# Requirement Specifications

# Project: Cursus

Version: 0.2

# Approval Page

The endorsement on this document, by the authorized <<client>> representative, indicates that <<client>> and FPT have agreed on the document "Cursus- Software Requirement Specifications".

**Prepared  by :**   FPT                                   Signature:

Business Analyst                                           _____

Date:

\_\_\_\_ /\_\_\_\_  /\_\_\_\_

**Reviewed  by :**   FPT                                   Signature:

Project Manager                                           _____

Consultant                                                Date:

\_\_\_\_ /\_\_\_\_  /\_\_\_\_

**Supported by :**   <<client>>                            Signature:

_____

Date:

\_\_\_\_ /\_\_\_\_  /\_\_\_\_

**Approved by :**   <<client>>                             Signature:

_____

Date:

\_\_\_\_ /\_\_\_\_  /\_\_\_\_

# Revision History

| Date | Version | Author | Change Description |
|---|---|---|---|
| 30 Apr- 22 | 0.1 | Nguyen Phuong Nhat Khoa | Create SRS for Cursus |
| 5 May –24 | 0.2 | Tuan Tran | Modify SRS use case, paint point. |

# MỤC LỤC

# 1. Introduction

## 1.1 Purpose

The Functional Requirements Specification will:

- ❖ Define the scope of business objectives, business functions, and organizational units covered,
- ❖ Identify the business processes that the solution must facilitate,
- ❖ Facilitate a common understanding of what the functional requirements are for all parties involved,
- ❖ Establish a basis for defining the acceptance tests for the solution to confirm that what is delivered meets requirements.
- ❖ The purpose of the document is to collect and analyse all assorted ideas that have come up to define the system, its requirements with respect to consumers. Also, we shall predict and sort out how we hope this product will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops.

## 1.2 Overview

Overview - Litware, Inc. is a medium-sized Education technology company that has a main office in Boston. Litware researched about the Edtech domain and want to build an Online Course Management with name **Cursus** as MVP (Minimum Value Project - simple) version to test the EU market trend.

Cursus system will allow user to be a part of Student/Studier and Instructor. Coacher will publish the course and student can purchase and study. Student could interact with Instructor as well.

Litware consider about the Client-Side architect for this Cursus to improve the user exeprience.

## 1.3 Intended Audience and Reading Suggestions

This document is intended for:

- ❖ Development team: Responsible to develop detailed design, implement and perform unit test. Also perform test manual.
- ❖ Project Owner: Responsible to manage the Product backlog for the application.

## Abbreviations

| Acronym | Reference |
|---------|-----------|
| SRS | Software requirement specification |
| UC | Use case |
| | |
| | |

## 2. High Level Requirements

This section describes the general overview of the system functions or business processes which are depicted in different diagrams. It shows the types of users, their permissions granted to perform specific system functions and the sequence required to complete a business workflow (if any). As the section name implies, it is high-level which means it is not detailed enough. For detailed requirement specification, please see Error! Reference source not found. Error! Reference source not found. section below.

### 2.1 System structure

The system architecture will be described in the software development phase.

### 2.2 Workflow

This section shows the flow of tasks or steps taken by each users of the system in-order to complete a business process. The user's actions are shown in each business process stage of the system and what happens before it can move to the next stage or revert to the previous.
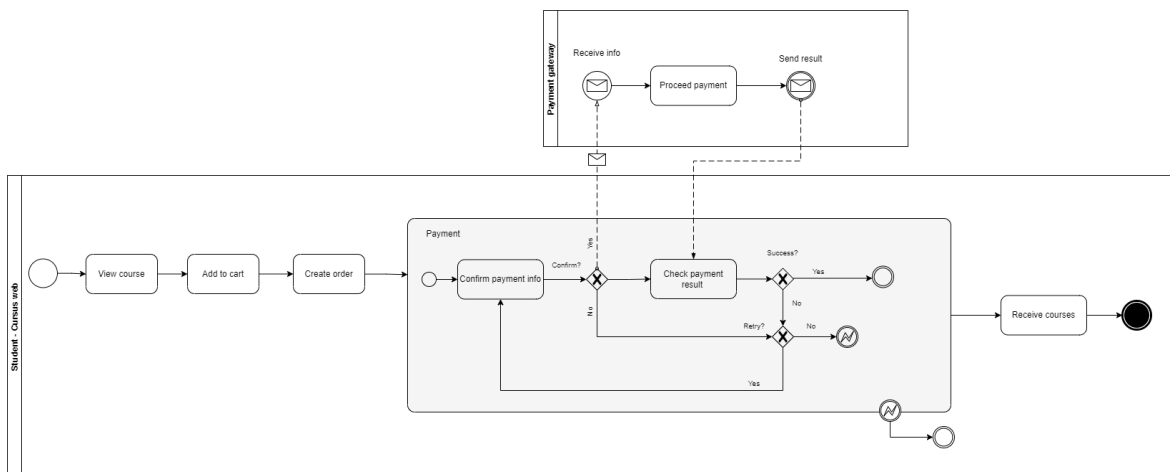


### 2.3 Use Case Diagram

The use case diagram here shows the specific goal and objective or how the user interacts with the system. The eclipse in the system boundary represents the system use case/functions while the stickman represents the actor/user of the system. The line connecting the actor and the use case shows that the actor can perform that function in the system to achieve a goal.

#### 2.3.1 General users

**Student:** the person who is in Cursus as Student Role, can view eligible courses, purchase and enroll into the courses, give feedback, manage personal user information.

**Instructor:** the person who provides the courses like teacher, mentor, support the course. This actor can get back the course fee that Student are paid.

**Guest:** the person who visits the Cursus and looks in to be the Student or Instructor.

View dashbord

CRUD course

View reviews

View earning analytics

Payout

Setting

View/ edit profile

Change password

Instructor

Autthenticate

Log in/ out/ register

Verify email

Create password

Guest

View homepage

Search/ View course

Search/ View enrolled course

Search/ View course by category

Setting

View/ edit profile

Change password

View/ search list of instructors

Buy course

Student

Enroll course

Review/ Report course

Saved course

### 2.3.2 Administrator

The person who managed the Cursus system. This role mainly can support to approve/reject the Instructor's courses and manage all courses, also can take care of the health of Cursus system.

System admin

- Log in/log out
- View dashboard
- CRUD categories
- CRUD sub-categories
- Manage student
  - View list of student
  - Block/ un-block student
- Manage instructor
  - View list of instructor
  - Block/ un-block instructor
  - Proceed payout
- Manage Course
  - View list of course
  - Review/ approve/ Reject/ block course
- Manage admin
  - View list of admin
  - Block/ unblock admin
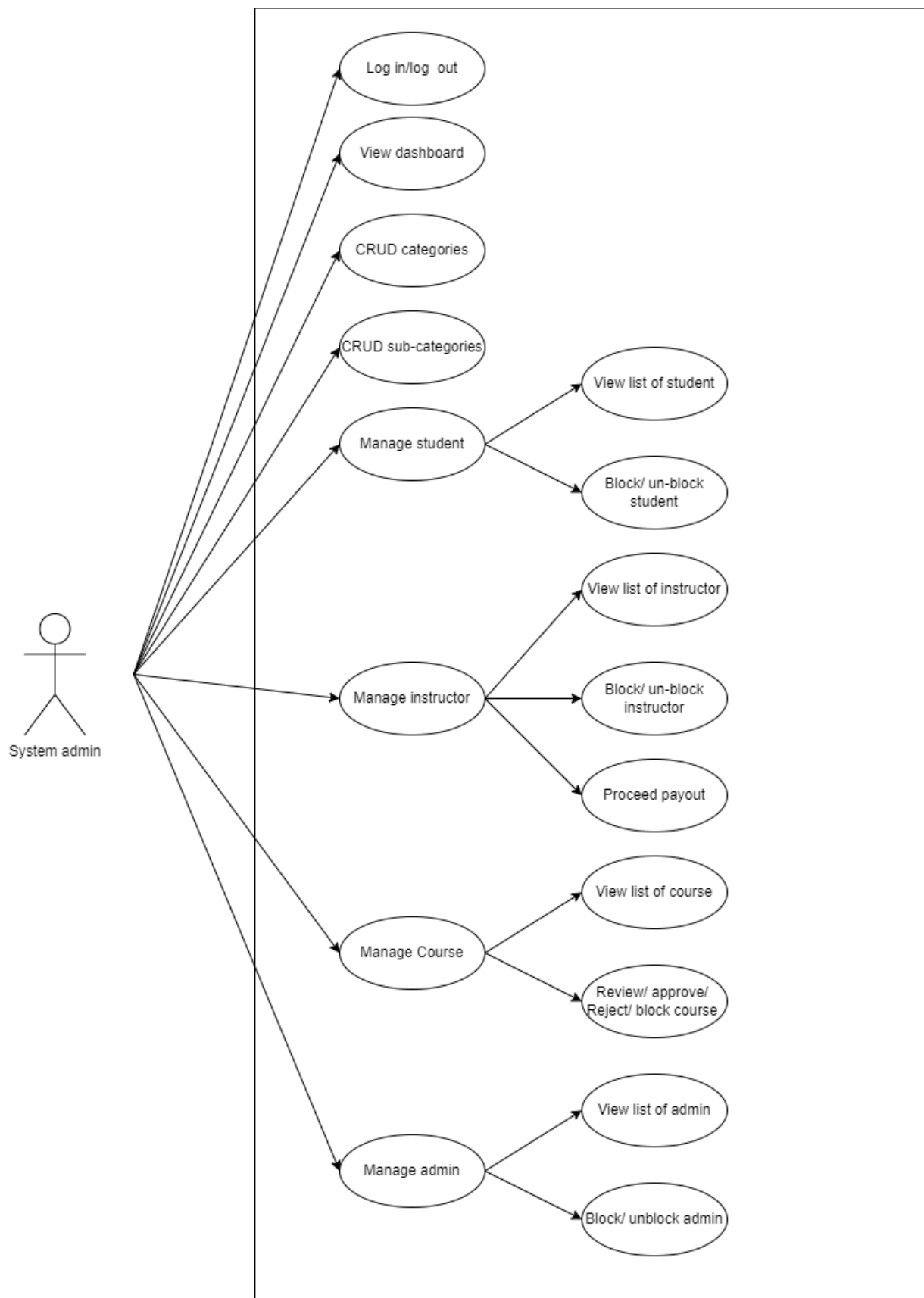
## 2.4 Permission matrix

