# Introduction

## Problem Identification

The task is to create a personalized news recommendation system that provides personalized news article recommendations to users.

* **Personalized News Article Recommendations**
* **User Profile Management**
* **Article Management**
* **User Management**
* **Interaction Tracking**
* **Article Categorization**

## Solution

* **Personalized News Article Recommendations – simple recommendation algorithm used.**
* **User Profile Management – can check details, update details and preferences.**
* **Article Management – can add edit and delete articles.**
* **User Management – can reset passwords of users and delete accounts.**
* **Interaction Tracking – read, liked and skipped articles of users are saved.**
* **Article Categorization – newly added articles are categorized by keyword extraction.**

## Timeline

Week 1: UML Design and Core Structure (Use case, class, activity, sequence diagrams). Basic user management and article fetching.

Week 2: Implement File Handling and Exception Handling. Set up concurrency for multi-user support.

Week 3: Integrate AI/ML model for recommendations. Test classification and recommendation accuracy.

Week 4: Finalize all functionalities, add logging, error handling, and improve UI (console-based or basic GUI).

Week 5: Comprehensive testing (unit testing, concurrency testing, file integrity checks, user acceptance testing).

## Project Scope

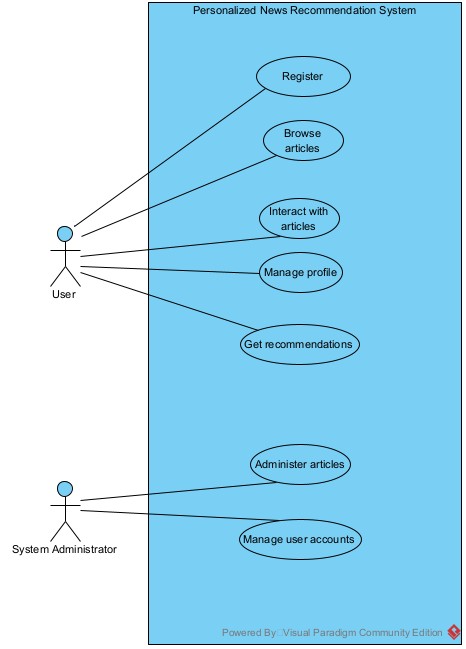
1. Support concurrency for multiple users.
2. Provides recommendations based on user preferences while dynamically updating preferences with user interactions.
3. The system has one administrator whose details are manually added to the database initially **(Username: admin123, Password: admin123)**.
4. Uses a python script to extract keywords from articles.
5. The articles stored in the database are initially assigned a category, only newly added articles will be automatically categorized.

# Requirement Analysis

Stakeholders with the requirements are mentioned.

|  |  |
| --- | --- |
| Stakeholder | Requirement |
| End User | Receive personalized news articles based on their preferences and interactions. |
| Register in the system with personal details and preferences. |
| Browse and interact with all articles in the system. |
| Manage profile. |
| System Administrator | Article management: add, edit, delete. |
| User management: reset password, deactivate user account. |
| Developer | Build a system capable of handling multiple users. |
| Store required data in databases and manage data retrieval and updates. |
| Implement functional recommendation algorithm. |
| Conduct necessary testing to ensure system integrity. |

Use Case Diagram



## Use case descriptions

**Use Case: Register**

Section: Main

System: Personalized News Recommendation System

Actor(s): User

Purpose: To allow a new user to create an account in the system.

Overview: A new user enters their username, password, full name and the data is validated and stored, then the user is given a confirmation about their registration.

Pre conditions: The user is previously unregistered.

**Typical Course of Events**

|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. Chooses to register. |  |
|  | 1. Prompts user for necessary details. |
| 1. Enters new username. |  |
| 1. Enters new password. |  |
| 1. Enters full name. |  |
| 1. Enters preferences. |  |
|  | 1. Creates new user account and stores the information. |
|  | 1. Confirms successful registration. |
|  | 1. Notify the user that they can login. |

**Alternative Courses**

* If a user enters a username that is already used by another user, the system prompts for a new username.
* If the password is invalid, the system prompts the user for a valid password.
* If the name is invalid, the system prompts the user for a valid name.

**Use Case: Browse Articles**

Section: Main

System: Personalized News Recommendation System

Actor(s): User

Purpose: To allow users to check and access all articles apart from recommendations.

Overview: A user can view all available articles if needed.

Pre conditions: The user is already registered and has logged into the system.

**Typical Course of Events**

|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. Chooses to browse articles. |  |
|  | 1. Display all articles(title). |
|  | 1. Prompt article id. |
| 1. Enter article id. |  |
|  | 1. Display selected article. |

**Alternative Courses**

* If there are issues when displaying articles, the user will be notified of the issue.

**Use Case: Interact with Articles**

Section: Main

System: Personalized News Recommendation System

Actor(s): User

Purpose: The user can read articles and then like or skip articles depending on their interest.

Overview: After reading an article the user can like the article or skip the article, this will help the recommendation system learn user preferences.

Pre conditions: The user is already registered and has logged into the system.

**Typical Course of Events**

|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. User selects an article from the article list. |  |
|  | 1. Loads the article content. |
| 1. Reads the article. |  |
|  | 1. Add article to read list. |
| 1. If user likes or skips the article. |  |
|  | 5.1) Records the interaction for recommendations. |
| 5.2) Else, no interaction. |  |

**Alternative Courses**

* If there is an issue in loading the article content, the user will be informed of the issue.

**Use Case: Manage Profile**

Section: Main

System: Personalized News Recommendation System

Actor(s): User

Purpose: The user can update their personal information or preferences.

Overview: The user can change their name, username, password and their preferred news article categories.

Pre conditions: The user is already registered and has logged into the system.

**Typical Course of Events**

|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. Chooses to manage their profile. |  |
|  | 1. Display account information. |
|  | 1. Prompt account detail changes. |
| 1. Updates necessary details. |  |
|  | 1. Prompt for preference changes. |
| 1. Updates necessary preference changes. |  |
|  | 1. Save changes. |
|  | 1. Confirmation message sent. |

**Alternative Courses**

* If the newly entered details are invalid, the user is asked to re-enter the details.

**Use Case: Get Recommendations**

Section: Main

System: Personalized News Recommendation System

Actor(s): User

Purpose: Provides recommended articles to the user based on their reading and interaction history.

Overview: The user can check the recommended articles to easily find articles to their liking without having to browse several articles. The recommended articles will change depending on user reading history and interactions.

Pre conditions: The user is already registered and has logged into the system.

**Typical Course of Events**

|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. Chooses to get Recommendations. |  |
|  | 1. Analyzes user interaction history and preferences. |
|  | 1. Update preferences. |
|  | 1. Generate a list of recommended articles. |
|  | 1. Displays the articles. |

**Alternative Courses**

* The user will be notified of any issues regarding recommendation or article displaying.

**Use Case: Administer Articles**

Section: Main

System: Personalized News Recommendation System

Actor(s): System Administrator

Purpose: Allows the system administrator to add new articles, update the title or the content of existing articles or completely remove existing articles.

Overview: The administrator has the choice to add, update, or remove articles. If the add option is used, a new article title and content can be added. After choosing the edit option, any article can be edited by choosing the article number. After choosing the delete option, any article can be removed by choosing the article number.

Pre conditions: The administrator is already registered as an administrator and has logged into the system.

**Typical Course of Events**

|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. Chooses to Administer Articles. |  |
|  | 1. Prompts for add, edit or delete option. |
| 1. If add is selected, |  |
|  | 3.1) Prompts new article details. |
| 3.2) Enters new article title and content. |  |
|  | 3.3) Categorize article. |
|  | 3.4) Saves the article. |
| 1. If edit is selected, |  |
|  | 4.1) Display article titles with article numbers. |
|  | 4.2) Prompts for article number. |
| 4.3) Enters article number. |  |
|  | 4.4) Display article title and content. |
|  | 4.5) Prompts changes. |
| 4.6) Make changes. |  |
|  | 4.7) Saves changes. |
| 1. If delete is selected, |  |
|  | 5.1) Display article titles with article numbers. |
|  | 5.2) Prompts for article number. |
| 5.3) Enters article number. |  |
|  | 5.4) Deletes article. |
|  | 1. Notify user. |

**Alternative Courses**

* If invalid article number is entered for editing or deleting, the administrator is asked to re-enter a valid number.
* If there is an issue during adding, editing or deleting, the administrator is informed and asked to re-perform the action.

**Use Case: Manage User Accounts**

Section: Main

System: Personalized News Recommendation System

Actor(s): System Administrator

Purpose: To manage user accounts, which includes resetting passwords and deactivating user accounts.

Overview: The system administrator views user accounts and user details. Depending on the requirement the passwords can be reset or accounts can be deactivated.

Pre conditions: The administrator is already registered as an administrator and has logged into the system.

