

**LEARNING ONLINE**

**Software Requirement Specification**

– Hanoi, Feb 2023 –

**Table of Contents**

[I. Overview 3](#_Toc127527850)

[1. Introduction 3](#_Toc127527851)

[2. System Functions 4](#_Toc127527852)

[a. Screen Flow 4](#_Toc127527853)

[b. Screen Details 4](#_Toc127527854)

[c. User Authorization 7](#_Toc127527855)

[II. Functional Requirements 10](#_Toc127527856)

[1. Home page 10](#_Toc127527857)

[a. Screen layout 10](#_Toc127527858)

[b. Use-case 10](#_Toc127527859)

[c. DAR model: 11](#_Toc127527860)

[2. Login 11](#_Toc127527861)

[a. Screen layout 12](#_Toc127527862)

[b. Use case 12](#_Toc127527863)

[c. DARmodel 13](#_Toc127527864)

[3. Login 13](#_Toc127527865)

[a. Screen layout 13](#_Toc127527866)

[b. Use-case 13](#_Toc127527867)

[c. DARmodel 14](#_Toc127527868)

[4. User Profile 15](#_Toc127527869)

[a. Screen layout 15](#_Toc127527870)

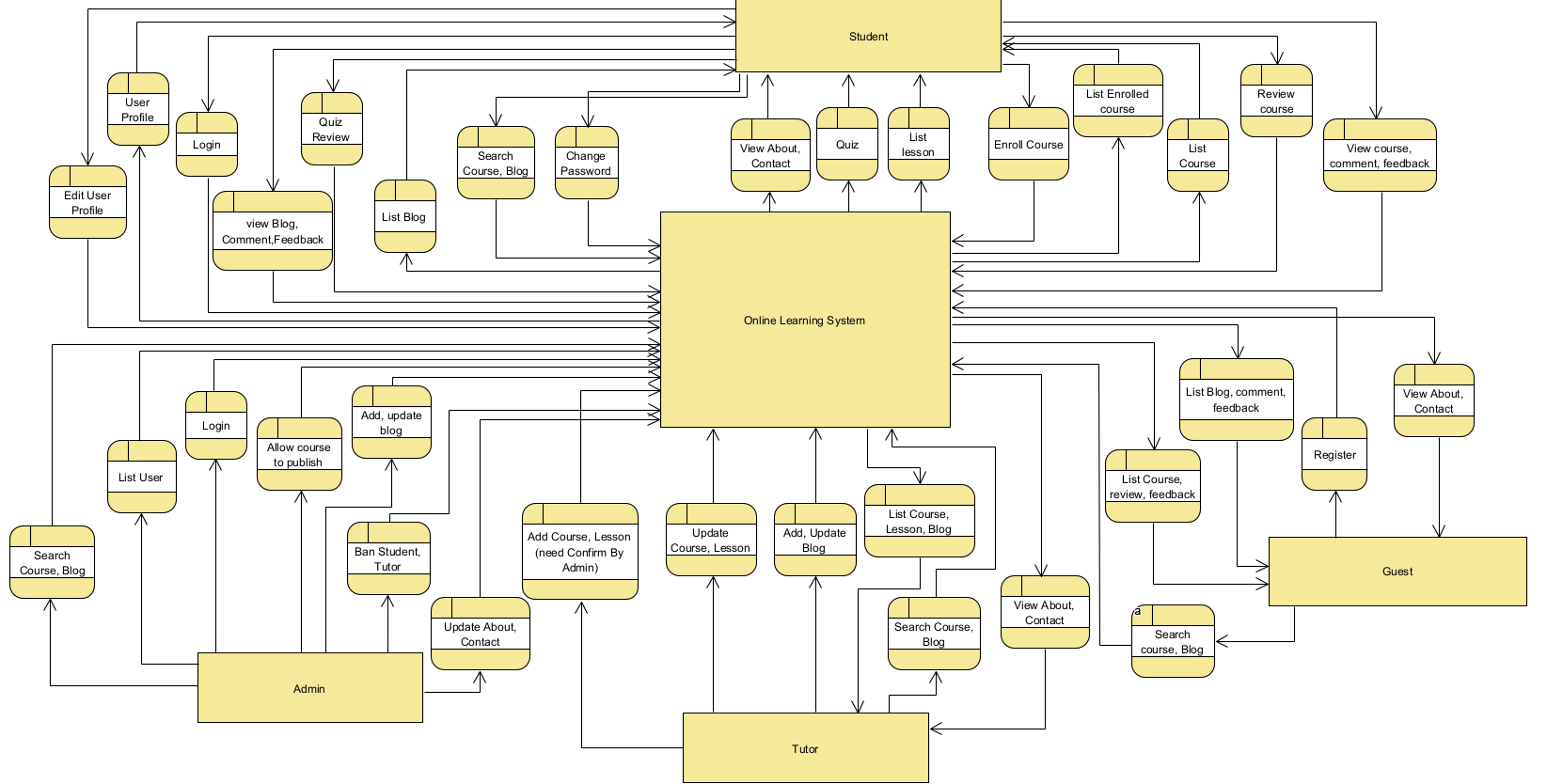
[b. Use-case 15](#_Toc127527871)

