

EX. NO.2 IDENTIFICATION OF USECASE AND DEVELOPMENT OF USECASE MODEL**DATE****AIM:** To identify the problem domain and develop UML diagrams.**PROBLEM STATEMENT:**

The current process of managing expiry dates of products is time-consuming and prone to errors, resulting in potential health hazards and financial losses for businesses. This problem is particularly acute for businesses that deal with perishable goods, such as food and pharmaceuticals. The Expiry Date Identifier project aims to address this problem by developing an application that can accurately detect and extract expiry dates from product labels using OCR technology, thereby streamlining the expiry date management process and improving accuracy.

IDENTIFICATION OF ACTORS:**PRIMARY ACTORS**

A) User

SECONDARY ACTORS

A) Admin

USECASE TEMPLATE:

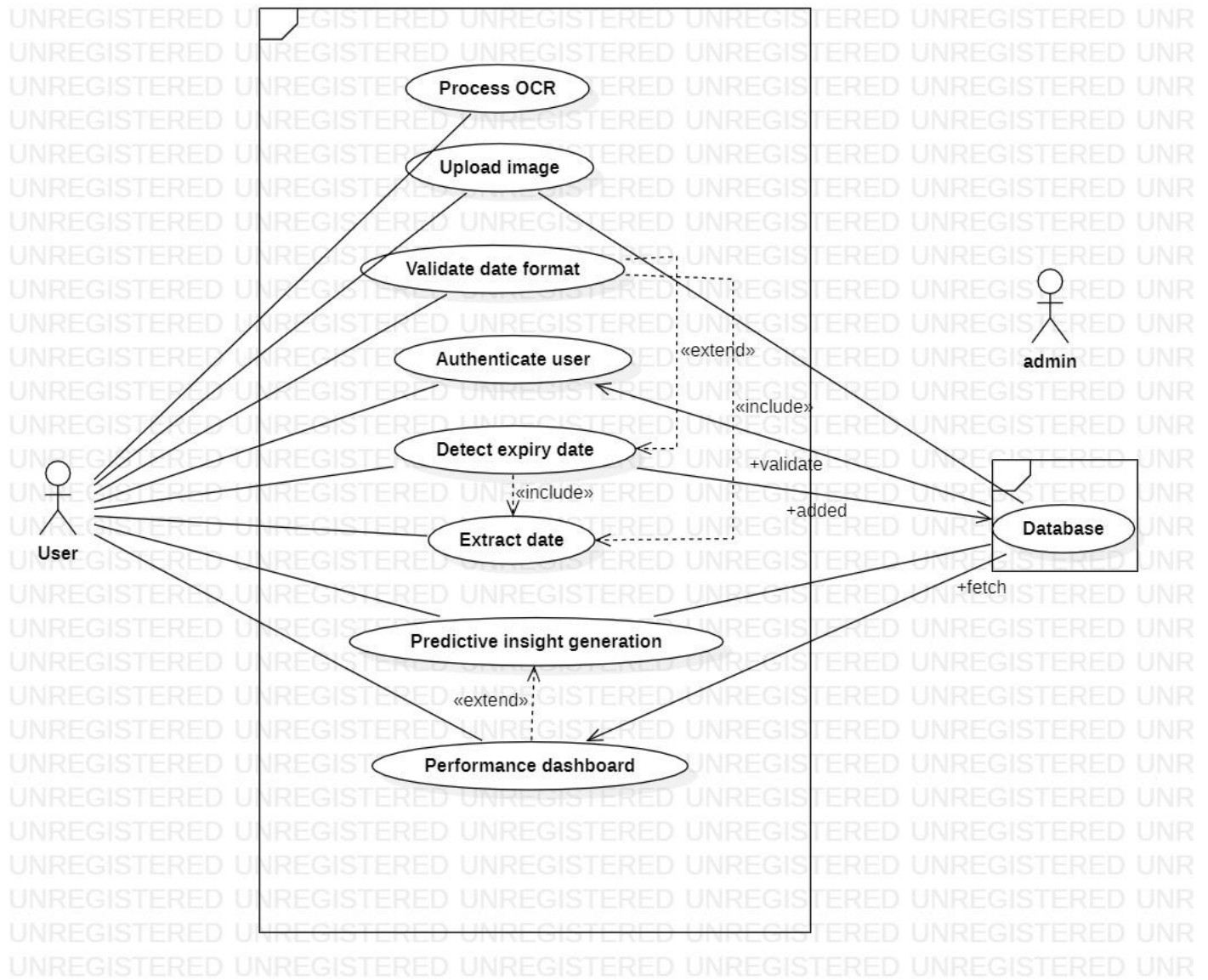
SI. NO.	Template	Description
1.	usecase	Authenticate User Upload Image Process OCR Detect Expiry Date Extract Date Validate Date Format Access Learning Performance Dashboard Generate Predictive Analytics Review Requirements Review Prototypes Generate Test Cases
2.	Primary actor	A) user
3.	goals of an usecase	The goals of the provided use cases are to: authenticate users, upload images for processing, convert text using OCR, identify expiry dates, extract date information, validate date formats, access learning performance dashboards, generate predictive