**Typical Course of Events**

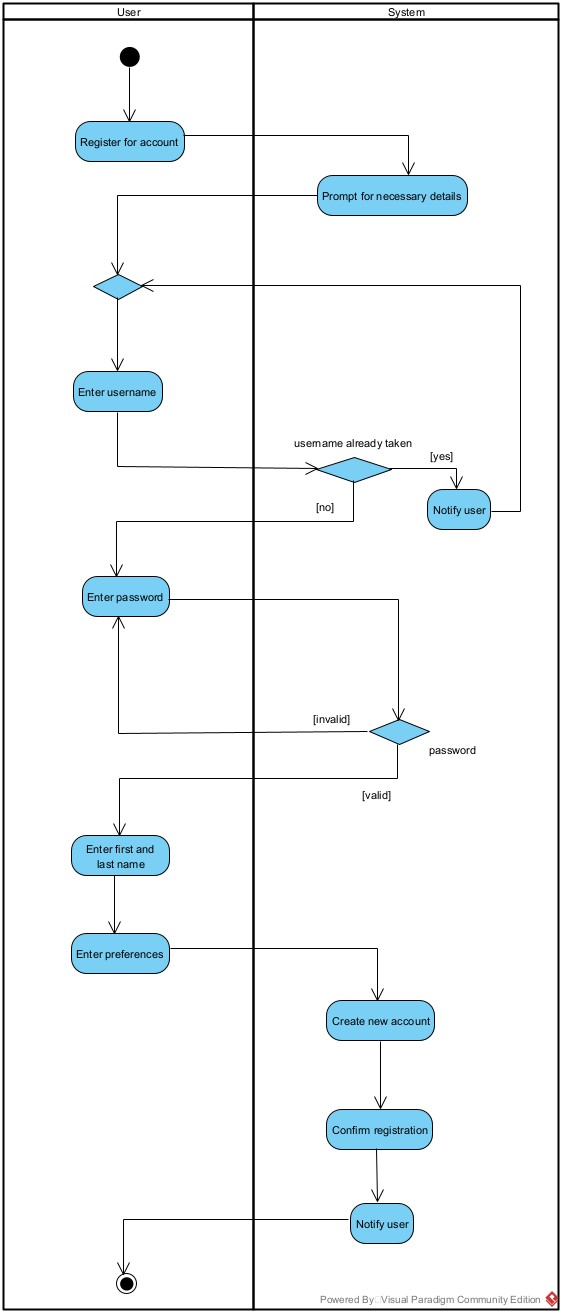
|  |  |
| --- | --- |
| Actor Action | System Response |
| 1. Chooses to Manage accounts. |  |
|  | 1. Prompts for reset password option or deactivate account option. |
| 1. If reset password selected, |  |
|  | 3.1) Displays a list of registered users. |
|  | * 1. Prompts for user ID. |
| 3.3) Enters user ID. |  |
|  | 3.4) Generates new password. |
|  | 3.5) Displays new password. |
| 1. Else, account deactivation selected, |  |
|  | 4.1) Displays a list of registered users. |
|  | 4.2) Prompts for user ID. |
| 4.3) Enters user ID. |  |
|  | 4.4) Prompts for re-confirmation. |
| 4.5) Confirms de-activation. |  |
|  | 4.6) Deactivates user profile. |

**Alternative Courses**

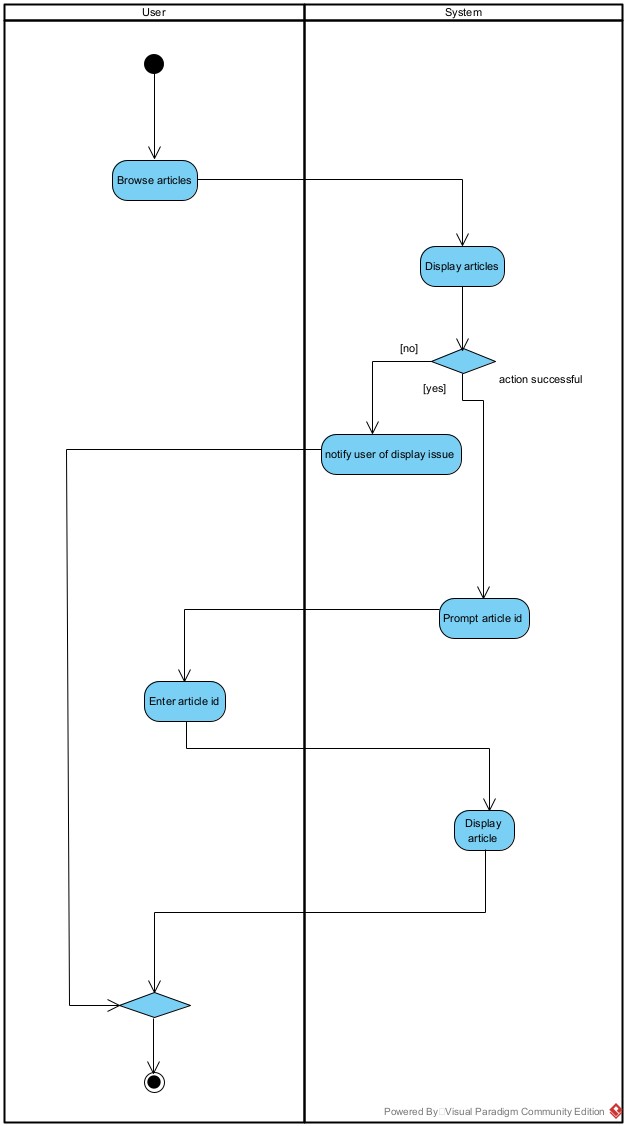
* If invalid user ID is entered, the system re-prompts for a valid user ID.

