404 Not Found

Digital Newspaper Website
Use-Case Specification: Submit Article

Version 1.0

Revision History

Date	Version	Description	Author
1/7/2025	1.0	Specific about Submit Article use case	Nguyễn Tấn Lộc

©<Company Name>, 2025

Table of Contents

1. Use-Case Name: Submit Article **4**

1.1 Brief Description 4

2. Flow of Events **4**

2.1 Basic Flow 4

2.2 Alternative Flows 4

2.2.1 Cancel Submission 4

2.2.2 Notification failure 4

3. Special Requirements **4**

3.1 Email format 4

3.2 UI standard: 4

3.3 Performance 4

4. Preconditions **5**

4.1 Writer is authenticated 5

4.2 The article exist and is in draft state 5

4.3 Target editor role exist in system 5

5. Postconditions **5**

5.1 Article status = pending review 5

5.2 A review task is created and assigned to the specified editor 5

5.3 Notification record is saved (even if failed) 5

6. Extension Points **5**

6.1 Select editor 5

Use-Case Specification: Submit Article

1. Use-Case Name: Submit Article

1.1 Brief Description

Write selects a draft article and submits it to the editor for review. The system changes state to " pending review" and notifies the editor by mail and in-app notification.

2. Flow of Events

2.1 Basic Flow

1. Initiates actor: writer logs and navigates to My Articles.
2. The writer selects a draft article and clicks submit for approval.
3. The system displays a confirmation dialog showing the article and target editor.
4. Writer clicks confirm.
5. System:
    a. Updates article status to pending review
    b. Create a review task to the editor
    c. Send an email and in-app notification to editor with a link to an article
6. The system displays a " submission successful" message to the writer.

2.2 Alternative Flows

2.2.1 Cancel Submission

- In step 3, writer clicks cancel instead of confirm

2.2.1.1 An Alternative Subflow

- System closes the confirmation dialog
- No changes are made, article remains in draft
- Return to my draft view

2.2.2 Notification failure

- In step 5.3, the system fails to send mail or in-app notifications due to network issues or services error

2.2.2.1 An Alternative Subflow

1. The system logs the notification error
2. Still updates article states to pending review and create review task
3. Display warning "submission successful, but unable to notify editor, please contact them manually"

3. Special Requirements

3.1 Email format

- Notification email must include article title, summary, and a direct link to the review page

3.2 UI standard:

- Confirmation dialog and success/warning messages must follow the style.

3.3 Performance

- Status update and task must be completed in 1 second time normal load

©<Company Name>, 2025

4. Preconditions

4.1 Writer is authenticated
4.2 The article exist and is in draft state
4.3 Target editor role exist in system

5. Postconditions

5.1 Article status = pending review
5.2 A review task is created and assigned to the specified editor
5.3 Notification record is saved (even if failed)

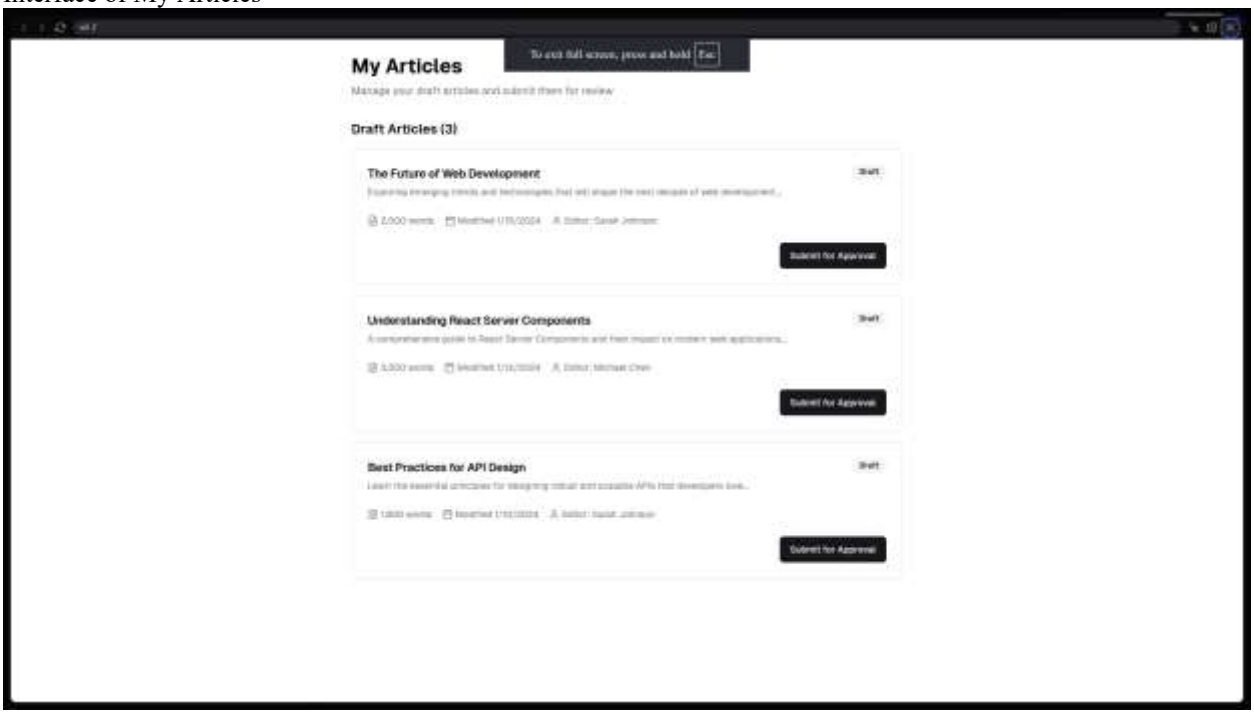
6. Extension Points

6.1 Select editor

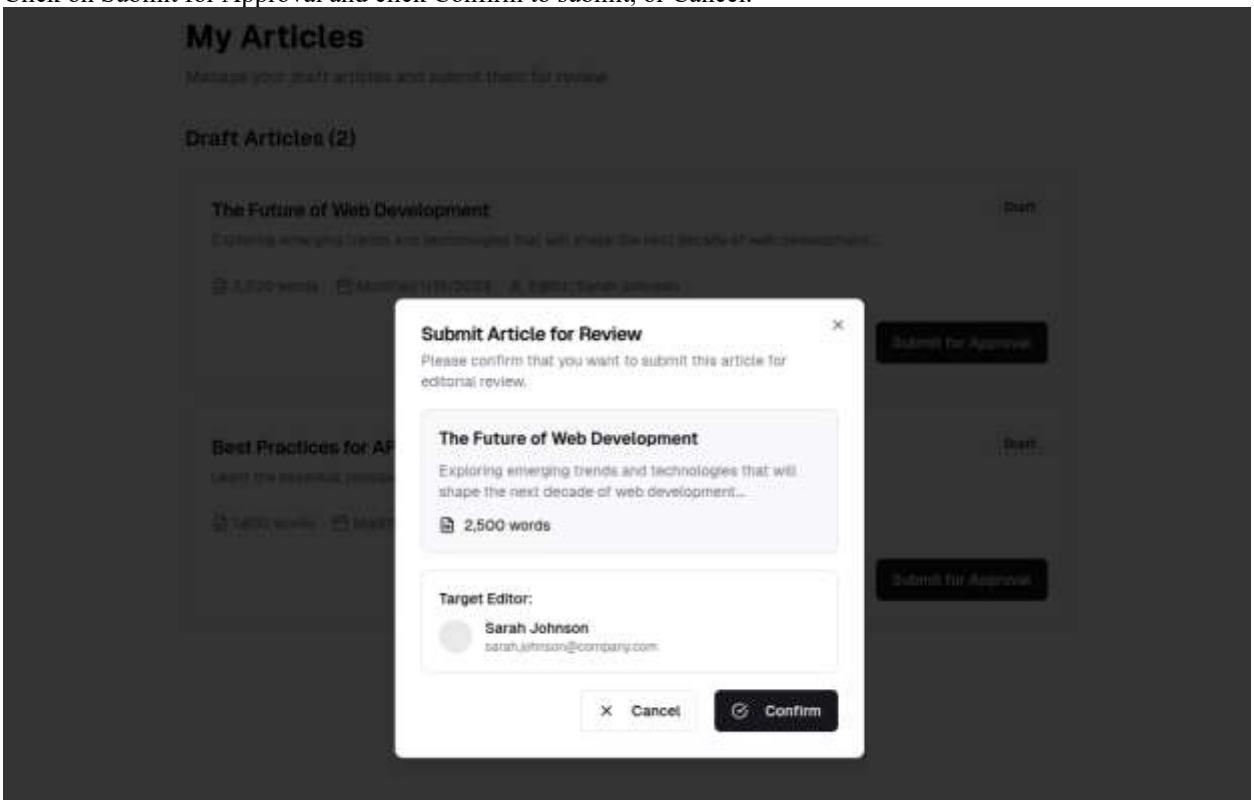
After step 3: writer may change the default editor before confimation.

7. Generate Interface

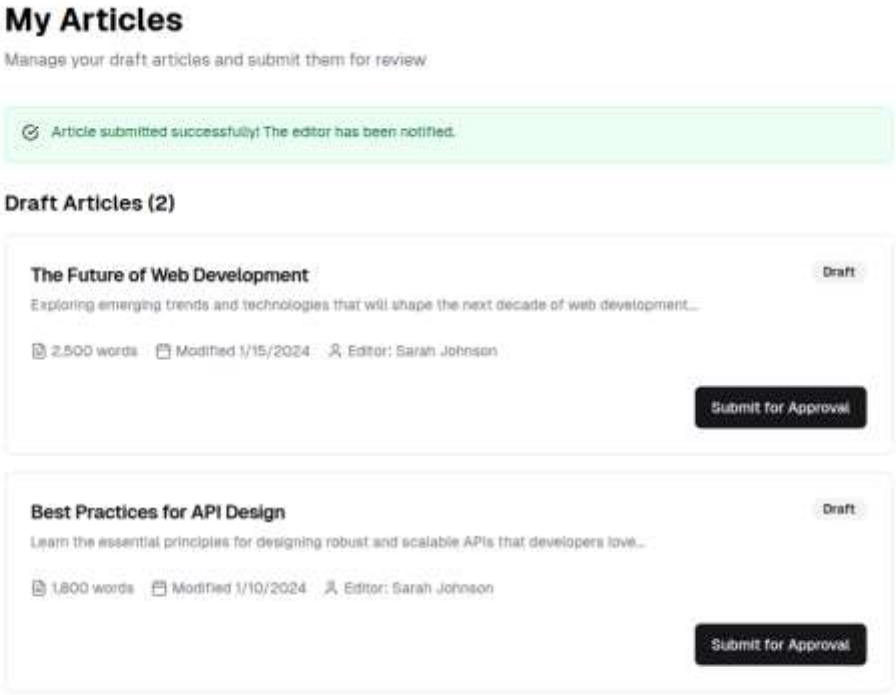
Interface of My Articles



Click on Submit for Approval and click Confirm to submit, or Cancel.



when we click ComFirm system will back to My Article page and with message: “article submitted successfully the editor has been notified”



When we click, the Cancel system will go back to the My Article page.

404 Not Found

Digital Newspaper Website
Use-Case Specification: Comment Article

Version 1.0

©<Company Name>, 2025

Revision History

Date	Version	Description	Author
12/7/2025	1.0	Specific about Comment Article use case	Nguyễn Tấn Lộc

©<Company Name>, 2025

Table of Contents

1. Use-Case Name: Comment Article 4
1.1 Brief Description 4
2. Flow of Events 4
2.1 Basic Flow 4
2.2 Alternative Flows 4
2.2.1 User not logged in 4
2.2.1.1 An Alternative Subflow 4
2.2.2 Empty or too long comment 4
2.2.2.1 An Alternative Subflow 4
2.2.3 Contains prohibited comments 4
2.2.3.1 An Alternative Subflow 4
2.2.4 Database save failed 5
2.2.4.1 An Alternative Subflow 5
3. Special Requirements 5
3.1 Profanity filter 5
3.2 Comment display time 5
3.3 Responsive UI 5
3.4 Edit/delete comment 5
3.5 Spam prevention 5
4. Preconditions 5
4.1 The user is authenticated 5
4.2 The article is in a “published” state and publicly viewable 5
4.3 The commenting feature is enabled for that article 5
5. Postconditions 5
5.1 A valid comment is stored in database, linked to the user and article 5
5.2 The comment appears on the article's page, visible to all readers. 5
5.3 The user may edit or delete the comment for a limited time (30 minutes) 5
6. Extension Points 5
6.1 Report comment 5
6.2 Like/Dislike 5
6.3 Nested comments 5

Use-Case Specification: Comment Article

1. Use-Case Name: Comment Article

1.1 Brief Description

This use case allows logged-in users to post a comment on a published article. The system validates the input, stores the comment, and displays it along with the user`s name and timestamp. The user can also edit or delete their own comments within a limited time.

2. Flow of Events

2.1 Basic Flow

7. The user navigates to an article page
8. The system displays the full article content
9. If the user is logged in, the system displays a comment input box below the article
10. The user types a comment and clicks the submit button
11. The system performs validation on the comment
    a. Ensures it is not empty
    b. Checks for prohibited words or content violations.
    c. Enforces the a maxim character limit (500 words)
12. If validation, the system
    a. Stores the comment in the database
    b. Associates the content with the article and the user`s ID
    c. Displays the comment below the article with the user`s name and timestamp
13. The user sees their comment appear in the comet section and is given the option to edit or delete (within allow time)

2.2 Alternative Flows

2.2.1 User not logged in

- At step 3, user is not logged in

2.2.1.1 An Alternative Subflow

1. The comment input is hidden
2. The system display a message: “you must be logged in to post a comment”
3. A login / register link is provided

2.2.2 Empty or too long comment

- At step 5, the comment is empty or exceeds the character limit

2.2.2.1 An Alternative Subflow

1. The system display an error message: "comment can not be empty or too long”
2. The user must rewrite the comment

2.2.3 Contains prohibited comments

- The comment contains profanity or banned words

2.2.3.1 An Alternative Subflow

1. The system blocks the comment and shows: “your comment contains prohibited content”
2. User is prompted to revise the comment and resubmit

2.2.4 Database save failed

- At step 6, the system fails to save the comment (e.g due to server/database error)

2.2.4.1 An Alternative Subflow

1. The system displays “unable to post comments at the moment. Please try again later”
2. The comment is not posted or stored

3. Special Requirements

3.1 Profanity filter

- The system must detect and block offensive, abusive or inappropriate language using a predefined list or AI-based moderation

3.2 Comment display time

- Approved comments should be visible within 1 second after submission

3.3 Responsive UI

- The comment input field and list must display correctly on all devices

3.4 Edit/delete comment

- Only author of a comment can edit or delete their own comments

3.5 Spam prevention

- The system should implement rate-limiting (e.g max 5 comment per minute per user)

4. Preconditions

4.1 The user is authenticated

4.2 The article is in a “published” state and publicly viewable

4.3 The commenting feature is enabled for that article

5. Postconditions

5.1 A valid comment is stored in database, linked to the user and article
5.2 The comment appears on the article's page, visible to all readers.
5.3 The user may edit or delete the comment for a limited time (30 minutes)

6. Extension Points

6.1 Report comment

- After step 7, other users can report a comment for moderation

6.2 Like/Dislike

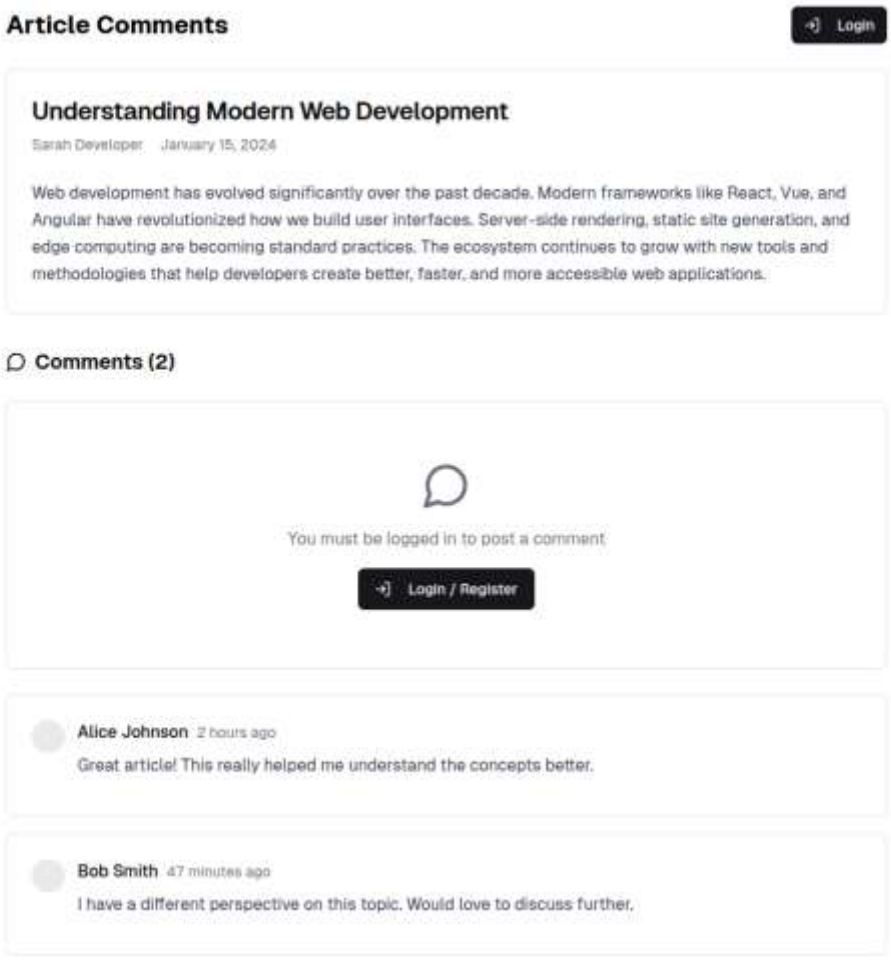
- After step 7, users may react to others`s comments using like/dislike buttons

