

**SOFTWARE REQUIREMENT SPECIFICATION**

**<<XÂY DỰNG ỨNG DỤNG QUẢN LÝ ĐÀO TẠO NỘI BỘ TẠI CÔNG TY VIVAS**

**– Project the Training Department at Vivas>>**

Version 1.0 approved

**Prepared by :**   
- Nguyễn Thế Hoàng Tùng  
- Dương Mạnh Tiến

- Nguyễn Anh Đức

-Nguyễn Hữu An Nguyên

– Hanoi, June 2025 –

**Table of Contents**

Version 1.0 approved. 1

I. Record of Changes. 3

II. Software Requirement Specification. 4

1. Requirement Overview.. 4

1.1 Context Diagram.. 4

1.2 User Requirements. 5

1.3 System Functionalities. 7

2. Functional Specifications. 9

2.1 <<Feature/Workflow Name>>. 9

2.2 Common Functions. 12

2.3 Patron Feature. 13

3. UI Requirements. 16

3.1 <<Feature Name>>. 16

4. Non-Functional Requirements. 18

4.1 External Interfaces. 18

4.2 Quality Attributes. 18

5. Requirement Appendix. 18

5.2 Common Requirements. 18

5.3 Application Messages List 18

5.4 Other Requirements….. 19

# **I. Record of Changes**

| Date | A\*  M, D | In charge | Change Description |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

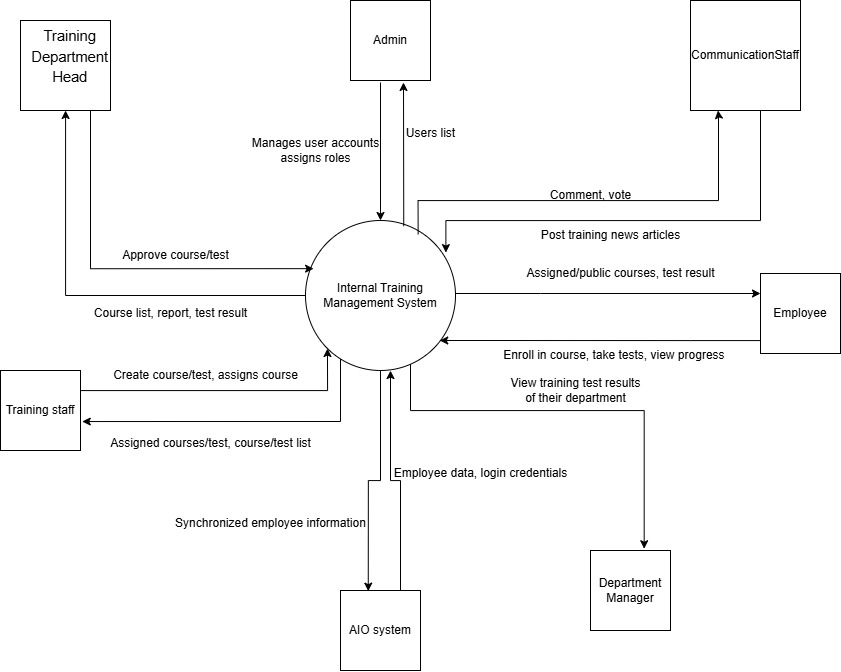
\*A - Added M - Modified D - Deleted

# **II. Software Requirement Specification**

## **1. Requirement Overview**

### **1.1 Context Diagram**

The Internal Training Management System (ITMS) is a new software system designed to support the Training Department at Vivas, a software company with 150 employees across 7 departments/branches. The system aims to streamline the creation, management, and tracking of internal training programs, courses, tests, and related communications. It replaces manual processes for managing training activities and integrates with the existing All-in-One (AIO) system to retrieve employee information, including login credentials. The context diagram below illustrates the external entities and system interfaces for release 1.0.