# Activity Diagrams

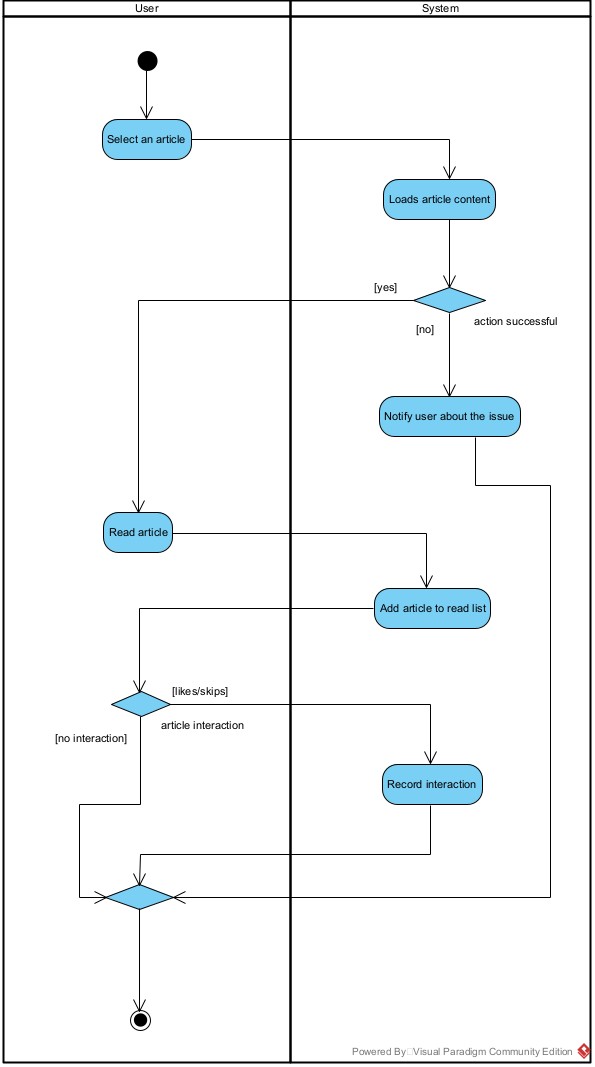
## Register



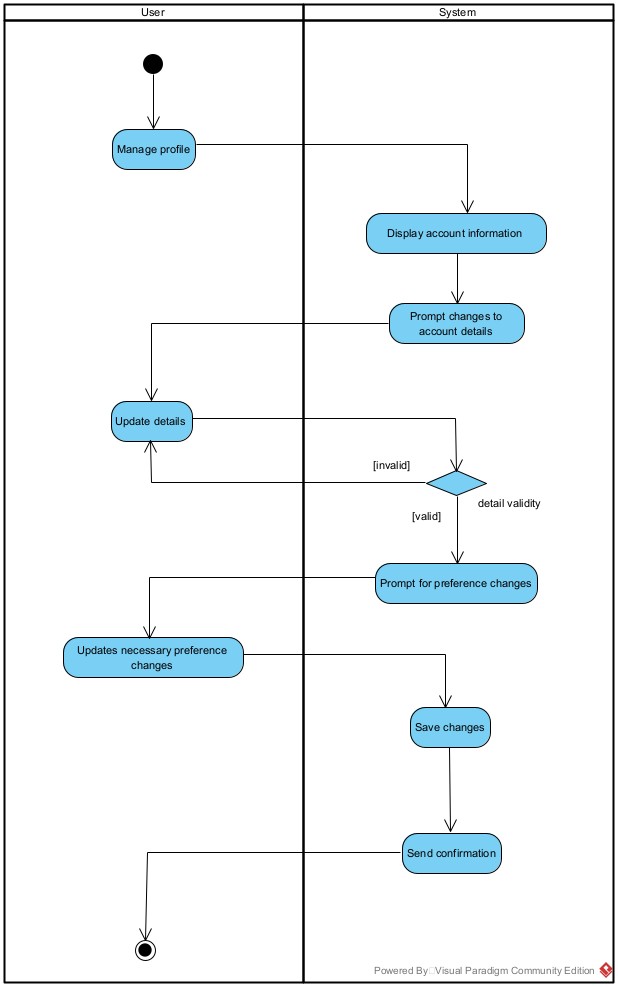
## Browse Articles



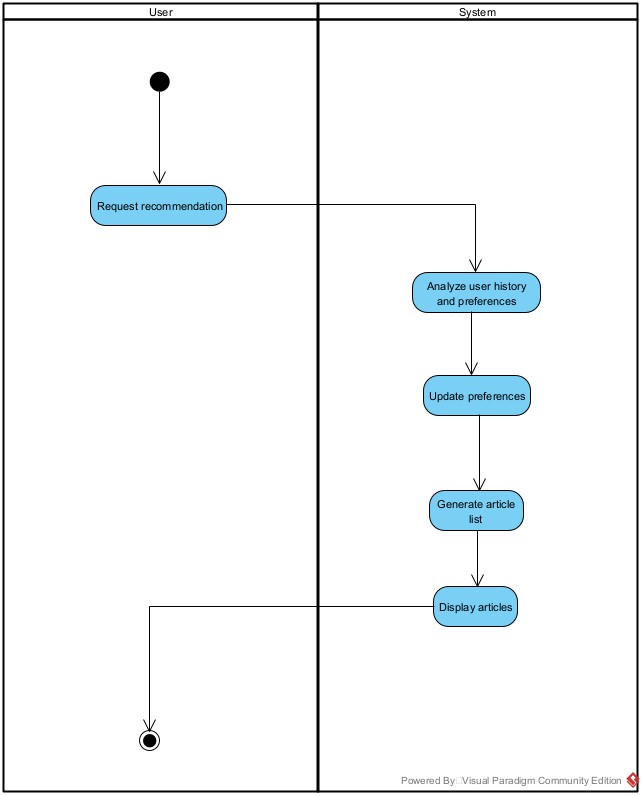
## Interact with Articles



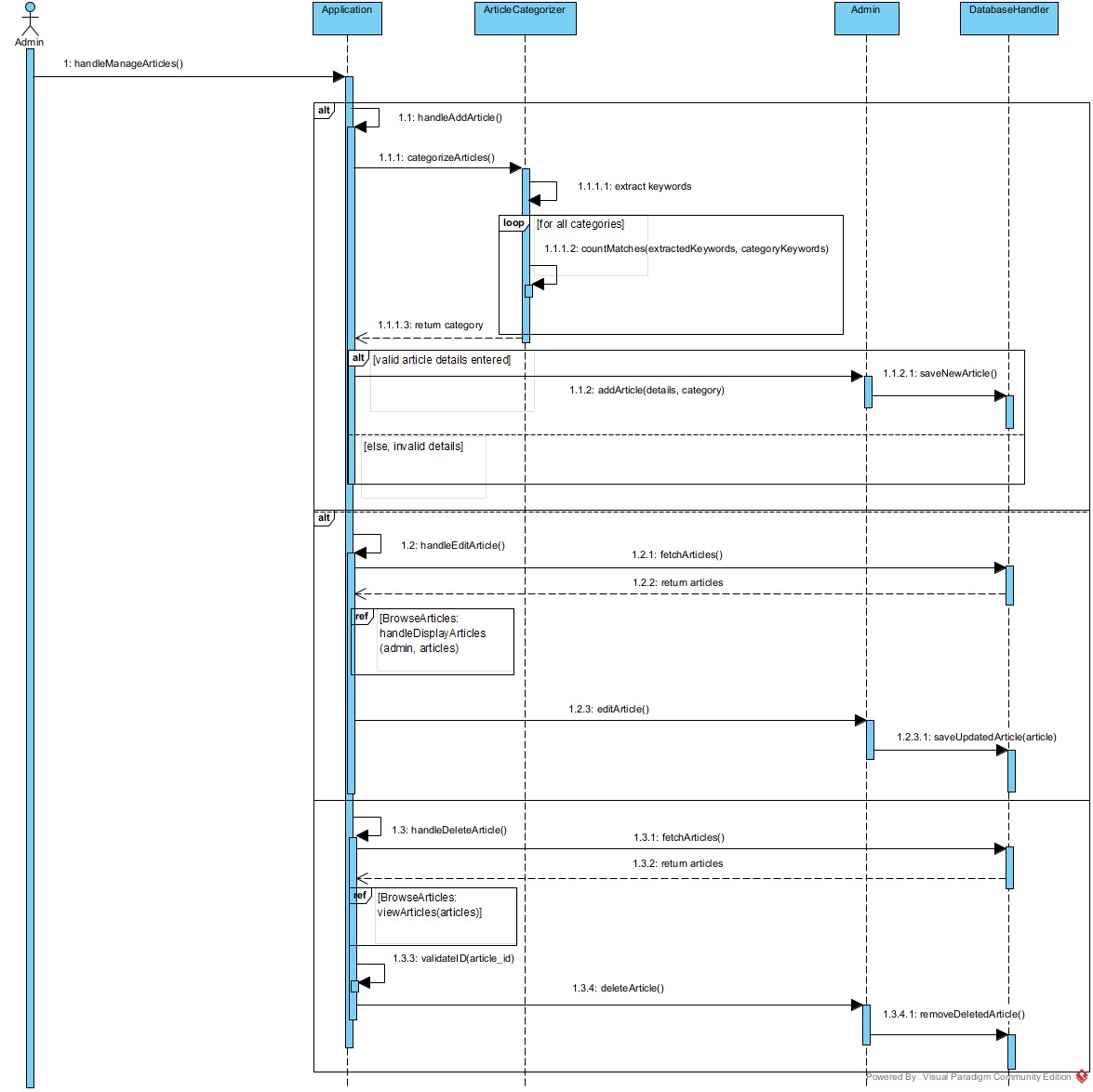
## Manage Profile



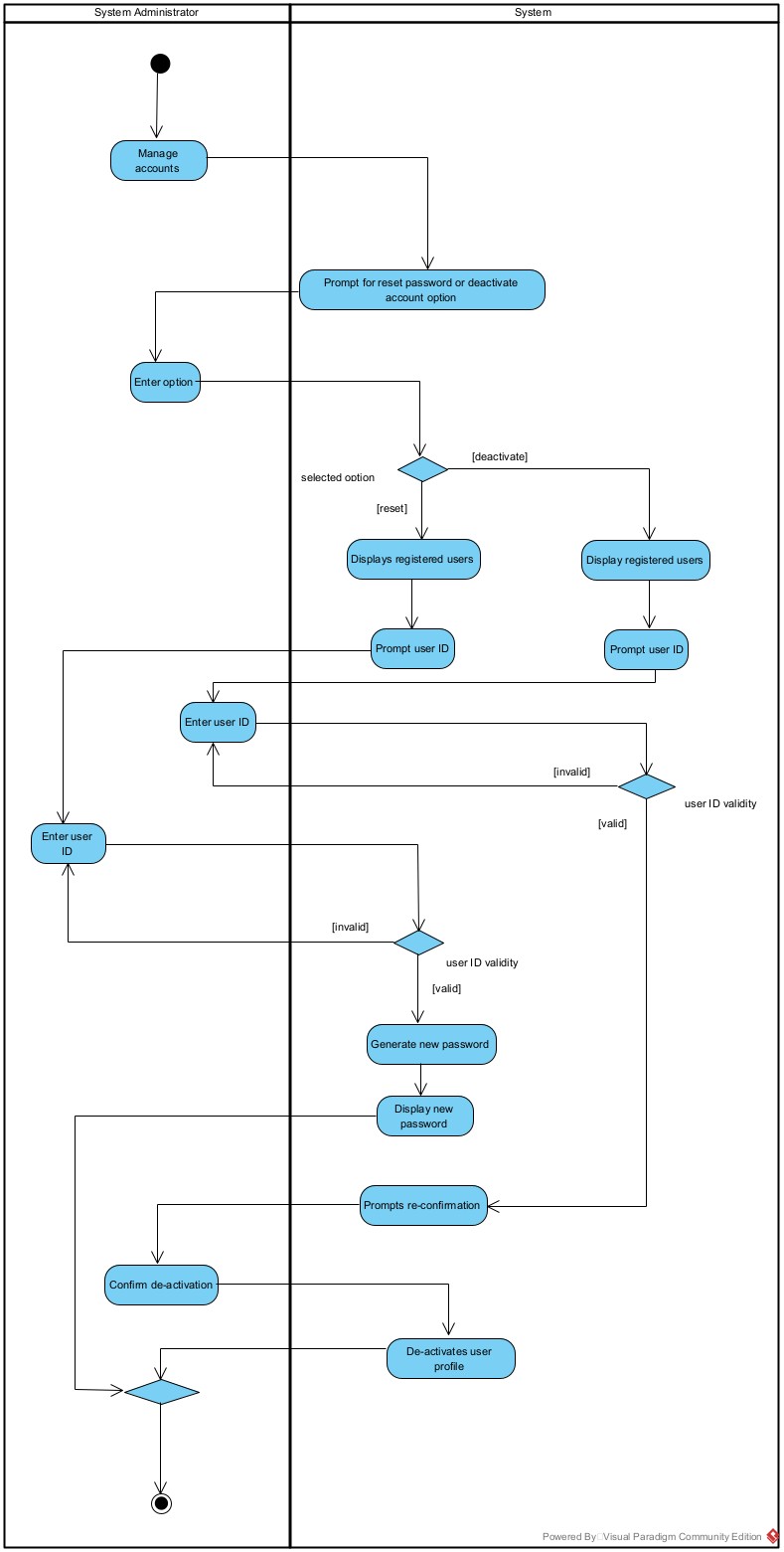
## Get Recommendations



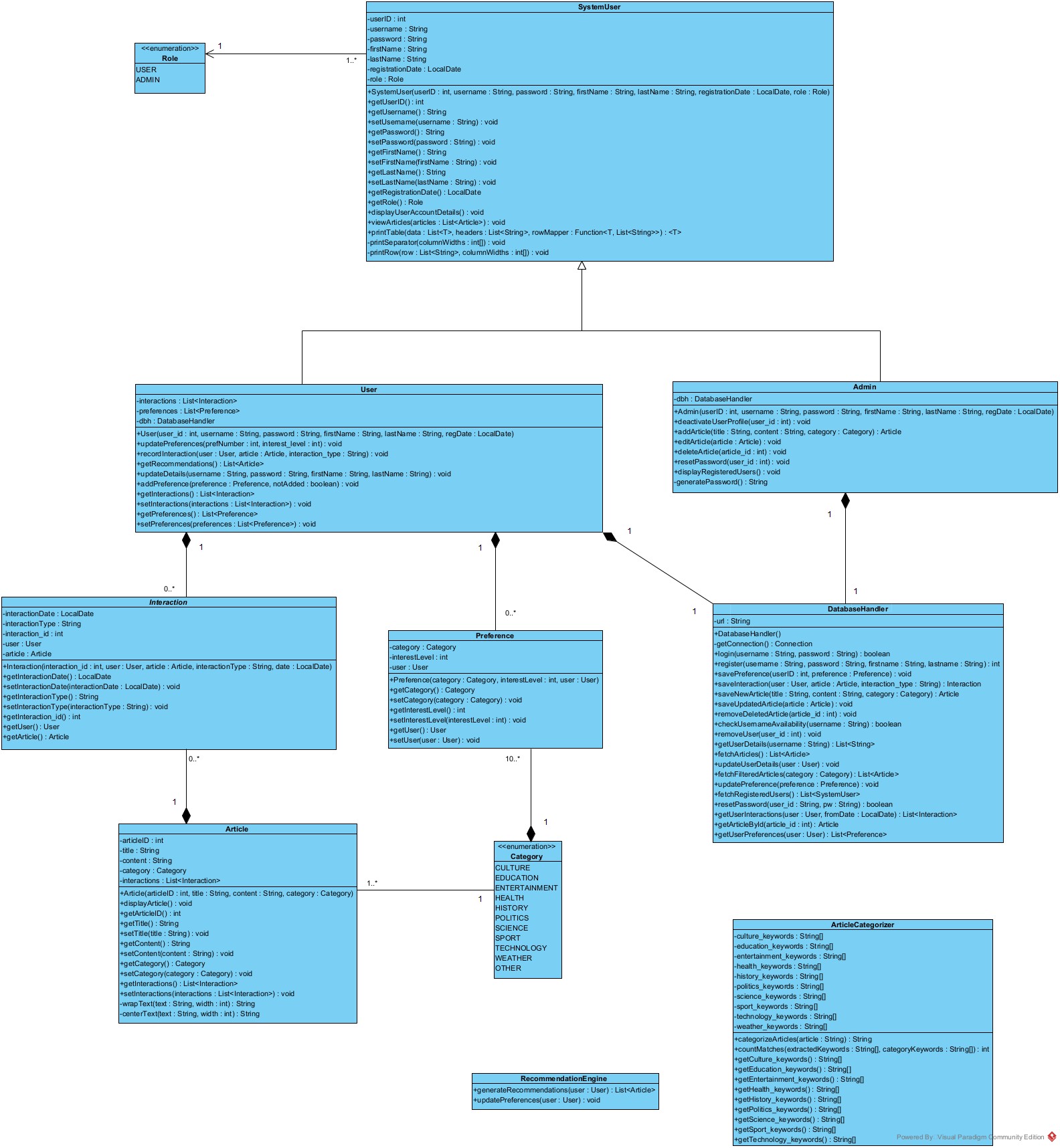
## Administer Articles



## Manage User Accounts



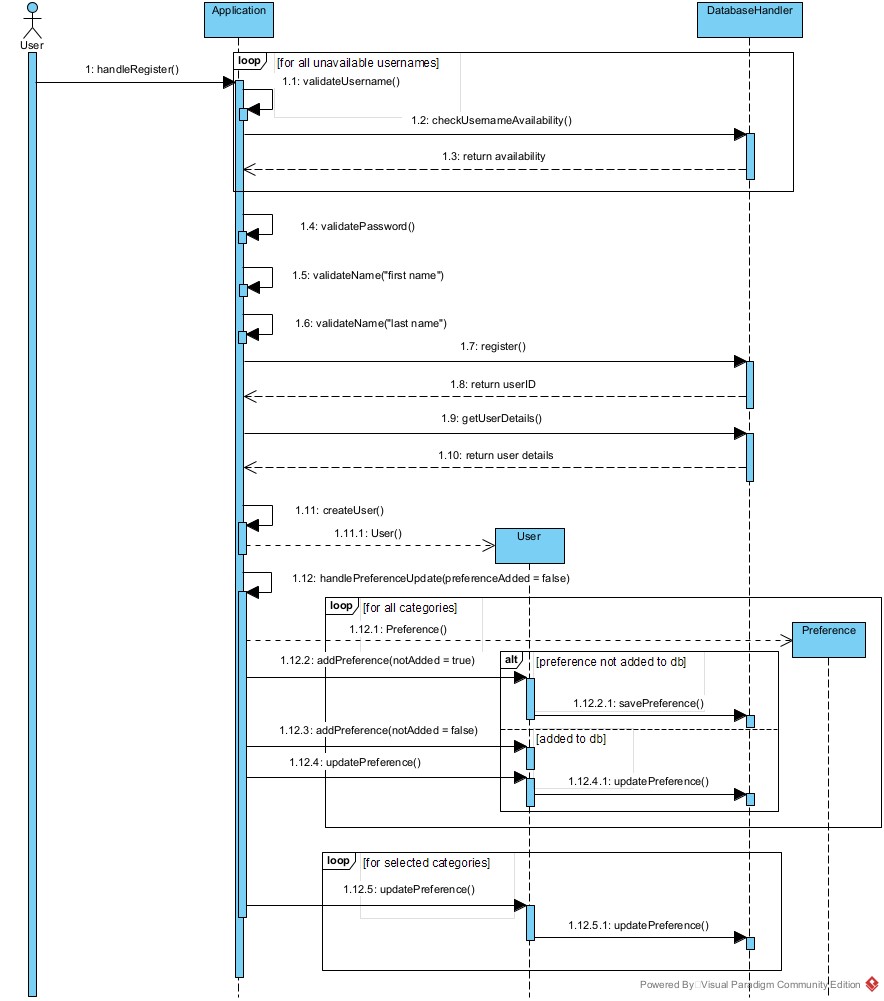