Permission Matrix mapping functions and user roles for <<App Name>> application is described as below:

Remark:

"O" means that user has permission on corresponding function. For more information about what the actor can do on that function, please refer to corresponding use case.

 "X" means that user does not have permission on corresponding function.

| Function | Guest | Student | Instructor | Admin |
|---|---|---|---|---|
| **Authentication** | | | | |
| Register | O | O | O | X |
| Create password | X | O | O | X |
| Log in/ log out | O | O | O | O |
| **Homepage/ Dashboard** | | | | |
| View homepage as guest | O | O | O | X |
| View homepage by student | X | O | X | X |
| View Dashboard | O | O | O | O |
| View course enrolled. | X | O | X | X |
| View total of student/ course | X | X | O | X |
| View profile analytics | X | X | O | X |
| **Setting** | | | | |
| View/ edit profile | X | O | O | X |
| Change password | X | O | O | X |
| **Purchase courses** | | | | |
| View/ add/ edit/ delete shopping cart | X | O | X | X |
| Proceed purchase course | X | O | X | X |
| **Enroll course** | | | | |
| Enroll into a course | X | O | X | X |
| View tracking of the course | X | O | X | X |
| Review course | X | O | X | X |
| Report course | X | O | X | X |
| Saved course | X | O | X | X |
| **Dashboard for Instructor** | | | | |
| View course/ student analytics | X | X | O | X |

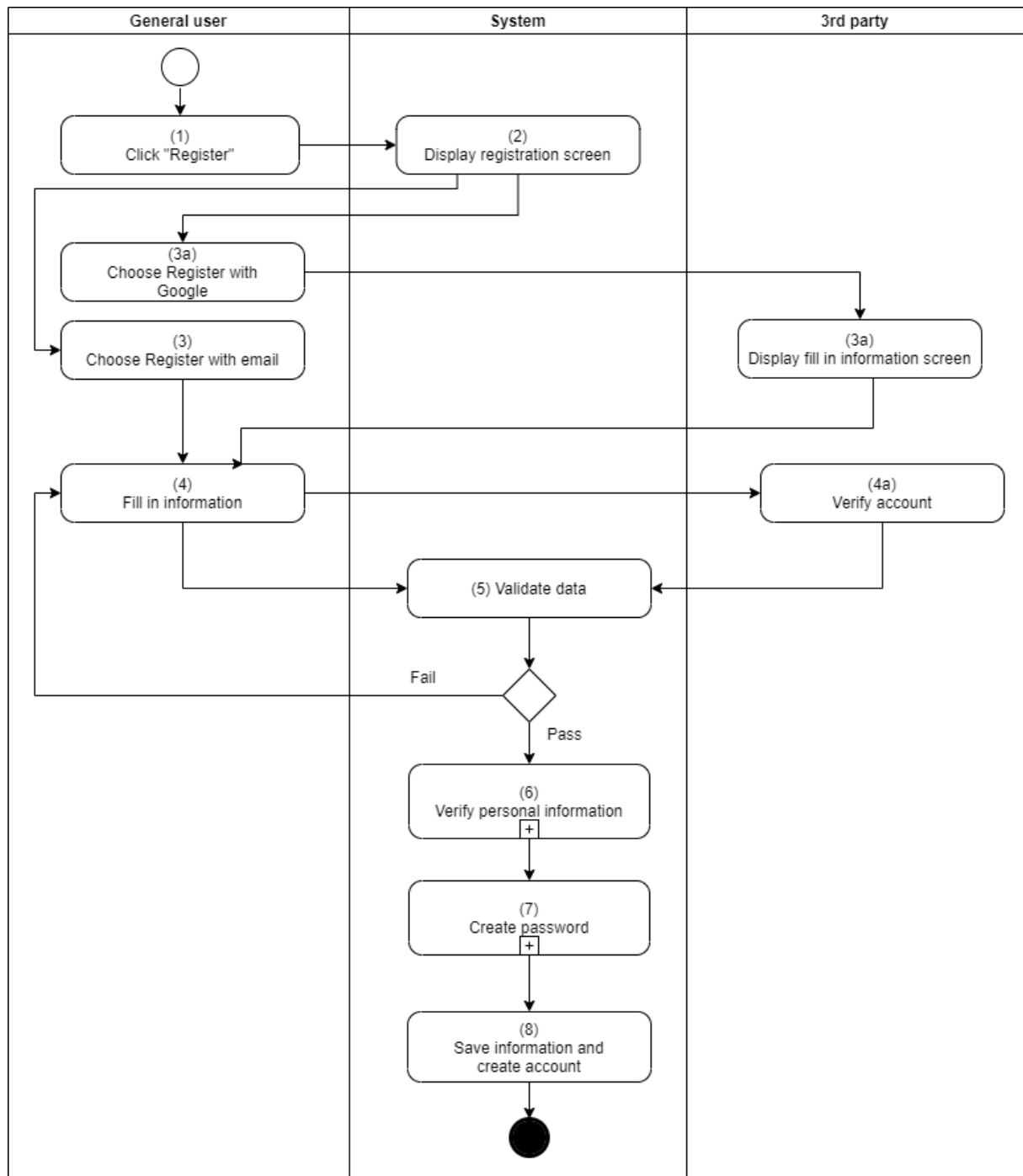| Manage course | | | | |
|---|---|---|---|---|
| View list course | X | X | O | X |
| Create/ edit/ delete course | X | X | O | X |
| Submit course for approval | X | X | O | X |
| **Earning Affiliate** | | | | |
| View earning money | X | X | O | X |
| Payout | X | X | O | X |
| **Dashboard for admin** | | | | |
| View dashboard | X | X | X | O |
| **Manage categories** | | | | |
| Create/ view/ search/ edit/ delete categories | X | X | X | O |
| **Manage sub-categories** | | | | |
| Create/ view/ search/ edit/ delete sub-categories | X | X | X | O |
| **Manage student** | | | | |
| View list of students | X | X | X | O |
| Block/ unblock student | X | X | X | O |
| **Manage instructor** | | | | |
| View list of instructors | X | X | X | O |
| Block/ unblock instructor | X | X | X | O |
| Approve/ reject the course. | X | X | X | O |
| **Manage course** | | | | |
| View list of course | X | X | X | O |
| Review/ Approve/ Reject/ Block/ Unblock course | X | X | X | O |
| **Manage admin** | | | | |
| View list of admins | X | X | X | O |

# 3.  Use Case Specifications

This section covers the system's functional requirements which details what the system must do in terms of input, behavior and the expected output. It elicits the interaction between the actor(s) and the system, the system's behavior and the results of their interactions.