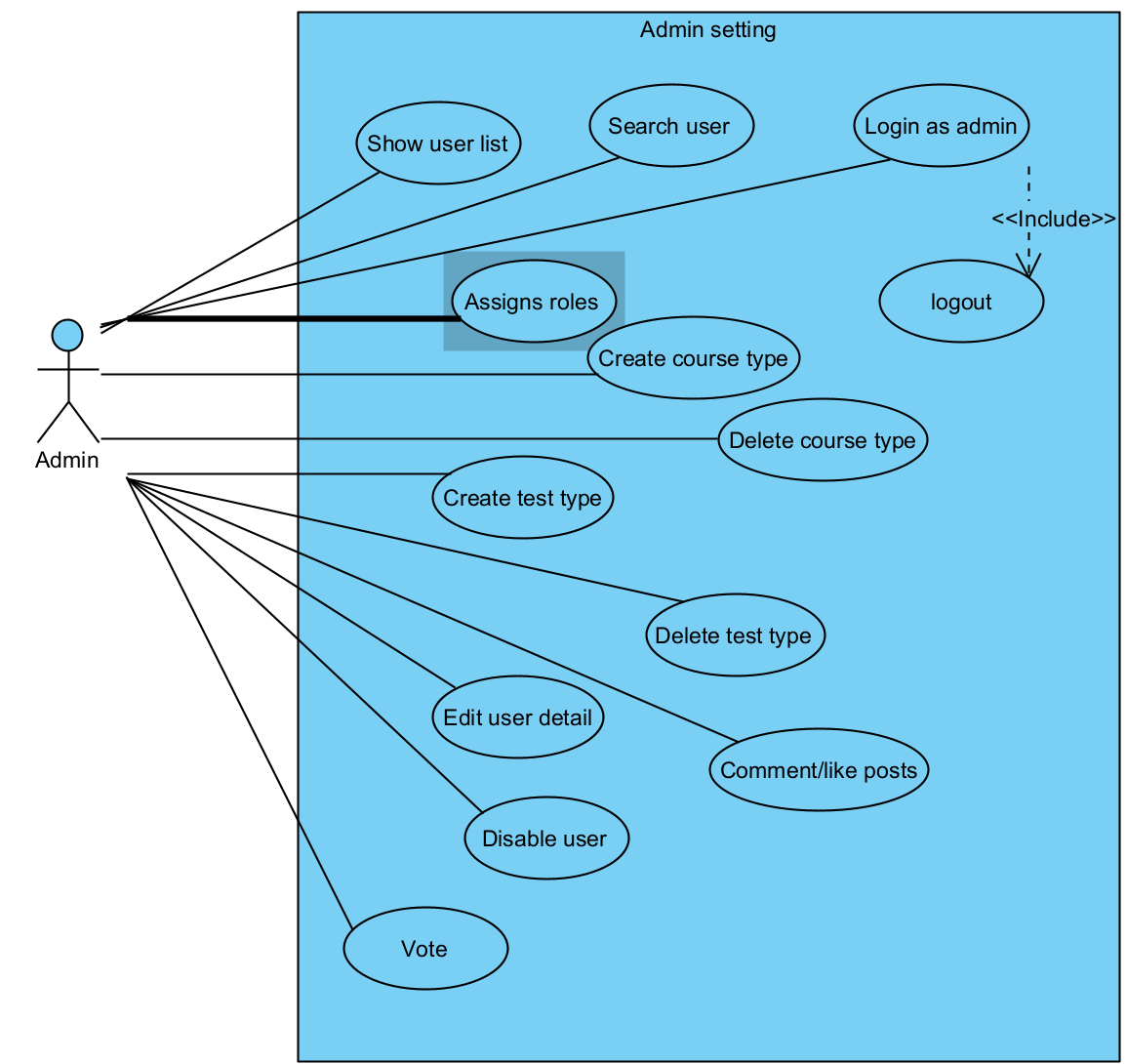


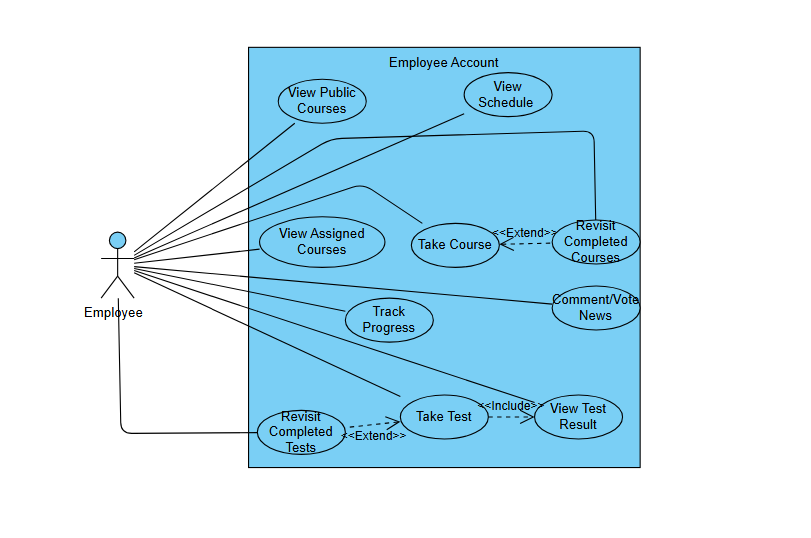
### **1.2 User Requirements**

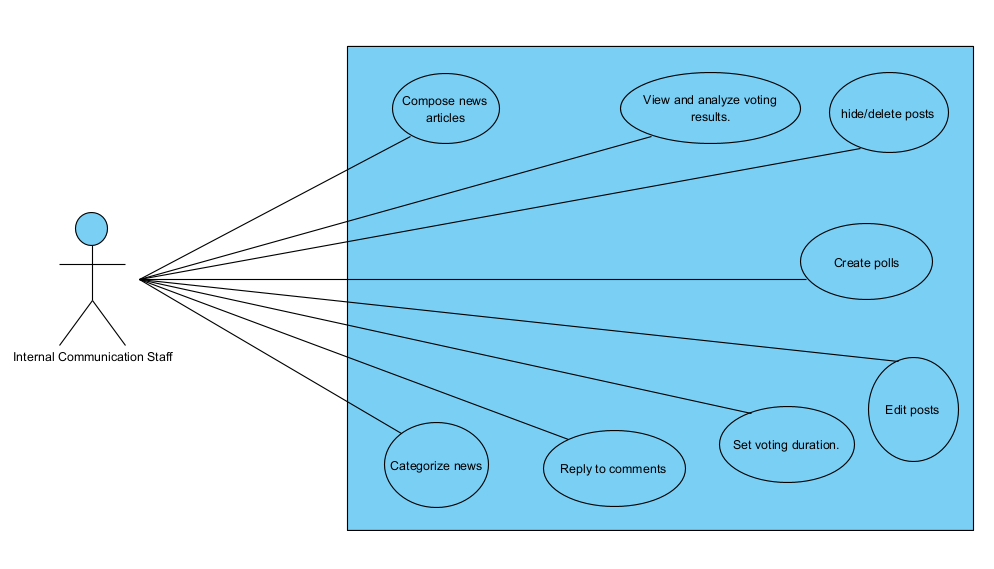
#### **1.2.1 Actors**

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | Administrator | Manages user accounts, assigns permissions, and configures system settings (e.g., course types, test types). |
| 2 | Training Department Head | Oversees training activities, approves courses and tests, and views training reports and employee progress. |
| 3 | Training Staff | Creates and assigns courses and tests to employees or departments. |
| 4 | Department Manager | Monitors training progress and test results for employees in their department. |
| 5 | Employee | Participates in assigned and public courses, takes tests, and tracks personal progress. |
| 6 | Internal Communication Staff | Posts and manages training-related news, including comments and votes. |
| 7 | AIO System | External system providing employee information and authentication credentials. |

