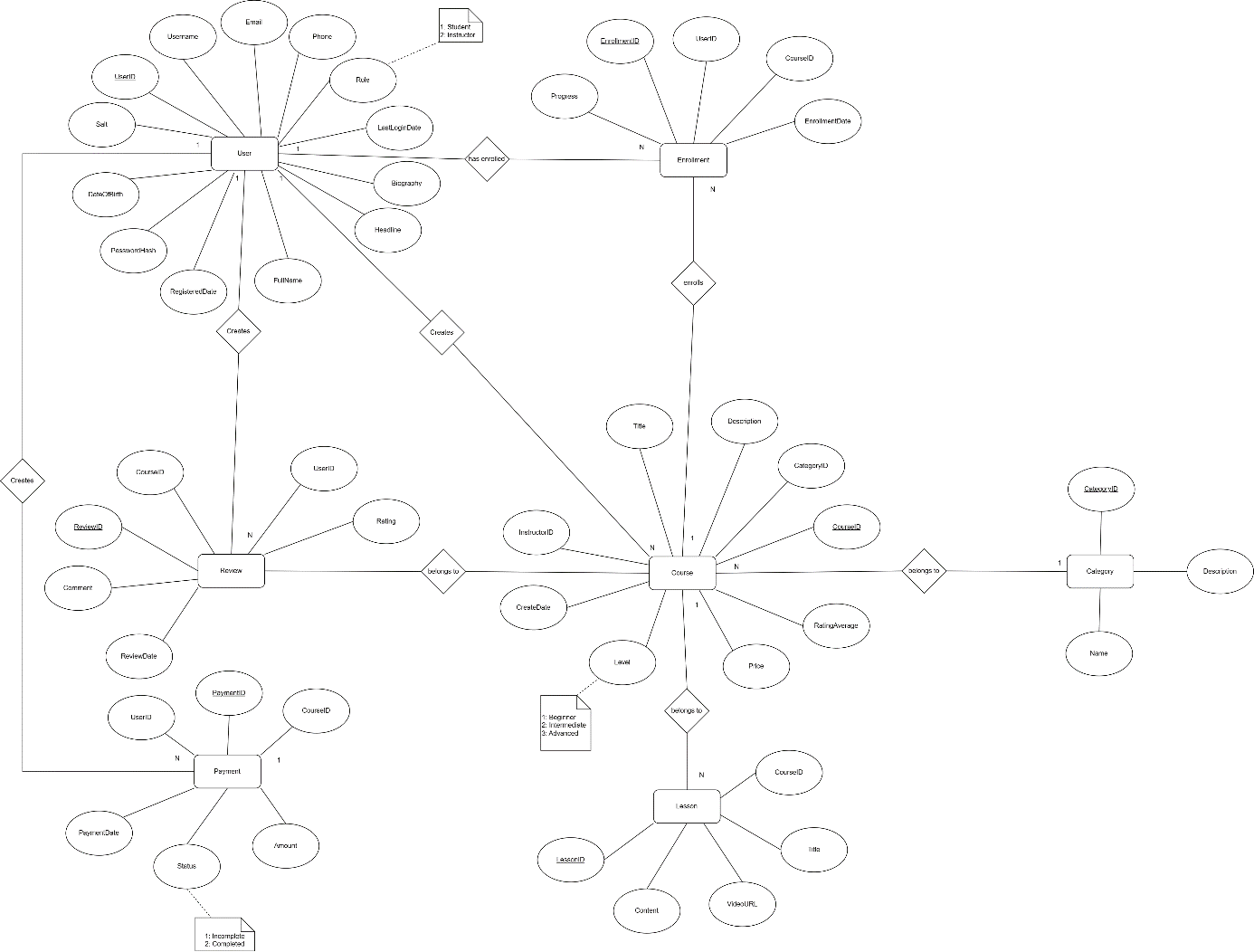
* 1. Create course

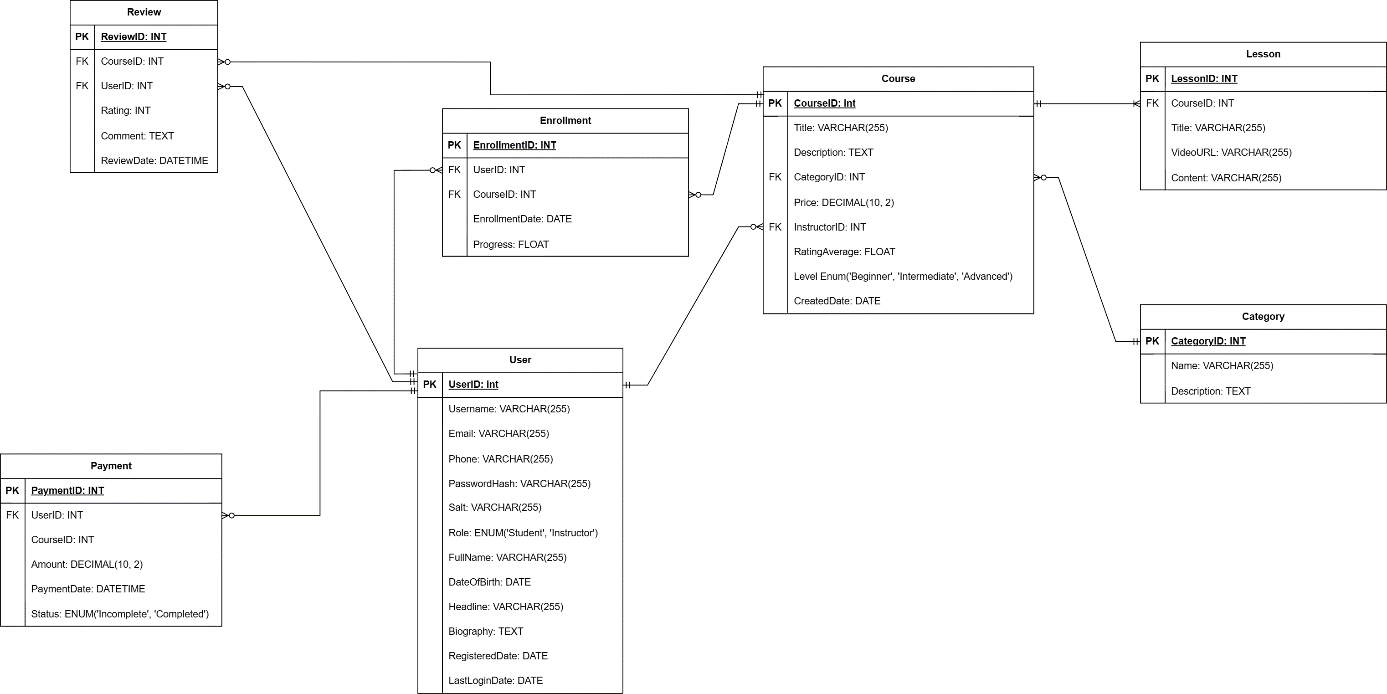


* 1. Evaluate a course

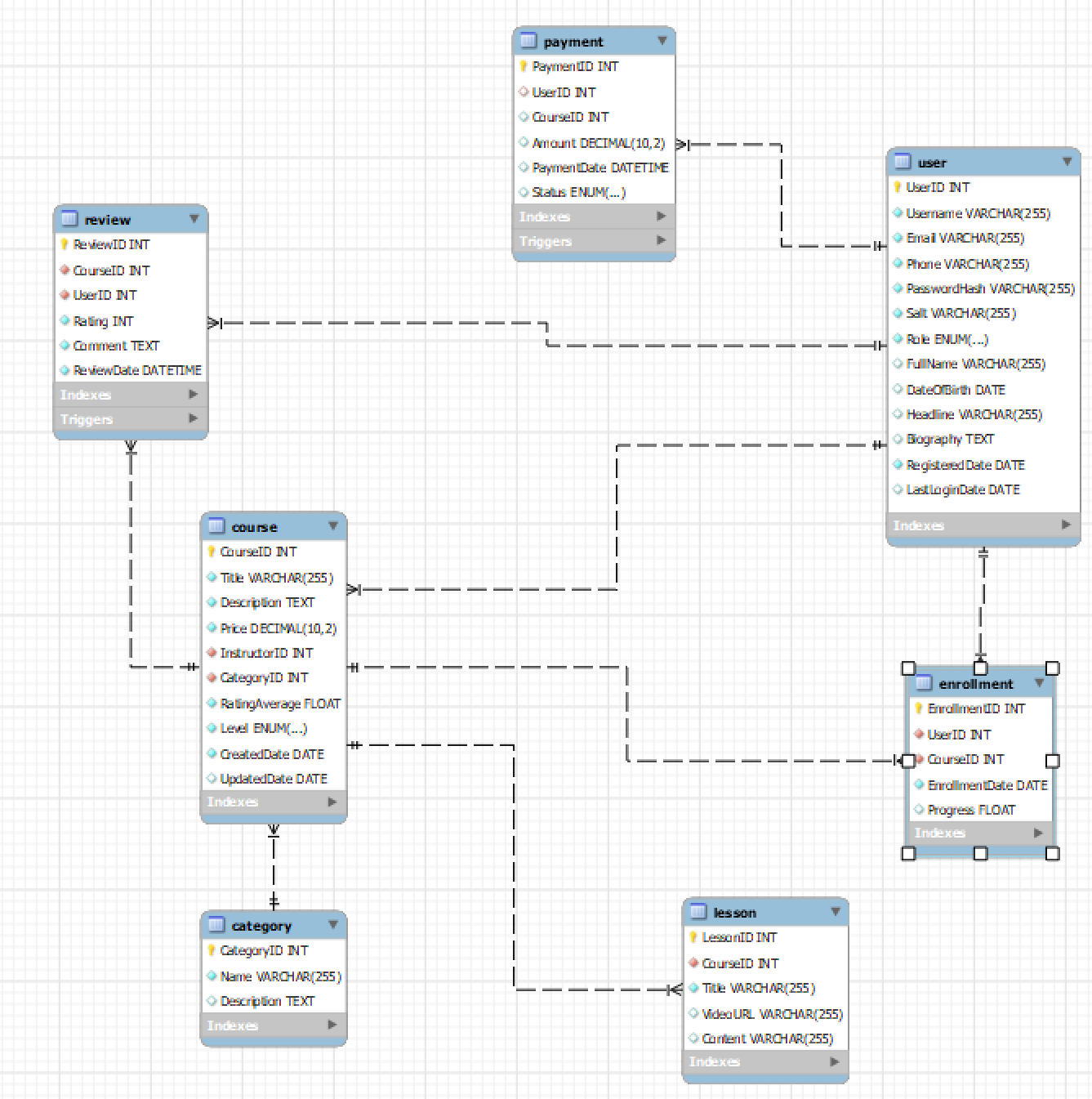


1. Database Design
   1. Entity Relationship Diagram





* 1. Database Design Details



|  |  |  |  |
| --- | --- | --- | --- |
| User | | | |
| Column Name | Data Type | Constraints | Description |
| UserID | int | Primary Key, Autoincrement |  |
| Username | varchar(255) | NOT NULL |  |
| Email | varchar(255) | NOT NULL |  |
| Phone | varchar(15) | NOT NULL |  |
| PasswordHash | varchar(100) | NOT NULL | User's password encrypted with Salt |
| Role | enum('Student', 'Instructor') | NOT NULL |  |
| FullName | varchar(255) |  |  |
| DateOfBirth | date |  |  |
| Salt | varchar(255) | NOT NULL |  |
| Headline | varchar(255) |  |  |
| Biography | Text |  |  |
| RegisteredDate | date | NOT NULL |  |