6.3 Nested comments

- Step 4, the user can reply directly to another comment

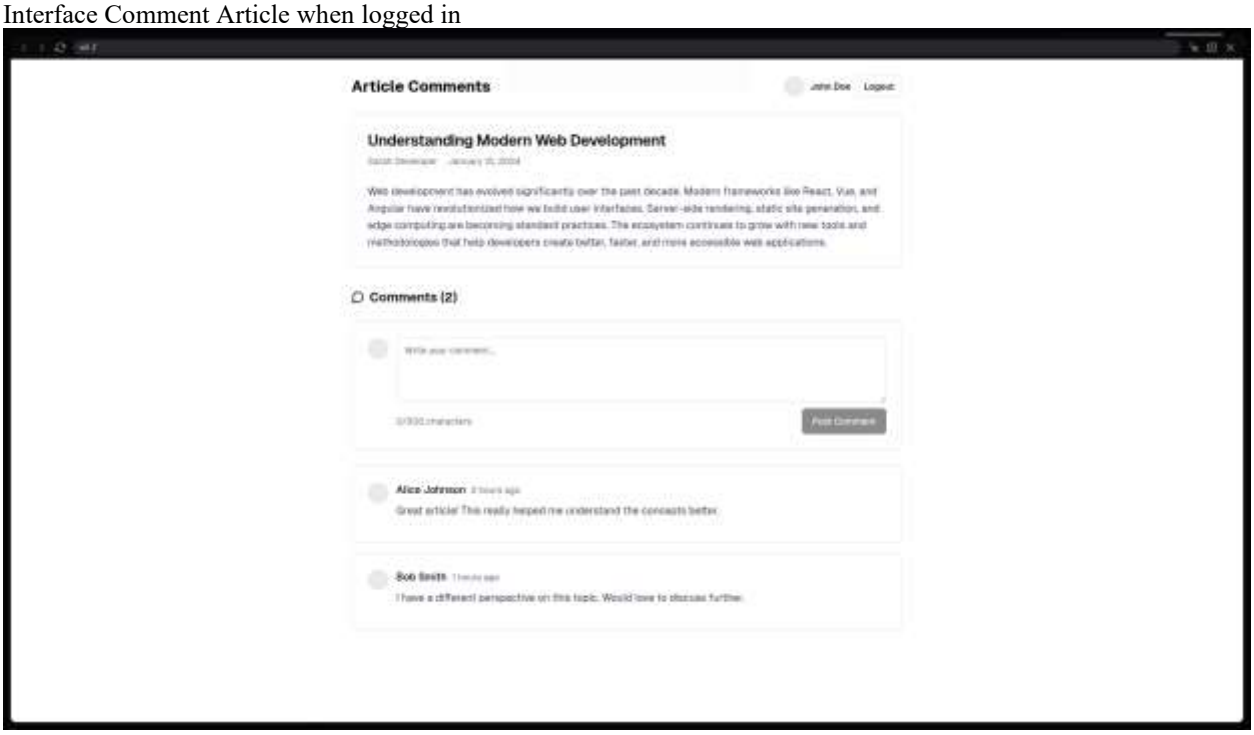
7. Generate Interface

Interface Comment Article when not logged in



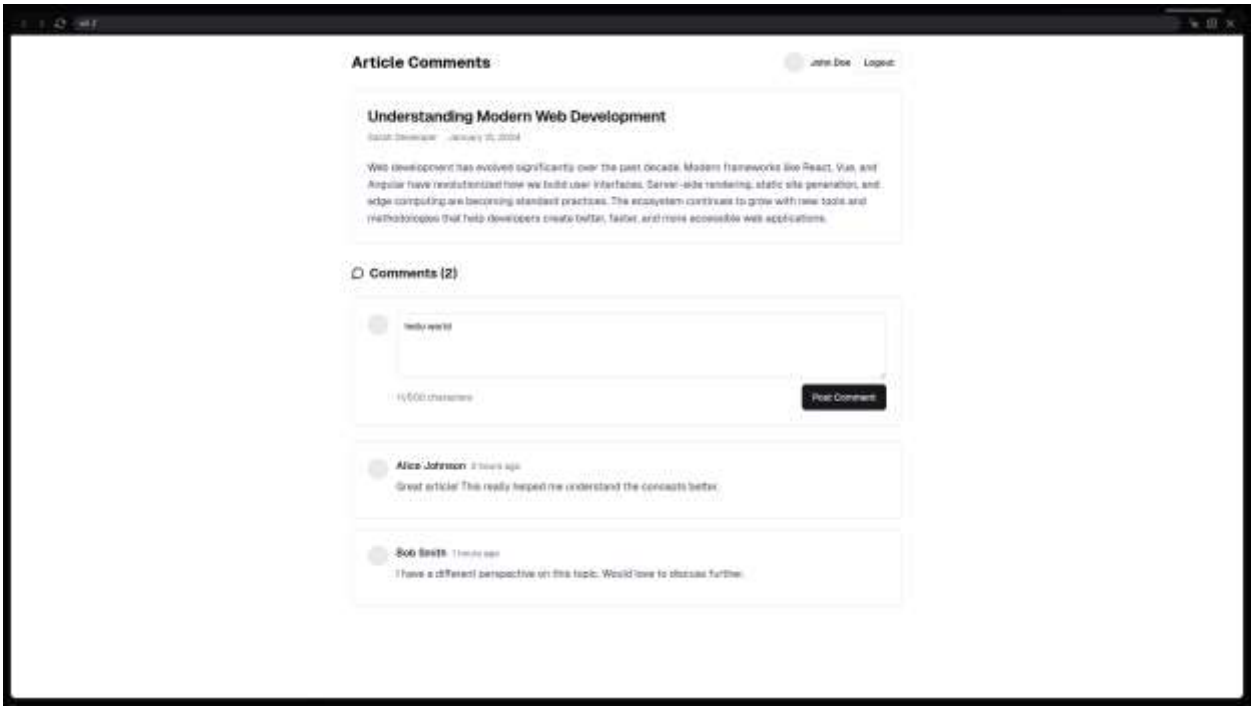
©<Company Name>, 2025

Interface Comment Article when logged in

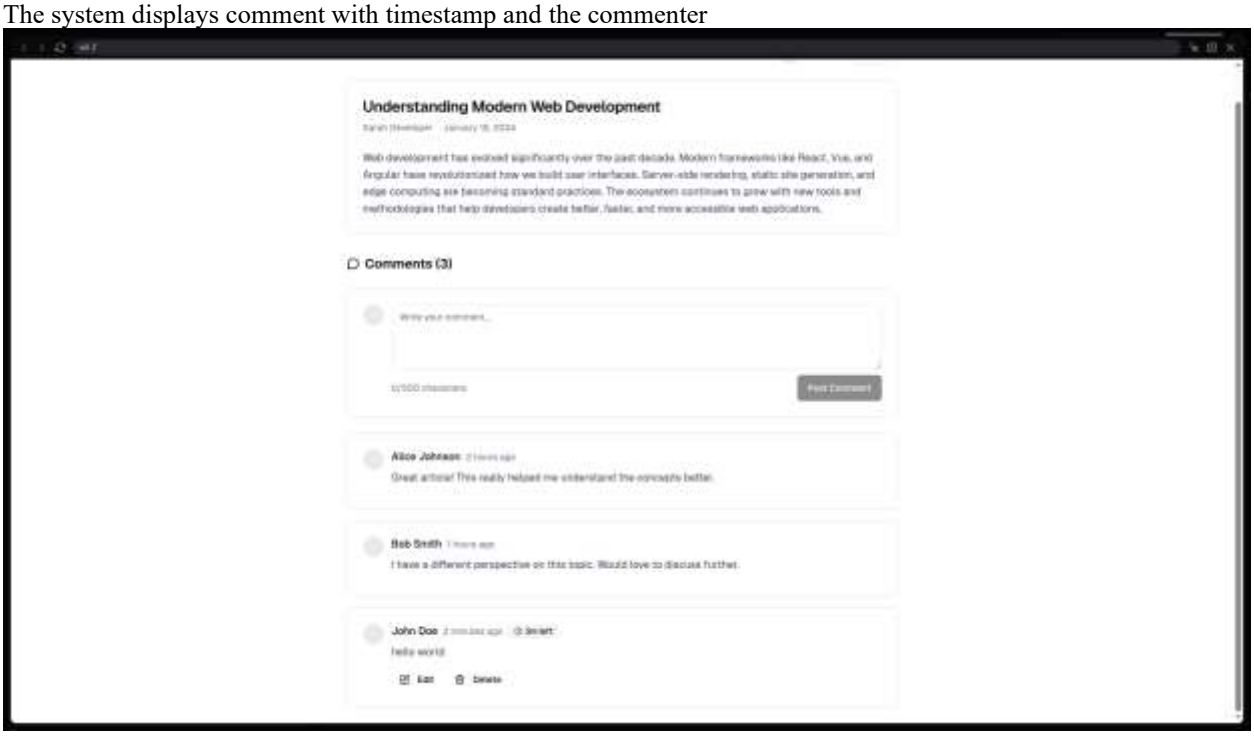


we can write comment (logged in) and click Post Comment to post

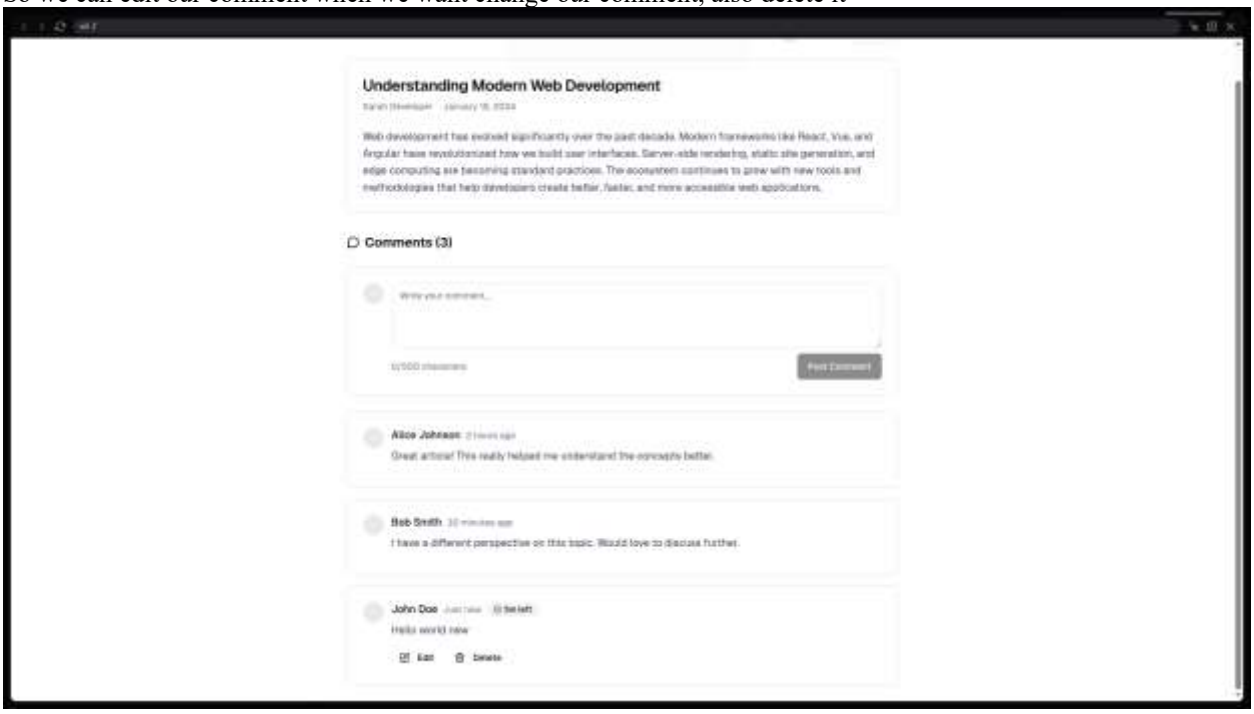
©<Company Name>, 2025



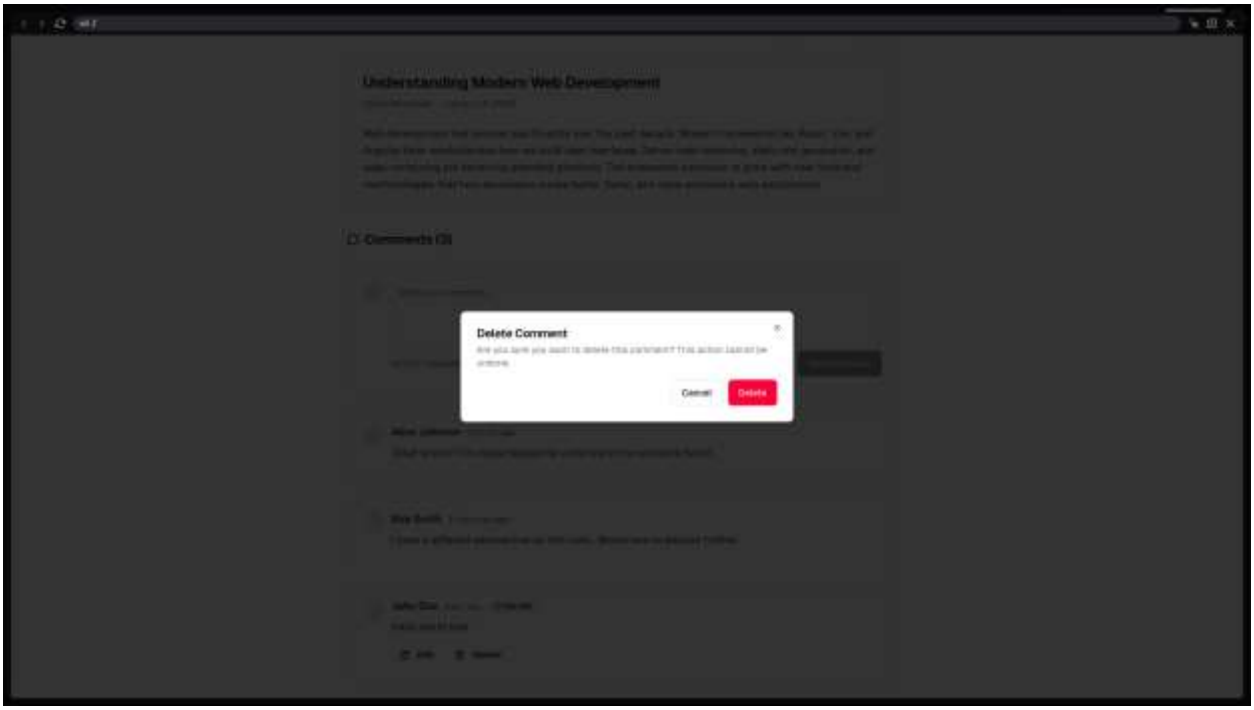
The system displays comment with timestamp and the commenter



So we can edit our comment when we want change our comment, also delete it



©<Company Name>, 2025



©<Company Name>, 2025

404 Not Found

Digital Newspaper Website
Use-Case Specification: Read Article

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
15/07/2025	1.0	Specific about Read Article use case	Nguyễn Tấn Lộc

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Read Article **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows 4
2.2.1 Article not found 4
2.2.2 No internet connection 4
3. Special Requirements **4**
3.1 Responsive Interface 5
3.2 Performance 5
3.3 Media optimization 5
4. Preconditions **5**
4.1 The article must be approved by the editor and in “Published” status. 5
4.2 The system is operational and the user has internet access 5
5. Postconditions **5**
5.1 The article is successfully displayed 5
5.2 If the user is logged in, the article is saved to their Reading History 5
5.3 Any user interaction (like, save, share) are recorded by the system 5
6. Extension Points **5**
6.1 Related articles 5

©Error! No property name supplied., 2025

Use-Case Specification: Read Article

1. Use-Case Name: Read Article

1.1 Brief Description

A user (reader) selects and views an article on the website. The system displays the article`s content, including title, body text, author, publishing date, image (video), related tags. If the user is logged in, they can like, comment, and save articles.

2. Flow of Events

2.1 Basic Flow

1. The user (logged in or not) accesses the homepage
2. The user selects an article by clicking its title or searching.
3. The system loads and displays the article`s details:
    a. Titile
    b. Published
    c. Author`s name
    d. Article content (text)
    e. Image or video (if any)
    f. Related tags.
4. If user has logged in, the system also displays:
    a. Comment box
    b. “Save to favorite” Button
    c. Share button
5. The user scrolls through the page to read the full article
6. If the user is logged in, the system saves the article into the user`s reading history

2.2 Alternative Flows

2.2.1 Article not found

- The user accesses a broken or deleted article link.