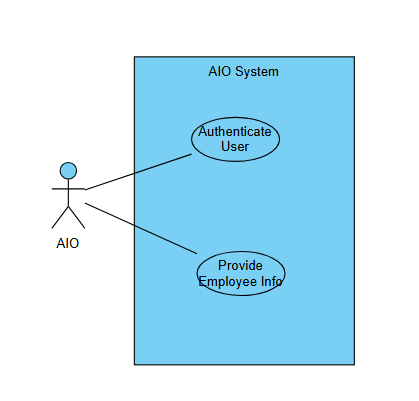
#### **1.2.2 Use Cases**

**

**

**

**

**

| **ID** | **Feature** | **Use Case** | **Description** |
| --- | --- | --- | --- |
| 01 | Authentication | Login | Login with correct account |
| 02 | Authentication | Logout | logout |
| 03 | Admin setting | Show User List | Administrator can see all user |
| 04 | Admin setting | Search User | Administrator can find user account with name, id, email or role |
| 05 | Admin setting | Assign Roles | Administrator can assign roles to user account |
| 06 | Admin setting | Create Course Type | Administrator can create course type with different type of training for training department to add |
| 07 | Admin setting | Delete Course Type | Administrator can delete course type |
| 08 | Admin setting | Create Test Type | Administrator can create test type (like quiz, FE, PE) |
| 09 | Admin setting | Delete Test Type | Administrator can remove |
| 10 | Admin setting | Edit User Detail | Administrator can edit user account like ID, email or department |
| 11 | Admin setting | Disable User | Admin can disable user account |
| 12 | Employee Account | View Assigned Courses | Employee views the list of courses assigned by the training department. |
| 13 | Employee Account | View Public Courses | Employee accesses public training courses available to all employees. |
| 14 | Employee Account | View Schedule | Employee views the schedule for courses, exams, or training events. |
| 15 | Employee Account | Take Course | Employee begins participating in a training course. |
| 16 | Employee Account | Revisit Completed Courses | Employee revisits content from previously completed courses. |
| 17 | Employee Account | Track Progress | Employee tracks progress of completed courses and tests. |
| 18 | Employee Account | Take Test | Employee takes a test assigned directly or linked to a course. |
| 19 | Employee Account | View Test Result | After submitting a test, the system displays the result immediately. |
| 20 | Employee Account | Revisit Completed Tests | Employee may review submitted tests (without seeing detailed answers). |
| 21 | Employee Account | Comment/Vote News | Employee comments or votes on internal training-related news posts. |
| 22 | Internal Communication Staff | Compose news articles | Create training-related news posts with formatted text, images, and file attachments. |
| 23 | Internal Communication Staff | View and analyze voting results. | Review poll outcomes with vote counts, percentages, and summary statistics. |
| 24 | Internal Communication Staff | hide/delete posts | Temporarily hide or permanently remove posts that are outdated or inappropriate. |
| 25 | Internal Communication Staff | Create polls | Add polls to news articles with customizable voting options (single/multiple/reactions). |
| 26 | Internal Communication Staff | Edit posts | Modify the content, title, attachments, or category of previously published news posts |
| 27 | Internal Communication Staff | Set voting duration | Define the start and end time for each poll to control participation periods. |
| 28 | Internal Communication Staff | Reply to comments | Respond to user comments under posts to clarify information or engage with readers. |
| 29 | Internal Communication Staff | Categorize news | Assign categories (e.g., announcements, events, motivation) to help users filter content. |
| 30 | Course Management | delete course | The Training Department Head can remove a course from the system if it is no longer relevant. |
| 31 | Course Management | View all course | The Training Department Head can view a comprehensive list of all courses |
| 32 | Course Management | Create course |  |
| 33 | Test Management | Add Question | The Training Department Head or Training Staff can add questions to a course or test, which will be included in the question list |
| 34 | Test Management | Create Test | The Training Department Head or Training Staff can create a new test by including a list of questions. |
| 35 | Test Management | Create Question List | The Training Department Head or Training Staff can create a list of questions to be used in tests or courses. |
| 36 | Assignment Management | Assign Course | The Training Department Head or Training Staff can assign a course to specific employees or departments. |
| 37 | Course Management | Browse Courses | The Training Department Head can view and browse all available courses to monitor or plan training activities. |
| 38 | Assignment Management | Assign Test | The Training Department Head or Training Staff can assign a test to specific employees or departments. |
| 39 | Employee Management | Assign Test | The Training Department Head or Training Staff can assign a test to specific employees or departments. |
| 40 | Employee Management | Get Employee List | The Training Department Head or Training Staff can retrieve a list of employees to assign courses or tests. |
| 41 | Test Management | View All Tests | The Training Department Head can view a comprehensive list of all tests in the system. |
| 42 | Test Management | Delete Test | The Training Department Head can remove a test from the system if it is no longer needed. |
| 43 | Test Management | Browse Tests | The Training Department Head can browse all available tests to monitor or manage testing activities. |
| 44 | Reporting | View Learning Report | The Training Department Head can view a report on employees’ learning progress for specific courses. |
| 45 | Reporting | View Statistical Report | The Training Department Head can view statistical reports on overall training and testing performance across the company. |
| 46 | AIO system | Authenticate User | AIO verifies user credentials during login to the training system (SSO or API). |
| 47 | AIO system | Provide Employee Info | |  | | --- |  | AIO shares employee profile data (e.g., name, department, email) via integration. | | --- | |

*1.2.3 Business rules*

| **ID** | **Category** | **Rule Definition** |
| --- | --- | --- |
| **BR-01** | Constraints | Only the Head of the Training Department has the authority to approve courses and tests before they are available to employees. |
| **BR-02** | Constraints | The Admin is responsible for managing user accounts, assigning roles, and configuring the system. |
| **BR-03** | Constraints | Training staff can only create courses or tests and assign them to employees within the system. |
| **BR-04** | Constraints | Department Managers can only view training progress and test results of employees under their supervision. |
| **BR-05** | Constraints | Communication Staff are only allowed to post training news and interact via comment/vote features. |
| **BR-06** | Constraints | Employees can only enroll in assigned or public courses and view their own training progress and test results. |
| **BR-07** | Constraints | All employee information, including login credentials, must be synchronized from the AIO system and cannot be edited directly in the training system. |
| **BR-08** | Facts | A course is considered completed only when the employee finishes all lessons and passes the final test (if applicable). |
| **BR-09** | Facts | An employee’s course progress and test results are automatically recorded in the system after completion and are viewable only by the employee and their department manager. |