		analytics, review requirements, review prototypes, and generate test cases.
4.	Pre-condition	Preconditions for the provided use cases include user authentication, successful image upload, completed OCR processing, detected expiry date, extracted date information, validated date format, authorized access to the learning performance dashboard, availability of relevant learning data for predictive analytics, presence of the software requirements document for review, readiness of prototypes for user feedback, and defined use cases and functional requirements for testing.
5.	scenario	Scenarios encompass user authentication, image upload, OCR processing, expiry date detection, date extraction, format validation, dashboard access, predictive analytics generation, requirements review, prototypes review, and test case generation across the provided use cases.
6.	Triggers	Triggers for the given use cases include user login initiation, image selection for upload, OCR processing start, completion of OCR analysis, expiry date identification, successful date extraction, format validation initiation, user dashboard access request, predictive analytics generation trigger, software requirements document review initiation, prototype review trigger, and test case generation initiation.
7.	Exception	Exceptions include failed user authentication, image upload errors, OCR processing issues, expiry date detection failures, unexpected text in date extraction, incorrect date format in validation, unauthorized dashboard access attempts, predictive analytics generation errors, inconsistencies in requirements review, usability

		problems in prototype review, and challenges in defining test scenarios for test case generation.
8.	Priority	Primary actors have main role in monitoring the application. user authentication and image upload having high priority due to their foundational role, followed by OCR processing and expiry date detection. Date extraction, format validation, dashboard access, and predictive analytics have moderate priority. Requirements and prototype reviews have lower priority, while test case generation is of higher priority for ensuring system functionality.
9.	Template issues	Template issues for the given project might include ensuring consistent user interface design, addressing compatibility with various devices and browsers, handling potential network connectivity issues during OCR processing, managing data security and privacy concerns, and optimizing performance for quick response times during dashboard access and analytics generation.

RELATION OF USECASE:

- The user interacts with use cases such as Authenticate User, Upload Image, Process OCR, Detect Expiry Date, Extract Date, Validate Date Format, View Learning Performance Dashboard, and Generate Predictive Insights. The "Detect Expiry Date" includes "Extract Date," "Validate Date Format" includes "Extract Date," and "View Learning Performance Dashboard" includes "Generate Predictive Insights." "Validate Date Format" extends "Detect Expiry Date," and "Generate Predictive Insights" extends "View Learning Performance Dashboard."

UML DIAGRAM:

PROBLEM STATEMENT:

The current process of unit conversion across categories is inefficient and time-consuming, requiring users to manually calculate and convert values. This problem is particularly acute for individuals who frequently need to convert units for work or personal use. The Unit Converter App project aims to address this problem by developing an application that can accurately convert units across categories on both Android and iOS platforms. The app will feature intuitive interfaces, reliable conversion formulas, real-time results, and history storage. Thorough testing and validation will be conducted to ensure accuracy and enhance user experience. By providing users with an efficient tool for unit conversion, the app can improve productivity and reduce errors.