[c. DARmodel 16](#_Toc127527872)

I. Overview

1. Introduction

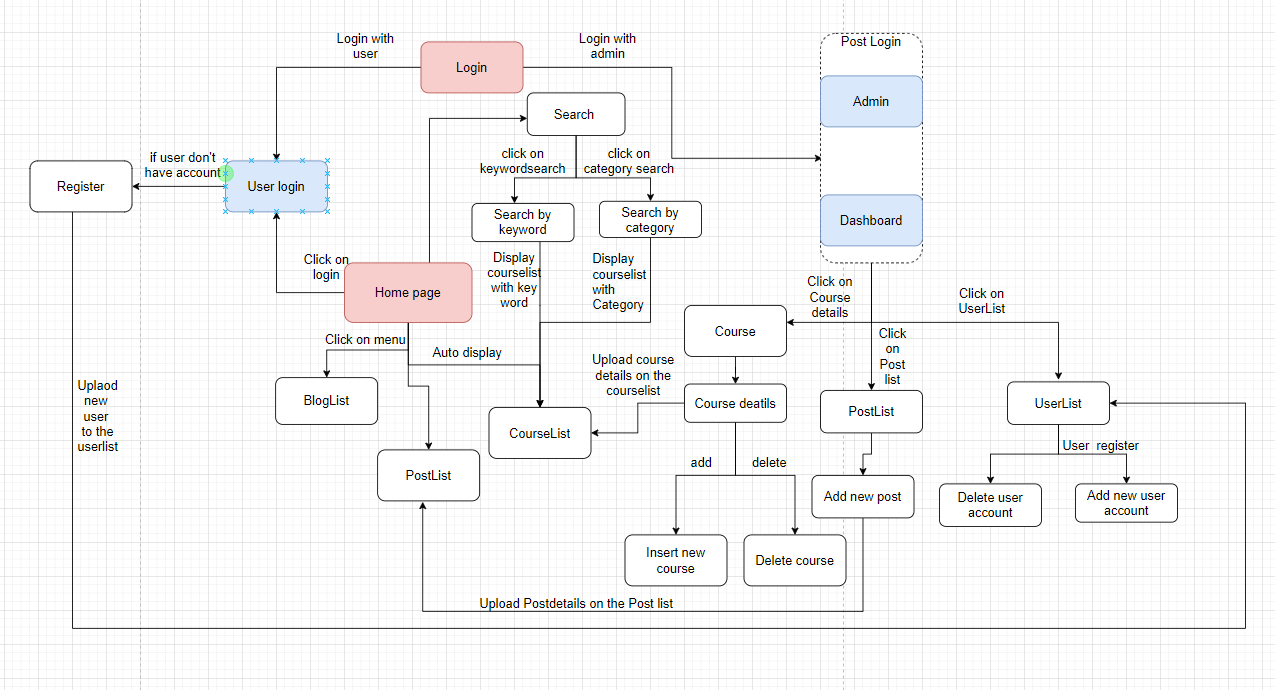
Online learning system is a system that allows users to access available courses, a system that provides information about the completion of courses, and discussions about questions.