# Class Diagram



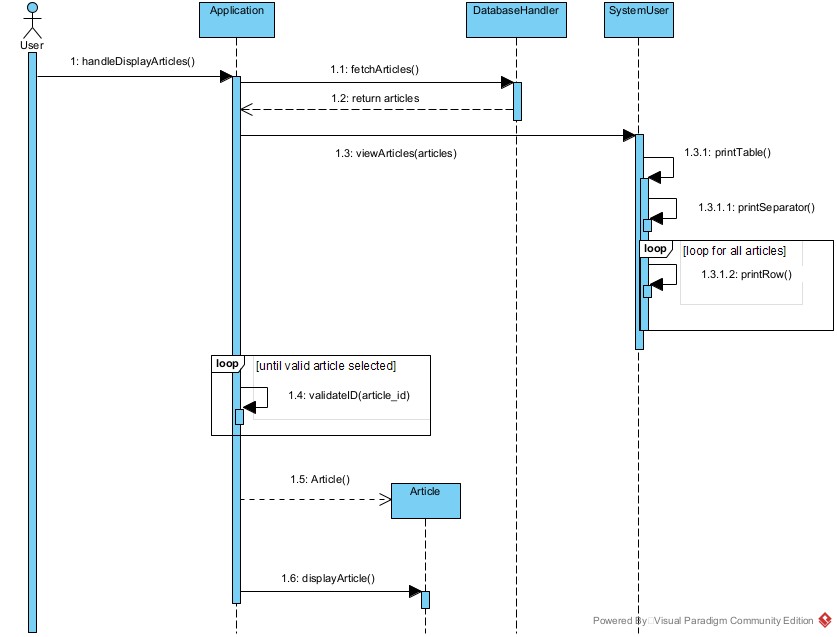
The ArticleCategorizer and RecommendationEngine classes are utility classes designed to provide core functionalities like article categorization and recommendation generation. They are invoked by other classes (e.g., User and Admin) when required but do not maintain direct associations with other classes in the system.