IDENTIFICATION OF ACTORS:

- User
- admin

USECASE TEMPLATE:

SI. NO.	Template	Description
1.	usecase	Perform Unit Conversion Display Unit Options Display Conversion History Select Source Category Select Target Category Enter Value Perform Conversion Retrieve History Display History
2.	Primary actor	<ul style="list-style-type: none"> • User
3.	goals of an usecase	The goal of these use cases is to provide users with a user-friendly and efficient tool for performing unit conversions across various categories. Users should be able to easily select source and target units, input values, perform conversions, and view their conversion history, enhancing their experience and accuracy in

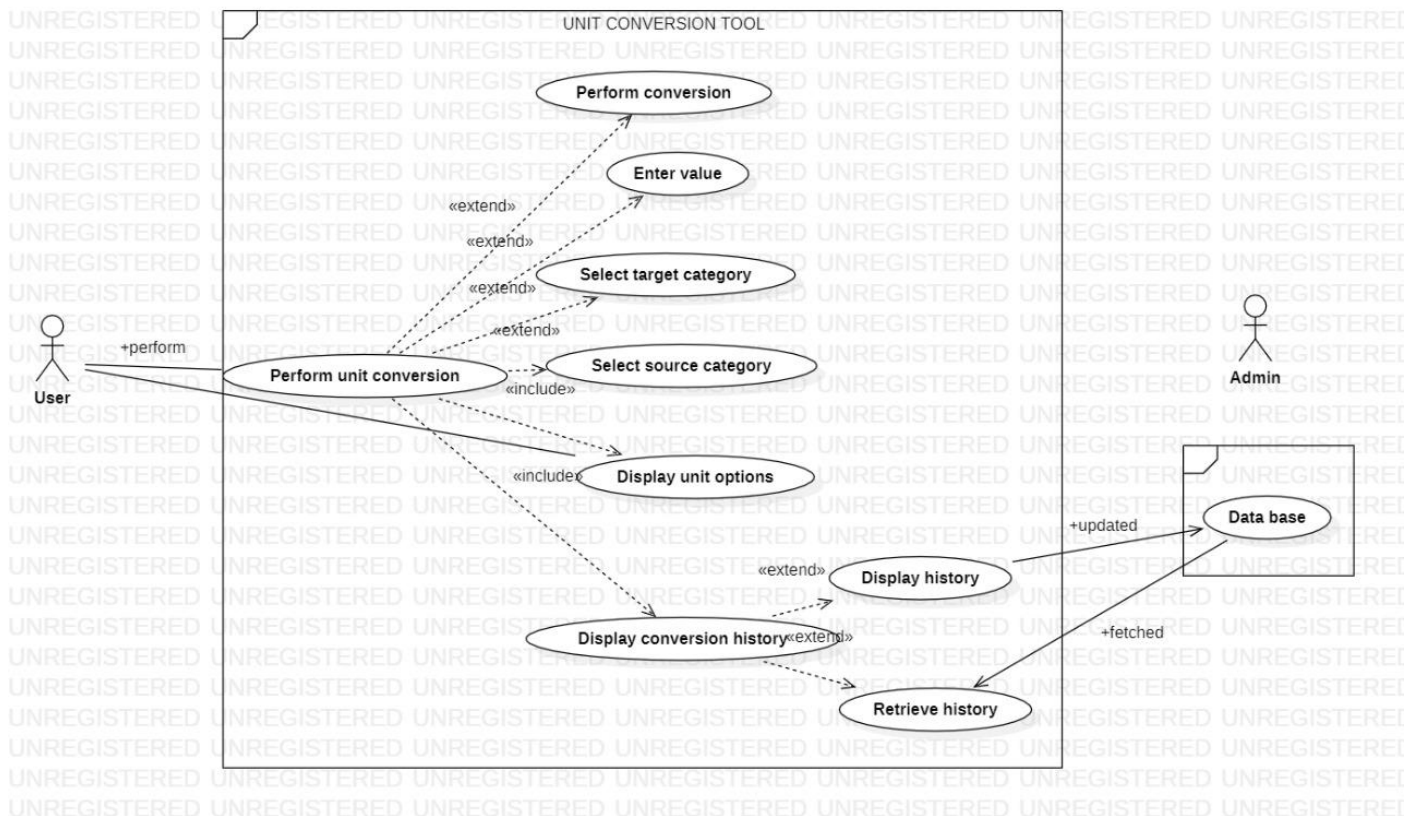
		unit conversions.
4.	Pre-condition	The precondition for these use cases is that the user has installed the Unit Converter App on their Android or iOS device and has access to an internet connection (for currency conversion and potential updates). The user should also be familiar with basic unit categories and their associated units.
5.	scenario	The user uses the Unit Converter App to perform unit conversions, view unit options, access conversion history, select source and target categories, enter values, and retrieve and display conversion history details.
6.	Triggers	User triggers the app to perform unit conversion, view unit options, access conversion history, select source and target categories, enter values, and retrieve and display conversion history details.
7.	Exception	Exceptions occur if the app encounters errors in retrieving unit options, performing conversions, or accessing conversion history. Additionally, incorrect inputs or unavailable data sources may lead to exceptions during the conversion process.
8.	Priority	The priority of these use cases depends on their importance in delivering a functional and user-friendly unit converter app. Typically, "Perform Unit Conversion," "Display Unit Options," and "Select Source Category" might have higher priorities to ensure core functionality, followed by other features like history retrieval and display. The priority can be determined based on user needs and project objectives.
9.	Template issues	Template issues encompass UI consistency, responsive design challenges, real-time currency rate accuracy, reliable conversion formulas, efficient data storage, localization for diverse users,