2. System Functions

a. Screen Flow

This part shows the system screens and the relationship among screens. You can draw the Screens Flow for the system in the form of diagram as below:



b. Screen Details

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Public | Home Page | Show sliders, hot posts, featured subjects + the sider with the latest posts, static contacts/links  - Shown slider information includes its image and title; the user is redirected to the slider’s backlink on his/her clicking  - Shown latest Blog information includes its thumbnail, title, post-date; the user is redirected to the Blog details on his/her clicking  - Shown top courses information includes its thumbnail, title, tag line; the user is redirected to the course's details on his/her clicking  - Shown all tutors information includes its thumbnail, title, tag line; the user is redirected to the tutor's CV on his/her clicking |
| 2 | Student | User Profile | Show detailed user information (avatar, full name, gender, email, mobile, role, address, status), from that allow the user to add new, view or edit user information |
| 3 | Public | Course register | This is a pop-up screen which allow the user to register for access a subject. On this screen, the user needs to choose appropriate subject. |
| 4 | Common | Change password | The user to input his/her email to receive the reset password link. This link is sent to the user’s email for his/her password resetting.  - On clicking the link, the user is redirected to the password resetting page in which s/he input new password (twice) for the system access  - Note that the link is only available for a specific time duration as configured in the system config file. |
| 5 | Common | Login | This is a pop-up screen which allows the user to enter username & password to login or user may login with email and Facebook. On this page, there are also options for user to register new information or reset the password for the case he forgets it |
| 6 | Common | Register | This is a pop-up screen which allows the user to enter username, date of birth, phone number, full name, password, confirm password, gender. |
| 7 | Public | BlogList | Show the posts paginated (sorted by post-date) + the sider with the post search box, posts categories, latest posts, static contacts/links; Shown post information includes its thumbnail, title, category, brief-info; the user is redirected to the post’s details on his/her clicking |
| 8 | Public | BlogDetail | Show post details (including title, author, post-date, category, post details, feedbacks and comments) + the sider with the posts categories, latest posts, latest courses |
| 10 | Public | CourseList | Show subject details (including title, tag line, brief info, lowest price package option (original price, sale price), product description, and a Register button) + the sider with the subject search box, subject categories, featured subjects, static contacts/links |
| 11 | Student | My Course | For the customer to view his/her access-allowed courses |
| 13 | Common | UserAthorizaiton | This is authorization mechanism in the system, including the specify the user's role, his/her authorized page links (building the displayed menu items and preventing unauthorized access via enter the links directly). The scope of this function also includes building the header, footer, the main function menu (for both Admin and NonAdmin parts) |
| 14 | Popular | Rating and review | The screen displays student ratings, comments, and votes for each course  Show the list of user's registration + the sider with the subject search box, subject categories, static contacts/links |
| 15 | Public | Sliders | Show the paginated list of sliders (including sliders' id, title, image, backlink, status)  - The user can filter the sliders by the status + search by title or backlink  - For each slider, the user can choose to hide, show, edit it |
| 16 | Public | Slider Details | Show detailed slider information (image, title, backlink, status, notes) |
| 17 | Admin | Course List | 'Show the list of course paginatedly. The shown information for each subject include the id, name, category, number of lessons and status  The user can:  - Filter the list by course category & status  - Search the course by the name |
| 18 | Admin | Course Detail | Admin can see course detail and can change the publish of the course |
| 19 | Tutor | Tutor screen | Tutor can view their course they created and can update it or create new course |
| 20 | Tutor | Create course screen | The screen show a form for tutor  The tutor can:   * Add course name * Add new course lession * Add lession detail * Add quiz and correct answer |
| 21 | Tutor | Update course screen | Tutor can update course that they created. The screen show all the current information of the course  The tutor can:   * Change course name, course detail, delete lession, delete quiz. * Add new course lession * Add lession detail * Add quiz and correct answer |
| 21 | Student | Course Detail | * The screen show the detail of the course right at the lession that student learn after, if student not learned anything after the screen will show the first lession |
| 22 | Student | Lession detail and quiz | * Student can see the lession detail of the lession they are learning and the previous lession * Student can take quiz at the an of the lession * Only if student complete quiz they can move to the next lession |
| 23 | Public | Search | * Users can search for the course they want by entering information |
| 24 | Public | Search with category | * Users can search for the course they want by selecting the available courses on the menu bar |