## 3.1 General functions

### 3.1.1 UC 01: Sign up for Student

| | |
|---|---|
| **Objective:** | Allow general user to create new account on Cursus |
| **Actor:** | 1. Guest |
| **Trigger:** | 1. User chooses Sign up on Log in pop up<br>2. User chooses "Start Learning" at homepage |
| **Pre-condition:** | User has no account yet. |
| **Post-condition:** | User creates new account successfully |
| **Basic flow:** | (1) At Log in pop up/ Start learning, user choose the Register.<br>(2) System displays registration screen.<br>(3) User chooses:<br>    - Register (with email), system will show the registration form.<br>    - Register with Google (External registration), system redirects to registration form with detected email from Google.<br>(4) User fills other information (email (if no), full name, address, password...) and submit.<br>(5) System processes the data with validation as .NET default.<br>(6) System response the result:<br>    - If success, send email confirmation (content of email: ET 1) & redirect to confirmation page<br>    - If failed, show the reason in the page. |
| **Alternative flow** | N/A |

**Activity flow**

**Business rules**

| Step | BR Code | Description |
|------|---------|-------------|
| (2) | BR 1 | **Display rules:**<br><br>- When user click Log in button at homepage, website displays Log in pop up. At this pop up, user click Register to show registration screen. |

| | | |
|---|---|---|
| | | - Button "Register with Google" always shows above other fields. User clicked this button, redirect to log in screen of 3<sup>rd</sup> party – Google<br><br>- Full name – max length: 200 characters<br><br>- Email: user's email. If user input, format must correct format of email as abc@abc.xyz. If user enters wrong format → show MSG1 in red and below the field.<br><br>- Button "Submit": disable until user fills all information |
| (5) | BR 2 | **Validate rules:**<br><br>When user fills all information and click "Submit" button, system validates data as the rules below:<br><br>- If any required fields are left blank, the system displays error message MSG 2. MSG 2 is displayed below the blank field in red, italic format.<br><br>- If username or email existed in system → show MSG 3 |
| (7) | BR 3 | |

### 3.1.2 UC 02: Sign up for Instructor

| | |
|---|---|
| **Objective:** | Allow general user to create new account on Cursus |
| **Actor:** | 2. Guest |
| **Trigger:** | 3. User chooses "Join to be Instructor" on Log in pop up |
| **Pre-condition:** | User has no account yet. |
| **Post-condition:** | User creates new account successfully |
| **Basic flow:** | 1. Click "Join to be Instructor" to redirect to Instructor registration screen.<br>2. System displays registration screen.<br>3. User chooses:<br>    - Register (with email), system will show the registration form.<br>    - Register with Google (External registration), system redirects to registration form with detected email from Google.<br>4. User fills other information (email (if no), full name, address, password, professional experience) and submit.<br>5. System processes the data with validation as .NET default.<br>6. System response the result:<br>    - If success, send email confirmation (content of email: ET 1) & redirect to confirmation page that inform that "Bạn đã đăng ký làm Giáo viên, chúng tôi sẽ xem xét thông tin của bạn trong X ngày."<br>    - If failed, show the reason in the page. |
| **Alternative flow** | N/A |

## Activity flow

## Business rules

| Step | BR Code | Description |
|------|---------|-------------|
| (2) | BR 1 | **Display rules:**<br>- When user click Log in button at homepage, website displays Log in pop up. At this pop up, user click Register to show registration screen.<br>- Button "Register with Google" always shows above other fields. User clicked this button, redirect to log in screen of 3$^{rd}$ party – Google<br>- Full name – max length: 200 characters<br>- Email: user's email. If user input, format must correct format of email as abc@abc.xyz. If user enters wrong format → show MSG1 in red and below the field.<br>- Button "Submit": disable until user fills all information |
| (5) | BR 2 | **Validate rules:**<br>When user fills all information and click "Submit" button, system validates data as the rules below:<br>- If any required fields are left blank, the system displays error message MSG 2. MSG 2 is displayed below the blank field in red, italic format.<br>- If username or email existed in system → show MSG 3 |
| (7) | BR 3 | **Create rules:**<br>When system validate information successfully, the system proceeds to create new account:<br>- Display message MSG 4<br>- Add record for new user in DB<br>- Save user information in DB<br>- Send confirmation email to be corresponding. |

### 3.1.3  UC 03: Sign in
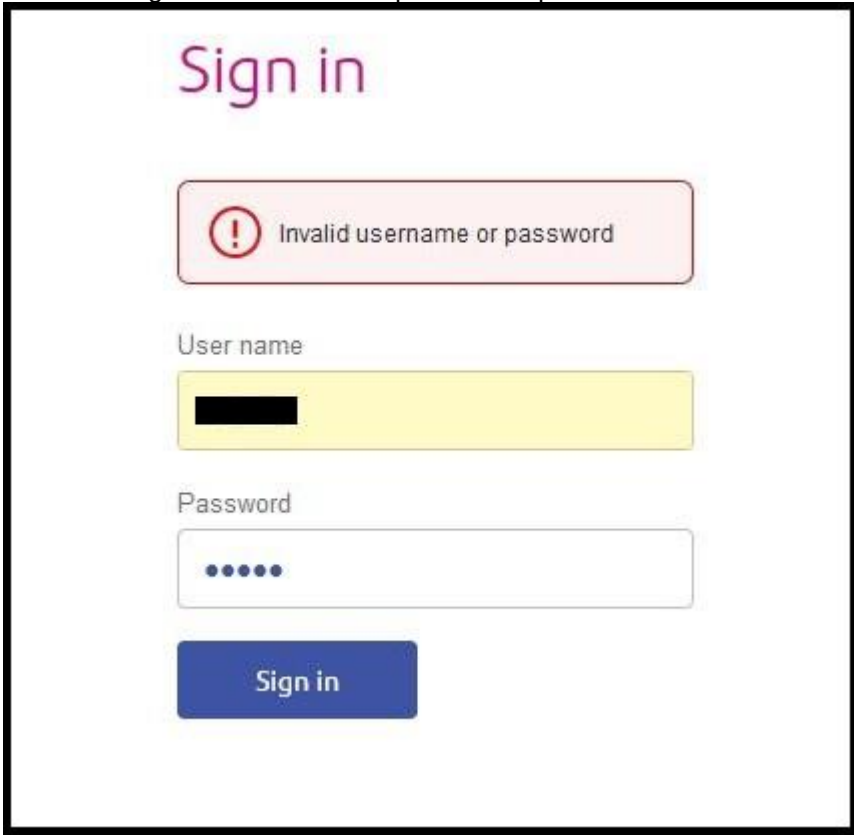
| | |
|---|---|
| **Objective:** | Allow user Login to website. |
| **Actor:** | All types of users |
| **Trigger:** | User clicks "Log in" button at homepage |
| **Pre-condition:** | User has already had account in system |
| **Post-condition:** | User signs in successfully |
| **Basic flow** | (1). User clicks Log in at homepage <br><br> (2). System displays Log in pop up. <br><br> (3). User fills information: email + password <br><br> (4). System validates data & response |
| **Alternative flow** | (1). User clicks Log in at homepage <br><br> (2). System displays Log in pop up <br><br> (3a). User chooses Log in with Google <br> (3b) 3<sup>rd</sup> party displays Log in screen <br> (3). User fills information <br><br> (4a). 3<sup>rd</sup> party validates data & response the data. <br><br> (5). System wi |

**Activity flow**

**Business Rules**

| Step | BR Code | Description |
|------|---------|-------------|
| (2) | BR 4 | **Display rules:**<br><br>System displays Log in pop up with these fields:<br><br>- Button "Login with Google" is always on the top of pop up<br>- Field "Email": free text. This field allows user to enter the email which user signed up before.<br>- Field "Password": free text.<br>- "Remember me": check box. If user ticks this box, user will not enter email and password again. System will remember user information and allow to log in automatically.<br>- Button "Log in": call API to validate user's information |
| (4) | BR 5 | **Validate rules:**<br><br>After user clicks "Log in" at Log in pop-up, system will validate based on the existing rules.<br><br>The validation logic is:<br><br>- If the value of any mandatory fields is blank, system will show an error message for the required fields as MSG<br>- If the email does not exist in database, the system will show an error message as MSG<br>- If the password is not correct, the system will show an error message as MSG. |

| | | 1. Error message will be show on top. For example: |
|---|---|---|
| | |  |
| (5) | BR 6 | **Save rules:** |
| | | System records information with last success login datetime for the user. |

