

VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY

UNIVERSITY OF SCIENCE

Faculty of Information Technology



CHAT BOT - PA2

USE CASES SPECIFICATION

by

Phan Thế Anh 22127021

Trần Quốc Huy 22127165

Trần Gia Khang 22127181

Võ Anh Quân 22127352

Nguyễn Hoàng Sang 22127361

Phan Văn Tài 22127372

Under the supervision of

Teacher: **Nguyễn Minh Huy**

Assistant Teacher: **Phạm Hoàng Hải**

Assistant Teacher: **Ngô Ngọc Đăng Khoa**

NOVEMBER 21th 2024

Group 02

**<Chat Bot Application>
Use-Case Specification**

Version <1.0>

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Revision History

Date	Version	Description	Author
<dd/mm/yy>	<x.x>	<details>	<name>

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Table of Contents

Revision History	3
1. Introduction	6
2. Use Case Model	6
2.1. Authentication and Authorization:	6
2.2. Chat with AI Agents:	7
2.3. Create and manage AI BOT:	8
2.4. Knowledge Data Management:	9
2.5. Prompt Management:	10
2.6. Account Management:	11
2.7. Photo Chating:	12
2.8. Email with AI Agents:	12
3. Use-case Specifications	13
3.1. Use-case: Authentication and Authorization:	13
3.1.1. <i>Register an account:</i>	13
3.1.2. <i>Login:</i>	14
3.1.3. <i>Recover Password:</i>	14
3.2. Use-case: Chat with AI Agents:	15
3.2.1. <i>Create New Thread Chat:</i>	15
3.2.2. <i>Chat with AI Agent:</i>	16
3.2.3. <i>History Browsing:</i>	16
3.3. Use-case: Create and manage AI BOT	17
3.3.1. <i>Create AI BOT:</i>	17
3.3.2. <i>Display/Find AI BOT:</i>	17
3.3.3. <i>Update/Delete AI BOT:</i>	18
3.3.4. <i>Update prompt for AI Bot:</i>	19
3.3.5. <i>Communicate with created AI BOT through Chat widget:</i>	19
3.3.6. <i>Add/Remove knowledge data:</i>	20
3.3.7. <i>Preview and Chat with AI BOT:</i>	21
3.3.8. <i>Publish AI Chat to Slack, Telegram, Messenger:</i>	21
3.4. Use-case: Knowledge Data Management	22
3.4.1. <i>Manage Data:</i>	22
3.4.2. <i>Disable/delete data:</i>	23
3.5. Use-case: Prompt Management	23
3.5.1. <i>Create a private prompt:</i>	23
3.5.2. <i>Filter prompts by category:</i>	24
3.5.3. <i>Update and delete private prompt:</i>	24
3.6. Use-case: Account Management	24

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.6.1 Upgrade Account to Pro	24
2.6.2 Monetization	26
3.7. Use-case: Photo Chating	26
3.7.1. Update picture to chat, Chat with screenshot:	26
3.7.2. Take picture and chat with taken picture	27
3.8. Use-case: Email with AI Model	29

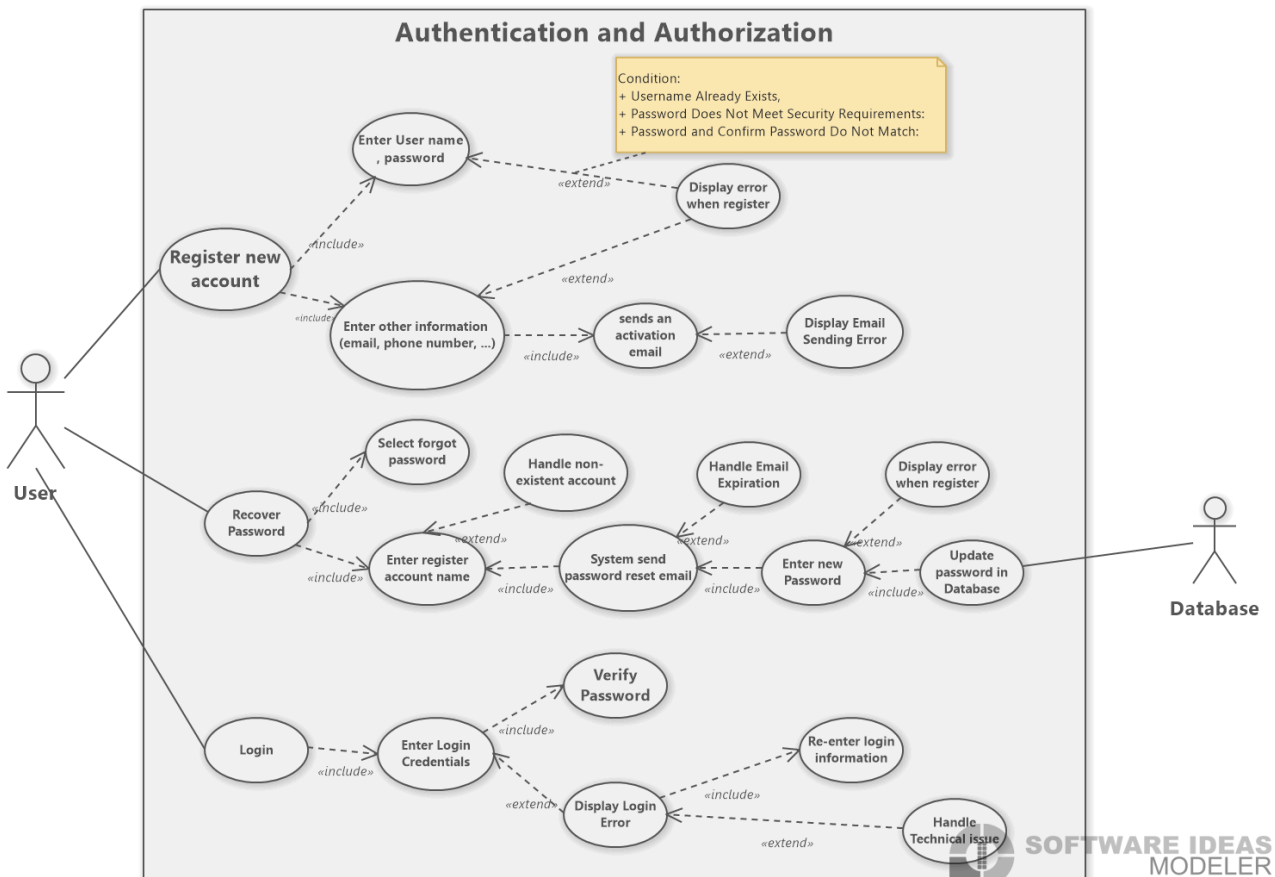
Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

1. Introduction

- This document outlines the use cases of the system through use-case diagrams, which depict the interactions between actors and use cases, along with the relationships between them.
- Document also includes use case specification for each use case.
- The provided template by the teacher will guide the completion of documents.
- **Drawing Tool:** Software Ideas Modeler.