# Sequence Diagrams

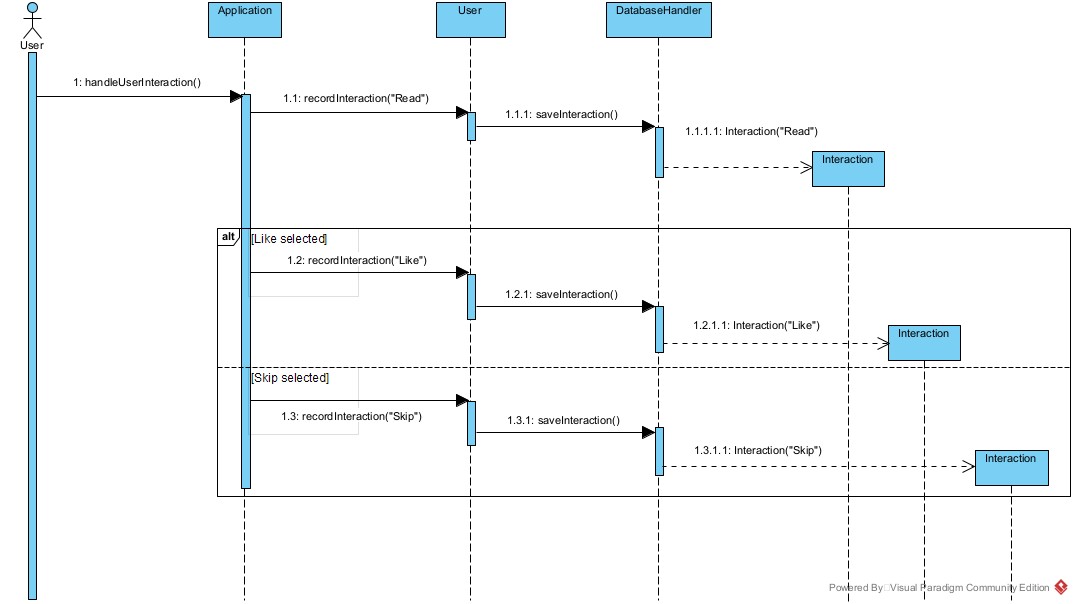
## Register



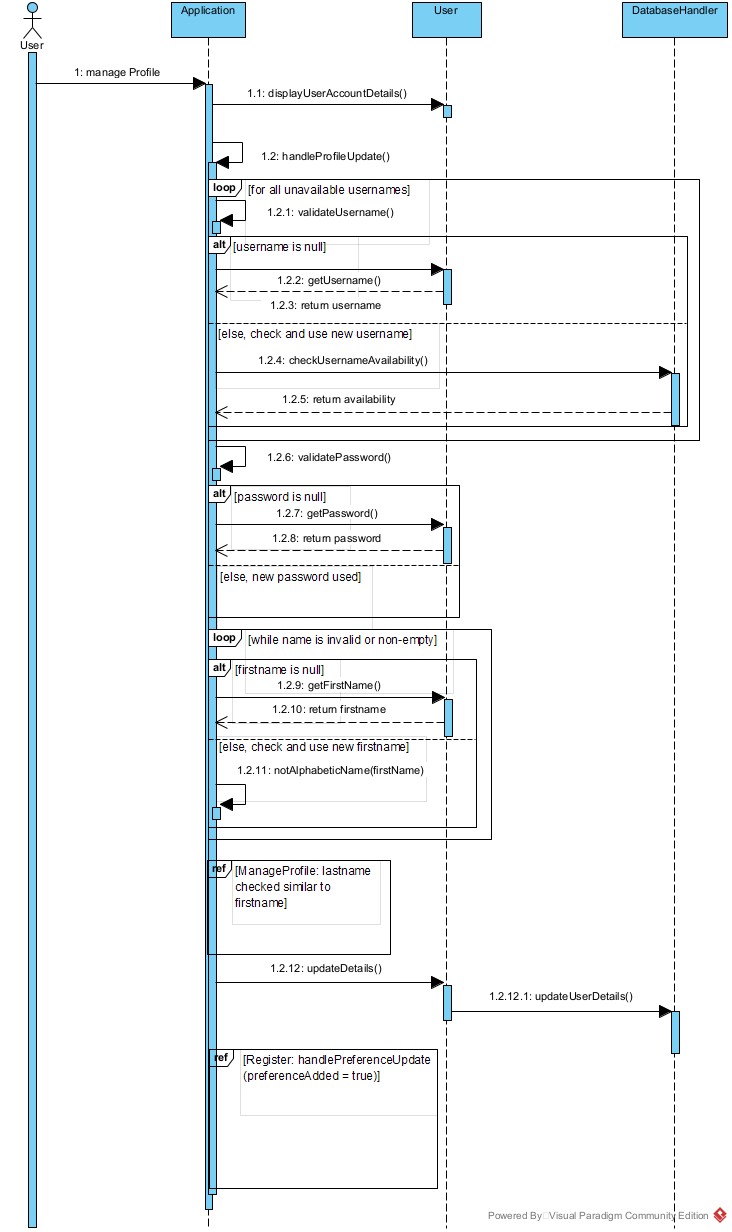
## Browse Articles



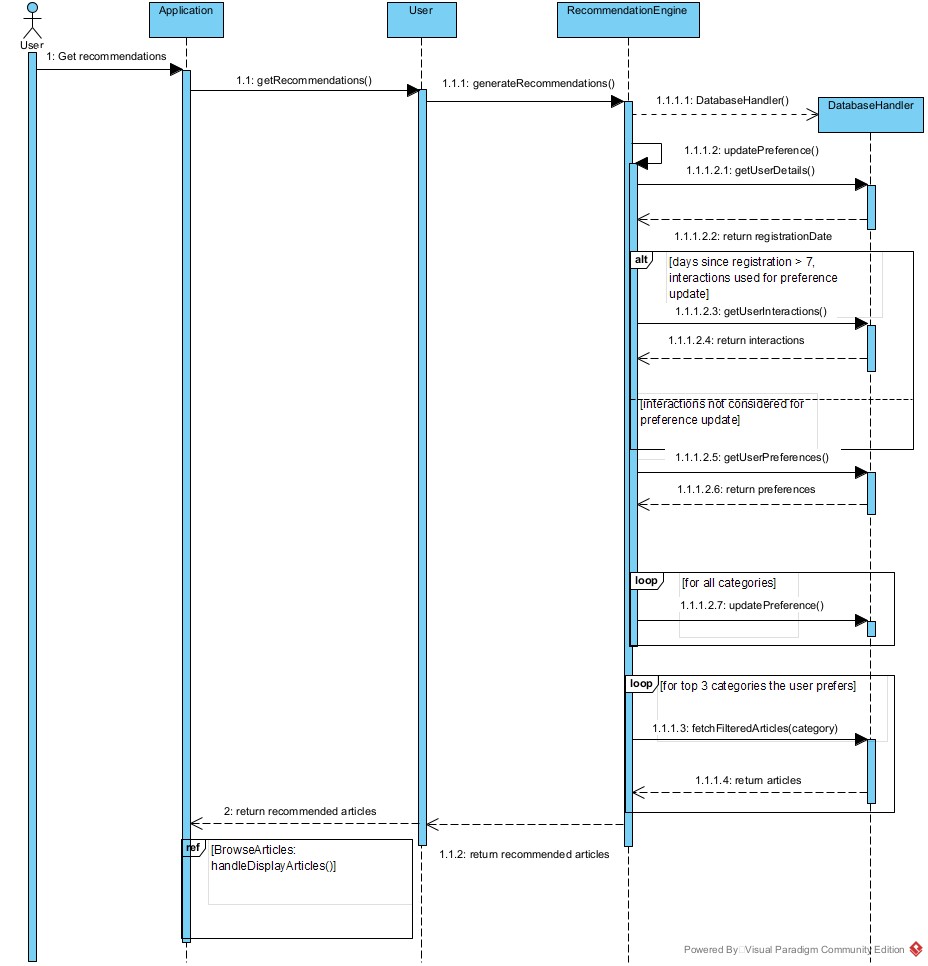
## Interact with Articles



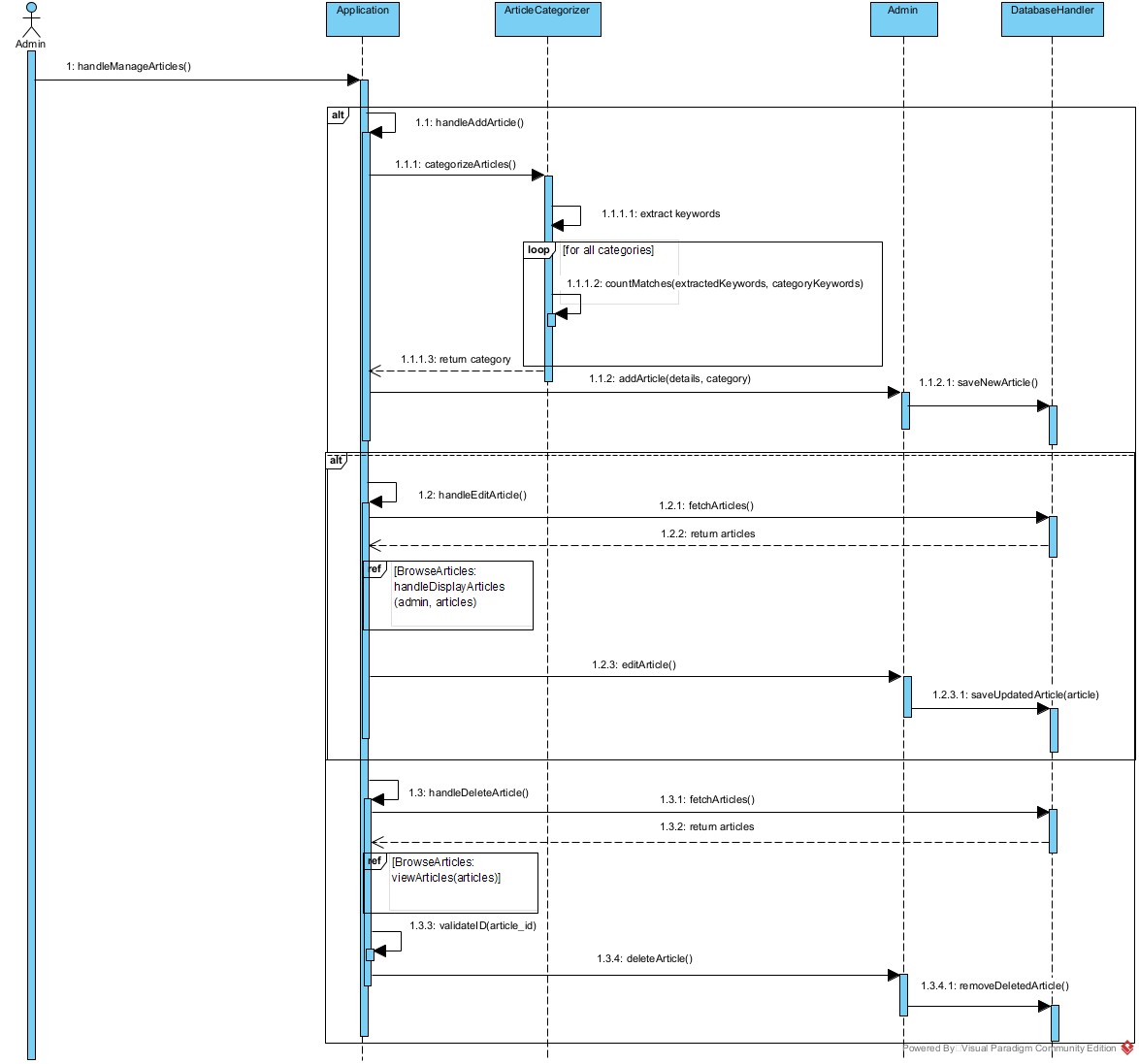