| LastLoginDate | date |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Course | | | |
| Column Name | Data Type | Constraints | Description |
| CourseID | int | Primary Key, Autoincrement |  |
| Title | varchar(255) | NOT NULL |  |
| Description | varchar(255) | NOT NULL |  |
| Price | DECIMAL(10,2) | NOT NULL |  |
| InstructorID | int | NOT NULL, FOREIGN KEY |  |
| CategoryID | int | NOT NULL, FOREIGN KEY |  |
| RatingAverage | float | NOT NULL |  |
| Level | enum('Beginner', 'Intermediate', 'Advanced') | NOT NULL |  |
| CreatedDate | date | NOT NULL |  |
| UpdatedDate | date |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Enrollment | | | |
| Column Name | Data Type | Constraints | Description |
| EnrollmentID | int | Primary Key, Autoincrement |  |
| UserID | int | NOT NULL, FOREIGN KEY |  |
| CourseID | int | NOT NULL, FOREIGN KEY |  |
| EnrollmentDate | DATE | NOT NULL |  |
| Progress | float |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Category | | | |
| Column Name | Data Type | Constraints | Description |
| CategoryID | int | Primary Key, Autoincrement |  |
| Name | varchar(255) | NOT NULL |  |
| Description | text |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Lesson | | | |
| Column Name | Data Type | Constraints | Description |
| LessonID | int | Primary Key, Autoincrement |  |
| CourseID | int | NOT NULL, FOREIGN KEY |  |
| Title | varchar(255) | NOT NULL |  |
| VideoURL | varchar(255) |  |  |
| Content | varchar(255) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Payment | | | |
| Column Name | Data Type | Constraints | Description |
| PaymentID | int | Primary Key, Autoincrement |  |
| CourseID | int | NOT NULL, FOREIGN KEY |  |
| UserID | int | NOT NULL, FOREIGN KEY |  |
| Amount | DECIMAL(10,2) | NOT NULL |  |
| PaymentDate | date | NOT NULL |  |