c. User Authorization

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Screen** | **Guest** | **Users** | **Admin** | **Tutor** |
| Home Page | X | X | X | X |
| User Profile |  | X |  |  |
| User Register | X | X |  |  |
| User List |  |  | X |  |
| User Password change |  | X |  |  |
| User Details |  |  | X |  |
| Login | X | X | X | X |
| BlogList | X | X |  |  |
| CourseList | X | X | X | X |
| Enroll course |  | X |  |  |
| View course detail, lession detail, take quiz |  | X |  |  |
| Add New Course |  |  |  | X |
| Update Course |  |  |  | X |
| Delete Course |  |  | X |  |
| Delete users |  |  | X |  |
| View list users |  |  | X |  |
| Search Course | X | X | X | X |
| Popular Online Course |  |  |  |  |
| BlogDetail | X | X |  |  |
| View list courses | X | X | X | X |
| Update information user |  | X |  |  |
| My Course |  | X |  |  |
| Feedback |  | X |  |  |
| My registration |  | X |  |  |
| Rating and review |  | X |  |  |

In which:

* Guest: User who does not login. Only view course information but not course details
* Users: who have registered, can enroll course
* Admin: The web manager who manages the web's contents, manage publish course,...
* Tutor : The person who can create new course and update course

**3. Common Requirements:**

- Users can search and view all of the course

- User can view the course detail

- User want to register into the website.

- User (except admin and guest) are able to change their account’s information such as Full Name, Phone, Address, Password,...

- User can login to their account

- User can view blogs list

- Guest can’t enroll course

- Student can give review after learn course

- Student can write discuss and can be ban if use offensive words

- Only Student can enroll course

- Admin can ban user.

- Admin can edit userlist

- Admin can allow which course can be publish

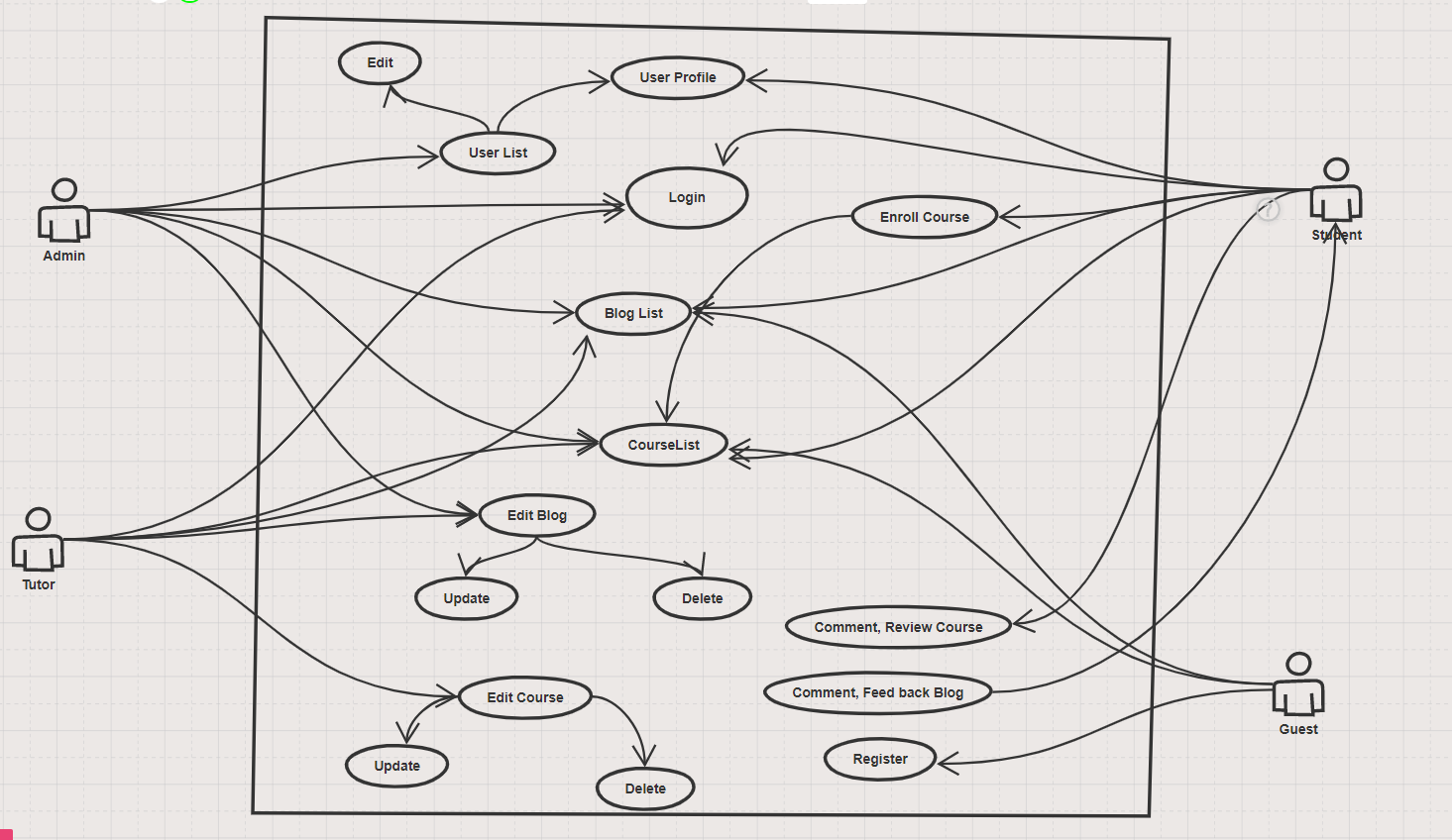
- Admin can update bloglist.