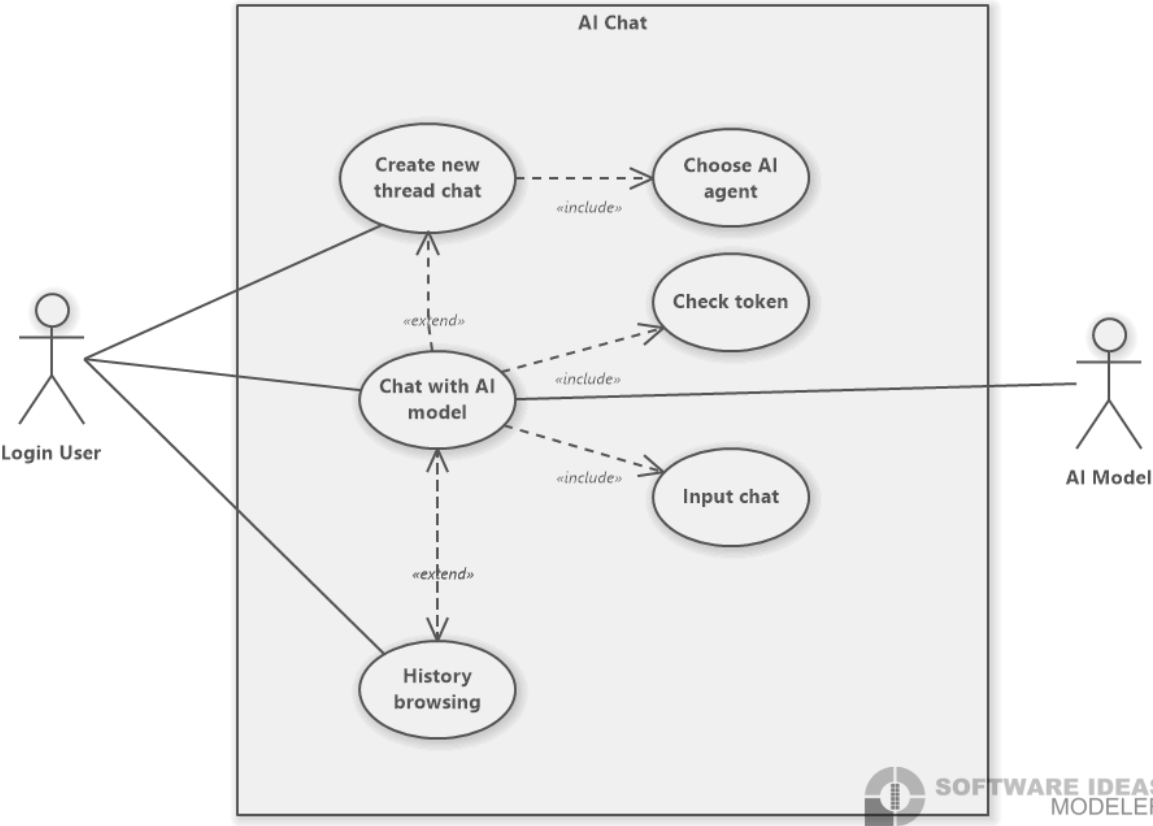
2. Use Case Model

2.1. Authentication and Authorization:



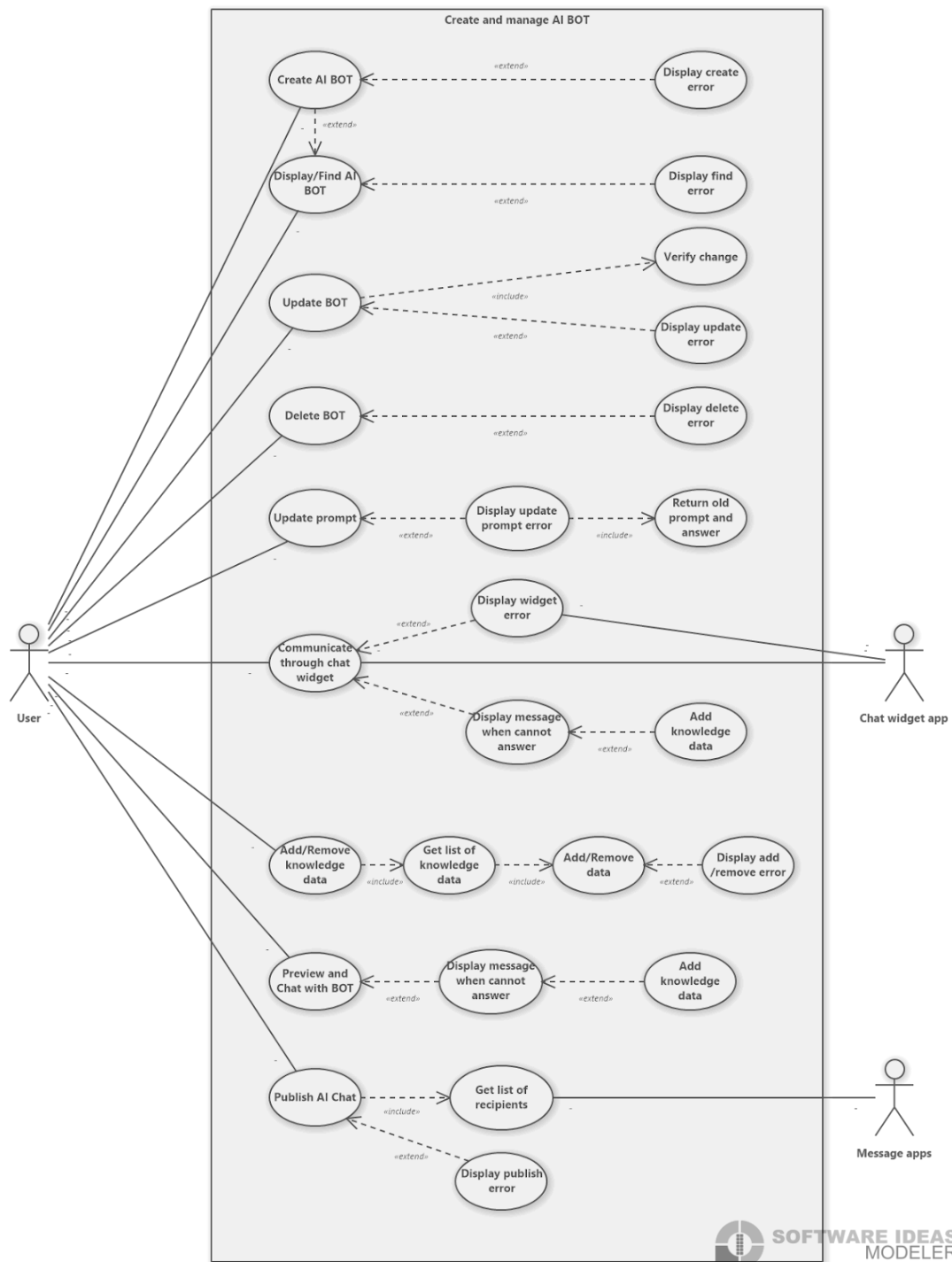
Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.2. Chat with AI Agents:



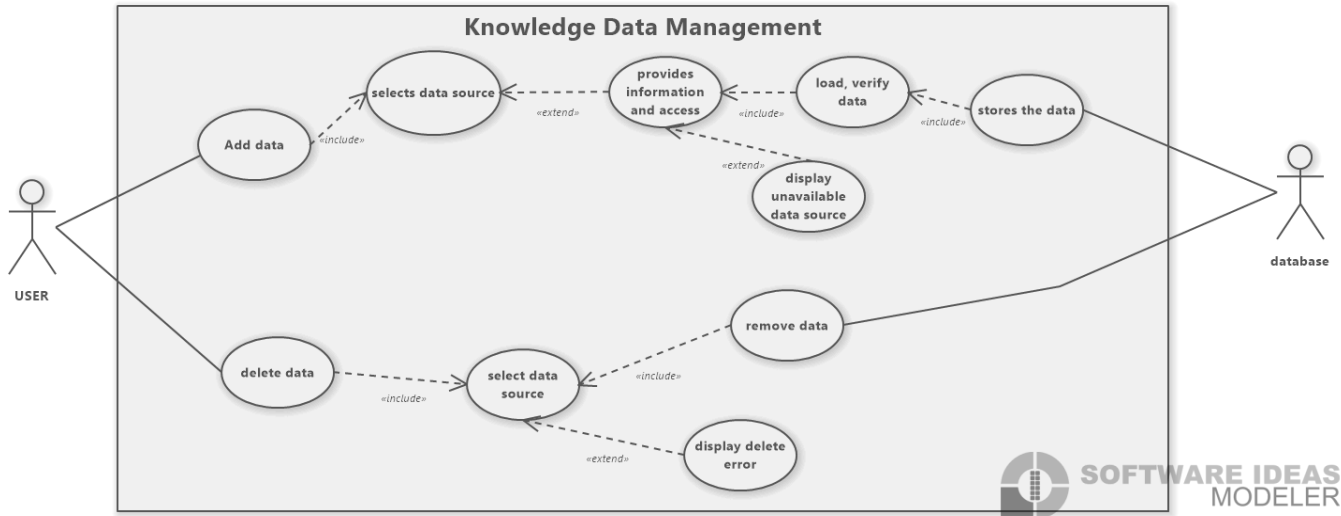
Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.3. Create and manage AI BOT:



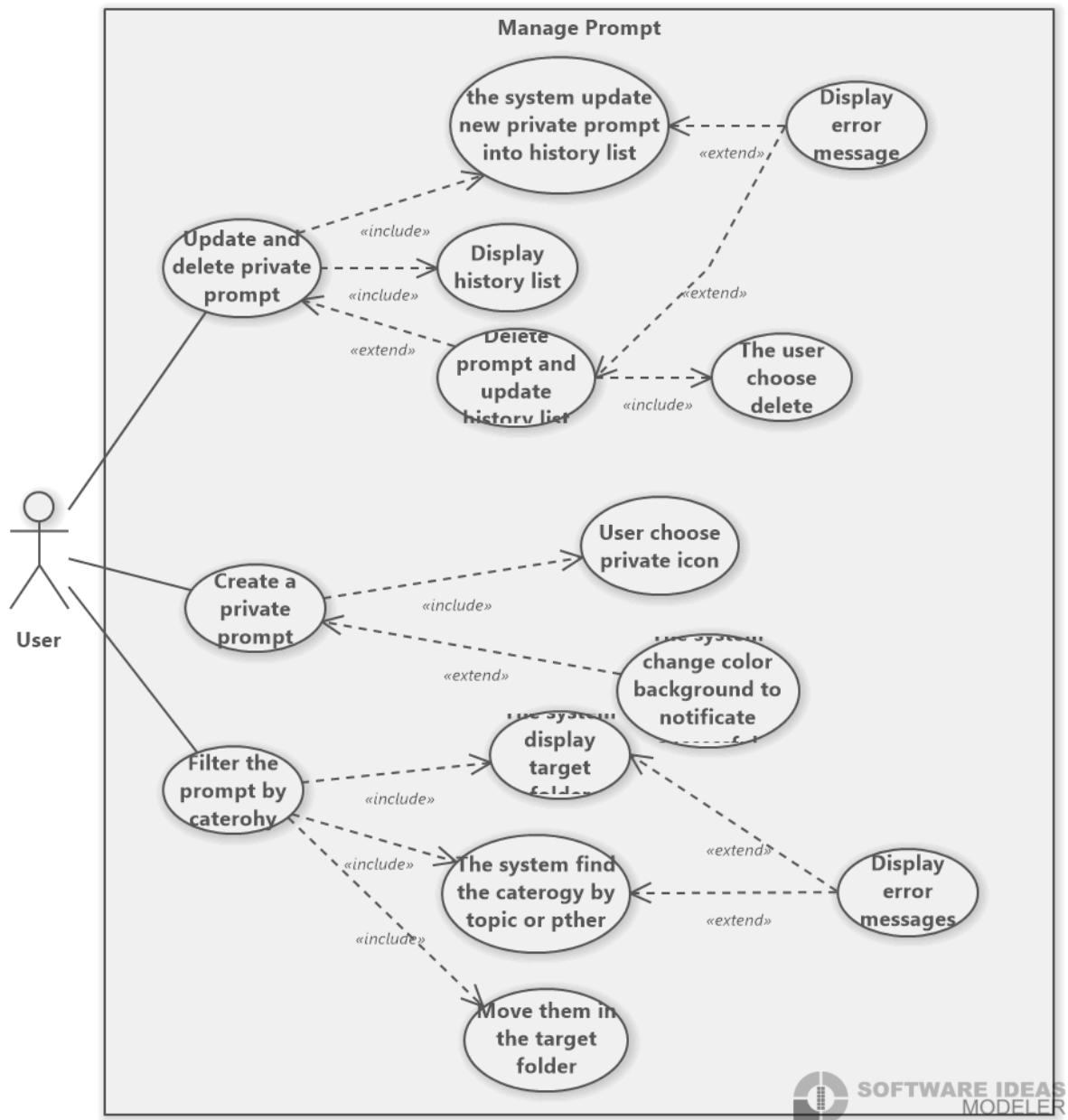
Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.4. Knowledge Data Management:



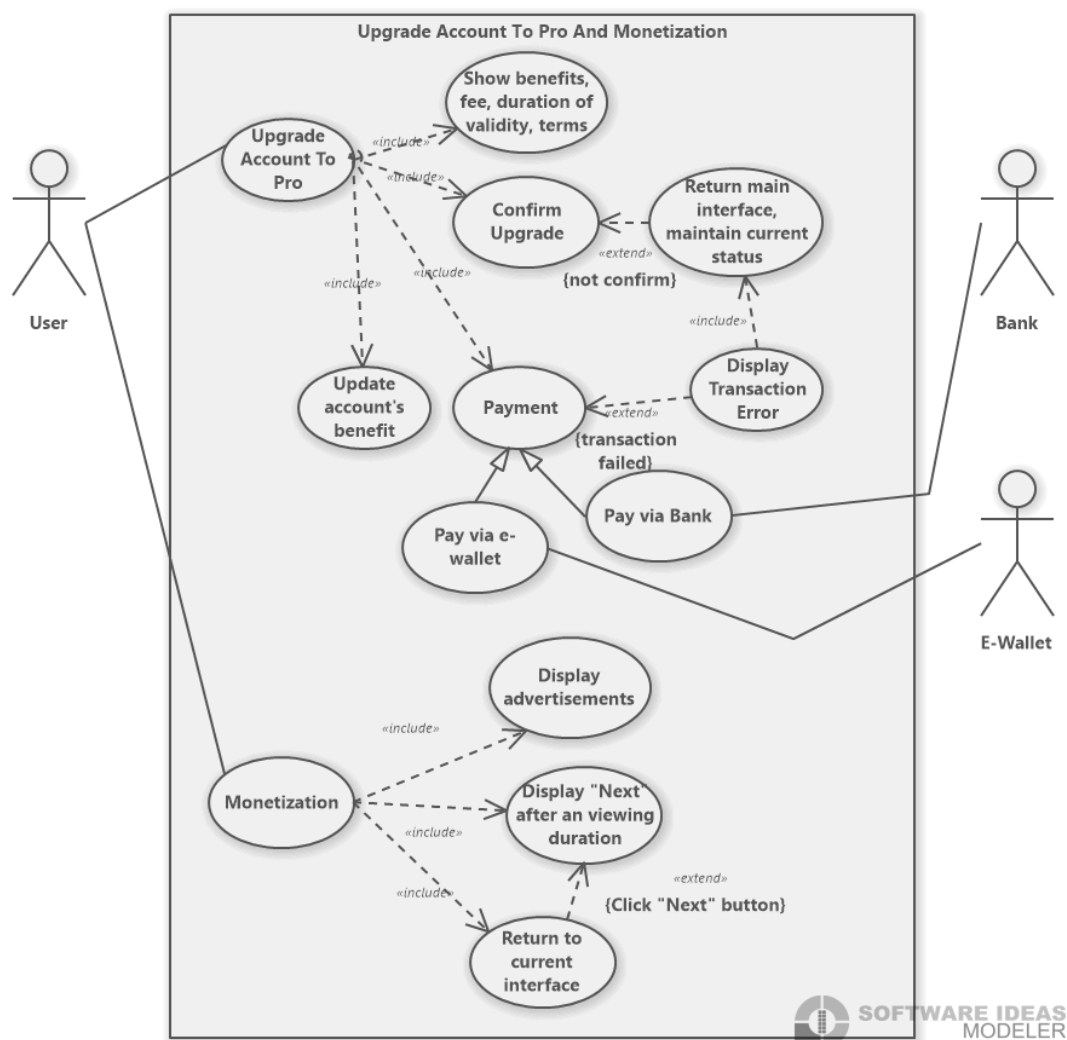
Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.5. Prompt Management:



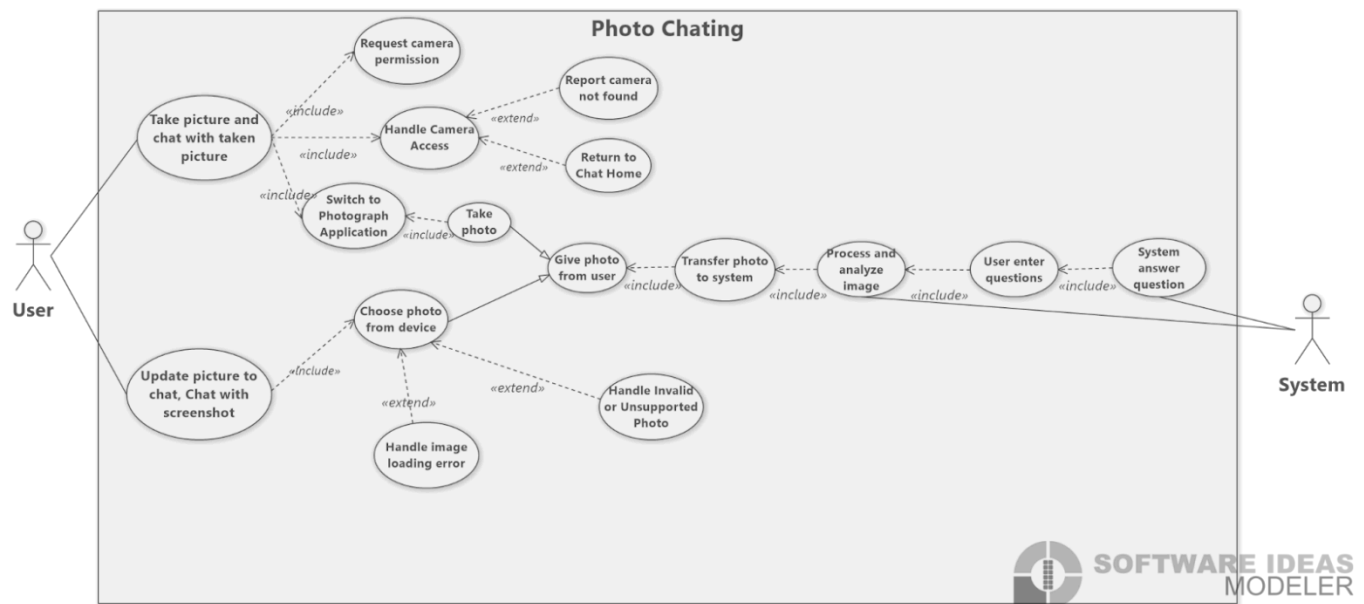
Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.6. Account Management:

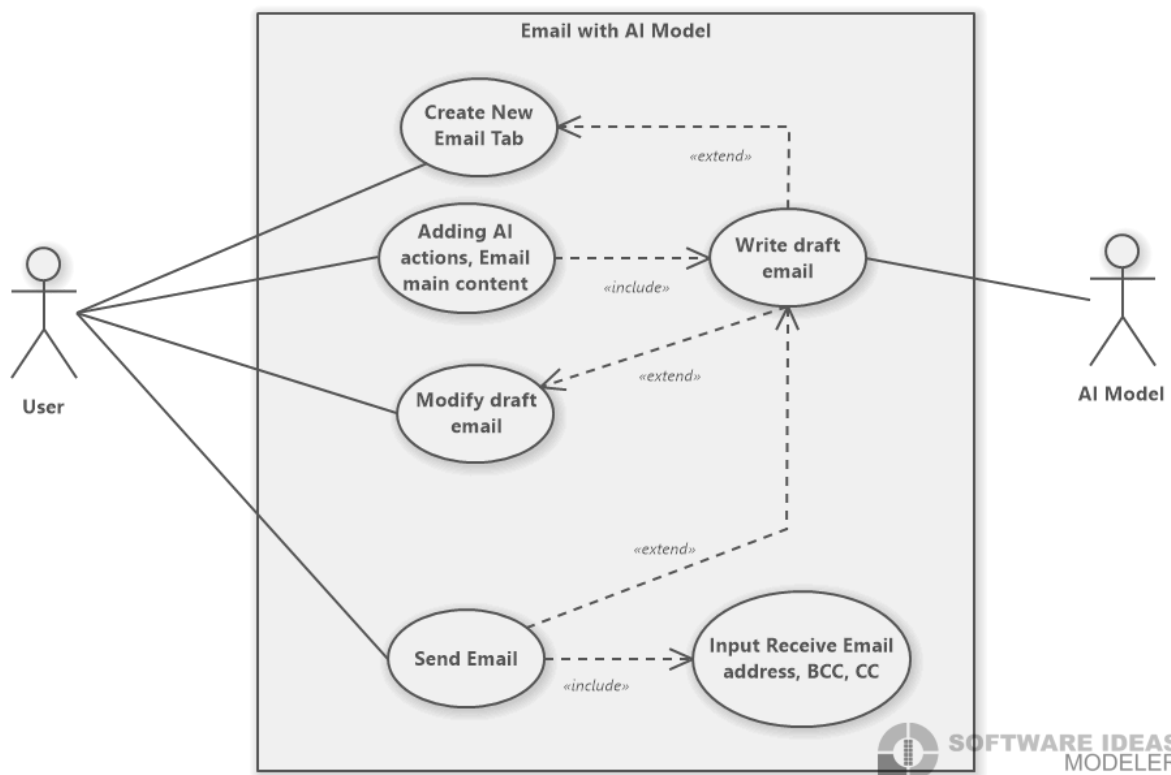


Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.7. Photo Chating:



2.8. Email with AI Agents:



Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

3. Use-case Specifications

3.1. Use-case: Authentication and Authorization:

3.1.1. Register an account:

Use Case	Register an account
Actor	User
Pre-conditions	<ul style="list-style-type: none"> - User do not have an account - User has access to a valid email address.
Main flow	<ol style="list-style-type: none"> 1. The user opens the account registration page. 2. User enters username, password, and other information (email, phone number, etc.). 3. System checks that binding. 4. The system sends an account activation email to the registered email address. 5. User confirms email to activate account.
Alternative Flow	<ol style="list-style-type: none"> 3a. Username Already Exists: <ol style="list-style-type: none"> 1. The system notifies: "The username has already been taken." 2. The user is asked to choose and provide a different username and try again. 3b. Password Does Not Meet Security Requirements: <ol style="list-style-type: none"> 1. The system indicates which requirements the password fails to meet (e.g., length, digits, special characters, etc.). 2. The user is asked to modify the password and retry. 3c. Password and Confirm Password Do Not Match: <ol style="list-style-type: none"> 1. The system notifies: "Passwords do not match." 2. The user is asked to re-enter the password in the "Confirm Password" field to proceed. 4a. Activation Email Sending Error: <ol style="list-style-type: none"> 1. The system notifies that there was an issue sending the email. 2. The system suggests the user try again later or contact support. 4b. Email Confirmation Took Too Long, Exceeding the Specified Time Limit: <ol style="list-style-type: none"> 1. The system notifies that the activation email has expired. 2. The user can request a new activation email. 5a. User Did Not Confirm Email Within the Specified Timeframe: <ol style="list-style-type: none"> 1. The account is not activated. 2. The user can request a new activation email or create a new account.
Post condition	<ol style="list-style-type: none"> 1. Successful Registration: <ul style="list-style-type: none"> • The user account is stored in the database. 2. Email Confirmation: <ul style="list-style-type: none"> • The email address provided is marked as verified in the system. 3. Registration Failures: <ul style="list-style-type: none"> • If the registration fails (e.g., username taken, invalid password), no new account is added to the database.
Special requirements	<ul style="list-style-type: none"> - Require password strength with the following criteria: minimum length of 8 characters, including uppercase letters, lowercase letters, numbers and special characters. - Processing time for registration and account activation via email is no more than 5 seconds. - The time to send the account activation email does not exceed 30 seconds (from the time the user requests to the time the email is received).

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

3.1.2. Login:

Use case	Login
Actor	User
Pre-conditions	User already has an account
Main Flow	<ol style="list-style-type: none"> 1. User opens the login page. 2. User enters login name and password. 3. The system checks login information. 4. If compatible, the system logs in the user and allows use of the application.
Alternative Flow	<ol style="list-style-type: none"> 3a. User Enters Incorrect Information: <ol style="list-style-type: none"> 1. The system notifies: "Incorrect password/username." 2. The user is prompted to re-enter the information and try again. 4a. System Encounters a Technical Issue (overload, etc.): <ol style="list-style-type: none"> 1. The system displays: "System error, unable to process request." 2. The system suggests that the user try again later.
Post condition	<ol style="list-style-type: none"> 1. Successful Login: <ul style="list-style-type: none"> • The user is authenticated and granted access to the system. • The session is initialized, and the user is redirected to the application's main interface. 2. Failed Login: <ul style="list-style-type: none"> • No session is created, and the user is prompted to try again.
Special Requirements	<ul style="list-style-type: none"> - Login processing time should not exceed 5 seconds. - The system can handle ~1000 users logging in simultaneously. - Specific error message when entering incorrect login information. - Fast response time (real-time) to avoid waiting time. - There is a "remember login" function for users on personal devices.

3.1.3. Recover Password:

User case	Recover Password
Actor	User
Pre-conditions	User had account in database
Main Flow	<ol style="list-style-type: none"> 1. User selects "Forgot password". 2. User enters registered account name. 3. The system sends an email with a link to reset the password via the email associated with the account. 4. The user accesses the link and enters a new password. 5. The user re-enters the new password and selects the update password.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Alternative Flows	<p>2a. Account does not exist in the system:</p> <ol style="list-style-type: none"> 1. The system announces "This account does not exist in the system". 2. Users can check the information again or register a new account. <p>3a. The time to send the password reset email is too long, beyond the specified time:</p> <ol style="list-style-type: none"> 1. Notification of email expiration date. 2. Users can request to resend a new activation email. <p>4a. Password does not meet security requirements:</p> <ol style="list-style-type: none"> 1. The system notifies that the password does not meet any requirements (length, numbers, special characters, ...). 2. Ask the user to change the password and try again. <p>5a. Password and re-entered password do not match:</p> <ol style="list-style-type: none"> 1. The system notifies "Password does not match". 2. Ask the user to re-enter the "Re-enter password" section correctly and continue.
Post condition	<ol style="list-style-type: none"> 1. Successful Password Recovery: <ul style="list-style-type: none"> • The user's password is securely updated in the database. • The user is notified of the successful password reset. • Any active sessions associated with the old password are terminated (if applicable). • The reset link (Email Confirmation) becomes invalid after use to prevent reuse. 2. Failed Password Recovery: <ul style="list-style-type: none"> • No updates are made to the database.
Special Requirements	- The time to send a password reset email must not exceed 30 seconds (from the time the user requests to the time the email is received).

3.2. Use-case: Chat with AI Agents:

3.2.1. Create New Thread Chat:

Use Case	Create New Thread Chat
Brief Description	Create a new thread chat with existing AI model
Actor	User
Basic Flow	<ol style="list-style-type: none"> 1. User selects the option create new thread chat 2. System create and store a new thread chat data 3. User selects the AI model from the list 4. System loading AI model
Alternative Flows	<p>Alternative flow 1:User do not select the AI agent</p> <ol style="list-style-type: none"> 1. From step 2 the system automatically choose system default AI model for the next operation 2. Loading AI model
Pre-conditions	User has signed up and logged into the web.
Post-condition	A new thread chat is created and stored into database

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

3.2.2. Chat with AI Agent:

Use Case	Chat With AI Agent
Brief Description	User chat with AI bot through ui interface, using their token
Actor	User, AI Agent
Basic Flow	<ol style="list-style-type: none"> 1. User types a message or query in the input chat box 2. System check the account for available tokens 3. System sends the input to the selected AI model 4. AI model retrieves a response and display on the chat window
Alternative Flows	Alternative flow 1: User account do not have enough tokens <ol style="list-style-type: none"> 1. From step #2, the system do not proceed with processing input 2. System display a notification indicating insufficient tokens and options for purchasing additional tokens
Pre-conditions	<ul style="list-style-type: none"> - User has signed up, logged in, and selected or created a chat thread. - User has sufficient tokens for interaction.
Post-condition	- New conversation starts between the user and the AI model.
Special Requirement	Response Time: The system must respond to user messages within 1 second

3.2.3. History Browsing:

Use Case	History Browsing
Brief Description	User search for existing chat, access the chat history feature
Actor	User
Basic Flow	<ol style="list-style-type: none"> 1. User choose Chat History feature 2. System retrieves a list of previous chat threads 3. User select a chat thread to view 4. System display the conversation content
Alternative Flows	Alternative Flow 1: User do not have any previous chat <ol style="list-style-type: none"> 1. From step #1 system displays a message: "No previous chat threads available" Alternative Flow 2: User continue to chat on the previous section <ol style="list-style-type: none"> 1. From step #4 , continue use case Chat with AI Model
Pre-conditions	<ul style="list-style-type: none"> - User has logged into their account on the website. - User has previously created one or more chat threads.
Post-condition	- User browsing on a previous conversation with AI models.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

3.3. Use-case: Create and manage AI BOT

3.3.1. Create AI BOT:

Use case	Create AI BOT
Brief - Descriptions	This feature begins when the user wants to create a new AI BOT.
Actors	User
Main Flow	<ol style="list-style-type: none"> 1. User chooses to create a new AI BOT. 2. System asks user to select a model for the chatbot (ChatGPT, Gemini, Copilot) 3. User selects a model. 4. User can optionally name the bot (default name: "{Model}_{ordinal}"). 5. User can optionally insert knowledge data. 6. System creates an AI BOT based on that model.
Alternative Flow	<ol style="list-style-type: none"> 4a. Bot's name already existed: <ol style="list-style-type: none"> 1. If it's the default name, increase the ordinal by 1 and create the bot. 2. If it's the name that the user added, displays an error message and asks the user to change the name. 6a. System cannot create AI BOT: <ol style="list-style-type: none"> 1. System displays a "fail" message. 2. System returns to the create menu.
Pre-conditions	User has to sign in first before creating an AI BOT.
Post-conditions	<ol style="list-style-type: none"> 1. Bot created: <ul style="list-style-type: none"> - New AI Bot is created with specified model and optional name and knowledge data. - System redirects user to new bot's chat UI. 2. Bot not created: <ul style="list-style-type: none"> - System notifies user of the error.
Special Requirements	<ul style="list-style-type: none"> - Simple and minimal UI. - System moves to the new bot's chat UI after the user created the bot.