		offline mode considerations, data security, validation measures, error handling, user feedback integration, and performance optimization.
--	--	---

RELATION OF USECASE:

User interacts with Perform Unit Conversion, Display Unit Options, Display Conversion History; Perform Unit Conversion includes Display Unit Options, Display Conversion History, Select Source Category, Select Target Category, Enter Value, Perform Conversion; Display Conversion History includes Retrieve History, Display History; Perform Unit Conversion extends Select Source Category, Select Target Category, Enter Value, Perform Conversion.

UML DIAGRAM:



PROBLEM STATEMENT:

In the face of growing environmental concerns, there is a lack of easily accessible and engaging platforms to educate individuals about pressing ecological issues and encourage them to adopt

sustainable behaviors. Current solutions often fail to provide a comprehensive way for users to access educational content, engage in actionable challenges, participate in meaningful discussions, personalize their experience, and perform practical unit conversions, all within a single user-friendly application. This gap highlights the need for an innovative Environmental Awareness App that seamlessly integrates these features, empowering users to enhance their environmental knowledge, foster positive actions, and actively contribute to a more sustainable future.

IDENTIFICATION OF ACTORS:

- User
- admin

USECASE TEMPLATE:

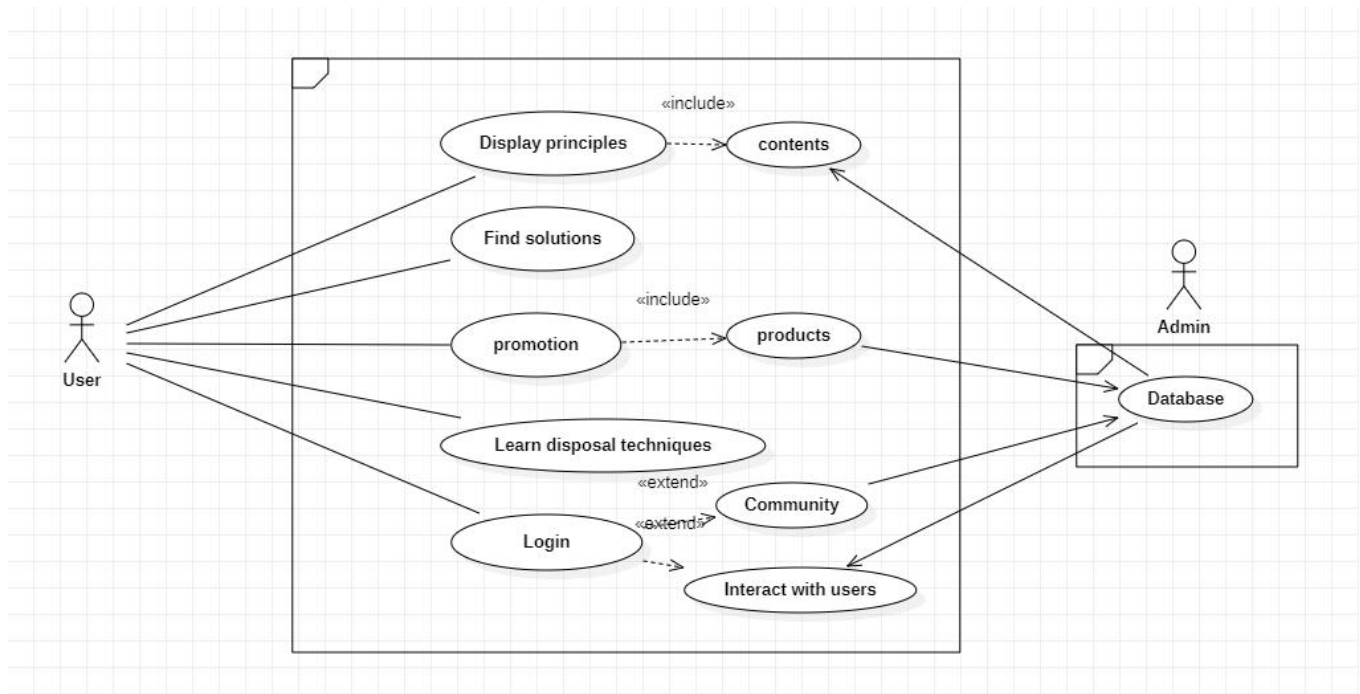
SI. NO.	Template	Description
1.	usecase	Display green principles Find solution Learn disposal techniques Promote principles Login Community Interact with other users
2.	Primary actor	<ul style="list-style-type: none"> • User
3.	goals of an usecase	The "View Content" use case aims to provide educational information on environmental topics. "Take Challenge" encourages eco-friendly actions through interactive tasks. "Participate in Discussion" facilitates user engagement in environmental conversations. "Customize Profile" allows personalization for enhanced user experience. "Access Conversion Tool" offers convenient unit conversions, supporting practical needs.