- Admin can view all courses details without enroll courses.

- Tutor can create course but need the allow of the admin to publish that course

- Tutor can update course.

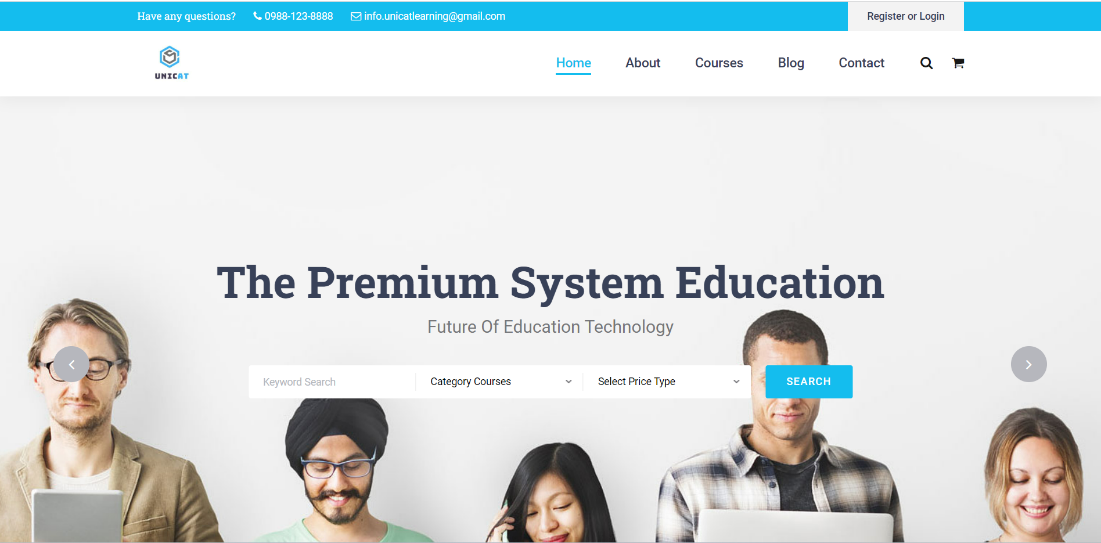
**4.User diagram**



II. Functional Requirements

1. Home page

a. Screen layout



b. Use-case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | | UC01 | |
| **Use Name** | | | **Home Page** | |
| **Actor** | | | User, System | |
| **Description** | | | The starting page of the system | |
| **Precondition** | | | Always | |
| **Trigger** | | | User access to the Learning Online website | |
| **Post-Condition** | | |  | |
| **Main flows** | | | | |
| **Step** | **Actor** | | **Action** | |
| 1 | User | | Access to the Learning Online website | |
| 2 | System | | Display Home Page screen with the following fields:  - sliders  - latest Blog  - Popular Course  - View tutors  - search by keyword | |
| 3 | User | | Click on any field has a backlink | |
| 4 | System | | Redirect user to the screen of field’s backlink | |
| **Alternative flows** | | | | |
| **AT1** | | At step 2 in the main flows**,** if there is an internal error in the system | | |
| **Sub step** | | **Actor** | | **Action** |
| 2.1 | | System | | Display "Error" page with error message |

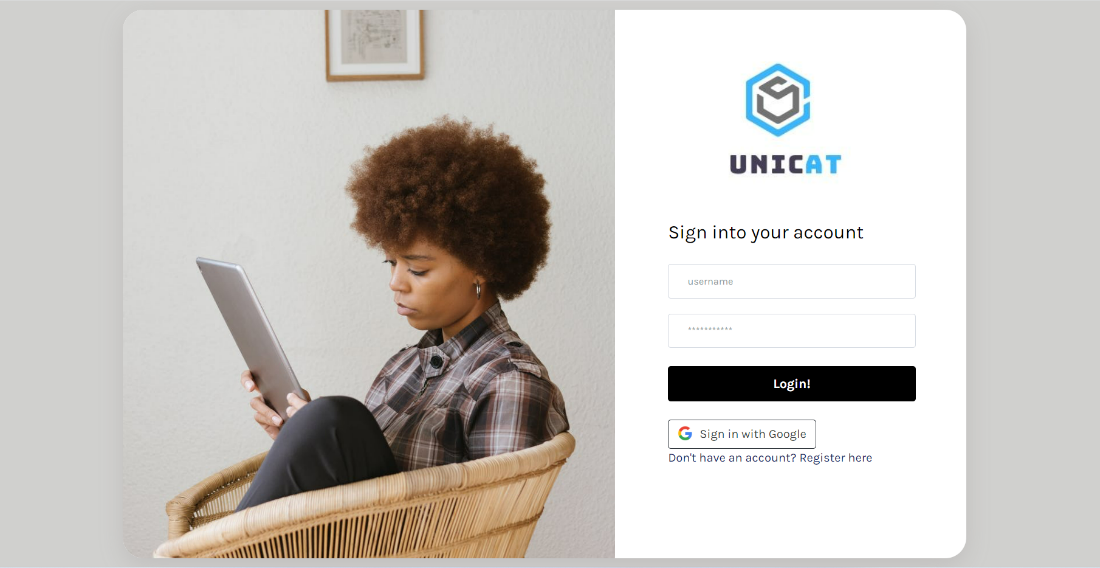
c. DAR model:

|  |  |  |  |
| --- | --- | --- | --- |
| **UI Element: Home Page** | | | |
| **UI Element Description** | | | |
| **ID** | FF01 | | |
| **Description** | The starting page of the system | | |
| **UI Element Display** | | | |
| **Precondition** | | **Display** | |
| Always | | Show sliders, hot posts, featured course + the slider with the latest posts, static contacts/links.  Shown slider information includes its image and title.  Shown course information includes its thumbnail, title, brief information. | |
| User login as Student | | Show home page course and enrol course | |
| USer login as Admin role | | Show the Admin Dashboard button on header | |
| **UI Element Behaviours** | | | |
| **Precondition** | | **Action** | **Response** |
| None | | User click on slider | The user is redirected to the slider's backlink |
| None | | User click on a post | The user is redirected to the post's details |
| None | | User click on course | The user is redirected to the course details |
| None | | User click on enrol | The course will be saved to my course |
| None | | User click on View all course | Show the menu categories of course |
| None | | User click on Account button | The user is redirected to the User Profile screen |
| User type in search bar | | User click on search button | Show the list of courses match with search key |
| User login as student role | | User click on User manager button | The user is redirected to the User's course list |
| User login as Admin role | | User click on Admin Dashboard button | The user is redirected to the Admin Dashboard screen |