### 3.1.4  UC 04: Verify email

| | |
|---|---|
| **Objective:** | Allow user to verify email used to register account |
| **Actor:** | Guest. |
| **Trigger:** | System verifies user information successfully |
| **Pre-condition:** | User registered successfully with UC 01 with Register (with email) type. |
| **Post-condition:** | User account is activated. |
| **Basic flow:** | (1). User received the confirmation email from UC 01 |
| | (2)  User clicks on the email link and redirect to the Cursus system confirmation page. |
| | (3). System response the result: |
| | - Success, active account. |
| | - Failed: show reason and do not activate the account. |

| **Alternative flow:** | *N/A* |
|---|---|
| **Exception flow:** | |

## Activity flow
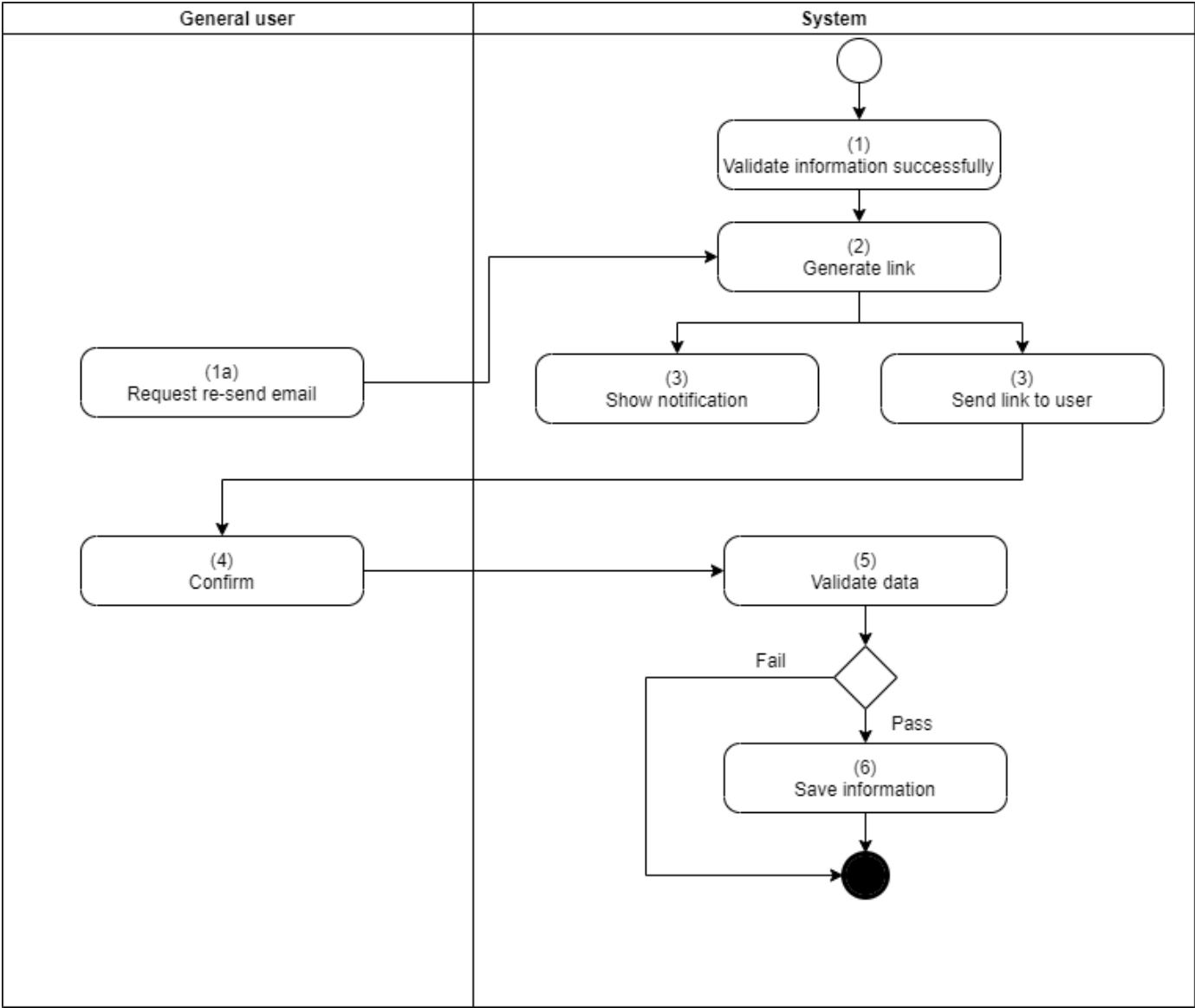


## Business Rules

| Step | BR Code | Description |
|---|---|---|
| (1) | BR 11 | Content of email need to follow the template - ET1 |

| (2) | BR 13 | - Email link should be valid within 24hrs, otherwise, return the MSG 5 in Confirmation Result page. |
| | | - If successful, return to the login page of UC 02 after 5 seconds. |

### 3.1.5 UC 05: Change password

| Objective: | Allow user changes the password. |
| --- | --- |
| Actor: | All types of users |
| Trigger: | User chooses "Password" at Setting screen. |
| Pre-condition: | User logs in successfully. |
| Post-condition: | User changes the password successfully. |
| Basic flow: | |

**Activity flow**
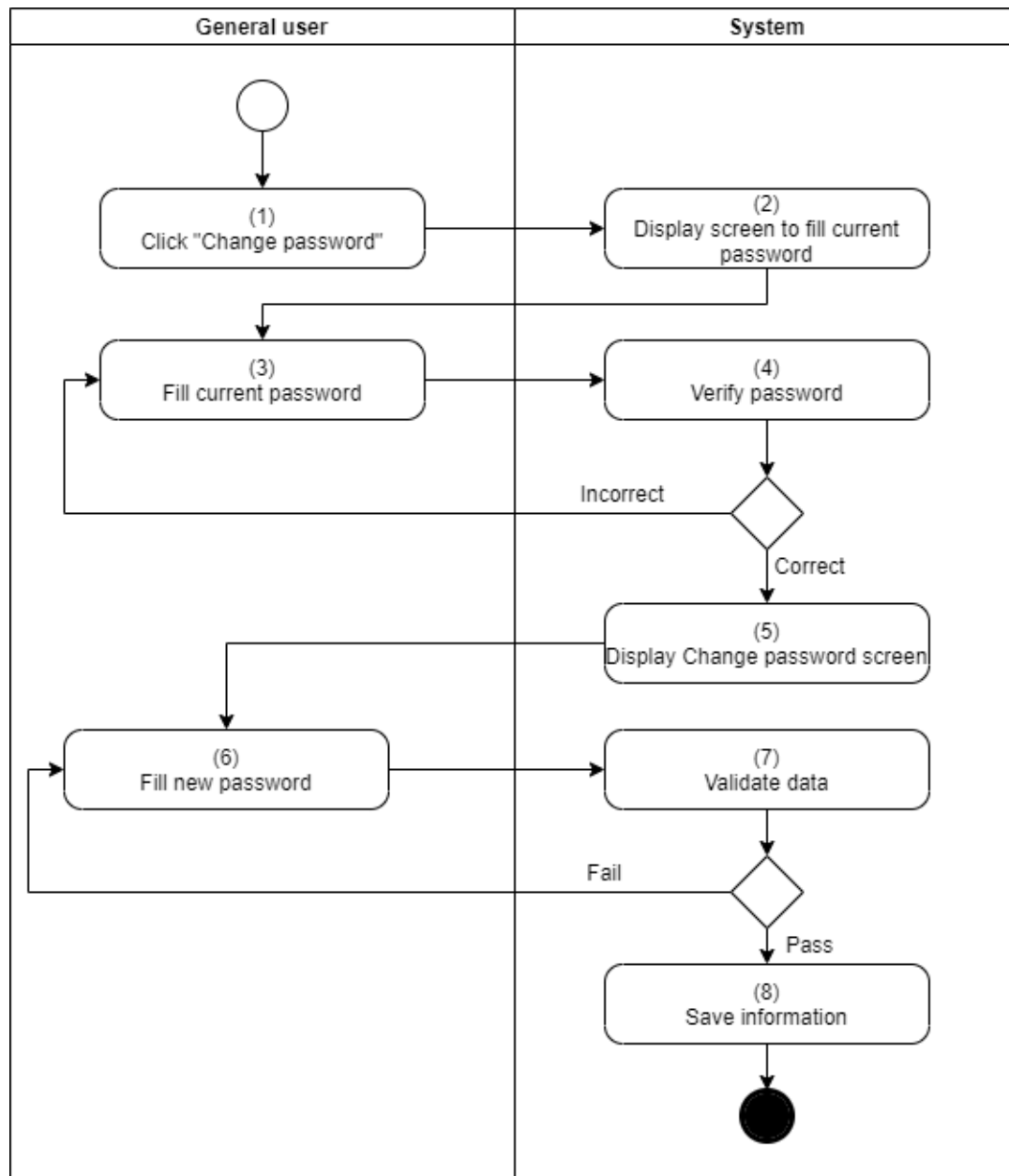
**Business Rules**

| Step | BR Code | Description |
|------|---------|-------------|
| (1) | BR 21 | Show Change password form with information:<br><br>- Current password.<br>- New password.<br>- Confirm password. |

| (2) | | **<u>Validation rule:</u>**<br><br>-      New password can't be same with current password. Show message MSG 22 if the same. |
| --- | --- | --- |