2.2.1.1 An Alternative Subflow

- The system display an error page: “article not found”
- Suggest going back to the homepage.

2.2.2 No internet connection

- The user lost internet connection while loading the article

2.2.2.1 An Alternative Subflow

1. the system displays the message: “unable to load article. Please check your connection”
2. allows the user to retry once the connection is restored.

3. Special Requirements

3.1 Responsive Interface

- Article content must be properly displayed on mobile phones, tablets and desktop devices.

©Error! No property name supplied., 2025

3.2 Performance

- Article must load within 1 second under normal conditions

3.3 Media optimization

- Images and videos must be compressed and optimized for last loading without sacrificing clarity.

4. Preconditions

4.1 The article must be approved by the editor and in “Published” status.

4.2 The system is operational and the user has internet access

5. Postconditions

5.1 The article is successfully displayed

5.2 If the user is logged in, the article is saved to their Reading History

5.3 Any user interaction (like, save, share) are recorded by the system

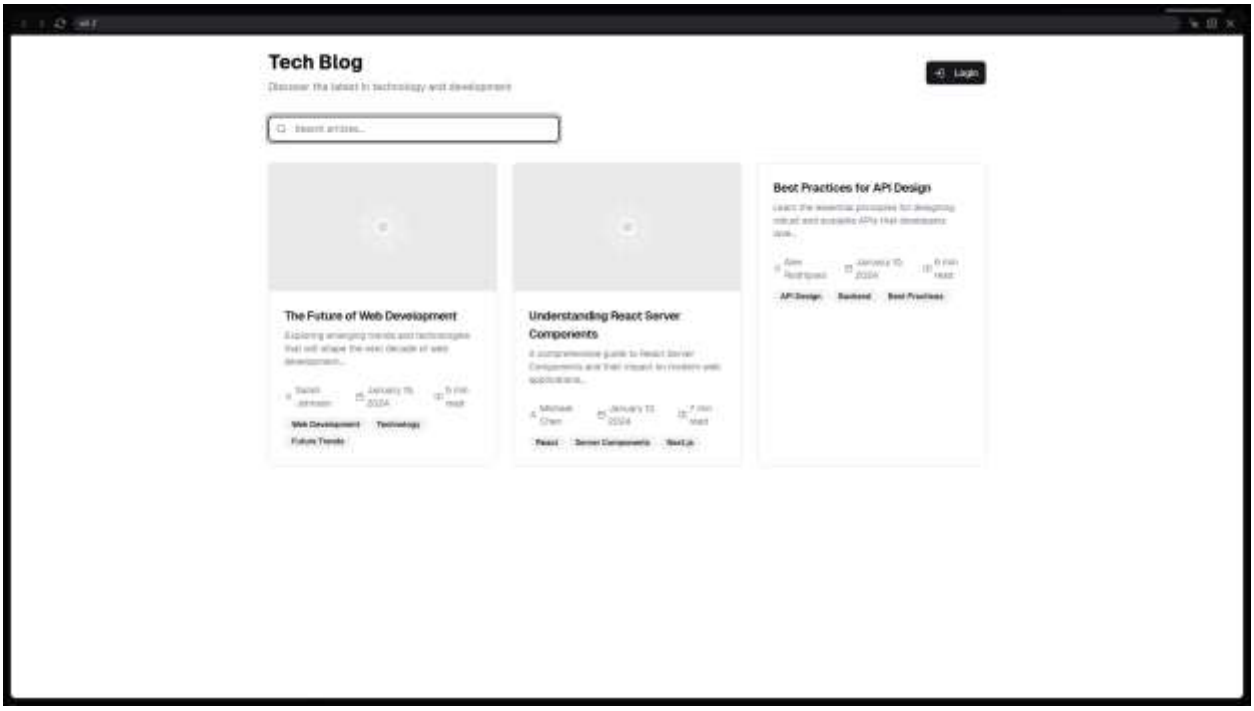
6. Extension Points

6.1 Related articles

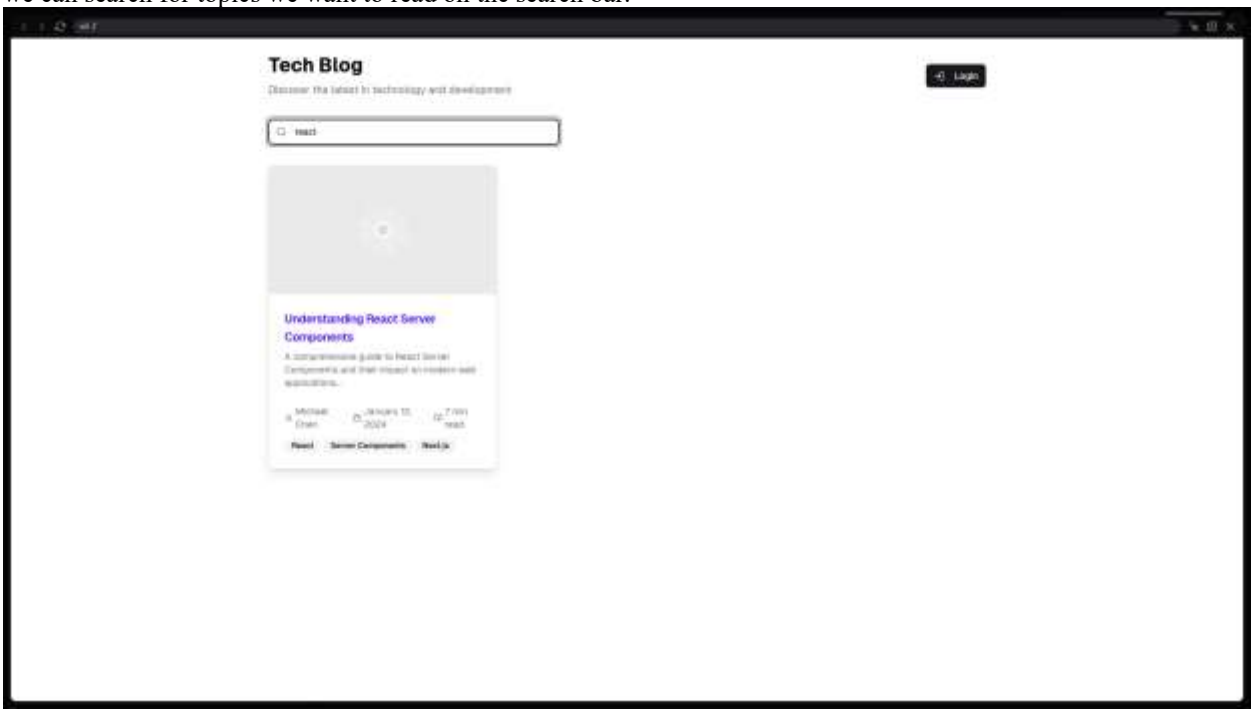
- After step 6: The system may show 3-5 articles with similar tags.

7. Generate Interface

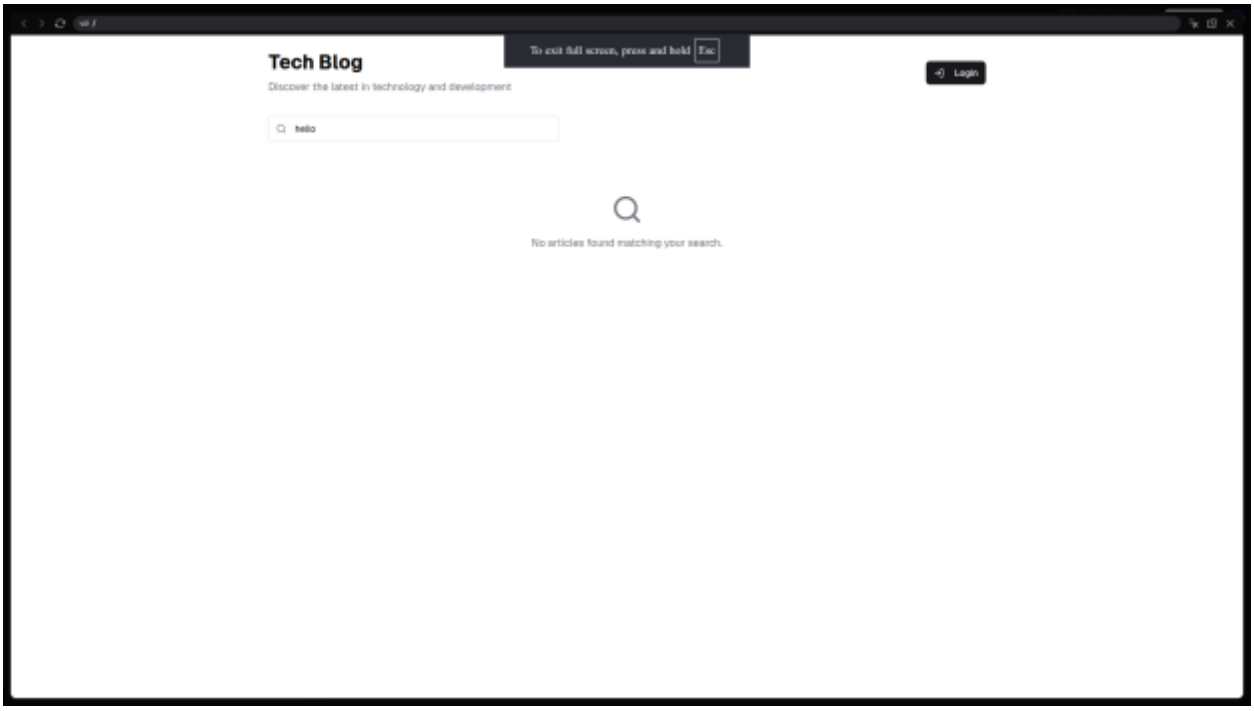
Interface Articles



we can search for topics we want to read on the search bar.



We can search for topics we want to read on the search bar, but the website don`t have our topic.



When we click article, we can scroll the page to read full article

©Error! No property name supplied., 2025

404 Not Found

Digital Newspaper Website
Use-Case Specification: Login

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
16/07/2025	1.0	Specific about Login use case	Nguyễn Tấn Lộc

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Login **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows 4
2.2.1 Invalid Credentials 4
2.2.2 Account Locked 4
3. Special Requirements **4**
3.1 Security 4
3.2 Session management 4
3.3 Responsiveness 5
3.4 Accessibility 5
4. Preconditions **5**
4.1 The user must already have a registered account in the system 5
4.2 The system is online and accessible 5
5. Postconditions **5**
5.1 The user is authenticated and session is started 5
5.2 The user gains access to restricted content and user-specific features 5
5.3 The login attempt is logged (for auditing and security tracking) 5

©Error! No property name supplied., 2025

Use-Case Specification: Login

1. Use-Case Name: Login

1.1 Brief Description

This use case describes how a registered user accesses their account by providing a valid username and password. If the credentials are correct, the system grants access and sets the session, if not, an error message is shown. Users may also request password recovery if needed.

2. Flow of Events

2.1 Basic Flow

7. The user click the login button from the homepage
8. The system displays the login form with fields: email/ username and password
9. The user enter their credentials and clicks submit button
10. The system validates the inputs:
    a. Check mail/ username is existed
    b. Verifies password
11. If the credentials are correct
    a. The system authenticates the user
    b. Sets a session token or login cookie
    c. The system displays the homepage with user` personal account
12. The user is now logged in and can access authenticated features

2.2 Alternative Flows

2.2.1 Invalid Credentials

- In step 4, creadential do not matching

2.2.1.1 An Alternative Subflow

- The system displays an error message “Invalid email or password. Please try again.”.
- Allow all users to re-enter credentials or choose password recovery

2.2.2 Account Locked

- The user enters incorrect credentials to much (e.g 5 times)

2.2.2.1 An Alternative Subflow

1. The system locks the account temporarily (e.g 5 minutes)
2. Display message “ your account is temporarily locked due to many failed login attempts”

3. Special Requirements

3.1 Security

- Passwords must be kept confidential to avoid information leakage.

3.2 Session management

- Session tokens or cookies must expire after a predefined period (eg. 1 hour)

©Error! No property name supplied., 2025

3.3 Responsiveness

- The login must render properly on all screen sizes

3.4 Accessibility

- The form must comply with WCAG 2.1 for screen readers and keyboard navigation

4. Preconditions

4.1 The user must already have a registered account in the system

4.2 The system is online and accessible

5. Postconditions

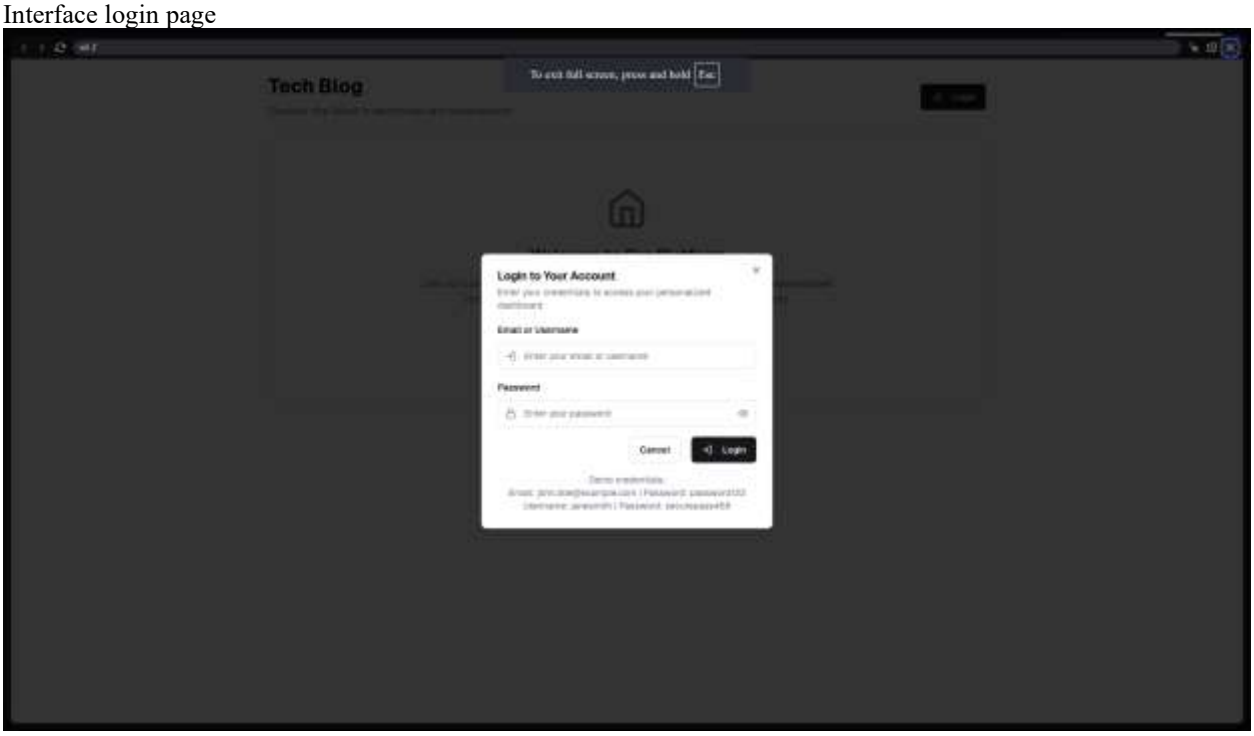
5.1 The user is authenticated and session is started

5.2 The user gains access to restricted content and user-specific features

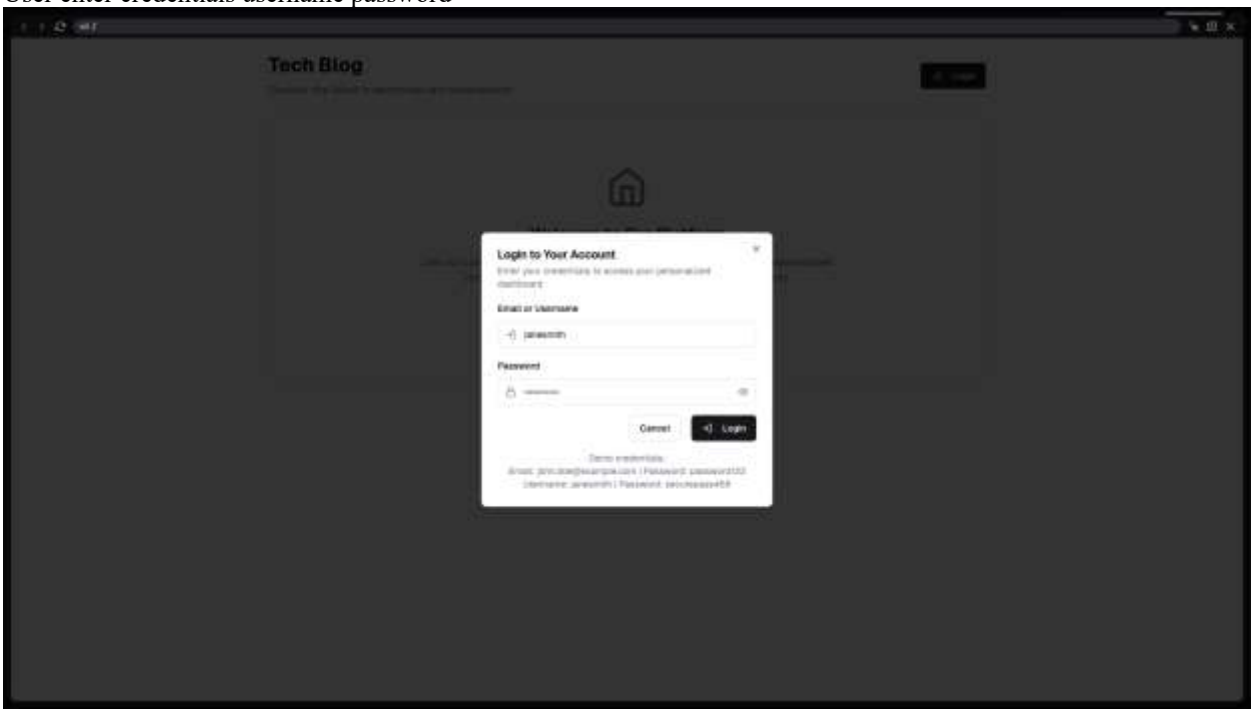
5.3 The login attempt is logged (for auditing and security tracking)