## Manage Profile



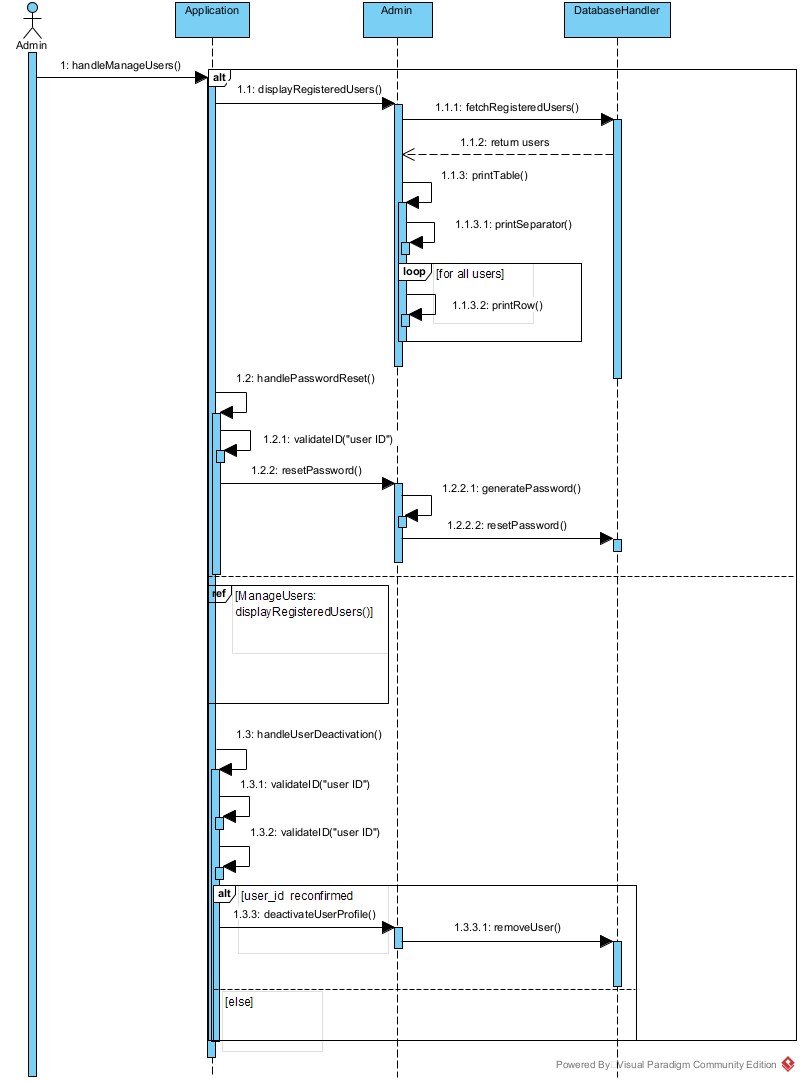
## Get Recommendations



## Administer Articles

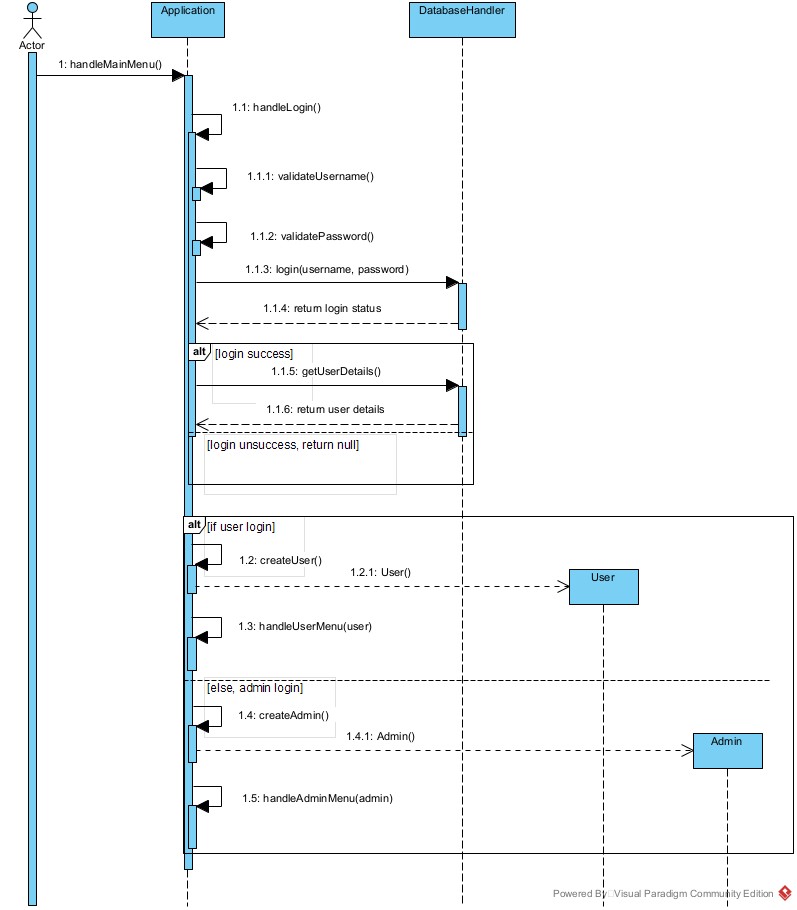


## Manage User Accounts



## Login

Optional sequence diagram to show the login flow.