3.3.2. Display/Find AI BOT:

Use case	Display/Find AI BOT
Brief - Descriptions	This feature begins when the user wants to find an AI BOT.
Actors	User

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Main Flow	<ol style="list-style-type: none"> 1. User hits the slide drawer icon. 2. System opens a new menu that displays a search bar and all the bots that the user has created. 3. User enters the information of the bot (name, model) in the search bar. 4. System finds the bots that have similar information. 5. System displays all the bots found.
Alternative Flow	<ol style="list-style-type: none"> 2a. User hasn't created any bots: <ol style="list-style-type: none"> 1. System displays a "create" symbol that leads to the create feature. 2b. Too many bots to appear on the menu.: <ol style="list-style-type: none"> 1. System displays a limited number of bots with the "..." symbol at the end. 4a. System cannot find any bot that satisfies the information: <ol style="list-style-type: none"> 1. System displays a message to indicate no bot was found.
Pre-conditions	User has to sign in first.
Post-conditions	A list of AI bots that the user has created.
Special Requirements	Minimal UI. The bots list should only contain the bots's icons and names.

3.3.3. Update/Delete AI BOT:

Use case	Update/Delete AI BOT
Brief - Descriptions	This feature begins when the user wants to update or delete an AI BOT.
Actors	User
Main Flow	<ol style="list-style-type: none"> 1. User right clicks on the bot. 2. System asks if the user wants to update or delete the bot. 3. User chooses delete: <ol style="list-style-type: none"> 1. System asks for the user's confirmation. 2. System deletes the bot's information and data. 3. System displays a message that the bot has been deleted. 4. User chooses update: <ol style="list-style-type: none"> 1. System moves to the update menu. 2. User changes the bot's information (name, knowledge data, etc). 3. User hits save. 4. System saves the bot's new information.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Alternative Flow	<p>3a. System cannot delete the bot:</p> <ol style="list-style-type: none"> 1. System displays a message to user. <p>4a. Bot's new name already existed:</p> <ol style="list-style-type: none"> 1. System displays a message. 2. System asks the user to enter a different name. <p>4b. System cannot save the bot's information:</p> <ol style="list-style-type: none"> 1. System displays a message.
Pre-conditions	<ul style="list-style-type: none"> - User has to sign in first. - The bot needs to exist before being updated or deleted.
Post-conditions	<ol style="list-style-type: none"> 1. Delete: <ul style="list-style-type: none"> - The bot and its data are fully deleted. - If the deletion process fails, the user is notified of the error. 2. Update: <ul style="list-style-type: none"> - The bot's information is successfully updated in the system. - If the update process fails, the user is notified of the error.
Special Requirements	System shall delete data thoroughly to protect the user's information.

3.3.4. Update prompt for AI Bot:

Use case	Update prompt for AI BOT
Brief - Descriptions	The feature begins when user updates the prompt for an AI bot to refine its responses and behavior.
Actors	User
Main Flow	<ol style="list-style-type: none"> 1. User hits the "change prompt" icon on the current prompt. 2. User can then edit the current prompt or completely rewrite it. 3. User confirms the change. 4. System replaces the old prompt and answer with the updated one.
Alternative Flow	<p>4a. System cannot replace new prompt:</p> <ol style="list-style-type: none"> 1. System displays a message. 2. System returns old prompt and answer.
Pre-conditions	Only the newest prompt can be updated.
Post-conditions	<ul style="list-style-type: none"> - The AI bot's prompt is successfully updated with the new prompt provided by the user. - The AI bot's responses and behavior are updated to reflect the new prompt.
Special Requirements	Allows users to view the old prompt and answer after updating them.

3.3.5. Communicate with created AI BOT through Chat widget:

Use case	Communicate with created AI BOT through Chat widget
----------	---

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Brief - Descriptions	This feature begins when the user interacts with the chat bot through a chat widget.
Actors	User, Chat widget app.
Main Flow	<ol style="list-style-type: none"> 1. User initiates a chat with the bot by clicking on the chat widget. 2. User types a query or message into the text area. 3. User hits send. 4. The bot processes the input and generates a response, which is displayed to the user on the widget.
Alternative Flow	<ol style="list-style-type: none"> 1a. Chat widget fails to open: <ol style="list-style-type: none"> 1. System displays a message to inform the user. 4a. The bot cannot respond to query: <ol style="list-style-type: none"> 1. System displays a message to inform the user. 2. System asks the user if they want to add knowledge data relating to that topic.
Pre-conditions	A functional chat widget integrated into the website or application.
Post-conditions	<ul style="list-style-type: none"> - The user's query is successfully processed by the AI bot. - The AI bot generates a relevant and informative response to the user's query.
Special Requirements	<ul style="list-style-type: none"> - The chat widget should be user-friendly and visually appealing. - The widget shall be able to redirect user to the main website or application if necessary.

3.3.6. Add/Remove knowledge data:

Use case	Add/Remove knowledge data
Brief -Descriptions	This feature begins when user adds knowledge data into the AI BOT or removes a knowledge data that the bot already learned.
Actors	User
Main Flow	<ol style="list-style-type: none"> 1. User right chooses to update the bot. 2. System moves to the update menu for the current bot. 3. User chooses the option to add/remove knowledge data. 4. System displays a list of available knowledge data. If the bot currently doesn't have a specific knowledge, displays "add" next to the knowledge data's name. Otherwise, it displays "remove". 5. User chooses "add" to add a new knowledge data or "remove" to remove an already learned knowledge data. 6. User confirms their options. 7. System inserts data into or removes data from the bot accordingly.
Alternative Flow	<ol style="list-style-type: none"> 7a. System cannot insert or remove data: <ol style="list-style-type: none"> 1. System displays a message.
Pre-conditions	The knowledge data has to exist in the system before adding it into the bot.
Post-conditions	The specified knowledge data is successfully added to / remove from the AI bot's knowledge base.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Special Requirements	System displays the name, source and status("add" / "remove") of a knowledge data in the knowledge data list.
-----------------------------	---

3.3.7. Preview and Chat with AI BOT:

Use case	Preview and Chat with AI BOT
Brief - Descriptions	This feature shows preview when the user chats with AI BOT.
Actors	User
Main Flow	<ol style="list-style-type: none"> 1. User initiates a chat session with AI BOT by entering a query in the text area. 2. User submits the query by hitting ENTER on their keyboard or the send icon on the UI. 3. The bot processes the input and generates a response, which is displayed to the user.
Alternative Flow	3a. The bot cannot respond to query: <ol style="list-style-type: none"> 1. System displays a message to inform the user. 2. System asks the user if they want to add knowledge data relating to that topic.
Pre-conditions	The bot needs to exist.
Post-conditions	<ul style="list-style-type: none"> - The user's query is successfully processed by the AI bot. - The AI bot generates a relevant and informative response to the user's query.
Special Requirements	System can display some recommended questions that are related to the current topic. User can click on these questions to ask the bot.