1.2.4 Business flow

- Tạo và duyệt khóa học

- Tạo và duyệt bài kiểm tra

- Đồng bộ dữ liệu giữa hai hệ thông

1.2.5 State machine diagram

*- Vòng đời trạng thái của khóa học*

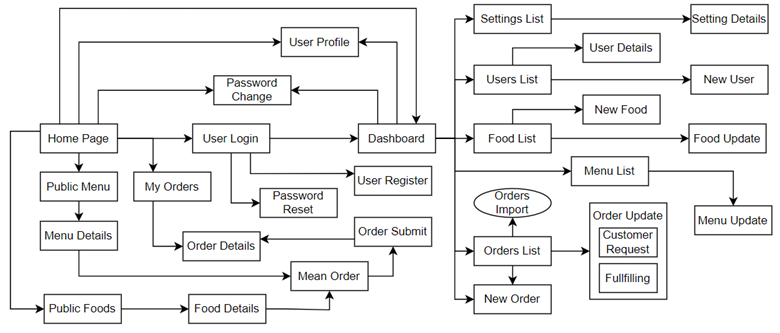
*- Vòng đời trạng thái của bài test*

### **1.3 System Functionalities**

*[Provide functionality overview of software system: screen flow, screen descriptions, system user roles, screen authorization, non-screen functions, ERD]*

#### **1.3.1 Screens 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. Please note that beside the normal flat screen, we might have the oval notation for pop-up screen (Orders Import) or a screen with multiple information tab (Order Update), etc. You may also use text or background format for different visuality purpose]*

**

#### **1.3.2 Screen Descriptions**

*[Provide the descriptions for the screens in the Screens Flow above]*

| **#** | **Feature** | **Screen** | **Description** |
| --- | --- | --- | --- |
| 1 | Order Meals | Create Order | <<Screen Brief description>> |
| 2 | Order Meals | Change Order |  |
| 3 | .. |  |  |

#### **1.3.3 Screen Authorization (not require)**

*[Provide the system roles authorization to the system features (down to screens, and event to the screen activities if applicable) in the table form as below – replace Role1, Role2,… with your specific system user role names]*

| **Screen** | **Role-Name1** | **Role-Name2** | **Role-Name3** | **…** |
| --- | --- | --- | --- | --- |
| <<Screen Name1>> | X |  | X | X |
| <<Screen Activity>> |  |  | X | X |
| <<Screen Name2>> | X |  | X |  |
| Query All Data | X |  |  |  |
| Query Own Data |  |  | X |  |
| Query Managed Data |  |  | X |  |
| Add New Data |  |  | X | X |
| Update All Data |  |  |  | X |
| Update Own Data |  |  |  | X |
| Update Managed Data |  |  |  | X |
| Delete Data |  |  |  |  |
| … |  |  |  |  |

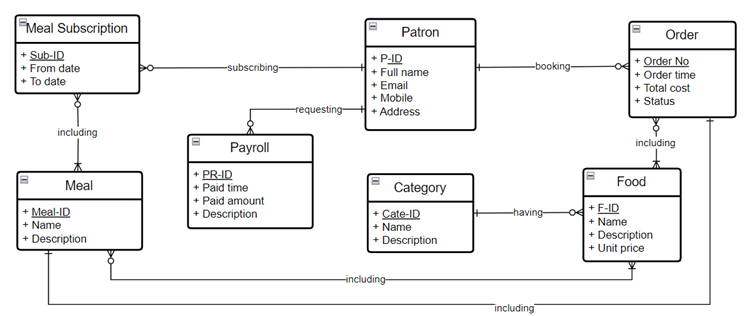
#### **1.3.4 Non-UI Functions (not require)**

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | <<Feature Name>> | <<Function Name1>> | <<Function Name1 Description>> |
| 2 | … |  |  |

#### **1.3.5 Entity Relationship Diagram**

*[Provide the entity relationship diagram and the entity descriptions in the table format as below]*

**

**Entities Description**

| **#** | **Entity** | **Description** |
| --- | --- | --- |
| 1 | Patrol | Cafeteria’s customer information |
| 2 | Meal | … |
| 3 | Meal Subscription |  |
| 4 | … |  |

## **2. Functional Specifications**

*[Provide the functional specification of the use cases (grouped by the features or workflows), in which you need to provide the use case specifications and the business rules in the form of table as guided]*

### **2.1 <<Feature/Workflow Name>>**

#### **2.1.1 <<UseCaseCode\_UC Name>>**

Provide the functional description for the use cases using the template/guides below

**Mockup**

**UI Requirement**

**Functional Description Template**

| UC ID and Name: |  | | |
| --- | --- | --- | --- |
| Created By: |  | Date Created: |  |
| Primary Actor: |  | Secondary Actors: |  |
| Trigger: |  | | |
| Description: |  | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Normal Flow: |  | | |
| Alternative Flows: |  | | |
| Exceptions: |  | | |
| Priority: | High (Medium, Low), Must Have (Should Have, Could Have),.. | | |
| Frequency of Use: |  | | |
| Business Rules: |  | | |
| Other Information: |  | | |
| Assumptions: |  | | |

**Functional Description Contents**

***Use Case ID and Name***

Give each use case a unique integer sequence number identifier. State a concise name for the use case that indicates the value the use case would provide to some user. Begin with an action verb, followed by an object.