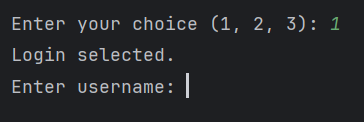


# Testing and evaluation

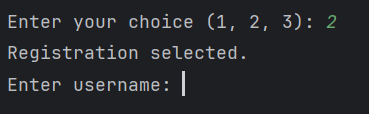
## Unit Testing

### Main Page

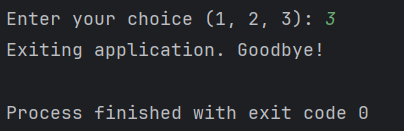
1. Login



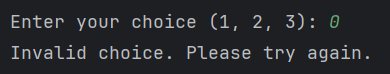
1. Register

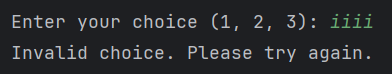


1. Exit



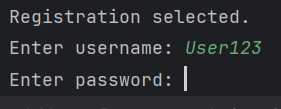
1. Invalid



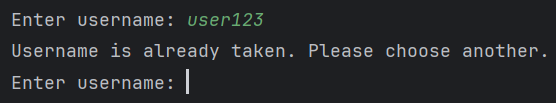


### Register Page

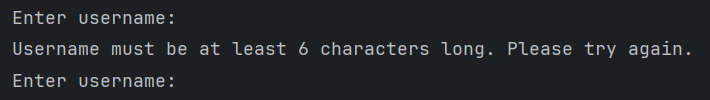
1. Username-Valid



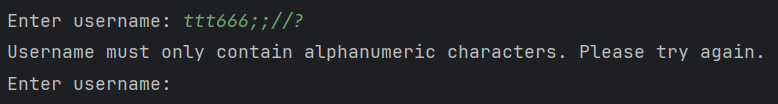
1. Username-Already taken



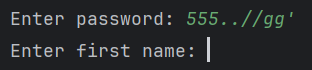
1. Username-Empty



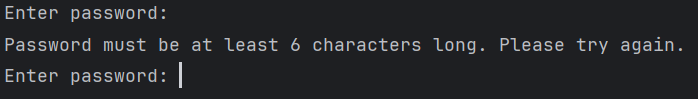
1. Username-Invalid



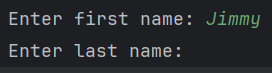
1. Password-Valid



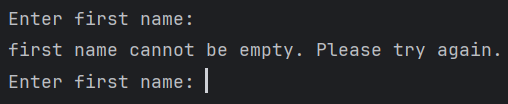
1. Password-Empty



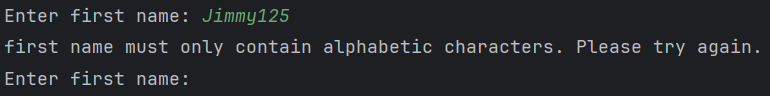
1. Firstname-Valid



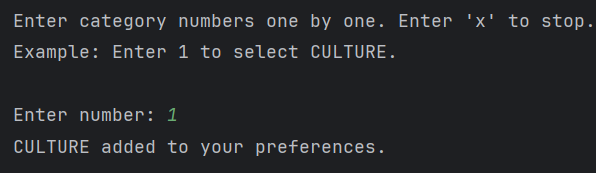
1. Firstname-Empty



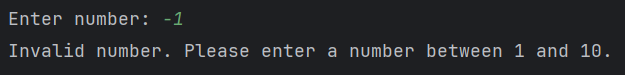
1. Firstname-Non-Alphabetic



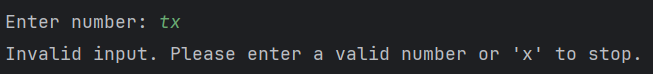
1. Similar testing for last name.
2. Category-Valid



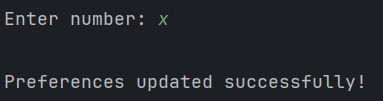
1. Category-Out-of-Range



1. Category-Invalid

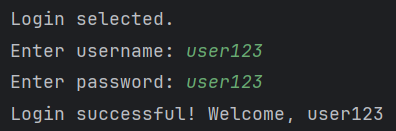


1. Category-Exit



### Login Page

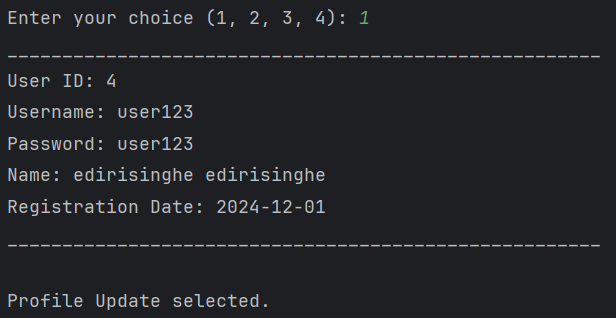
1. Username checked as before.
2. Password checked as before.
3. Username and Password – Valid and Match



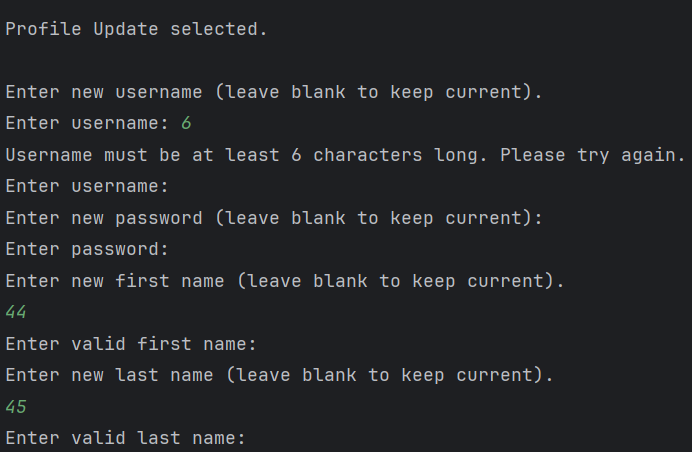
Based on credentials, user is redirected to USER menu or ADMIN menu.

### User menu

1. Profile Update



* 1. Manage Profile

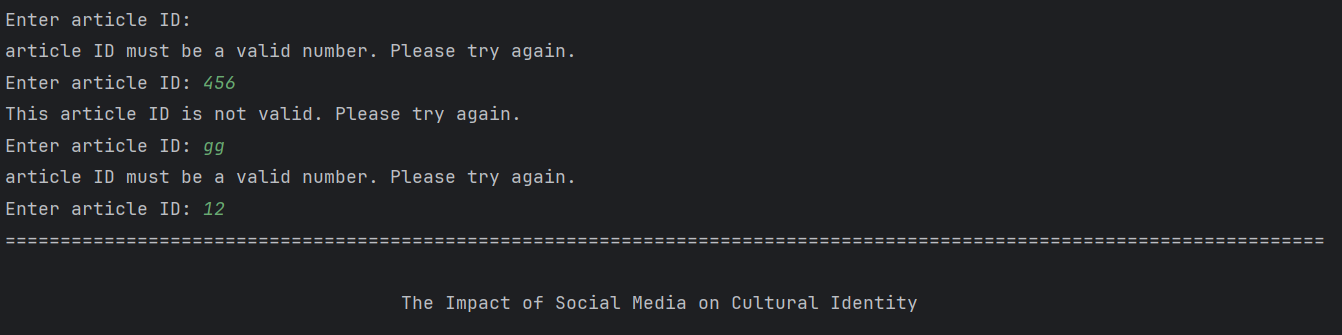


Same category checking as in register page.

1. View Articles

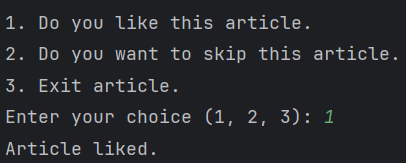


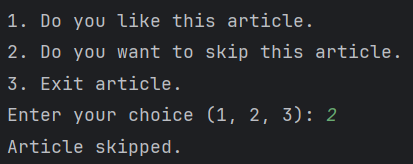
All articles displayed and asked for ID.

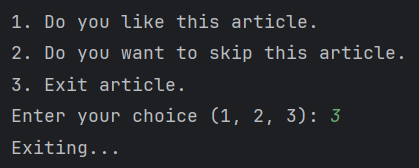


Article displayed.