3.3.8. Publish AI Chat to Slack, Telegram, Messenger:

Use case	Publish AI Chat to Slack, Telegram, Messenger
Brief - Descriptions	This feature begins when user wants to publish an AI Chat log to platforms like Slack, Telegram, Messenger.
Actors	User, Message apps
Main Flow	<ol style="list-style-type: none"> 1. User chooses to share the conversation with the current bot. 2. System displays options to share to other platforms. 3. User selects a platform. 4. System asks the user to choose specific recipients. 5. System transforms the logged chat into a suitable format for the selected platform. 6. System sends the logged chat to the platform.
Alternative Flow	5a. System cannot format chat correctly: <ol style="list-style-type: none"> 1. System displays a message to inform the user. 6a. System cannot send the logged chat: <ol style="list-style-type: none"> 1. System displays a message to inform the user.
Pre-conditions	<ul style="list-style-type: none"> - API keys for the desired messaging platforms. - User needs to have an account of the platform.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Post-conditions	<ul style="list-style-type: none"> - The AI chat log is successfully published to the selected platforms. - The user is notified of the successful publication of the chat log.
Special Requirements	<ul style="list-style-type: none"> - System shall be able to handle large chat logs. - System shall be able to handle different formatting and messaging protocols for different platforms. - System should limit the amount of messages sent if the chat data is too large.

3.4. Use-case: Knowledge Data Management

3.4.1. Manage Data:

Use case	Manage Data Knowledge
Brief - descriptions	This functionality allows users to add, display, and manage data knowledge, including loading data from various sources such as files, URLs, Google Drive, Slack, and Confluence.
Actor	User
Basic flow	<ol style="list-style-type: none"> 1. User selects the "Add Data" option. 2. The system prompts the User to select a data source. 3. User selects the desired data source. 4. The system provides information and access to the selected data source. 5. The system loads and verifies the validity of the data. 6. The system stores the data in the database.
Alternative Flow	<p>Alternative Flow 1: Data Source is Unavailable</p> <ol style="list-style-type: none"> 1. At Step #3, the system displays a notification indicating that the selected data source is unavailable. 2. The system suggests alternative actions (select a different data source or check the data source's availability)
Pre-conditions	The User has successfully logged into the system
Post-conditions	The user successfully adds new data knowledge to database
Result	Organized and standardized knowledge datasets

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

3.4.2. Disable/delete data:

Use case	Disable/ delete data
Brief - descriptions	This function allows the user to disable or remove a data source from the system's knowledge base when the data source is no longer needed or is invalid.
Actor	User
Basic Flow	<ol style="list-style-type: none"> 1. User selects the "Delete Data" option. 2. The system prompts the User to select a data source for deletion. 3. User selects the data source to be deleted. 4. The system removes the selected data from the database.
Alternative Flow	<p>Alternative Flow 2: Error During Deletion</p> <ol style="list-style-type: none"> 1. At Step #4, an error occurs during the deletion process, The system displays an error message indicating that the deletion failed. 2. The system provides options to retry the deletion
Pre-conditions	<ul style="list-style-type: none"> - The User has successfully logged into the system - The data source already exists and is accessible.
post-conditions	Data was successfully deleted from the database.
result	data resource is deleted.

3.5. Use-case: Prompt Management

3.5.1. Create a private prompt:

User case	Create a private prompt
Brief description	This case describes how the user can creative a private prompt.
Actors	User
Basic flow	<ol style="list-style-type: none"> 1. Users can choose the Private chat icon in the message box. 2. The system display private prompt mode by change color background to black or other color different public prompt.
Alternative flow	<p>Alternative flow 2: System can not display private prompt mode.</p> <ol style="list-style-type: none"> 1. The system display error. 2. System try again.
Pre-conditions	User login in chatbot website
Post-condition s	The user can create a private prompt and can watch private list

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

3.5.2. Filter prompts by category:

User case	Filter prompts by category
Brief description	This case describes how the user can filter prompts by category, the system divide prompt by topic or other ...
Actors	User
Basic flow	<ol style="list-style-type: none"> The system identify category of the prompts The system moves and display target folder.
Alternative flow	Alternative flow 2: The system cannot display the results. <ol style="list-style-type: none"> The system displays errors. The system is going to be tries again.
Pre-conditions	The user must write prompt with chat bot.
Post-conditions	The user can filter prompts by category.

3.5.3. Update and delete private prompt:

User case	Update and delete private prompt
Brief description	This case describes how the user can update and delete private prompt.
Actors	User, the system
Basic flow	<ol style="list-style-type: none"> When the user creates a new prompt, the system updates the prompt into history list. The system displays history list of prompts. When the user wants delete the prompt, the user can choose 3-dots in prompt in history list. The system display the list choose. The user chooses delete. The system delete this prompt out the history list and re-update the history list. Return to step 2.
Alternative flow	Alternative flow 1: The system cannot update the history list. <ol style="list-style-type: none"> The system displays errors. The system is going to be tries again. Alternative flow 6: The system cannot delete the prompt. <ol style="list-style-type: none"> The system displays errors. The system is going to be tries again.
Pre-conditions	The user chat with chat bot.
Post-conditions	The user can update and delete the prompt.

3.6. Use-case: Account Management

2.6.1 Upgrade Account to Pro

Use Case	Upgrade Account to Pro
----------	------------------------

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Brief Descriptions	The user intends to upgrade their account to Pro, gaining benefits such as unlimited tokens and an ad-free experience.
Actor	User, Bank, e-Wallet, ADMIN System
Main flow	<ol style="list-style-type: none"> 1. User selects the option to upgrade the account. 2. System displays the benefits, fees, duration of validity, terms, and commitments related to the Pro account. 3. User confirms the upgrade to Pro. 4. System presents the payment interface. 5. User completes the payment, choosing a preferred payment method (Bank or e-Wallet). 6. System verifies the transaction's success in real-time. 7. System updates the account to Pro, enabling Pro features such as unlimited tokens and an ad-free experience.
Alternative flow	<ol style="list-style-type: none"> 3a. User chooses not to confirm the upgrade to Pro: <ol style="list-style-type: none"> 1. The system returns to the main interface, maintaining the current account status. 6a. Transaction failure: <ol style="list-style-type: none"> 1. System displays the message: "Transaction failed. Please check your payment transaction." 2. The System checks with the Bank or e-Wallet to determine the reason for the transaction failure. 3. System notifies the User of the reason for the failure. 4. Troubleshooting: <ul style="list-style-type: none"> + If the transaction issue is resolved, proceed to step 6. + If the transaction issue persists, inform the user to await further resolution.
Pre-conditions	<ul style="list-style-type: none"> - The account is currently logged in. - The account has not been upgraded to Pro.
Post-conditions	The account is upgraded to Pro, and the user experiences the agreed-upon benefits.
Special Requirement	Real-time quality: Transaction confirmation between the System and Bank or e-Wallet should occur in real-time to ensure immediate response and processing.