6. Generate Interface

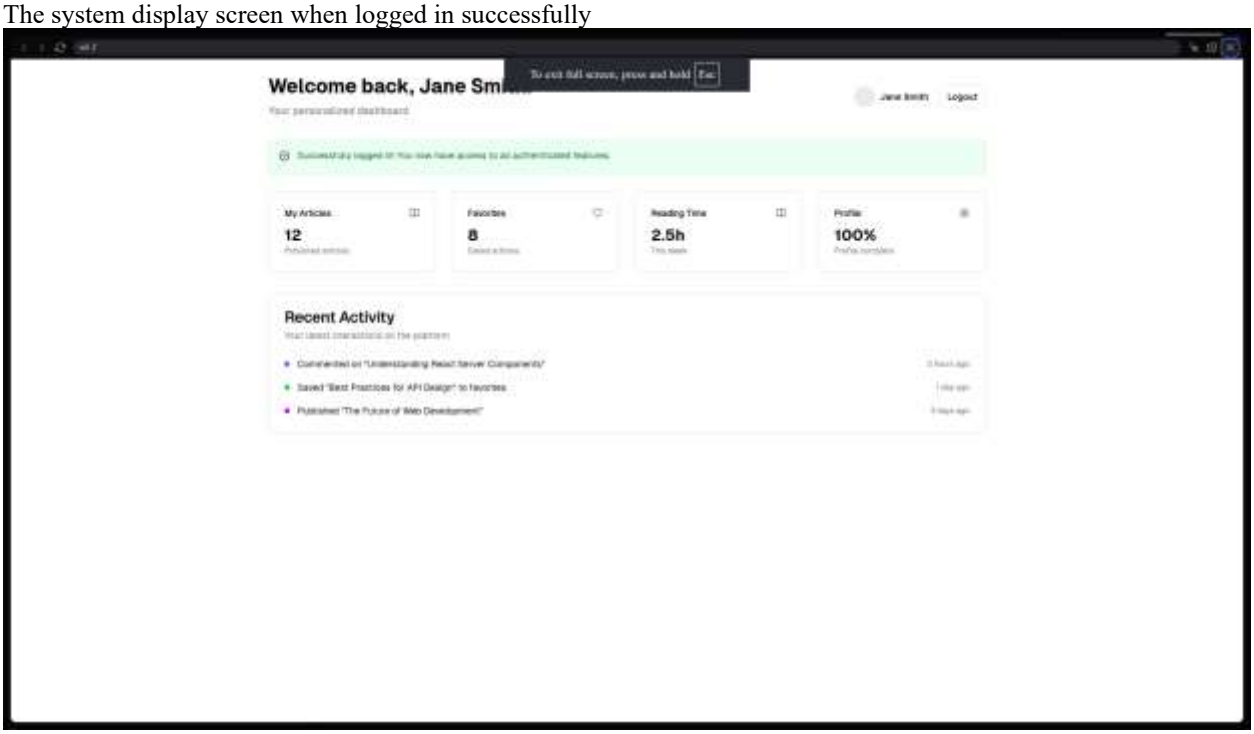
Interface login page



User enter credentials username password

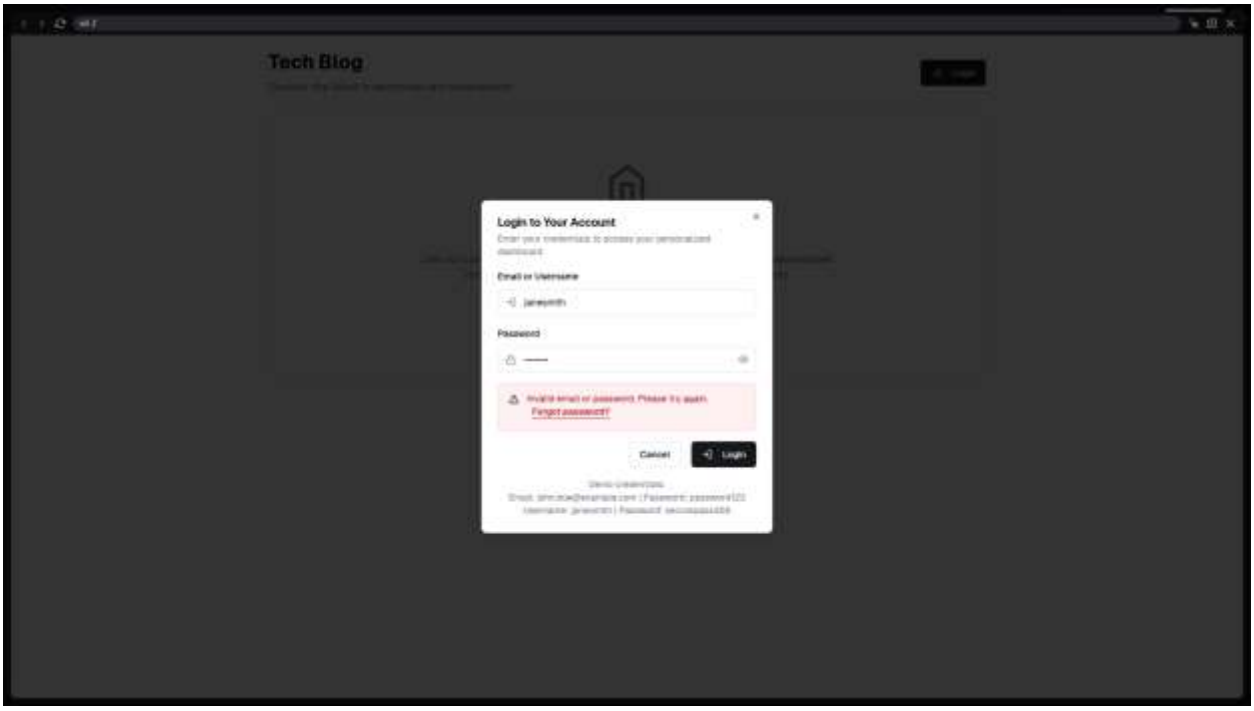


The system display screen when logged in successfully



If system check username or password are false

©Error! No property name supplied., 2025



©Error! No property name supplied., 2025

404 Not Found

Digital Newspaper Website
Use-Case Specification: Register

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
16/07/2025	1.0	Specific about Register use case	Nguyễn Tấn Lộc

Table of Contents

1. Use-Case Name: Register **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows 4
2.2.1 Email already registered 4
2.2.2 Passwords do not matching 4
2.2.3 Weak password 5
2.2.4 Invalid or expired verification code 5
3. Special Requirements **5**
3.1 Verification code timeout 5
3.2 Password encryption: 5
3.3 Rate limiting 5
3.4 Email format validation 5
3.5 Device compatibility 5
4. Preconditions **6**
4.1 The user doesn`t already have an account 6
4.2 The system is connected to the internet and email service is functioning 6
5. Postconditions **6**
5.1 A new user account is created and stored in the database 6
5.2 The user receives confirmation of registration success 6
5.3 The user is redirected to the login page 6
6. Extension Points **6**
6.1 Resend verification code 6
6.2 Auto login after verify 6

Use-Case Specification: Register

1. Use-Case Name: Register

1.1 Brief Description

This use case describes how a new user creates an account by entering personal information (full name, email, password) and complete email verification through a verification code, when the code is verified, the system registers user and redirects them to the login page

2. Flow of Events

2.1 Basic Flow

13. The user click the Register button from the homepage
14. The system displays the login form with fields:
    a. Full name
    b. Email
    c. Password
    d. Confirm password
15. The user enter their credentials and clicks submit button
16. The system validates the inputs:
    a. Check mail is valid and not already registered
    b. Confirms that password and confirm password match
    c. Verifies password strength (e.g min 8 character, have uppercase, number, special characters)
17. The system sends a verification code to mail which has been received
18. The system display new form requirement user to enter verification code
19. The user check mail and find verification code is sent
20. The user write verification code and clicks submit button
21. If valid, the system
    a. Store the user`s information in database
    b. Send a verification email(if applicable)
    c. Displays a message “registration successful"
    d. Redirects the user to the login page

2.2 Alternative Flows

2.2.1 Email already registered

- At step 4, the mail already registered exists in the database

2.2.1.1 An Alternative Subflow

- The system displays an error message “this email is already registed, please use another email or log in”

2.2.2 Passwords do not matching

- At step 4, the password and confirm password do not match

2.2.2.1 An Alternative Subflow

1. The system displays an error message “ passwords do not match”

2. User must re-enter the fields

2.2.3 Weak password

- The password doesn`t meet strength requirements

2.2.3.1 An Alternative Subflow

1. The system display error message: “password must be at least 8 character and include uppercase, numbers, symbols"
2. User must re-enter stronger password

2.2.4 Invalid or expired verification code

- At step 8, the verification code is invalid or has expired

2.2.4.1 An Alternative Subflow

1. The system displays error message: “valid or expired verification code”
2. Provides an option to resend the verification code and re-enter verification code.

3. Special Requirements

3.1 Verification code timeout

- The verification code expires after 5 mintutes.

3.2 Password encryption:

- All password must store using secure hashing algorithms

3.3 Rate limiting

- Limit the number of verification attempts to avoid abuse

3.4 Email format validation

- Ensure email format is RFC-compliant before sending code

3.5 Device compatibility

- Forms must work seamlessly with mobile devices, tablet and desktop.

4. Preconditions

4.1 The user doesn`t already have an account

4.2 The system is connected to the internet and email service is functioning

5. Postconditions

5.1 A new user account is created and stored in the database

5.2 The user receives confirmation of registration success

5.3 The user is redirected to the login page

6. Extension Points

6.1 Resend verification code

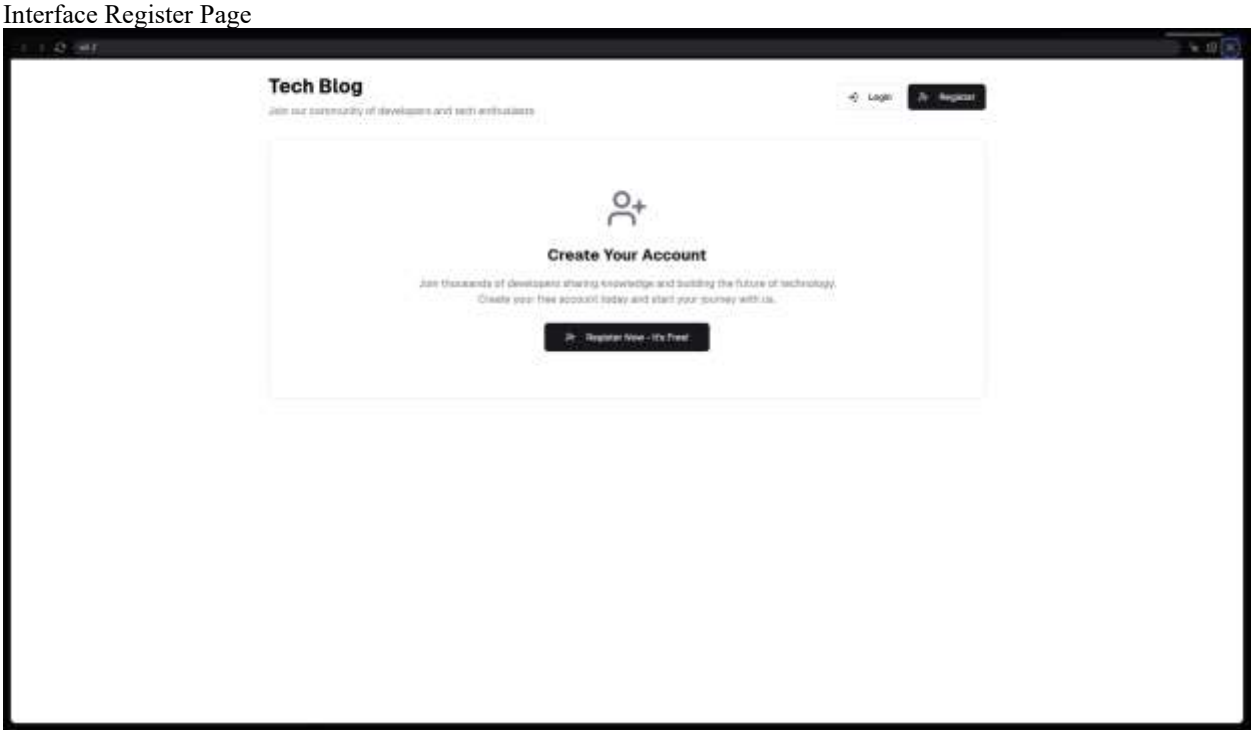
- After step 8, user may request to resend the verification code if needed

6.2 Auto login after verify

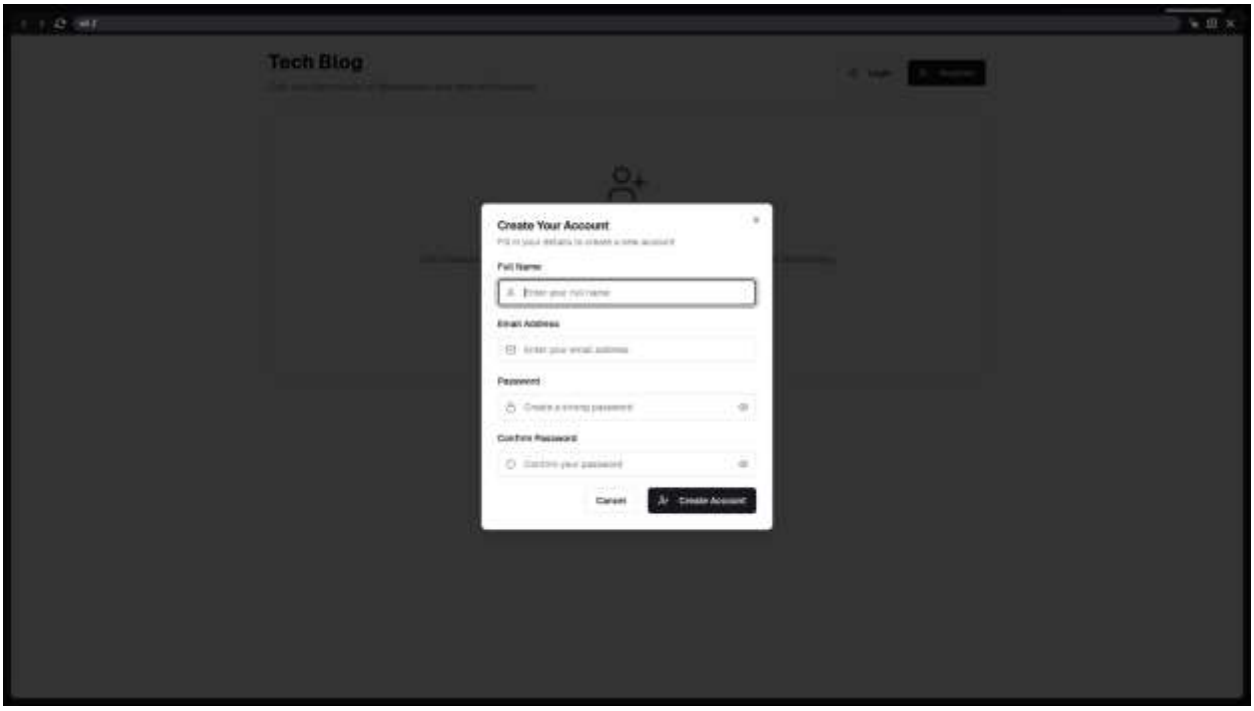
- After step 9, the system may optionally log the user in immediately