***Author and Date Created***

Enter the name of the person who initially wrote this use case and the date it was written.

***Primary and Secondary Actors***

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the primary actor that will be initiating this use case and any other secondary actors who will participate in completing execution of the use case.

***Trigger***

Identify the business event, system event, or user action that initiates the use case. This trigger alerts the system that it should begin testing the preconditions for the use case so it can judge whether to proceed with execution.

***Description***

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

***Preconditions***

List any activities that must take place, or any conditions that must be true, before the use case can be started. The system must be able to test each precondition. Number each precondition. Example: PRE-1: User’s identity has been authenticated.

***Postconditions***

Describe the state of the system at the successful conclusion of the use case execution. Label each postcondition in the form POST-X, where X is a sequence number. Example: POST-1: Price of item in the database has been updated with the new value.

***Normal Flow***

Provide a description of the user actions and corresponding system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. Show a numbered list of actions performed by the actor, alternating with responses provided by the system. The normal flow is numbered “X.0”, where “X” is the Use Case ID.

***Alternative Flows***

Document other successful usage scenarios that can take place within this use case. State the alternative flow, and describe any differences in the sequence of steps that take place. Number each alternative flow in the form “X.Y”, where “X” is the Use Case ID and Y is a sequence number for the alternative flow. For example, “5.3” would indicate the third alternative flow for use case number 5. Indicate where each alternative flow would branch off from the normal flow, and if pertinent, where it would rejoin the normal flow.

***Exceptions***

Describe any anticipated error conditions that could occur during execution of the use case and how the system is to respond to those conditions. Number each alternative flow in the form “X.Y.EZ”, where “X” is the Use Case ID, Y indicates the normal (0) or alternative (>0) flow during which this exception could take place, “E” indicates an exception, and “Z” is a sequence number for the exceptions. For example “5.0.E2” would indicate the second exception for the normal flow for use case number 5. Indicate where in the normal (or an alternative) flow each exception could occur.

***Priority***

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. Use the same priority scheme as that used for the functional requirements.

***Frequency of Use***

Estimate the number of times this use case will be performed per some appropriate unit of time. This gives an early indicator of throughput, concurrent usage loads, and transaction capacity.

***Business Rules***

List any business rules that influence this use case. Don’t include the business rule text here, just its identifier so the reader can find it in another repository when needed.

***Other Information***

Identify any additional requirements, such as quality attributes, for the use case that may need to be addressed during design or implementation. Also list any associated functional requirements that aren’t a direct part of the use case flows but which a developer needs to know about. Describe what should happen if the use case execution fails for some unanticipated or systemic reason (e.g., loss of network connectivity, timeout). If the use case results in a durable state change in a database or the outside world, state whether the change is rolled back, completed correctly, partially completed with a known state, or left in an undetermined state as a result of the exception.

***Assumptions***

List any assumptions that were made regarding this use case or how it might execute.

### **2.2 Common Functions**

#### **2.2.1 UC-2\_Login System**

**Mockup**

**A screenshot of a login form

Description automatically generated**

**UI Requirements**

| **Field Name** | **Field Type** | **Description** |
| --- | --- | --- |
| Email\* | Text Box | This is for user to input valid email address for logging in |
| Password\* | Password Box | This is for user to input password for logging in |
| Login | Button | User clicks to authenticate him/herself into the system with provided email & password |
| Register | Button | User clicks to redirect to the User Register page for registering new user account to access the system |
| Forgot Password? | Hyperlink | User clicks to redirect to the Password Reset page for resetting his/her forgot password |
| Login with Google | Hyperlink | Allow user to login with his/her Google account |
| Login with Facebook | Hyperlink | Allow user to login with his/her Facebook account |