Table 19. Upgrade Account To Pro

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

2.6.2 Monetization

Use Case	Monetization
Brief Descriptions	The system displays advertisements to free users and earns revenue from third-party advertisers.
Actor	User, ADMIN System.
Main flow	<ol style="list-style-type: none"> 1. System displays an advertisement to free users after a specified period or following a certain number of chat interactions, requiring the user to view the ad. 2. After an agreed-upon viewing duration (as per the advertisement agreement), the System displays a "Next" button. 3. User clicks "Next" to continue using the service. 4. System receives revenue from third-party advertisers based on ad views.
Alternative flow	<ol style="list-style-type: none"> 2a. User watches the entire ad: <ol style="list-style-type: none"> 1. The System closes the advertisement. 2. The System returns to the previous user interface.
Pre-conditions	<ul style="list-style-type: none"> - The account is currently logged in. - The account has not been upgraded to Pro.
Post-conditions	The User experiences the advertisement, and the System earns revenue based on views tracked by the third-party advertisers.
Special Requirement	<ul style="list-style-type: none"> - Video Quality: Ensures smooth and clear playback of video ads. - Advertisement Quality: Maintains engagement standards to enhance user experience and meet third-party expectations.

Table 20. Monetization

3.7. Use-case: Photo Chating

3.7.1. Update picture to chat, Chat with screenshot:

Use case	Update picture to chat, Chat with screenshot
Actor	User
Goal	Upload photos and chat about photo content
Pre-conditions	User has logged into their account
Main Flow	<ol style="list-style-type: none"> 1. User selects the "Upload photo" function. 2. User selects a photo from the device (available photo / photo just taken screenshot) and uploads it to the chat box in the system.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

	<p>3. The system processes the photo, using AI to analyze the content in the photo.</p> <p>4. User enters a question about the photo content.</p> <p>5. The system answers the question based on the analyzed photo content and searches for information on the internet.</p>
Alternative Flow	<p>2a. Invalid or unsupported photo:</p> <ol style="list-style-type: none"> The system reports the error "Invalid image" or "Image format not supported". User can select another photo and try again. <p>2b. Image loading error:</p> <ol style="list-style-type: none"> The system crashes when the user uploads images and is interrupted (internet, overload, ...). The system reports an error and asks the user to try again. <p>3a. Users just upload photos and don't ask any questions:</p> <ol style="list-style-type: none"> The system will process and load all content into the learning data set. Then give an overview/summary of the photo's content. The user can continue to ask about that content and the system will continue the response flow → proceed to main flow 4. <p>3b. The user just uploaded the photo and asked a question: → Proceed to main flow 4.</p>
Post condition	<ol style="list-style-type: none"> Successful Photo Upload: <ul style="list-style-type: none"> The uploaded photo is securely stored in the system's database or temporary storage for further processing. The photo is successfully analyzed by the AI module: <ul style="list-style-type: none"> Relevant content from the photo is extracted and stored in the system's learning dataset. If the user asks a question: <ul style="list-style-type: none"> The system logs the user's question and response. An answer related to the photo content is generated and sent back to the user. If the user does not ask a question: <ul style="list-style-type: none"> A summary or overview of the photo's content is provided. The user can optionally query the system further, which loops back to processing the question. Invalid or Unsupported Photo: <ul style="list-style-type: none"> The system provides feedback to the user: "Invalid image" or "Image format not supported". Image Loading Error: <ul style="list-style-type: none"> If the image fails to load due to network interruptions or system overload: The system display the failure for troubleshooting and monitoring.

3.7.2. Take picture and chat with taken picture

Use Case	Take picture and chat with taken picture
Actor	User
Goal	Take photos directly and get answers to questions about the photos you just took.
Pre-conditions	User is logged in

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Main Flow	<ol style="list-style-type: none"> 1. The user selects the “Take photo” function. 2. The system requests permission to use the device's camera and allows the user to take photos. 3. User takes a photo and sends it to the system. 4. Image content processing and analysis system. 5. The user enters a question related to the photo taken. 6. The system answers questions based on analyzed image content and searches for information on the internet.
Alternative Flow	<ol style="list-style-type: none"> 2a. Users do not agree to access and use the camera: <ol style="list-style-type: none"> 1. The system closes the photo capture window. 2. Return to the chat home page interface. 2b. User allows access: <ol style="list-style-type: none"> 1. Switch the interface to the user device's photography application. 2. After taking the photo, the photo will automatically be transferred to the chat frame. 3. The user clicks send to send the photo to the system. 2c. Devices that do not support camera photography: <ol style="list-style-type: none"> 1. The system reports an error that the camera cannot be found. 2. Return to the chat home page interface. 3a. Users just upload photos and don't ask any questions: <ol style="list-style-type: none"> 1. The system will process and load all content into the learning data set. Then give an overview/summary of the photo's content. 2. The user can continue to ask about that content and the system will continue the response flow → proceed to main flow 5. 3b. The user just uploaded the photo and asked a question: → Proceed to main flow 4.

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Post condition	<ol style="list-style-type: none"> Successful Photo Capture and Interaction: <ul style="list-style-type: none"> The photo taken by the user is securely uploaded to the system. The system successfully processes and analyzes the photo content using AI: <ul style="list-style-type: none"> Extracted data (tags, objects, metadata) from the photo are stored in the system's learning dataset for future use. If the user asks a question: <ul style="list-style-type: none"> The system generates and provides a relevant response based on the analyzed photo content. The user's question and the system's response are logged for auditing and potential improvement of the AI model. If the user does not ask a question: <ul style="list-style-type: none"> A summary or overview of the photo's content is generated and displayed to the user. The user is given the opportunity to ask questions, continuing the chat flow. Camera Access Denied or Not Supported: <ul style="list-style-type: none"> If the user denies camera access: <ul style="list-style-type: none"> No photo is taken or uploaded. The system closes the photo capture window and navigates back to the chat home interface. If the device does not support a camera: <ul style="list-style-type: none"> An error message is displayed: "Camera not found". No further action is taken, and the user is returned to the chat home interface. Technical or Upload Errors: <ul style="list-style-type: none"> If there is an error in photo upload or processing: <ul style="list-style-type: none"> The photo is not saved or analyzed. The system provides feedback to the user with error details and suggests retrying or troubleshooting. No impact is made on the learning dataset or existing interactions.
-----------------------	---

3.8. Use-case: Email with AI Model

Use Case	Write Email with AI Model
Brief description	Using AI model to write email base on the action added by users
Actor	User, AI Model

Chat Bot	Version: <1.0>
Use-Case Specification	Date: 21/11/2024

Basic Flow	<ol style="list-style-type: none"> 1. User navigates to the email drafting tab 2. System opens the new email tab. 3. User selects actions for AI model (e.g., Thanks, Sorry, Yes, No, Follow-Up, Request for more information) 4. System generate a draft email and display on the screen 5. User reviews and modifies the email if necessary 6. User input receiver information 7. System sending email
Alternative Flows	<p>Alternative Flow 1: User choose an existing email draft</p> <ol style="list-style-type: none"> 1. System open the existing email draft . 2. Continue step #2 in the basic flow <p>Alternative Flow 2: User saving email instead of sending</p> <ol style="list-style-type: none"> 1. From step 5, user close the email tab 2. System auto store the draft email in history
Pre-conditions	<ul style="list-style-type: none"> - Users has logged into their account on the website. - User account has enough tokens to use AI Model
Post-conditions	<ul style="list-style-type: none"> - Create a draft email successfully
Special Requirement	<ul style="list-style-type: none"> - Security:The system must securely handle and encrypt email content to protect user privacy. - Performance:Draft generation should not exceed 2 seconds to provide a responsive experience. - User Experience:Provide an intuitive, clear interface, with easy-to-navigate options for selecting AI actions and editing drafts.