7. Generate Interface

Interface Register Page



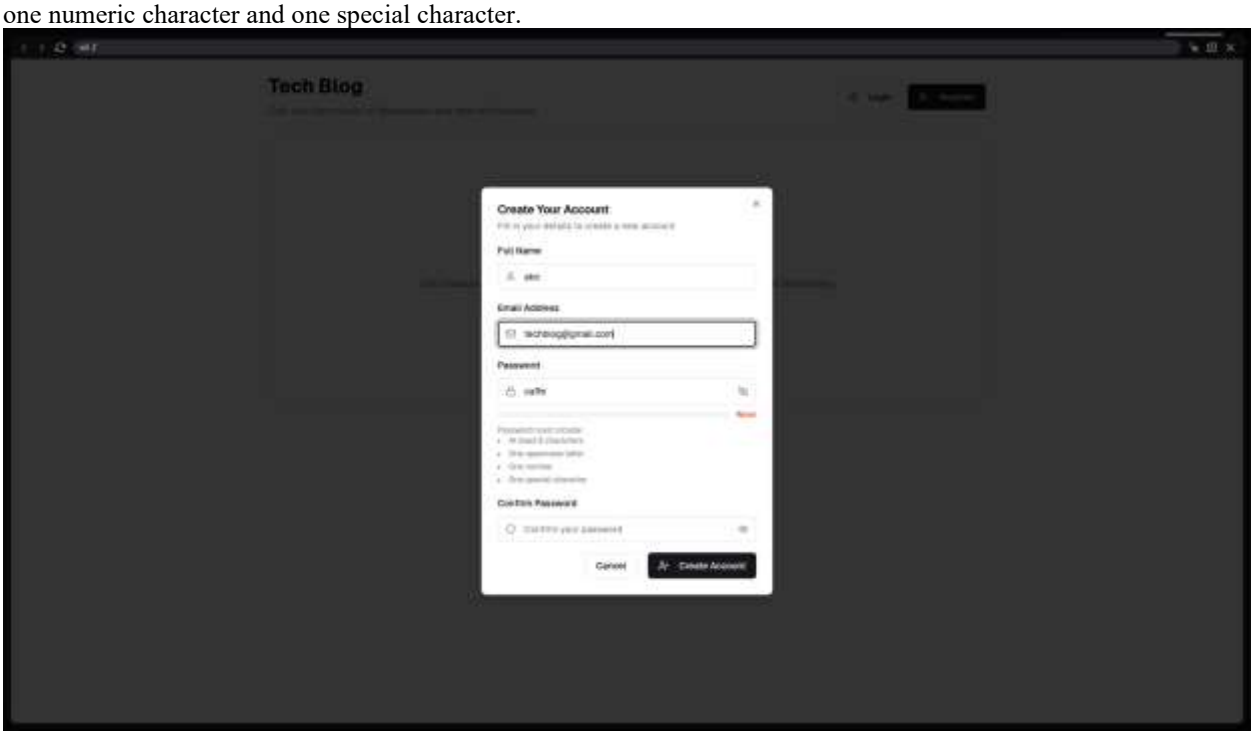
User click Register button, and the system dislays:



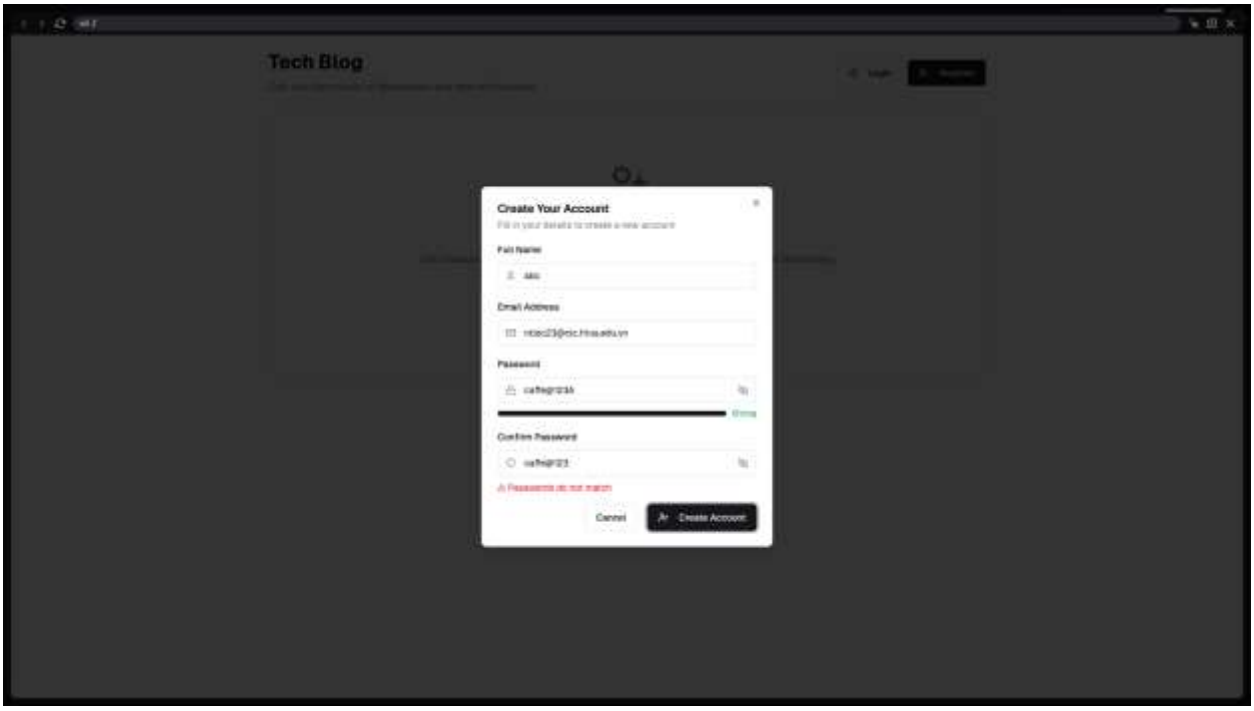
When the user enters a password, he must enter the same in the confirm password box.

with registering, the user must comply with the following requirements: At least 8 characters, one uppercase letter,

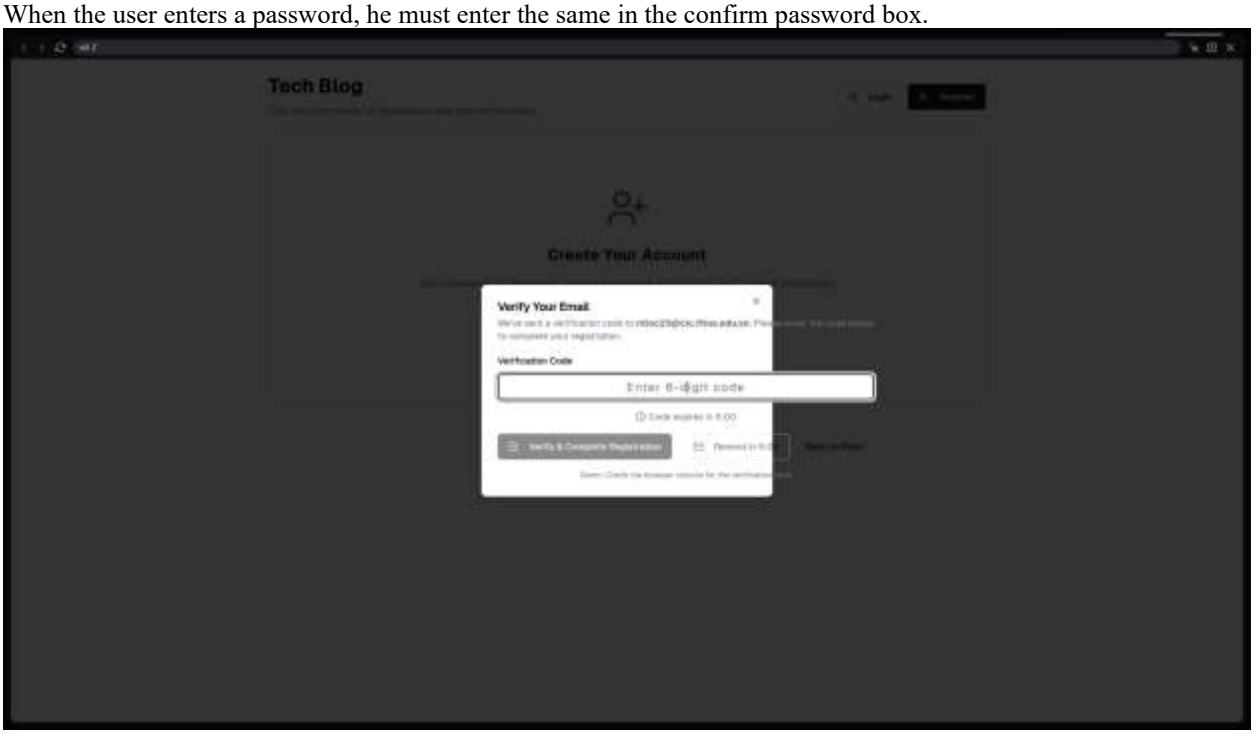
one numeric character and one special character.



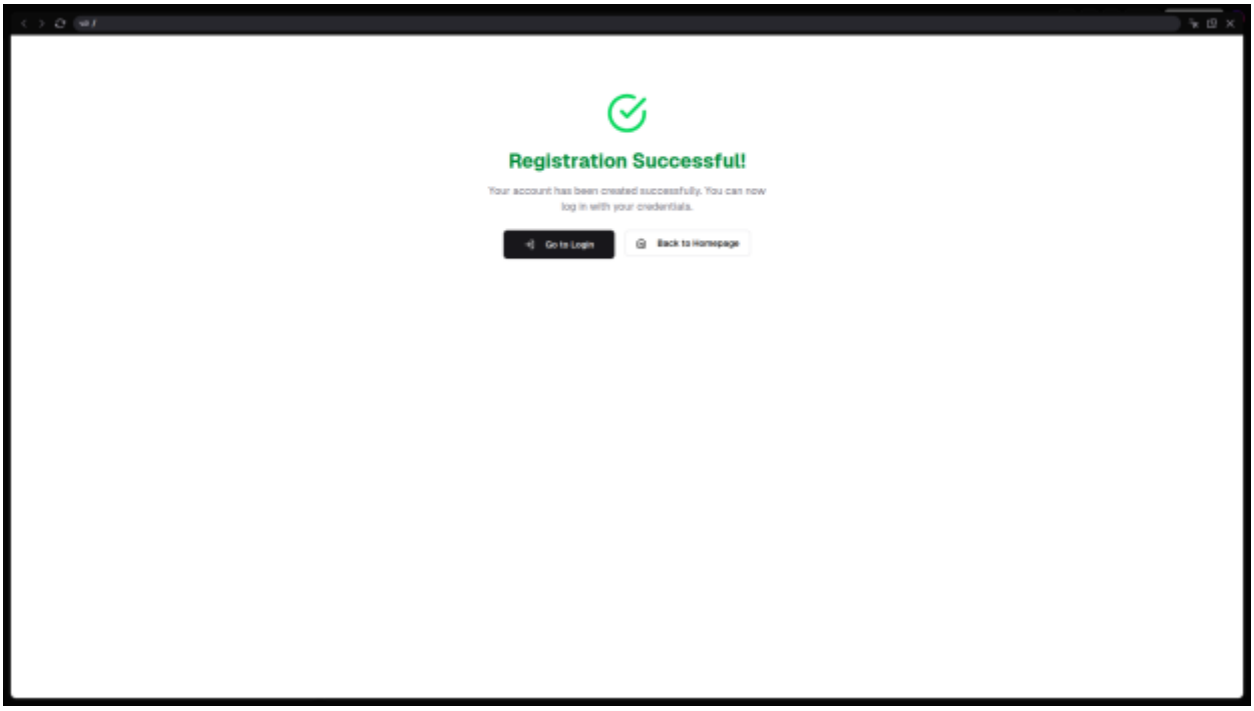
When the user enters a password, he must enter the same in the confirm password box.



When the user enters a password, he must enter the same in the confirm password box.

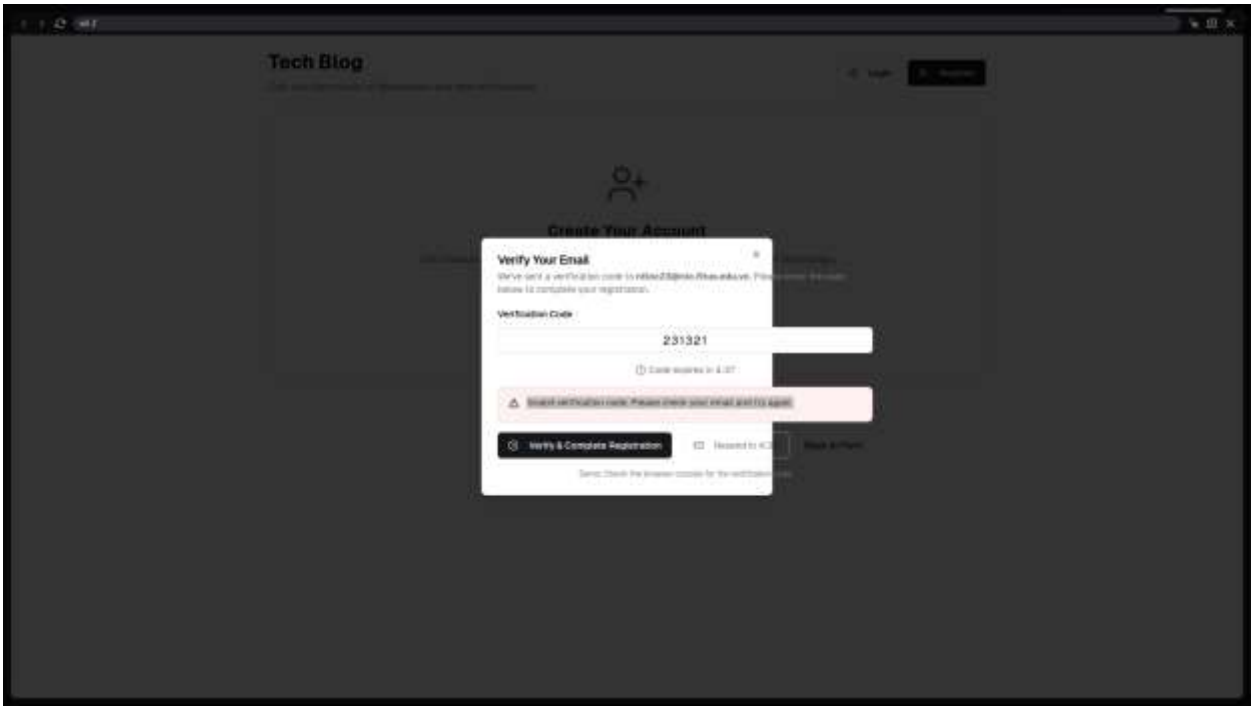


The post code will send to user mail and the user will write that into and click the Verify & Complete Registration button. The system will display a log in page.



And if the Verify & Complete Registration is wrong, the system will display message: “Invalid verification code. Please check your email and try again.”

©Error! No property name supplied., 2025



©Error! No property name supplied., 2025

404 Not Found

Digital Newspaper Website
Use-Case Specification: Rate Article

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
07/07/2025	1.0	Specific about Rate Article use case	Lê Thị Minh Thư

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Rate Article **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows 4
2.2.1 Anonymous Rating 4
2.2.2 Incomplete Rating 4
3. Special Requirements **4**
3.1 System Performance 4
3.2 Security & Privacy 5
3.3 Rating Integrity 5
4. Preconditions **5**
4.1 User Authentication 5
4.2 Valid Article. 5
5. Postconditions **5**
5.1 Rating result is shown 5
5.2 Average rating star of the article changes 5
6. Extension Points **5**
6.1 Anonymous Submission Option 5
7. Low Fidelity Prototype For Use Case **5**
7.1 Prompt Used for AI Tool 5
7.2 Generated Interface 6

©Error! No property name supplied., 2025

Use-Case Specification: Rate Article

1. Use-Case Name: Rate Article

1.1 Brief Description

This use case describes the functionality that allows users to rate the articles for evaluating the quality of the articles. Ratings help other readers and editors understand how the content is received.

2. Flow of Events

2.1 Basic Flow

1. The user navigates to a specific article.
2. The user scrolls to the rating section and selects a star rating (1 to 5 stars) for the article (main rating)
3. Optionally, the system displays sub-categories for more detailed rating such as image quality, content, accuracy.
4. The user may rate sub-categories or leave them empty.
5. The user clicks to the Submit button for sending their rating..
6. The system checks if at least main rating is selected:
    a. If yes: rating is submitted
    b. If not: the Submit button remains disabled.
7. The system stores the rating and the new average rating of the article is updated and displayed.
8. The new rating is shown on the rating section.
9. A confirmation message is shown to the user:
    a. "Your rating has been submitted"
    b. Or: "Thank you for your feedback!"

2.2 Alternative Flows

2.2.1 Anonymous Rating

- At step 5, the user chooses to submit the rating anonymously.

2.2.1.1 *Anonymous Rating* Subflow

1. The user checks the "Rate anonymously" checkbox
2. The system records the rating without linking it to the user's ID
3. The rating is displayed as submitted by "Anonymous User".
4. The rest of the rating flow proceeds normally

2.2.2 Incomplete Rating

- At step 6, the user clicks the Submit button without selecting the main rating.
- If only sub-categories are rated and main rating is left empty. The Submit button remains disabled. A tooltip may appears: "Please rate the article overall before submitting"
- If neither the main rating nor sub-categories are selected, no action is allowed.
- If the main rating is selected, regardless of sub-categories, submission proceeds normally.