### 3.1.6  UC 06: Forgot password

| | |
| --- | --- |
| **Objective:** | Allow user creates new password if user forgot the current password. |
| **Actor:** | General user |
| **Trigger:** | User clicks "Forgot password" at Log in screen. |
| **Pre-condition:** | User already has account on Cursus. |
| **Post-condition:** | User creates new password successfully. |
| **Basic flow:** | (1). The user selects Forgot password.<br><br>(2). The system displays a screen with email input.<br><br>(3). User inputs the email then submits.<br><br>(4). System checks the email:<br><br>- If existed in system, sends the email ET 2 to the input email.<br><br>- If not exist, response message MSG 8. |

**Activity flow**
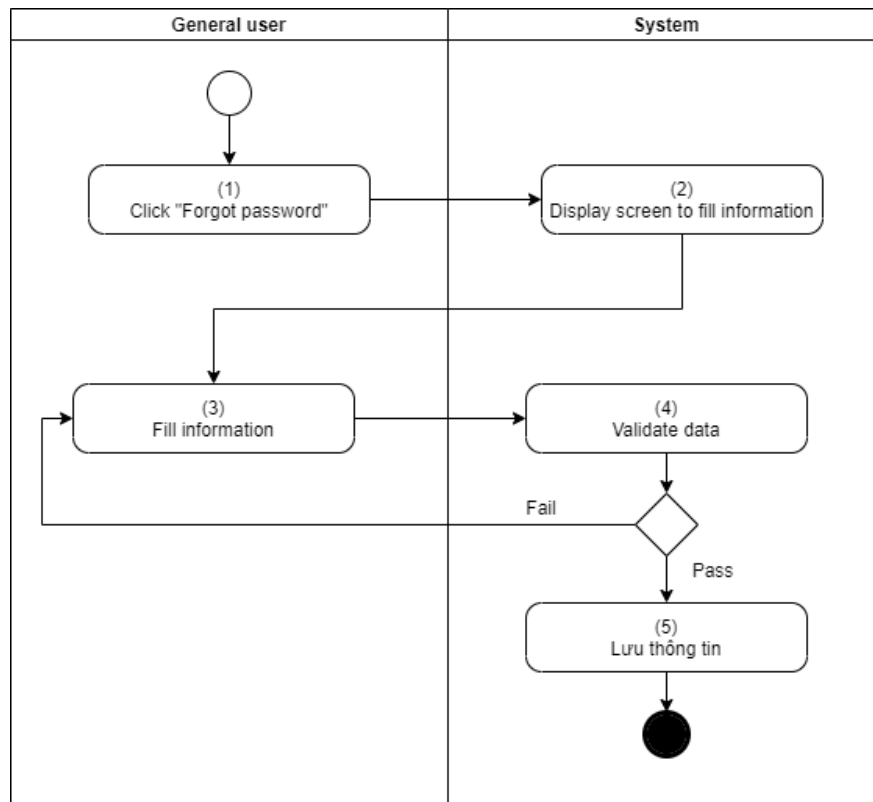
## 3.2 General user

### 3.2.1 UC 07: View Homepage

| | |
|---|---|
| **Objective:** | Allow user to view information at homepage |
| **Actor:** | General user |
| **Trigger:** | 1. User logs in successfully<br>2. User accesses website successfully |
| **Pre-condition:** | User accesses website successfully |
| **Post-condition:** | User views information at homepage |
| **Basic flow:** | (1). User can click on the logo<br><br>(2). System redirects to the user info page. |

**Activity flow**
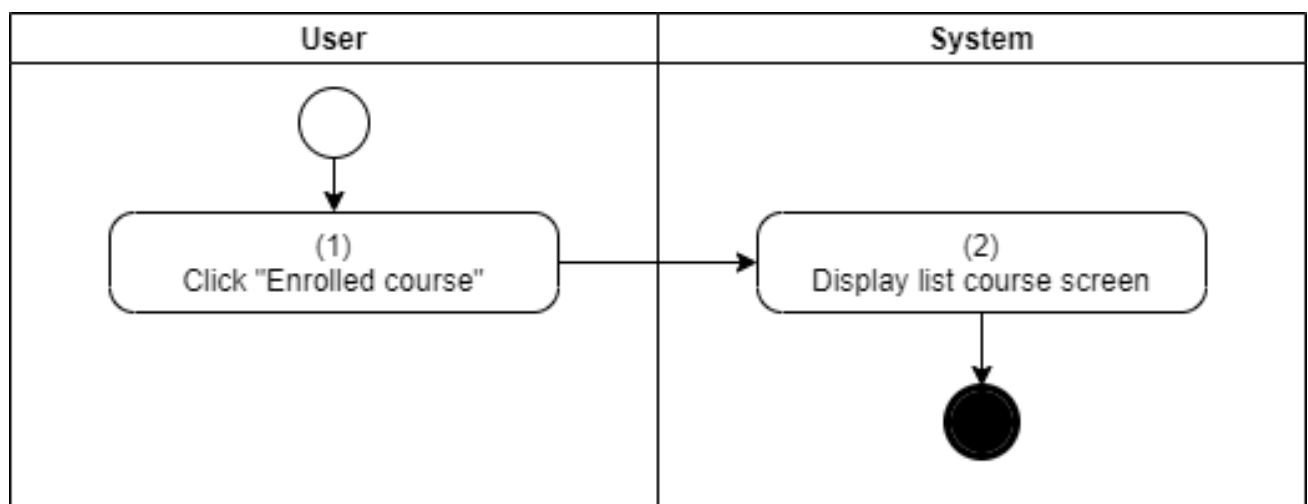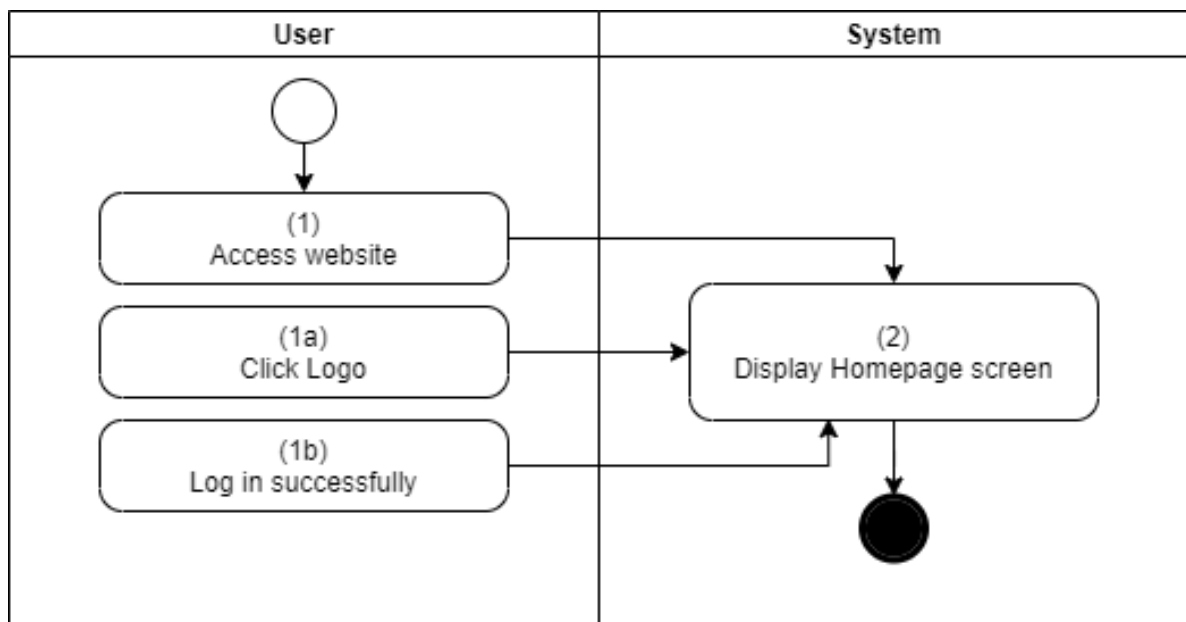
**Business Rules**

| Step | BR Code | Description |
| --- | --- | --- |
| (2) | BR 35 | - |

### 3.2.2  UC 08: Update profile

| Objective: | Allow user to update phone number/email to personal information above |
| --- | --- |
| Actor: | Regular users |
| Trigger: | User selects email at the account screen |

| Pre-conditions: | User logged in successfully |
|---|---|
| Post-conditions: | User successfully added phone number/email |
| Basic flows: | (1). User selects email |
| | (2). Screen display system Enter email |
| | (3). User enters email |
| | (4). Information checking system |
| | (5). The system performs verification via email |
| | (6). The system stores information |