2. Login

**4. Login**

a. Screen layout



b. Use case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | | UC05 | |
| **Use Name** | | | **Login** | |
| **Actor** | | | User, System | |
| **Description** | | | Allow the user to login to the system by using an account that already exists in the system | |
| **Precondition** | | | Always | |
| **Trigger** | | | User clicks on Account button on the header of website | |
| **Post-Condition** | | |  | |
| **Main flows** | | | | |
| **Step** | **Actor** | | **Action** | |
| 1 | User | | User clicks on registerOrlogin button on the header of website | |
| 2 | System | | Display Login screen with the following fields:  - User Name  - Password  - Login with Google  - Login button  - Register | |
| 3 | User | | Enter username & password into Username & Password fields on the Login screen, then click on the Login button or Click 'Sign in with Google' button. | |
| 4 | System | | Validate the entered user name & password or GmailID and then display Homepage screen | |
| **Alternative flows** | | | | |
| **AT1** | | At step 2 in the main flows**,** if there is an internal error in the system | | |
| **Sub step** | | **Actor** | | **Action** |
| 2.1 | | System | | Display error message |
| **AT2** | | At step 4 in the main flows**,** if system failed to validate username & password | | |
| **Sub step** | | **Actor** | | **Action** |
| 4.1 | | System | | Display "Login" page with message "Your user name or password is not correct." |

c. DARmodel

|  |  |  |  |
| --- | --- | --- | --- |
| **UI Element Description** | | | |
| **ID** | FF05 | | |
| **Description** | Allow the user to login to the system by using an account that already exists in the system | | |
| **UI Element Display** | | | |
| **Precondition** | | **Display** | |
| Always | | Display the required fields (username and password to let the user fill it in), Login button and Home button. | |
| **UI Element Behaviours** | | | |
| **Precondition** | | **Action** | **Response** |
| None | | User clicks Login | The system will check the username and password. If true, it leads to the homepage screen; if false, it will send user an error message and ask user to re-enter |
| None | |  |  |
| None | | User click Do not have an account ? Sign in heare! | Leads to the User Register screen |

3. Login

a. Screen layout



b. Use-case

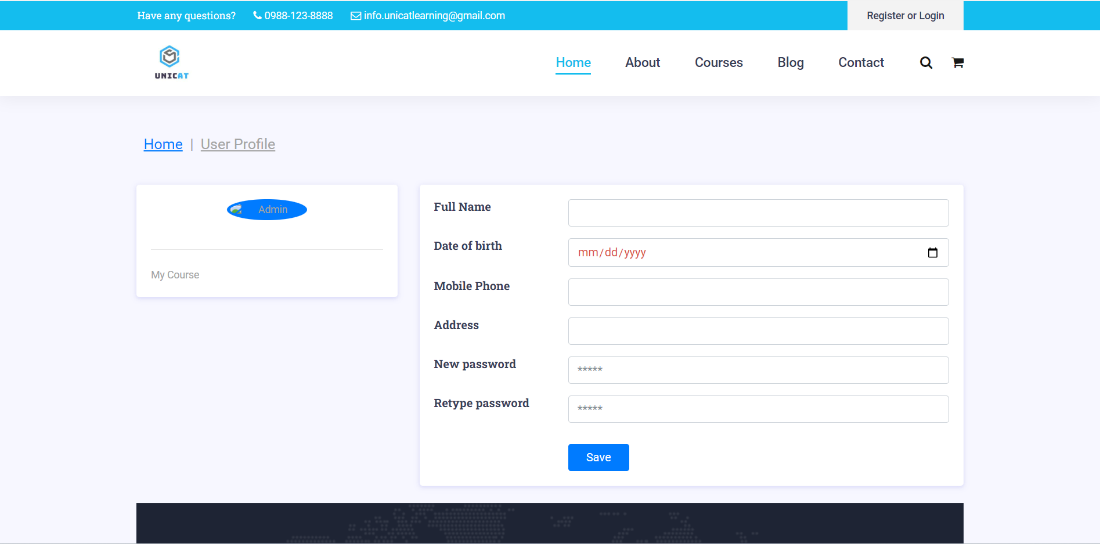
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | | UC06 | |
| **Use Name** | | | **Register** | |
| **Actor** | | | Guest, System | |
| **Description** | | | Register new user into the system | |
| **Precondition** | | | Always | |
| **Trigger** | | | Guest clicks on Create an account on the Login page | |
| **Post-Condition** | | |  | |
| **Main flows** | | | | |
| **Step** | **Actor** | | **Action** | |
| 1 | User | | Guest clicks on Create an account on the Login page | |
| 2 | System | | Display User Register screen with the following fields:  - User Name  - DOB  - Phone number  - FullName  - Address  - Password  - Re-password | |
| 3 | User | | Enter all fields on the Register screen, then click on the Create button. | |
| 4 | System | | Validate the entered fields and add to the database then go back to User Login page | |
| **Alternative flows** | | | | |
| **AT1** | | At step 2 in the main flows**,** if there is an internal error in the system, | | |
| **Sub step** | | **Actor** | | **Action** |
| 2.1 | | System | | Display error message |
| **AT2** | | At step 4 in the main flow**,** if the username or phone already exists. | | |
| **Sub step** | | **Actor** | | **Action** |
| 4.1 | | System | | Display Register page with error message “username exist “ or “phone exist “ or "Gmail exist" |