4.	Pre-condition	For the "View Content" use case, the precondition necessitates the user to have initiated the app while being connected to the internet. In the case of "Take Challenge," users need to have actively chosen a challenge within the "Engage" section. Similarly, for "Participate in Discussion," users must be within the "Engage" area and have selected a discussion topic. The "Customize Profile" use case demands that

		users be logged in and accessing the "Profile" section.
5.	scenario	In the "View Content" scenario, a user opens the app, selects the "Learn" section, and clicks on the "Climate Change" category. They read an informative article about reducing carbon footprint, gaining insights into eco-friendly practices.
6.	Triggers	The Environmental Awareness App is triggered when users launch it and select categories like "Learn" or "Engage" to access educational content, challenges, discussions, and tools, facilitating engagement and learning on environmental issues.
7.	Exception	Content and features may be inaccessible if the user lacks an internet connection. Server Issues: Interruptions in content delivery or updates may occur due to server-related problems.
8.	Priority	High priority, as it provides foundational educational content for users to learn about environmental topics. Significant, as it enhances user engagement by allowing personalization and tailored preferences. Substantial, as interactive challenges actively promote eco-friendly actions and engagement. Important, as it fosters community involvement and knowledge sharing among users. Moderate priority, offering a valuable utility but not central to the app's core goals.
9.	Template issues	Compatibility, performance, integration, and security hurdles. Inconsistent design, poor navigation, and accessibility problems. Scope creep, changing requirements, and communication gaps. Miscommunication, delays, inefficient workflow, and collaboration issues. Inadequate testing, regression issues, and user acceptance shortcomings.

RELATION OF USECASE:

The user interaction with use cases such as Viewing green principles, Finding solutions for the existing problems, can learn about various waste disposal techniques, and then user can view the promoted products and their implementation and their uses for the problems. And the user can login to the community and then interact with the other users and communicate about problems faced in the locality.

UML DIAGRAM:



OBSERVATION	(20)	
RECORD	(05)	
TOTAL	(25)	

Result:

Thus ,the creation of uml usecase diagram is successfully completed and verified