3. Special Requirements

3.1 System Performance

- Rating submission and average update must complete within **2 seconds** under normal load.
- The system must support concurrent ratings from multiple users without loss or duplication of data.

3.2 Security & Privacy

- For anonymous ratings, the system must ensure no personally identifiable information is stored with the rating.
- All rating data must be stored securely in accordance with data protection policies

3.3 Rating Integrity

- Each user can rate an article only **once.** Re-rating may be allowed if the previous rating is overwritten.
- Ratings must be protected against bots or automated spam via CAPTCHA.

4. Preconditions

4.1 User Authentication

- The user who wants to rate the articles must have a registered and logged-in account to rate articles.

4.2 Valid Article.

- The article must be visible and allow user interaction(not archived or locked)

5. Postconditions

5.1 Rating result is shown

- If not anonymous: The user's rating is shown with the user's profile (name and avatar) on the rating section.
- If anonymous: The user's rating is shown with the user name "Anonymous user" and their profile is hidden.

5.2 Average rating star of the article changes

- The system recalculates, updates and shows the article's average rating accordingly.

6. Extension Points

6.1 Anonymous Submission Option

- **Location in flow:** After step2 in the Rating flow.
- **Description:** If the user selects to submit anonymously, the system saves the rating without attaching any user identity. The rating will appear in the rating section under the name "Anonymous User", and the rater's profile is hidden**.**

7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

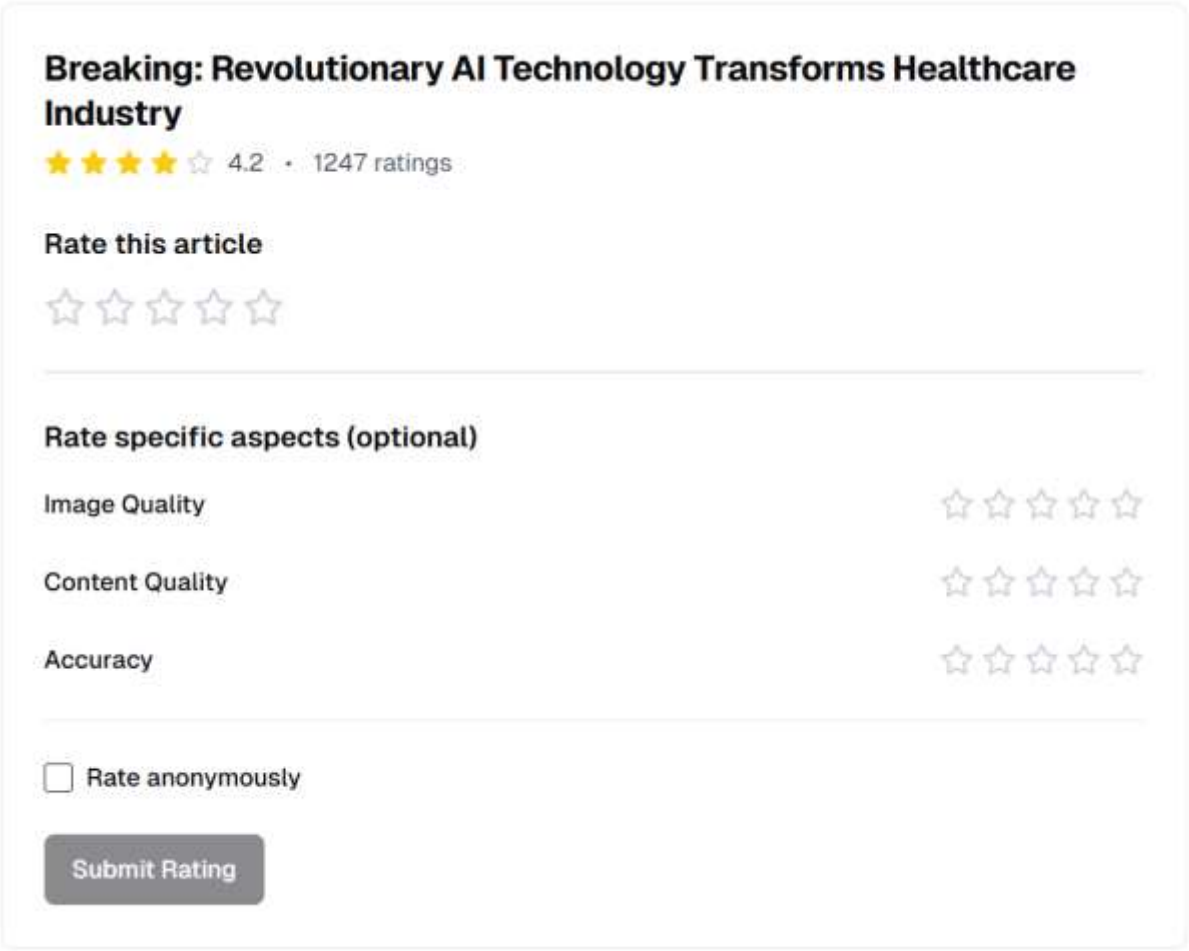
Create a user interface for rating a news article on a digital newspaper website.

The interface should display:
- The article title at the top.
- A 5-star rating input as the main rating.
- Optional sub-category ratings (e.g., image quality, content quality, accuracy), also using star inputs.
- A checkbox labeled "Rate anonymously".
- A Submit button, which is disabled unless the main rating is selected.
- After submission, display a confirmation message: "Your rating has been submitted" or "Thank you for your feedback!"
- If the user checked "Rate anonymously", show the rating as submitted by "Anonymous User"; otherwise,

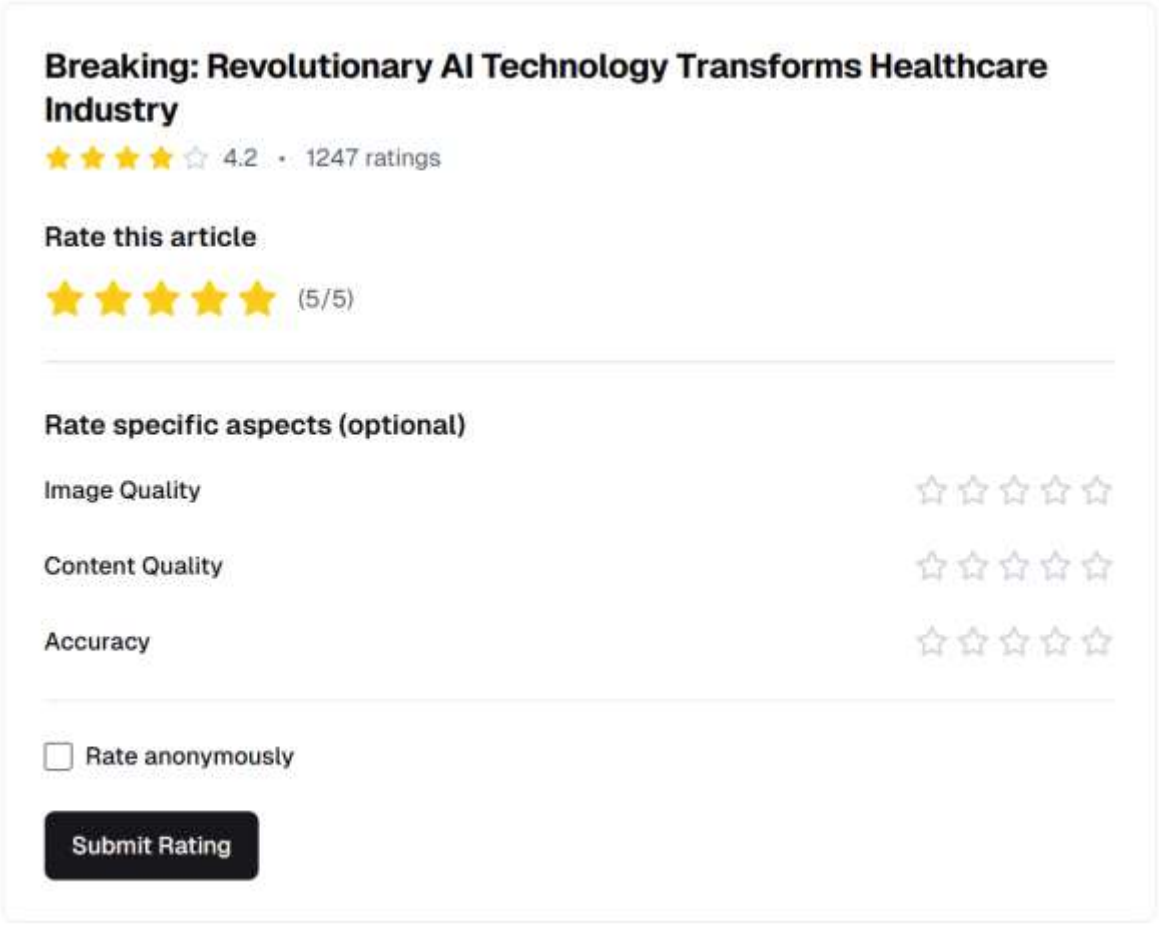
show the user's name and avatar.
- Display the updated average rating of the article after submission.

Make sure the layout is clean and easy to understand, and suitable for both desktop and mobile screens.

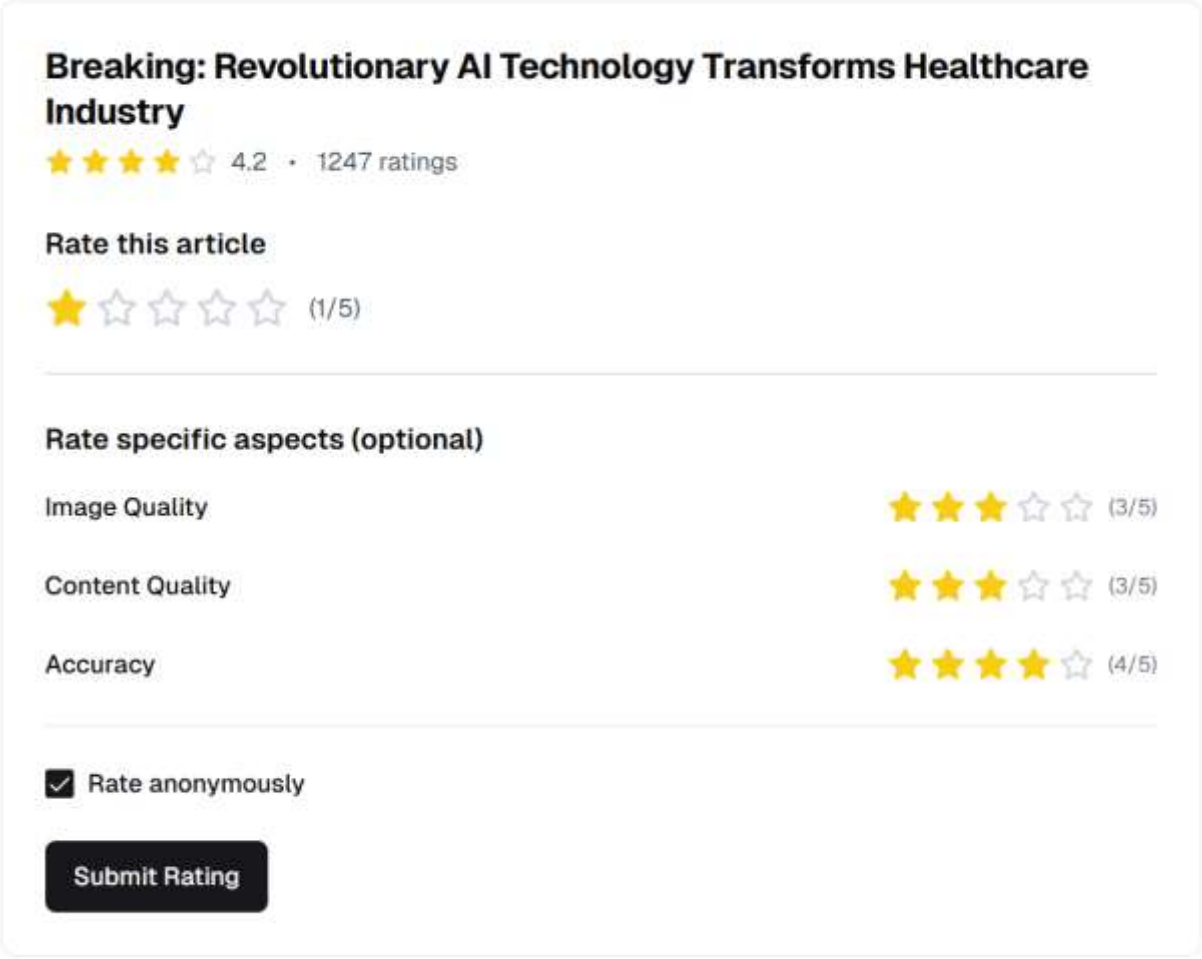
7.2 Generated Interface



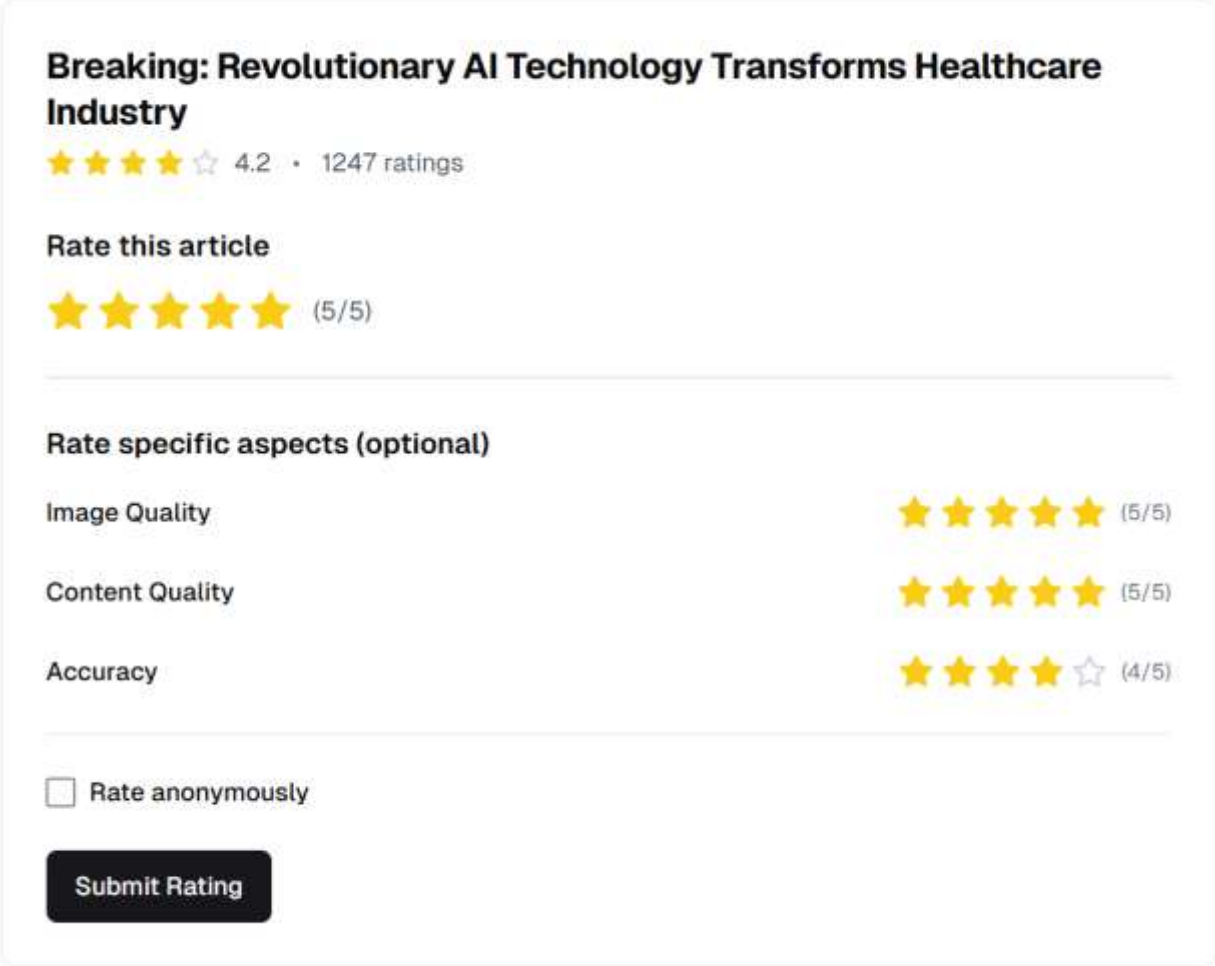
This is the main screen of Rate Article use case