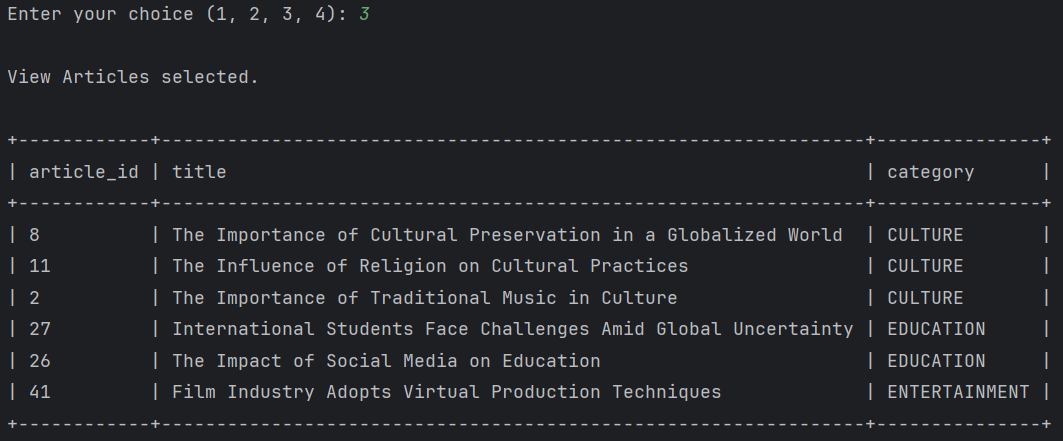
* 1. Interactions





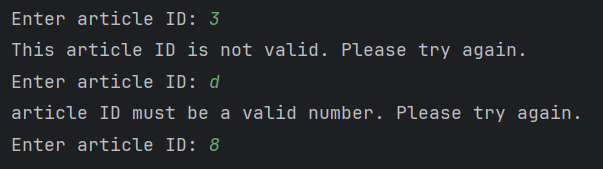


1. Get Recommendations

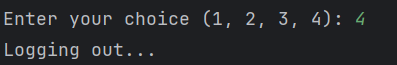


3.1) Validate article ID

The recommended articles can be viewed. Limited to the articles in recommended list.



1. Logout

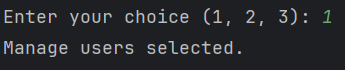


1. Invalid

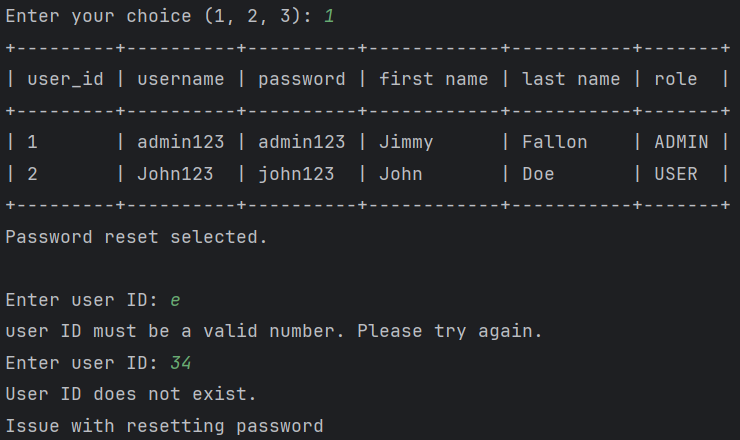


### Admin menu

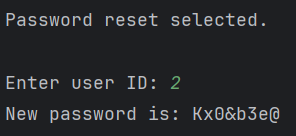
1. Manage Users



* 1. Reset Password



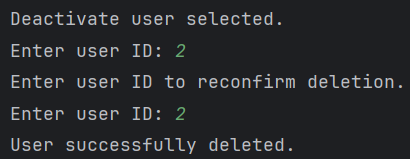
Password changed.



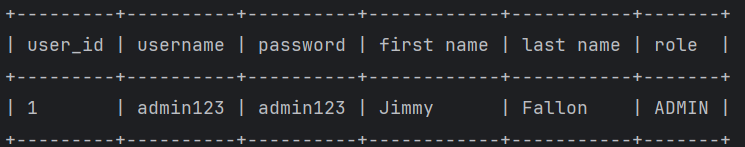


* 1. Deactivate Account

Validation added for user ID.



User removed.

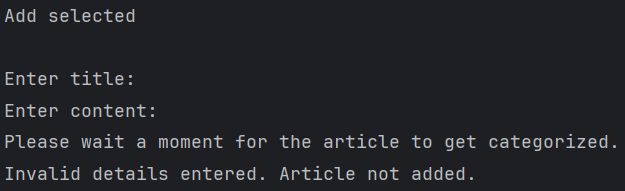


1. Manage Articles



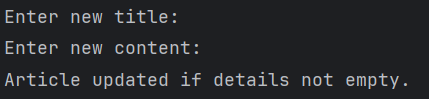
2.1) Add article

Validate details, categorize article and add if details are valid.



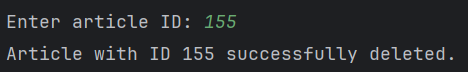
* 1. Edit article

Article details are edited if title or content is provided.

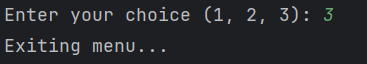


* 1. Delete article

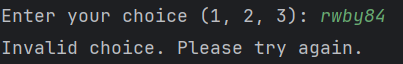
Validation added for article ID.



1. Logout



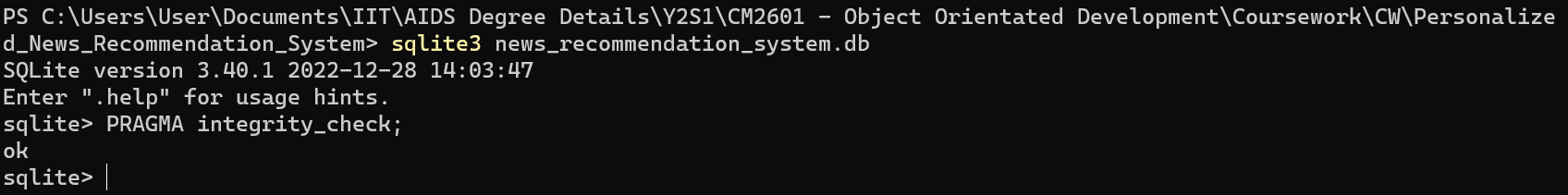
1. Invalid



Validations added for all pages, back options given for some pages as well. All functionality of the application is fully validated.

## File integrity checks

SQLite database integrity check in command prompt. Additional test case in project test suite.



## User acceptance testing

### Recommendation Engine Testing

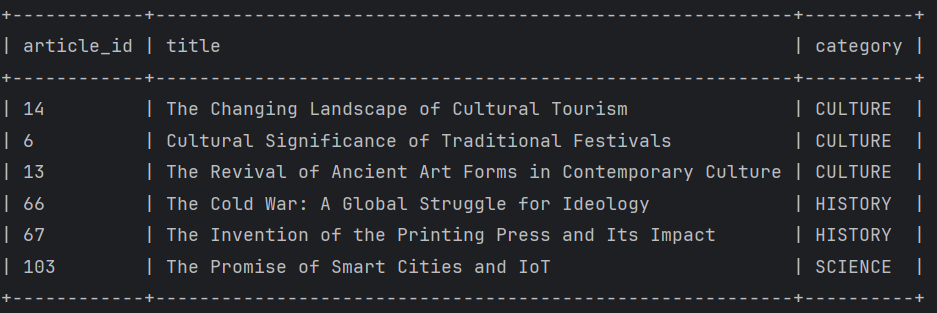
Test User



Initial user preferences during registration



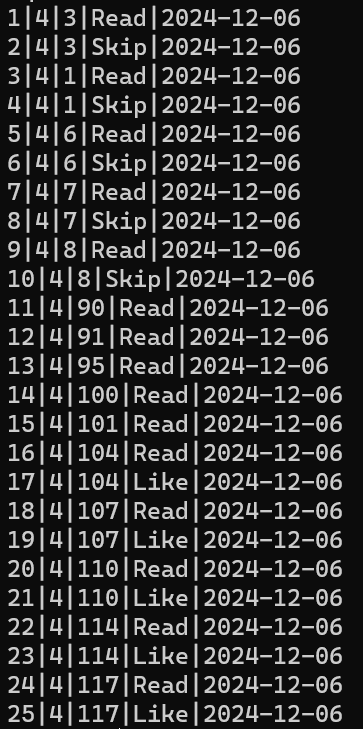
Articles recommended from selected categories.



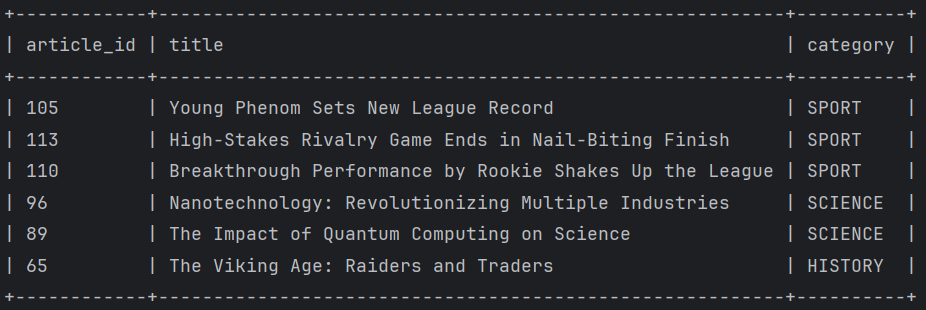
For users of more than 1 week, interactions used to update preferences and recommend articles.

Interactions

* 5 culture articles were skipped.
* 5 sport articles were liked.
* 5 science articles were read.



Recommended articles



A decay factor is used to keep the upper bound limited.

### Accuracy of the article categorizer

The code is available in the article categorizer class in the main method.



# References

# Appendix

#github commits