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (1) | BR 30 | **Display rules:** <br><br> The system displays the email input screen |
| (2) | BR 31 | **Rules for checking information:** <br><br> - Users can change the information, except email. <br> - After changed, the system should be updated. |
| (3) | BR 32 | **Save information** <br><br> After the email authentication process is successful, the system adds email information to the DB. |

### 3.2.3  UC 09: View and Search course

| Objective: | Allow users to view all courses on Cursus. |
|---|---|
| Actor: | General users |
| Trigger: | User clicks "Courses" or click on each category. |
| Pre-conditions: | User accesses the website successfully. |
| Post-conditions: | User views course information successfully. |
| Basic flows: | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|------|---------|-------------|
| (1) | BR 33 | Only activated courses can be shown in the list. |
| (2) | BR 34 | Can filter by search keyword (course name, instructor name) |
| (3) | BR 35 | Default has paging (10, 25, 50) per page. Short default as descending by Course Name, review points. |

### 3.2.4  UC 10: View enrolled course

| | |
|---|---|
| **Objective:** | Allow user to view all courses that user registered before |
| **Actor:** | student |
| **Trigger:** | User clicks "Enrolled course" on the left menu |
| **Pre-conditions:** | User logs in successfully as actor above. |
| **Post-conditions:** | User can view all of enrolled courses |
| **Basic flows:** | |
| **Alternative flows:** | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 35 | – |

### 3.2.5  UC 11: View course by category

| | |
|---|---|
| **Objective:** | Allow user to view courses by category |

| Actor: | student |
|---|---|
| Trigger: | User clicks "Category" on the left menu |
| Pre-conditions: | User logs in successfully as actor above. |
| Post-conditions: | User view courses by category successfully |
| Basic flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 36 | – |
| (4) | BR 37 | |
| (6) | BR 38 | |

### 3.2.6   UC 12: View list of subscriptions

| Objective: | Allow user to view list of instructors that user subscribed to before. |
|---|---|
| Actor: | student |
| Trigger: | User clicks "Subscriptions" on the left menu. |
| Pre-conditions: | User logs in successfully as actor above. |
| Post-conditions: | User views course by category successfully. |
| Basic flows: | |
| Alternative flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|------|---------|-------------|
| (2)  | BR 39   |             |

### 3.2.7  UC 13: Search instructor

| | |
|---|---|
| **Objective:** | Allow user to search instructor by name |
| **Actor:** | student |
| **Trigger:** | User clicks Search icon. |
| **Pre-conditions:** | User logs in successfully as actor above. |
| **Post-conditions:** | User searches instructor by namw successfully |
| **Basic flows:** | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|------|---------|-------------|
| (4)  | BR 40   | o |
| (6)  | BR 41   | ❖ |

### 3.2.8  UC 14: Purchase the course

| | |
|---|---|
| **Objective:** | Allow user to purchase courses |
| **Actor:** | student |
| **Trigger:** | User clicks Add to cart icon |
| **Pre-conditions:** | User logs in successfully as actor above. |
| **Post-conditions:** | User buys courses successfully |

| Basic flow | |
|---|---|
| | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (4) | BR 42 | |
| (6) | BR 43 | ❖ |

### 3.2.9  UC 15: Enroll course

| Objective: | Allow user to Enroll course that user bought before. |
|---|---|
| Actor: | student |
| Trigger: | User chooses a course which was bought before |
| Pre-conditions: | 1. User logs in successfully as actor above.<br>2. The course was bought successfully |
| Post-conditions: | User can enroll the course |
| Basic flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 44 | |
| (4) | BR 45 | |

### 3.2.10 UC 16: Review course

| | |
|---|---|
| **Objective:** | Allow user to review the course that user has finished. |
| **Actor:** | student |
| **Trigger:** | User clicks "Review the course" at thank you screen |
| **Pre-conditions:** | 1. User logs in successfully as actor above.<br>2. The course has been completed by user |
| **Post-conditions:** | User reviews the course successfully |
| **Basic flows:** | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|---|---|---|
| (2) | BR 85 | |
| (4) | BR 86 | |

### 3.2.11 UC 17: Report course

| | |
|---|---|
| **Objective:** | Allow user to report the courses which are violating the policy or the content does not match the description. |
| **Actor:** | student |
| **Trigger:** | User clicks "Report" button at each course. |
| **Pre-conditions:** | User logs in successfully as actor above. |
| **Post-conditions:** | User reports the course successfully. |
| **Basic flows:** | |

**Activity flow**

## Business Rules

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 133 | |
| (4) | BR 134 | |

### 3.2.12 UC 18: View Dashboard

| | |
|------|------|
| **Objective:** | Allow users to view overall information at dashboard |
| **Actor:** | Instructor |
| **Trigger:** | User logs in successfully. |
| **Pre-conditions:** | User logs in successfully as actor above. |
| **Post-conditions:** | User views dashboard successfully |
| **Basic flows:** | |
| **Alternative flows:** | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 60 | |
| (4) | BR 61 | ❖ |

### 3.2.13 UC 19: View course/ student analytics

| | |
|------|------|
| **Objective:** | Allow user to view all information about user's course and view analytics about students |
| **Actor:** | Instructor |
| **Trigger:** | User chooses Course/ Analytics |

| Pre-conditions: | User logs in successfully as actor above. |
|---|---|
| Post-conditions: | User views course/ analytic successfully. |
| Basic flows: | |
| Alternative flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 60 | |
| (4) | BR 61 | |

### 3.2.14  UC 20: Create course

| Objective: | Allow users to create new courses |
|---|---|
| Actor: | Instructor |
| Trigger: | 1. User clicks "Create course" button at homepage<br>2. User clicks "Create course" on the left menu |
| Pre-conditions: | User logs in successfully as actor above. |
| Post-conditions: | User Create course successfully |
| Basic flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|

| (2) | BR 48 | |
|-----|-------|---|
| (5) | BR 49 | ❖ |

### 3.2.15  UC 21: Edit course

| Objective: | Allow users to edit course information |
|------------|---------------------------------------|
| Actor: | Instructor |
| Trigger: | User clicks Edit icon at list course screen |
| Pre-conditions: | User logs in successfully as actor above. |
| Post-conditions: | User edits the course successfully. |
| Basic flows: | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 50 | |
| (4) | BR 51 | ❖ |
| (6) | BR 52 | |

### 3.2.16  UC 22: Re-submit course

| Objective: | Allow users to modify and resubmit course |
|------------|--------------------------------------------|
| Actor: | Instructor |
| Trigger: | User goes to the course |
| Pre-conditions: | Course is submitted and got rejection from admin. |
| Post-conditions: | User resubmit the course successfully. |

| Basic flows: | 1. User access the rejected course. |
| --- | --- |
| | 2. User modifies the information. |
| | 3. User submit the course again. |

## Activity flow

## Business Rules

| Step | BR Code | description |
| --- | --- | --- |
| (2) | BR 50 | |
| (4) | BR 51 | ❖ |
| (6) | BR 52 | |