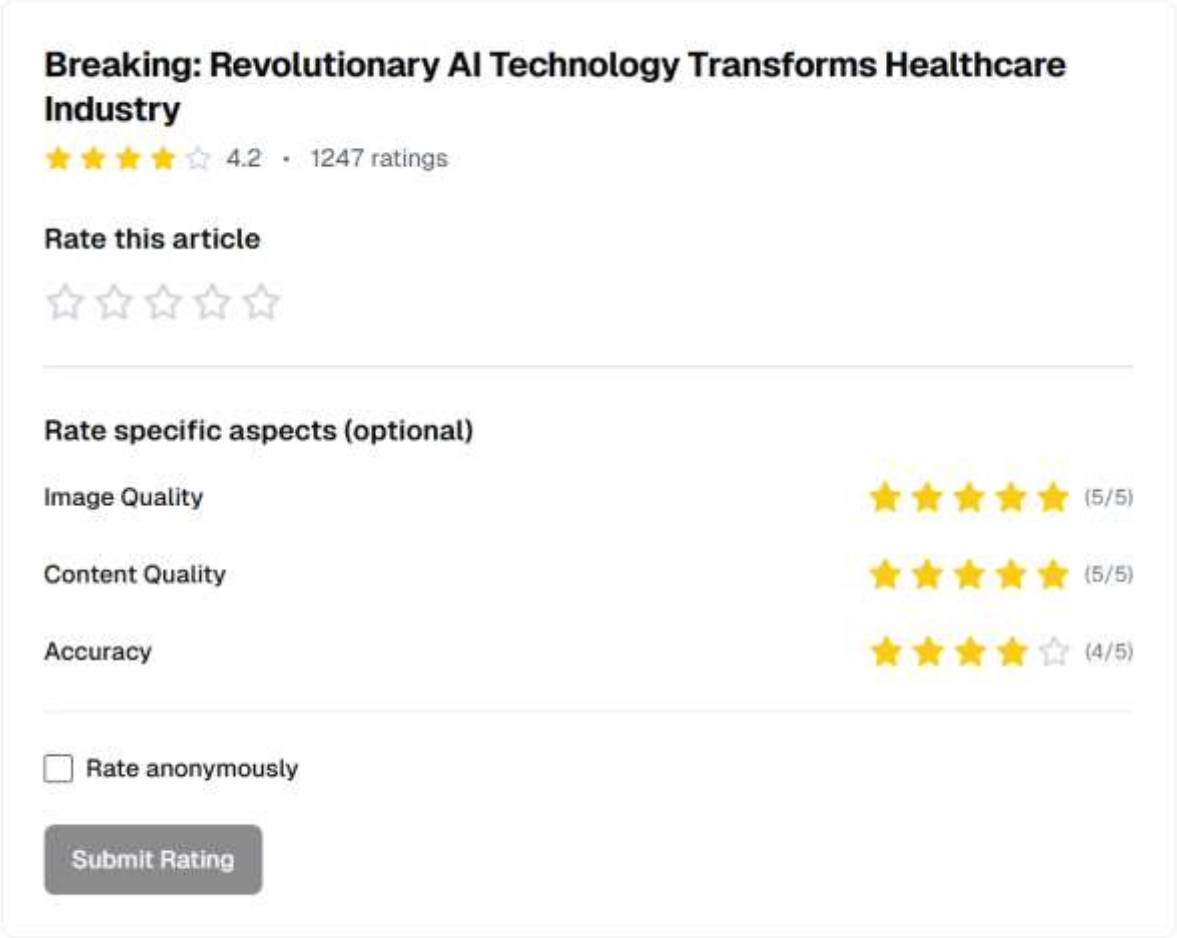
User choose star on the main rating below "Rate this article", without choose star at the sub-categories and "Rate anonymously" checkbook, the "Submit Rating" button is valid



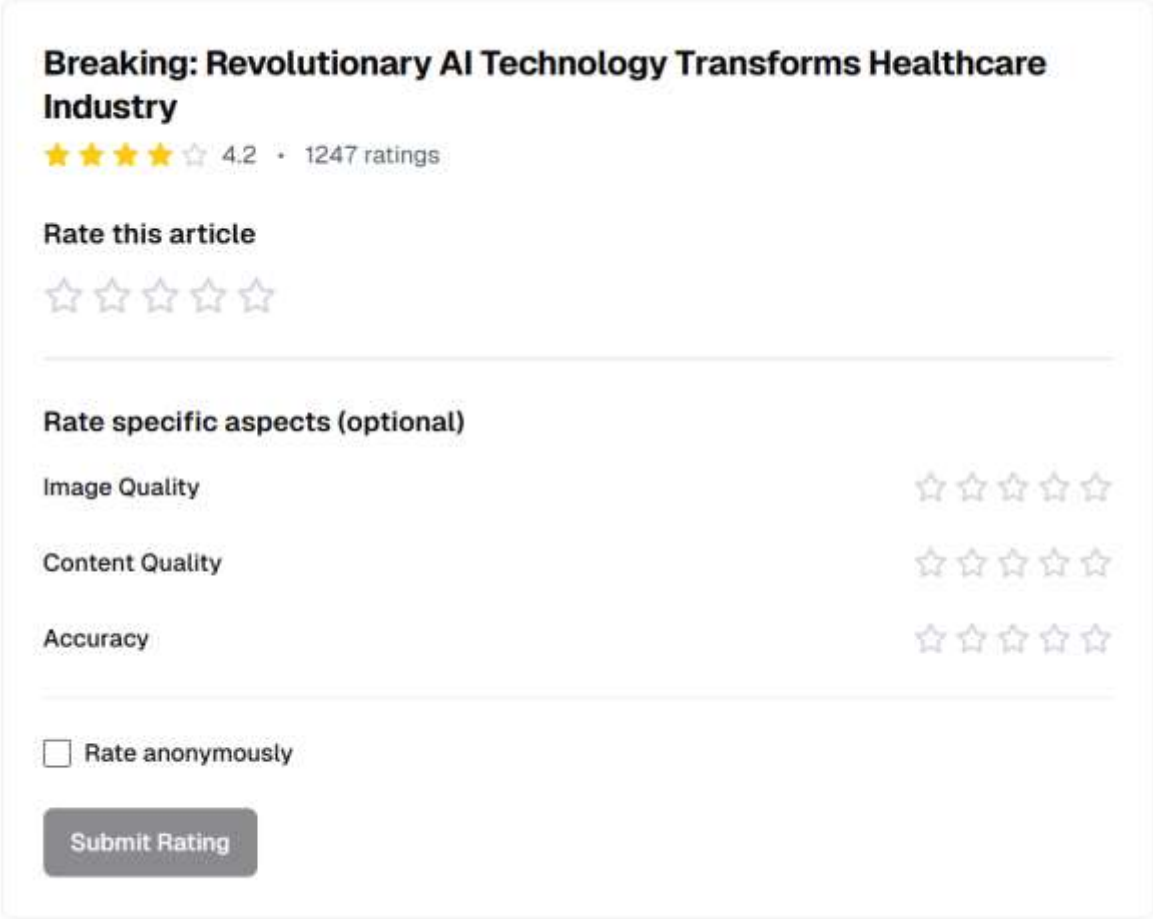
User choose star on the main rating below "Rate this article" and choose star at sub-categories and "Rate anonymously" checkbook, the "Submit Rating" button is valid



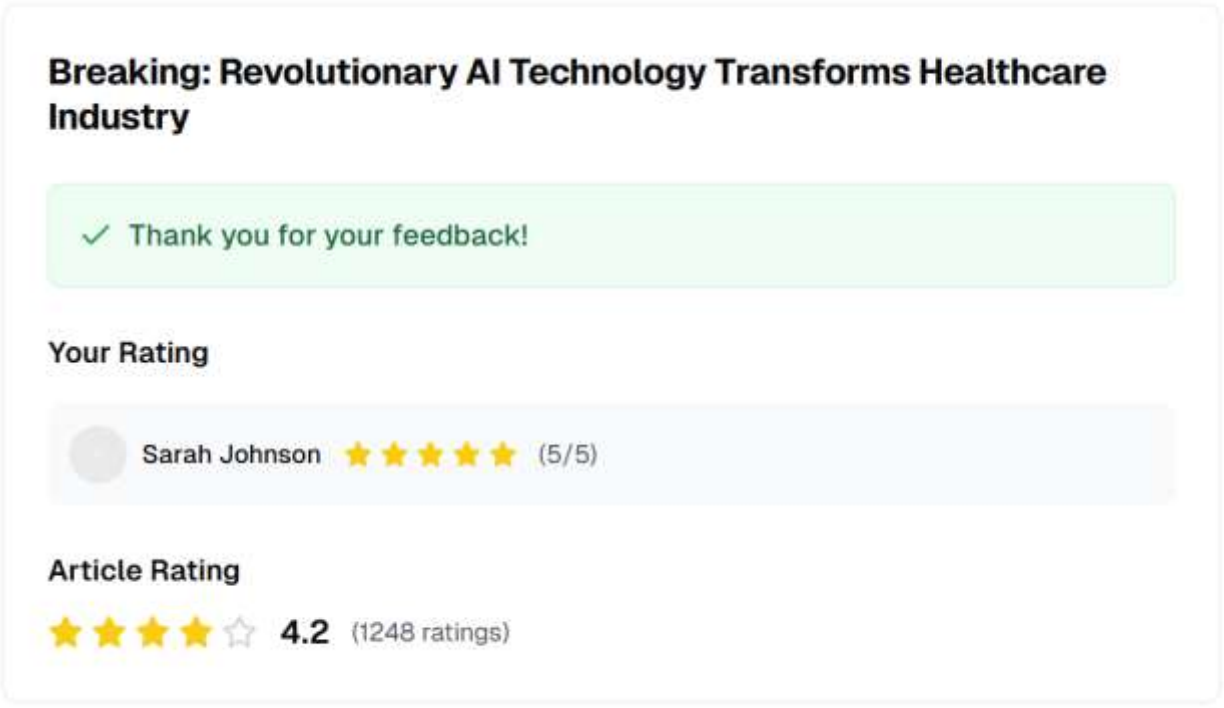
User choose star on the main rating below "Rate this article" and choose star at sub-categories, the "Submit Rating" button is valid



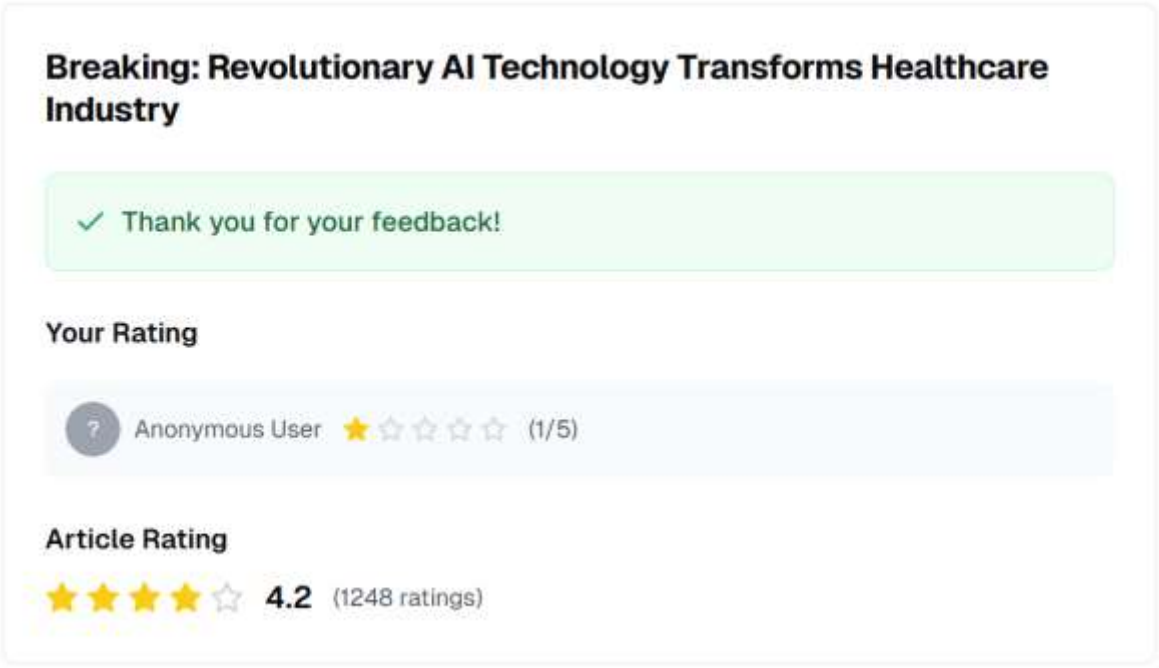
User choose stars at sub-categories but doesn't choose stars at the main rating"Rate this article". So that, the "Submit Rating" button is invalid



User doesn't chose stars, so the "Submit Rating" button is invalid.



The confirmation message, rating result is shown, average rating and number of ratings is updated and displayed after Submit Rating, and the user name and avatar is displayed when the user doesn't tick at "Rate Anonymously" checkbox.



The confirmation message, rating result is shown, average rating and number of ratings is updated and displayed after Submit Rating, and the user name is "Anonymous Usẻ" and avatar is hidden when the user ticks at "Rate Anonymously" checkbox.

404 Not Found

Digital Newspaper Website
Use-Case Specification: Share Article

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Specific about Share Article use case	Lê Thị Minh Thư

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Share Article 4
    1.1 Brief Description 4
2. Flow of Events 4
    2.1 Basic Flow 4
    1. The user views an article on the website. 4
    2.2 Alternative Flows 4
        2.2.1 User Cancels Sharing 4
        2.2.2 Social Account Having Been Logged In 4
        2.2.3 Platform Error 4
3. Special Requirements 4
    3.1 System Performance 4
    3.2 Compatibility 5
4. Preconditions 5
    4.1 User Authentication 5
    4.2 Social Media Login 5
5. Postconditions 5
    5.1 Article Shared 5
6. Extension Points 5
    6.1 Share Method Options 5
7. Low Fidelity Prototype For Use Case 5
    7.1 Prompt Used for AI Tool 5
    7.2 Generated Interface 5

©Error! No property name supplied., 2025

Use-Case Specification: Share Article

1. Use-Case Name: Share Article

1.1 Brief Description

This use case describes the functionality that allows users to share the articles via social media such as Facebooks, Zalo, Instagram,...It provides choosing sharing method (e.g., timeline, private, message) and visibility (e.g. public, private)

2. Flow of Events

2.1 Basic Flow

1. The user views an article on the website.
2. The user scrolls to the end of the article and selects the “Share” option.
3. The system displays a list of available platforms such as Facebook, Zalo, Instagram, Messenger,...
4. The user selects a platform and confirms sharing.
5. The system prompts the user to choose share method options, including:
    a. Share method: timeline/feed, private message, group.
    b. Visibility: public, friends, private, or custom.
    c. Caption or personal message (optional).
6. The article link is shared to the selected platform.
7. A confirmation or error message is shown to the user:
    a. If successful: “Sharing article successfully!”
    b. If failed: “Failed to share the article. Please try again later”

2.2 Alternative Flows

2.2.1 User Cancels Sharing

- At step 4, the user cancels before confirming.
- The system returns to the article page without sharing

2.2.2 Social Account Having Been Logged In

- At step 5, if the user has not logged into the selected social media platform.
- The system prompts the user to log in before proceeding with the sharing process.
- If they exit log in, the sharing options will be canceled or fail.

2.2.3 Platform Error

- If an error occurs while sharing(e.g: API failure, no internet connection, invalid token,..)
- The system displays an error message: “Failed to share the article. Please try again later.”
- The user can retry or cancel.

3. Special Requirements

3.1 System Performance

- The sharing operation should complete within 3 seconds under normal conditions.

3.2 Compatibility

- The system must support all major browsers and mobile platforms.

4. Preconditions

4.1 User Authentication

- The user must be logged in with a registered account.

4.2 Social Media Login

- The user must have logged into a supported social media account (e.g.. Facebook, Instagram,..) when they want to take this sharing action.

5. Postconditions

5.1 Article Shared

- The selected article is successfully shared to the chosen platform, or the user receives an appropriate error message.

6. Extension Points

6.1 Share Method Options

- **Location in flow:** After step 4 in the Basic Flow (user selects a platform and confirms sharing).
- **Description:** Once a platform is selected, the system may allow the user to choose how the article is shared. Depending on each platform, options may include:
    1. Share to timeline/ feed
    2. Send via private message
    3. Share to a group or multiple recipients.
    4. Choose visibility: public, friends, private, or custom by yourself.
    5. Add a personal message or caption.