**UC detail**

| UC ID and Name: | **UC-2\_Login System** | | |
| --- | --- | --- | --- |
| Created By: | MinhNNT | Date Created: | 16/Jun/2023 |
| Primary Actor: | Customer | Secondary Actors: | None |
| Trigger: | User clicks Login button from the page header, or  User accesses an authenticated feature (from a link or type the page URL directly into the address bar) | | |
| Description: | *As a user, I want to be able to log into the system so that I can use the system’s authenticated features and access my personalized account.* | | |
| Preconditions: | User account has been created & authorized | | |
| Postconditions: | · User logs in the system successfully  · The system tracked successful login into the Activity Log | | |
| Normal Flow | **2.0 Login System**  1. User accesses the User Login screen  2. User types in the login details or choo other login options (see 2.1 and 2.2)  3. User clicks the Login button  4. System validates the login details (see 2.0.E1)  5. System allows user to access  6. System tracks user’s success login to the Activity Log  7. System accesses the Home Page (or the previous calling page if any) | | |
| Alternative Flows: | ***2.1 Google Login***  1. User chooses to login system using Google account  2. System redirects the user to the Google’s Login screen  3. User types in the Google account details and chooses to login  4. Google validates user’s login information successfully and redirect him/her back to the system  5. Return to step 5 of normal flow.    ***2.2 Facebook Login***  1. User chooses to login system using Facebook account  2. System redirects the user to the Facebook’s Login screen  3. User types in the Facebook account details and chooses to login  4. Facebook validates user’s login information successfully and redirect him/her back to the system  5. Return to step 5 of normal flow. | | |
| Exceptions: | ***2.0.E1 System can’t authenticate the user***  1. The Error Message screen is shown to the user  2. User cancels the logging in *=> UC stops, change to UC-1\_View Home Page*  3. User clicks “Forgot Password?” link *=> change to UC-3\_Reset Password*  4. User clicks “Register” link *=> change to UC-4\_Register User Account* | | |
| Priority: | Must Have | | |
| Frequency of Use: |  | | |
| Business Rules: | FR1, FR2, FR3 | | |
| Other Information: |  | | |
| Assumptions: |  | | |

#### **2.2.2 UC-3\_Abc Xyz**

…

### **2.3 Patron Feature**

#### **2.3.1 UC-5\_Order a Meal**

Mockup

UI Requirements

UC detail

| ID and Name: | **UC-5 Order a Meal** | | |
| --- | --- | --- | --- |
| Created By: | Prithvi Raj | Date Created: | 10/4/13 |
| Primary Actor: | Patron | Secondary Actors: | Cafeteria Inventory System |
| Description: | A Patron accesses the Cafeteria Ordering System from the corporate intranet or from home, views the menu for a specific date if desired, selects food items, and places an order for a meal to be delivered to a specified location within a specified 15-minute time window. | | |
| Trigger: | A Patron indicates that he wants to order a meal | | |
| Preconditions: | PRE-1. Patron is logged into COS.  PRE-2. Patron is registered for meal payments by payroll deduction. | | |
| Postconditions: | POST-1. Meal order is stored in COS with a status of “Accepted”.  POST-2. Inventory of available food items is updated to reflect items in this order.  POST-3. Remaining delivery capacity for the requested time window is updated. | | |
| Normal Flow: | **5.0 Order a Single Meal**  1. Patron asks to view menu for a specific date. (see 5.0.E1, 5.0.E2)  2. COS displays menu of available food items and the daily special.  3. Patron selects one or more food items from menu. (see 5.1)  4. Patron indicates that meal order is complete. (see 5.2)  5. COS displays ordered menu items, individual prices, and total price, including taxes and delivery charge.  6. Patron either confirms meal order (continue normal flow) or requests to modify meal order (return to step 2).  7. COS displays available delivery times for the delivery date.  8. Patron selects a delivery time and specifies the delivery location.  9. Patron specifies payment method.  10. COS confirms acceptance of the order.  11. COS sends Patron an email message confirming order details, price, and delivery instructions.  12. COS stores order, sends food item information to Cafeteria Inventory System, and updates available delivery times. | | |
| Alternative Flows: | **5.1 Order multiple identical meals**  1. Patron requests a specified number of identical meals. (see 5.1.E1)  2. Return to step 4 of normal flow.  **5.2 Order multiple meals**  1. Patron asks to order another meal.  2. Return to step 1 of normal flow. | | |
| Exceptions: | **5.0.E1 Requested date is today and current time is after today’s order cutoff time**  1. COS informs Patron that it’s too late to place an order for today.  2a. If Patron cancels the meal ordering process, then COS terminates use case.  2b. Else if Patron requests another date, then COS restarts use case.  **5.0.E2 No delivery times left**  1. COS informs Patron that no delivery times are available for the meal date.  2a. If Patron cancels the meal ordering process, then COS terminates use case.  2b. Else if Patron requests to pick the order up at the cafeteria, then continue with normal flow, but skip steps 7 and 8.  **5.1.E1 Insufficient inventory to fulfill multiple meal order**  1. COS informs Patron of the maximum number of identical meals he can order, based on current available inventory.  2a. If Patron modifies number of meals ordered, then Return to step 4 of normal flow.  2b. Else if Patron cancels the meal ordering process, then COS terminates use case. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 300 users, average of one usage per day. Peak usage load for this use case is between 9:00 A.M. and 10:00 A.M. local time. | | |
| Business Rules: | BR-1, BR-2, BR-3, BR-4, BR-11, BR-12, BR-33 | | |
| Other Information: | 1. Patron shall be able to cancel the meal ordering process at any time prior to confirming it.  2. Patron shall be able to view all meals he ordered within the previous six months and repeat one of those meals as the new order, provided that all food items are available on the menu for the requested delivery date. (Priority = M)  3. The default date is the current date if the Patron is using the system before today’s order cutoff time. Otherwise, the default date is the next day that the cafeteria is open. | | |
| Assumptions: | Assume that 15 percent of Patrons will order the daily special (source: previous 6 months of cafeteria data). | | |