### 3.2.17  UC 23: Delete course

| Objective: | Allow user to delete course. |
| --- | --- |
| Actor: | Instructor |
| Trigger: | User clicks Delete icon at list course screen. |
| Pre-conditions: | User logs in successfully as actor above. |
| Post-conditions: | User deletes course successfully |
| Basic flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
| --- | --- | --- |
| | | |

| (2) | BR 53 | **Display rules** |
| --- | --- | --- |
|     |       | The system displays a confirmation message MSG 13. MSG 13 has 2 buttons "Agree" and "Cancel" |
| (4) | BR 54 | **Rules for saving information**<br>❖ If the user selects "Cancel", the use case stops and the system displays the previous screen.<br>❖ If the user chooses "Agree", the system will: Delete community information from the system<br>❖ Cannot delete the course that has student enrollment. |

### 3.2.18  UC 24: View reviews

| | |
| --- | --- |
| **Objective:** | Allow user to view all reviews about user's course |
| **Actor:** | Instructor |
| **Trigger:** | User clicks "review" on the left menu |
| **Pre-conditions:** | User logs in successfully as actor above. |
| **Post-conditions:** | User views all reviews successfully |
| **Basic flows:** | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
| --- | --- | --- |
| (2) | BR 58 | |
| (4) | BR 59 | ○ |

### 3.2.19  UC 25: View earnings analytics

| | |
| --- | --- |
| **Objective:** | Allow user to view the amount of money that earned from courses |
| **Actor:** | Instructor |
| **Trigger:** | User clicks "Earning" on the left menu |
| **Pre-conditions:** | User logs in successfully as actor above. |

| Post-conditions: | User view earnings analytic successfully |
|---|---|
| Basic flows: | 1. *User goes to Earning page.*<br>2. *User can see the earning list by months as list.* |
| Alternative flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 60 | |
| (4) | BR 61 | ❖ |

### 3.2.20 UC 26: Payout

| Objective: | Allow user to withdraw money to the bank account |
|---|---|
| Actor: | Instructor |
| Trigger: | User clicks "Payout" on the the earnings analysis page. |
| Pre-conditions: | User logs in successfully as actor above. |
| Post-conditions: | User withdraws money successfully |
| Basic flows: | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 62 | User cannot withdraw money if zero. |

| (4) | BR 63 | User will receive the email for the Payout |
|-----|-------|-------------------------------------------|

## 3.3 Admin

### 3.3.1 UC 27: Login

| Objective: | Allow admin to login into administration site. |
|------------|------------------------------------------------|
| Actor: | System admin |
| Trigger: | Admin logs in successfully. |
| Pre-conditions: | N/A |
| Post-conditions: | Admin has account. |
| Basic flows: | (1). User access to admin site (ex: abc.com\admin) then input the email & password.<br><br>(2). User can access the admin site. |

### 3.3.2 UC 28: View dashboard
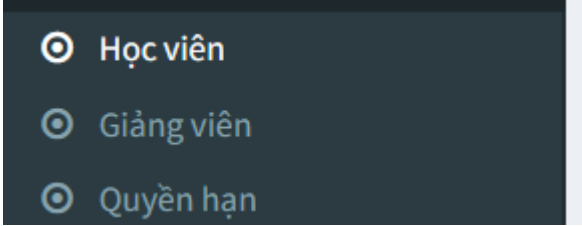
| Objective: | Allow admin to view dashboard |
|------------|-------------------------------|
| Actor: | System admin |
| Trigger: | Admin logs in successfully. |
| Pre-conditions: | Admin logs in successfully. |
| Post-conditions: | Admin views dashboard successfully. |
| Basic flows: | (1). User logs in successfully<br><br>(2). System displays dashboard screen |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 87 | |

### 3.3.3 UC 29: View list of students/instructors

| Objective: | Allow admin to manage the user of this system by each type. |
|---|---|
| Actor: | Admin |
| Trigger: | |
| Pre-conditions: | Admin is logged in. |
| Post-conditions: | |
| Basic flows: | 1. Admin accesses the admin page.<br><br>2. User can see the Users menu. At here:<br><br>- Choose the "Student" submenu to manage student.<br><br>- Choose the "Instructor" submenu to manage the instructor.<br><br> |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|---|---|---|
| (2) | BR 91 | |
| (4) | BR 92 | ❖ |
| (6) | BR 93 | |
| (8) | BR 94 | |

### 3.3.4 UC 30: Approve or Reject Instructor registration

| Objective: | Allow admin to approve/reject the instructor of this system by each type. |
|---|---|
| Actor: | Admin |

| Trigger: | |
|---|---|
| Pre-conditions: | Admin access to the instructor submenu under User menu. |
| Post-conditions: | |
| Basic flows: | 1. Admin selects the instructor that is registered and has not activated at the instructor list. 2. Admin choses: - Approve: instructor account is activated with email ET 3 - Reject: instructor account keeps inactivate with ET 4. |

### 3.3.5  UC 31: View list of categories

| Objective: | |
|---|---|
| Actor: | |
| Trigger: | |
| Pre-conditions: | |
| Post-conditions: | |
| Basic flows: | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|---|---|---|
| (2) | BR 95 | |
| (4) | BR 96 | |

### 3.3.6  UC 32: Create a category

| Objective: | |
|---|---|
| Actor: | |
| Trigger: | |

| | |
|---|---|
| **Pre-conditions:** | |
| **Post-conditions:** | |
| **Basic flows:** | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 97 | |
| (4) | BR 98 | |

### 3.3.7  UC 33: Edit a category

| | |
|---|---|
| **Objective:** | |
| **Actor:** | |
| **Trigger:** | |
| **Pre-conditions:** | |
| **Post-conditions:** | |
| **Basic flows:** | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 97 | |
| (4) | BR 98 | |

### 3.3.8 UC 34: Delete category

| | |
|---|---|
| **Objective:** | |
| **Actor:** | |
| **Trigger:** | |
| **Pre-conditions:** | |
| **Post-conditions:** | |
| **Basic flows:** | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|
| (2) | BR 97 | |
| (4) | BR 98 | |

### 3.3.9 UC 35: Block/ unblock student/ instructor

| | |
|---|---|
| **Objective:** | |
| **Actor:** | |
| **Trigger:** | |
| **Pre-conditions:** | |
| **Post-conditions:** | |
| **Basic flows:** | |

## Activity flow

## Business Rules

| Step | BR Code | description |
|---|---|---|

| (2) | BR 106 | |
|---|---|---|
| (4) | BR 107 | ❖ |

### 3.3.10  UC 36: View list of courses

| Objective: | |
|---|---|
| Actor: | |
| Trigger: | |
| Pre-conditions: | |
| Post-conditions: | |
| Basic flows: | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|---|---|---|
| (2) | BR 108 | |
| (4) | BR 109 | |

### 3.3.11  UC 37: View details of a course

| Objective: | |
|---|---|
| Actor: | |
| Trigger: | |
| Pre-conditions: | |
| Post-conditions: | |
| Basic flows: | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 108 | |
| (4) | BR 109 | |

### 3.3.12  UC 38: Approve/Reject course

| | |
|------|---|
| **Objective:** | |
| **Actor:** | |
| **Trigger:** | |
| **Pre-conditions:** | |
| **Post-conditions:** | |
| **Basic flows:** | |

**Activity flow**

**Business Rules**

| Step | BR Code | description |
|------|---------|-------------|
| (2) | BR 108 | The course is rejected will back to the previous version. |
| (4) | BR 109 | The course is approved will applied the changes. |

# 4. Non-functional requirements

## 4.1  Performance requirements

❖ Screen loading time should not exceed 5 seconds.

## 4.2 Security Requirements

❖ The system must encrypt the user's personal information: password

❖ The system must comply with applicable security standards and policies

❖ The system should be designed to ensure that data stored in the system is protected against data loss, unauthorized access, use, modification, disclosure or misuse. Only authorized people have access to the data

## 4.3 Software quality attributes

❖ The system can handle the amount of data as the number of users increases by optimizing storage and access methods.

❖ The application is guaranteed to run on all operating systems
  - o Android: from 8.0 and up
  - o iOs: from 13.1 and above

❖ The admin portal is guaranteed to run on:
  - o Windows 7 and above
  - o Chrome v88.0

# 5. Other requirements

## 5.1 Localize rules

❖ Number format:
  - ➢ Decimal digits: 2
  - ➢ Date format: DD/MM/YYYY
❖ Application language: Vietnamese

## 5.2 Common field

### 5.2.1 Freetext

All free text (single line of text input) displayed in the application, unless otherwise stated, has max length of 255 characters. This includes all text fields & standard search boxes. Once 255 characters have been inputted into the field, additional input is no longer received (not receiving input despite button pressed).

This input only accepts ASCII characters, and does not accept special character types (eg Unicode).