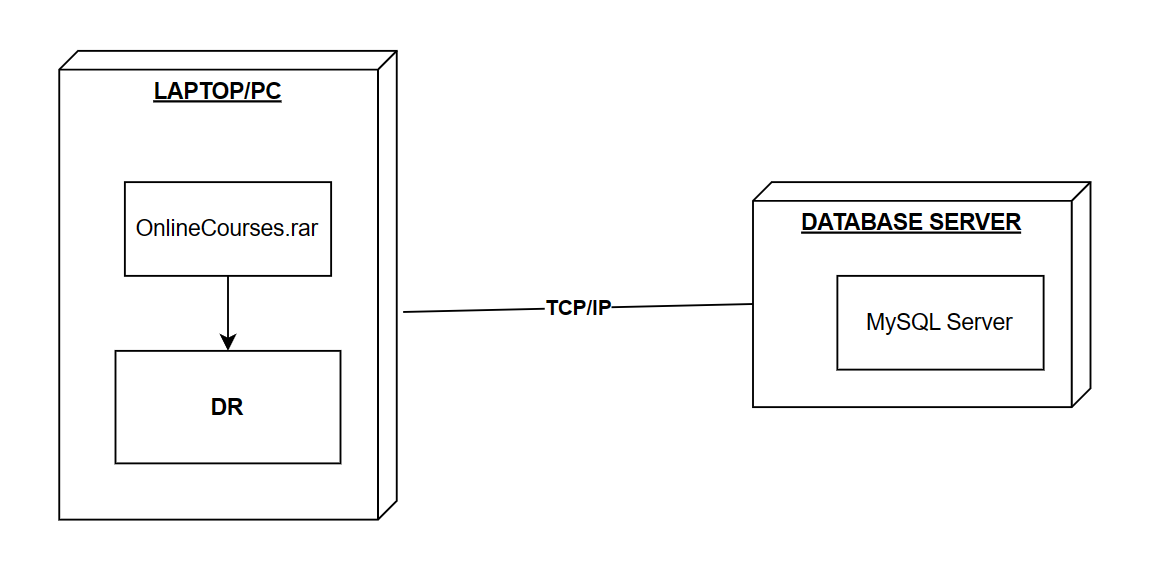
|  |  |  |  |
| --- | --- | --- | --- |
| Review | | | |
| Column Name | Data Type | Constraints | Description |
| ReviewID | int | Primary Key, Autoincrement |  |
| CourseID | int | NOT NULL, FOREIGN KEY |  |
| UserID | int | NOT NULL, FOREIGN KEY |  |
| Rating | int | NOT NULL |  |
| Comment | text | NOT NULL |  |
| ReviewDate | datetime | NOT NULL |  |

# Task Assign (to each team member)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Group Name | Project Name | | | | | |
| No | Task name | Description | Start Date | End Date | Member | Self assessment |
| 1 | Write report |  | 15 February | 28  March | Vi & Anh |  |
| 2 | Code |  | 15 February | 28  March | Vi & Anh |  |
| 3 | Drawing design |  | 15 February | 28  March | Vi & Anh |  |

# Installation Instructions

1. Deployment Diagram



1. Installation steps
   * Database Install
   * Server install
   * Application install