#### **2.3.2 UC-6\_Register for Payroll Deduction**

| ID and Name: | **UC-6 Register for Payroll Deduction** | | |
| --- | --- | --- | --- |
| Created By: | Nancy Anderson | Date Created: | 9/15/13 |
| Primary Actor: | Patron | Secondary Actors: | Payroll System |
| Description: | Cafeteria patrons who use the COS and have meals delivered must be registered for payroll deduction. For noncash purchases made through the COS, the cafeteria will issue a payment request to the Payroll System, which will deduct the meal costs from the next scheduled employee payday direct deposit. | | |
| Trigger: | Patron requests to register for payroll deduction, or Patron says yes when COS asks if he wants to register | | |
| Preconditions: | PRE-1. Patron is logged into COS. | | |
| Postconditions: | POST-2. Patron is registered for payroll deduction. | | |
| Normal Flow: | **6.0 Register for Payroll Deduction**  1. COS asks Payroll System if Patron is eligible to register for payroll deduction.  2. Payroll System confirms that Patron is eligible to register for payroll deduction.  3. COS asks Patron to confirm his desire to register for payroll deduction.  4. If so, COS asks Payroll System to establish payroll deduction for Patron.  5. Payroll System confirms that payroll deduction is established.  6. COS informs Patron that payroll deduction is established. | | |
| Alternative Flows: | None | | |
| Exceptions: | 6.0.E1 Patron is not eligible for payroll deduction  6.0.E2 Patron is already enrolled for payroll deduction | | |
| Priority: | High | | |
| Business Rules: | BR-86 and BR-88 govern an employee’s eligibility to enroll for payroll deduction. | | |
| Other Information: | Expect high frequency of executing this use case within first 2 weeks after system is released. | | |

## **4. Non-Functional Requirements**

### **4.1 External Interfaces**

*[This section provides information to ensure that the system will communicate properly with users and with external hardware or software/system elements.]*

### **4.2 Quality Attributes**

*[List all the required system characteristics (quality attributes) specification. Some of the possible attributes are provided with the guide/descriptions are mentioned here]*

#### **4.2.1 Usability**

*[This section includes all those requirements that affect usability. For example, specify the required training time for a normal users and a power user to become productive at particular operations specify measurable task times for typical tasks or base the new system’s usability requirements on other systems that the users know and like specify requirement to conform to common usability standards, such as IBM’s CUA standards Microsoft’s GUI standards]*

#### **4.2.2 Performance**

*[The system’s performance characteristics are outlined in this section. Include specific response times. Where applicable, reference related Use Cases by name.*

*Response time for a transaction (average, maximum)*

*Throughput, for example, transactions per second*

*Capacity, for example, the number of customers or transactions the system can accommodate*

*Resource utilization, such as memory, disk, communications, and so forth.]*

#### **4.2.3 …**

## **5. Requirement Appendix**

*[Provide common requirements, or other extra requirements information here]*

### **5.2 Common Requirements**

*[Fill all the common requirements here..]*

### **5.3 Application Messages List**

| **#** | **Message code** | **Message Type** | **Context** | **Content** |
| --- | --- | --- | --- | --- |
| 1 | MSG01 | In line | There is not any search result | *No search results.* |
| 2 | MSG02 | In red, under the text box | Input-required fields are empty | *The \* field is required.* |
| 3 | MSG03 | Toast message | Updating asset(s) information successfully | *Update asset(s) successfully.* |
| 4 | MSG04 | Toast message | Adding new asset successfully | *Add asset successfully.* |
| 5 | MSG05 | Toast message | Confirming email of asset hand-over is sent successfully | *A confirmation email has been sent to {email\_address}.* |
| 6 | MSG06 | Toast message | Resetting asset information successfully | *Return asset(s) successfully.* |
| 7 | MSG07 | Toast message | Deleting asset information successfully | *Delete asset(s) successfully.* |
| 8 | MSG08 | In red, under the text box | Input value length > max length | *Exceed max length of {max\_length}.* |
| 9 | MSG09 | In line | Username or password is not correct when clicking sign-in | *Incorrrect user name or password. Please check again.* |

### **5.4 Other Requirements…**