### 5.2.2 Dropdown list

❖ Auto focus and display the selected value when clicked to show value list.
❖ Default text when there is no selected value: "Please select item".
❖ Display 5 records, if there are more than 5, show vertical scrollbar.
❖ Sorted alphabetically.

- ❖ When the text to be displayed (in each entry, and also in the field after value has been selected) is longer than the text display box, cut the text at the end of first line, and replace the missing characters with "…". Also applicable with text suggestions during Brand input.
- ❖ Show tooltip to when hovering the text to display full text.
- ❖ Does not allow duplicate entries.
- ❖ Add 1 blank selection on top, when selected, set the place holder as "Please select item".

### 5.2.3 Scroll list for selecting Category

- ❖ Auto focus and display the selected value.
- ❖ When parent category is selected, the value list of child category will be displayed.
- ❖ Sort by alphabet order.
- ❖ Display 6 records, if there are more than 6, show vertical scrollbar.
- ❖ When click again to a selected value, reload the selection of category tree. Select on the selected value and all its child categories will be non-selected.

### 5.2.4 Image attachment

- ❖ Allow selecting and uploading 1 image at a time.
- ❖ When selecting another image to upload, this will overwrite the old image.
- ❖ After image is uploaded, display the image as thumbnail above the button.
- ❖ Allow only image files:
  - ➢ Bitmap (.bmp)
  - ➢ JPEG (.jpg, .jpeg)
  - ➢ PNG (.png)
  - ➢ GIF (.gif)
- ❖ User can delete uploaded image by clicking on option to delete it.
- ❖ Maximum allowed size: 5MB, show message "Maximum image attachment size is 5MB." under the field when user uploaded an image which its size over 5MB.
- ❖ If user selects not supported file type or format, show message MSG 28

### 5.2.5 Rich text

- ❖ Allow copy and paste with html format.
- ❖ Allow to format text with header.
- ❖ Allow text format as:
  - ➢ Bold
  - ➢ Italic
  - ➢ Underline
  - ➢ Strikethrough
- ❖ Allow text alignment.
  - ➢ Right
  - ➢ Left
  - ➢ Middle
- ❖ Allow change text color.
- ❖ Allow bulleted lists, numbered lists.
- ❖ Allow attach link (URL hyper link).
  - ➢ Anchor Link
  - ➢ URL
  - ➢ Mail Link
- ❖ Allow "Block Quote"
- ❖ Allow insert images
- ❖ Allow insert table.
- ❖ Show tooltip when hovering to each action
- ❖ Allow user edit and format using keyboard.

Last modified on 5 May –24

### 5.2.6 Text area

### 5.2.7 Date picker

❖ Validate immediately after losing focus.
❖ Format MM/DD/YYYY
  ➢ Show message "The format is invalid." under the field if user inputted date value with wrong format.
❖ Place holder: MM/DD/YYYY
❖ Default value when clicking on :<<Today>>.

### 5.2.8 Label display

❖ When the text to be displayed is longer than the text display box (Width), move the redundancy to the line below. When moving the text to new line, keep the whole word.
❖ When the text to be displayed is longer than the text display box (Height), cut the text at the end of last line, and replace the missing characters with "…".
❖ When the text to be displayed is longer than the text display box (Width) in case of display only one line, replace the missing characters with "…".
❖ Show tooltip to when hovering the text to display full text

## 5.3 Message Configuration

1. In-field error messages: Displayed in red italic text.
2. All pop-up messages: Always displayed with an "X" button on the top-right corner of the pop-up. Clicking outside of the pop-up does not close it, only the "X" button or corresponding buttons on the pop-up does.
3. After closing the popup, return to the current webpage.

## 5.4 Internal System Error

All internal system errors will share the same error message, as was defined in MSG 3

## 5.5 Pagination

Unless stated otherwise, search-related behavior in all views of the system will follow the following

specifications:

❖ When user performs a new search (both completely new and searching within the current search results), refresh table, go back to page 1.
❖ When user changes the pagination setting (number of items to show per page), refresh table, go back to page 1.
❖ When user in page 1, disable buttons of "<<" and "Previous"
❖ When user in last page, disable buttons of ">>" and "Next"
❖ Otherwise, enable all buttons.

If there is 1 page, still display the pagination with 1 page.

# 6. Appendices

## 6.1 Message list

| # | Mã số thông báo | Nội dung |
|---|---|---|
| 1. | MSG 1 | Vui lòng điền đầy đủ thông tin |

| 2.  | MSG 2  | Người dùng đã tồn tại trong hệ thống |
|-----|--------|--------------------------------------|
| 3.  | MSG 3  | Mã xác thực đã hết hiệu lực |
| 4.  | MSG 4  | Mã xác thực không chính xác. Vui lòng thử lại |
| 5.  | MSG 5  | Xác thực không thành công. Vui lòng thử lại |
| 6.  | MSG 6  | Xác thực thành công |
| 7.  | MSG 7  | Đăng ký thành công. Vui lòng kiểm tra email để hoàn thành xác nhận tài khoản. Nếu bạn không nhận được email. Ấn vào đây. |
| 8.  | MSG 8  | Tài khoản không tồn tại |
| 9.  | MSG 9  | Mật khẩu không chính xác. Bạn đã nhập sai mật khẩu <số lân> lần. Tài khoản của bạn sẽ bị vô hiệu hóa nếu nhập sai mật khẩu 3 lần |
| 10. | MSG 10 | Mật khẩu không chính xác. Tài khoản của bạn đã bị vô hiệu hóa. Vui lòng xác thực tài khoản. |
| 11. | MSG 11 | Mật khẩu không trùng khớp |
| 12. | MSG 12 | Mật khẩu không chính xác |
| 13. | MSG 13 | Bạn có chắc chắn xóa < thông báo> không? |
| 14. | MSG 14 | Bạn có muốn lưu thay đổi không? |
| 15. | MSG 15 | Bạn nhận được lời mời tham gia vào nhóm < tên nhóm> |
| 16. | MSG 16 | [tên người dùng] đã yêu cầu tham gia nhóm |
| 17. | MSG 17 | [tên người dùng] đã yêu cầu nhận sản phẩm [tên sản phẩm] |
| 18. | MSG 18 | [Tên người dùng] đã chấp nhận yêu cầu nhận sản phẩm của bạn |
| 19. | MSG 19 | [Tên người dùng] đã từ chối yêu cầu nhận sản phẩm của bạn |
| 20. | MSG 20 | Nhóm quyền này đã tồn tại |
| 21. | MSG 21 | Đã xảy ra lỗi. Vui lòng thử lại sau |
| 22. | MSG 22 | Mật khẩu mới không được trùng với mật khẩu hiện tại. |

## 6.2  Email template

ET 1: Confirmation template

| Gửi đến | <email người dùng> |
|---------|--------------------|
| Subject | Xác nhận địa chỉ email của bạn |
| Body    | Dear <email>, <br> Để xác thực địa chỉ email đã đăng ký vui lòng ấn **vào đây.** <br> <Chữ ký> |

ET 2: forgot password template.

| Gửi đến | <email người dùng> |
|---------|--------------------|
| Subject | Khôi phục mật khẩu |
| Body | Dear <email>,<br><br>Vui lòng nhấn **vào đây** để khôi phục tài khoản của bạn.<br><br><Chữ ký> |

ET 3:  approve Instructor template.

| Gửi đến | <email người dùng> |
|---------|--------------------|
| Subject | Chúc mừng bạn đã trở thành giáo viên tại Cursus. |
| Body | Dear <email>,<br><br>Chúc mừng bạn đã trở thành giáo viên tại Cursus. Hãy truy cập vào hệ thống để tiến hành cung cấp, phát triển khóa học của bạn.<br><br><Chữ ký> |

ET 4:  reject Instructor template.

| Gửi đến | <email người dùng> |
|---------|--------------------|
| Subject | Cảm ơn bạn đã chọn Cursus. |
| Body | Dear <email>,<br>Cursus cảm ơn bạn đã chọn, chúng tôi đã xem xét và đánh giá bạn chưa phù hợp cho những sản phẩm lần này.<br>Chúng tôi sẽ hy vọng sẽ làm việc với bạn trong tương lai sắp tới.<br><br><Chữ ký> |

ET 5:  Payout result

| Gửi đến | <email người dùng> |
|---------|--------------------|
| Subject | Bạn đã rút tiền thành công |
| Body | Dear <email>, |

Bạn đã rút {Tiền} thành công.

<Chữ ký>