c. DARmodel

|  |  |  |  |
| --- | --- | --- | --- |
| **UI Element: Home Page** | | | |
| **UI Element Description** | | | |
| **ID** | FF06 | | |
| **Description** | Register new user into the system | | |
| **UI Element Display** | | | |
| **Precondition** | | **Display** | |
| Always | | Display the required fields(username, password, re-enter the password, Name, DOB, phone numbers) and Create button. | |
| **UI Element Behaviours** | | | |
| **Precondition** | | **Action** | **Response** |
| None | | User clicks Register | The system will check the information the user gives. If true, it will send a notification; If false,it will send the user an error message  and ask the user to re-enter. |

4. User Profile

a. Screen layout



b. Use-case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID** | | | UC07 | |
| **Use Name** | | | **User profile** | |
| **Actor** | | | user, System | |
| **Description** | | | user information and edit | |
| **Precondition** | | | login successful | |
| **Trigger** | | | user clicks on user icon in header | |
| **Post-Condition** | | |  | |
| **Main flows** | | | | |
| **Step** | **Actor** | | **Action** | |
| 1 | User | | click on user icon in header | |
| 2 | System | | Display User Profile screen with the following fields:  - Full Name  - DOB  - Phone number  - Address  - Password  - Re-password | |
| 3 | User | | User can change one or all fields and click ‘save change’ button  user can unenrolled course when they click ‘unenroll’ | |
| 4 | System | | Validate the entered fields and add to the database or unenroll courses taken then go back to User Login page | |
| **Alternative flows** | | | | |
| **AT1** | | At step 2 in the main flows**,** if there is an internal error in the system | | |
| **Sub step** | | **Actor** | | **Action** |
| 2.1 | | System | | Display error message |
| **AT2** | | At step 4 in the main flow**,** if the username or phone already exists. | | |
| **Sub step** | | **Actor** | | **Action** |
| 4.1 | | System | | Display user profile page with error message “username exist “ or “phone exist “ |

c. DARmodel

|  |  |  |  |
| --- | --- | --- | --- |
| **UI Element: Home Page** | | | |
| **UI Element Description** | | | |
| **ID** | FF07 | | |
| **Description** | update userprofile | | |
| **UI Element Display** | | | |
| **Precondition** | | **Display** | |
| Always | | Display the required fields(username, password, repassword, DOB, phone numbers) and save change. | |
| **UI Element Behaviours** | | | |
| **Precondition** | | **Action** | **Response** |
| None | | User clicks save change | The system will check the information the user gives. If true, it will send a message “update successful”; If false,it will send the user an error message “...exist “  and ask the user to re-enter. |
| None | | User click unenroll | The system will delete the enrolled course and update the user information |