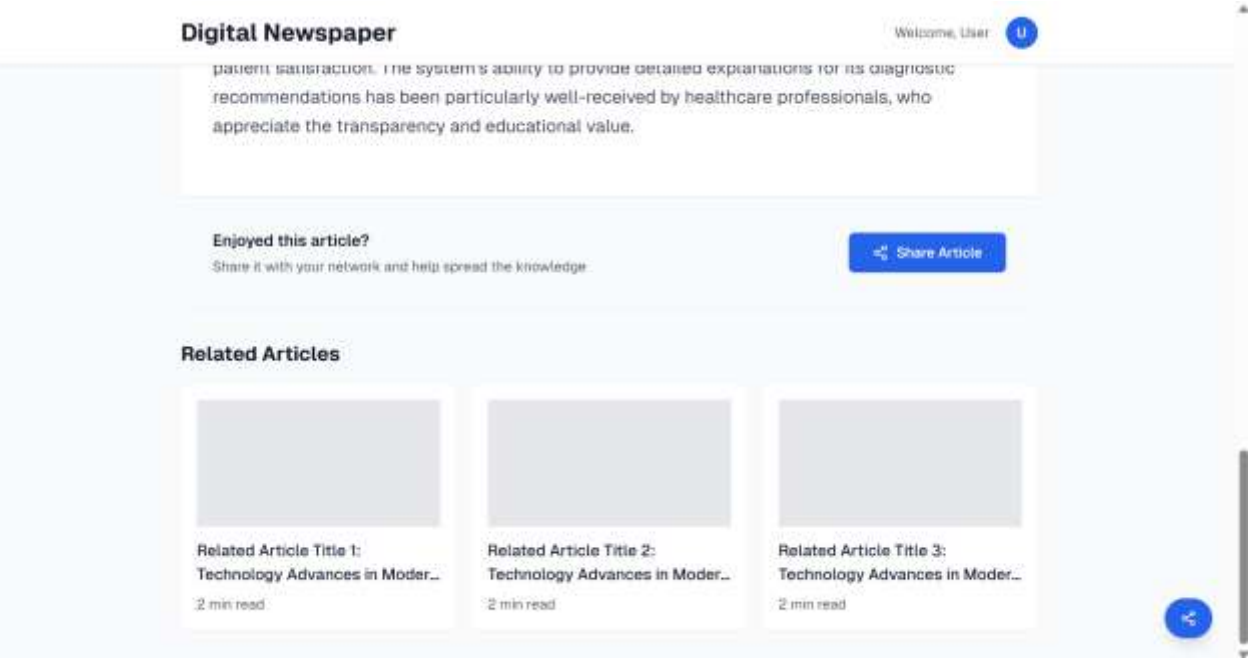
7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

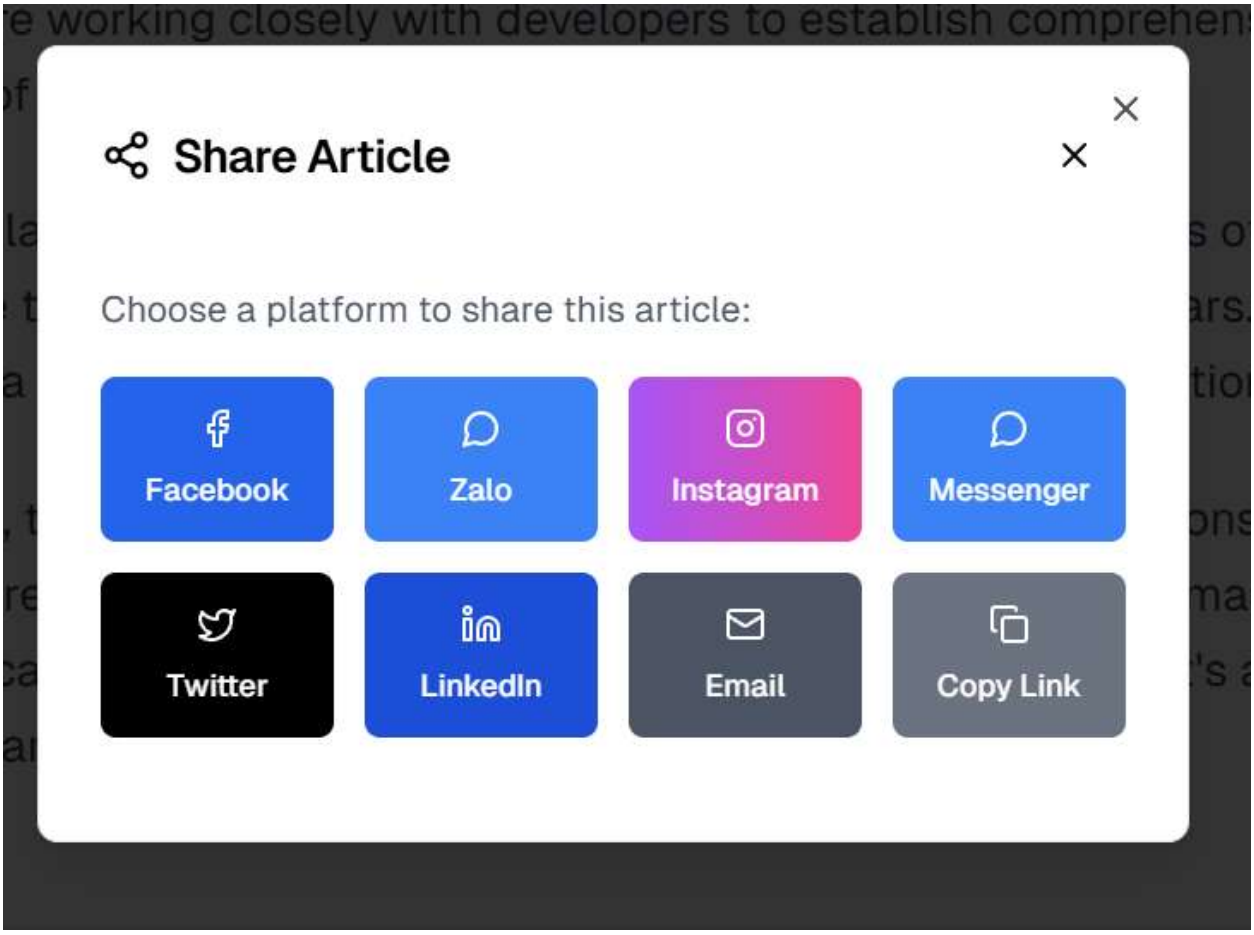
- Create a responsive web UI for a digital newspaper article page with a “Share Article” feature.
- Requirements: Meets all that have been specified in the use-case specification.
  [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*

- The UI should be mobile-friendly and support modern browsers. Prioritize usability and simplicity.

7.2 Generated Interface

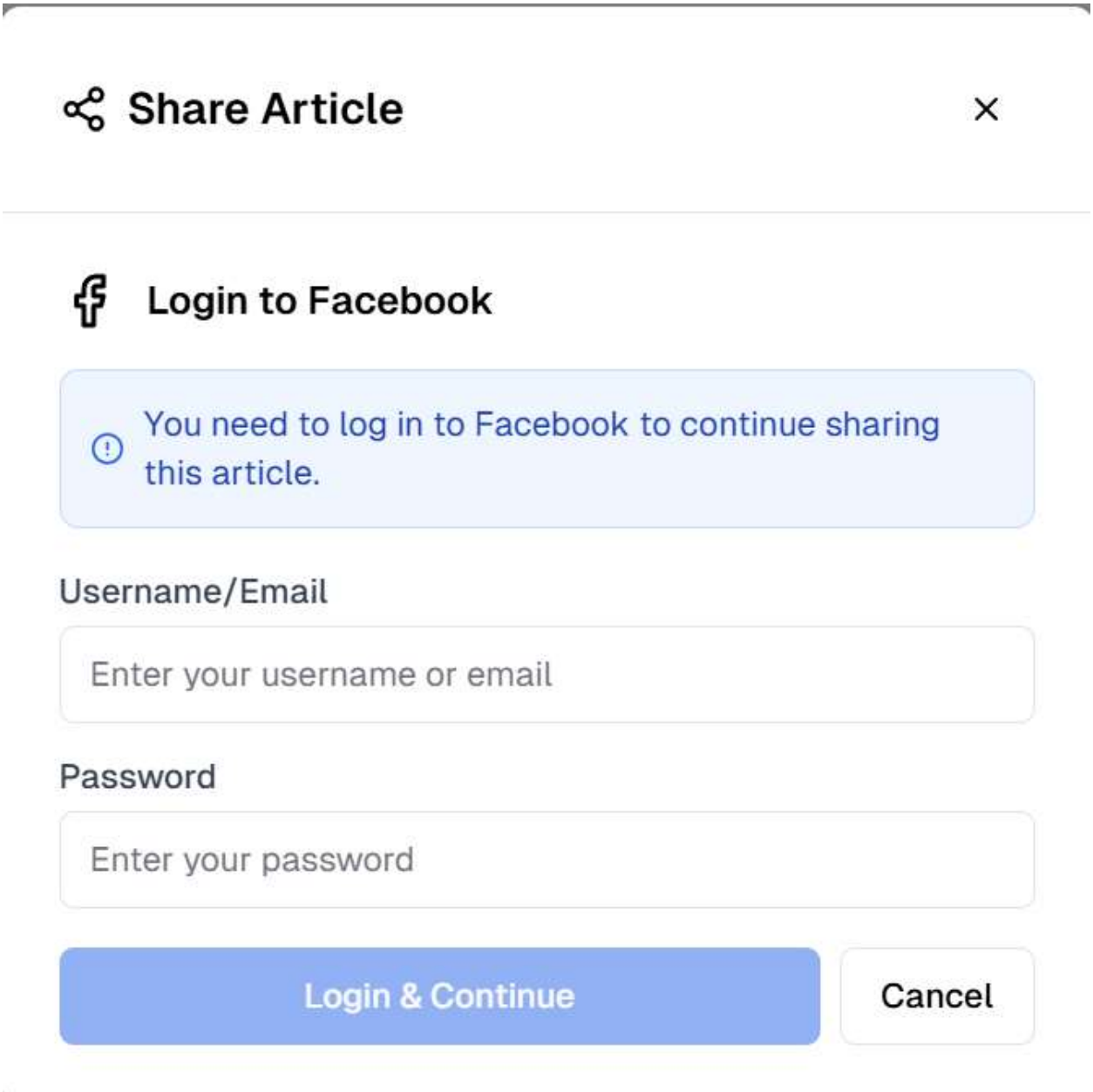


At the end of the article, the Share button is shown in the above picture.



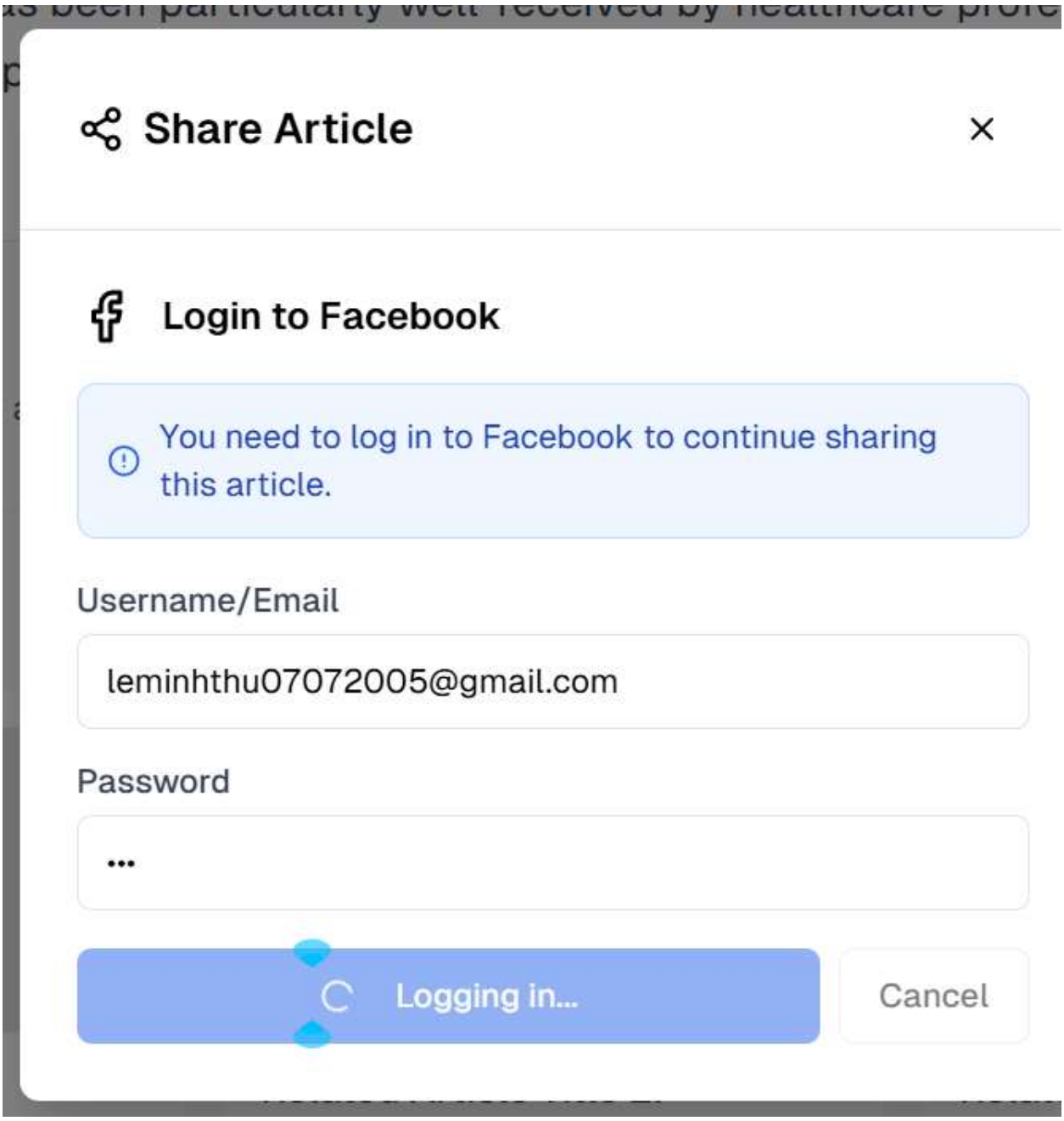
The Share Article box shown as soon as the user clicks the Share button.

After the user clicks to the Facebook platform. If the user hasn't logged in on that platform, the system requires the user to log in Facebook platform "You need to log in to Facebook first". The Back button will lead the user back to the previous screen.

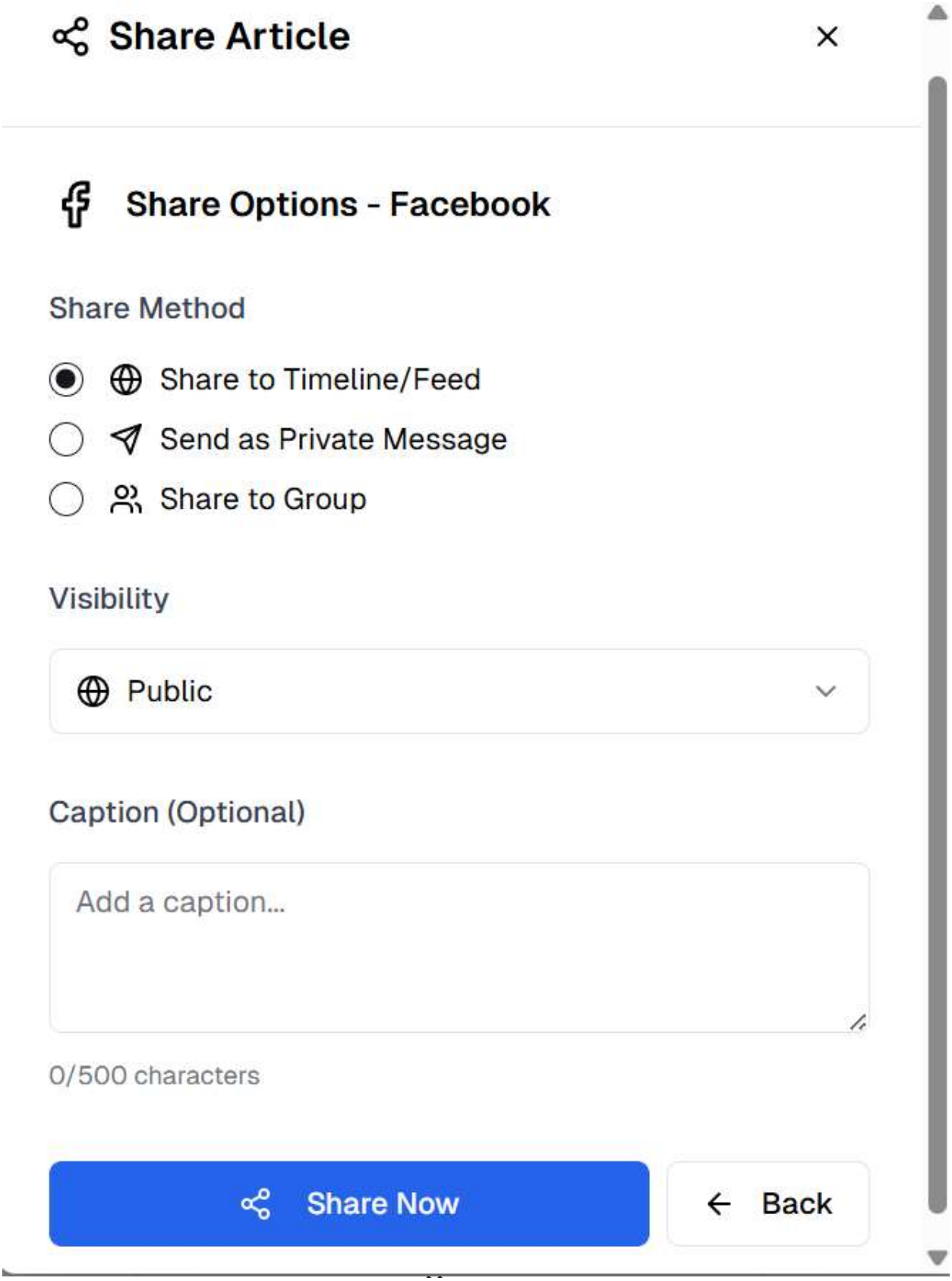


After clicking the Continue button, the Log in Screen is shown. Because no information is input, the Login&Continue button is invalid.

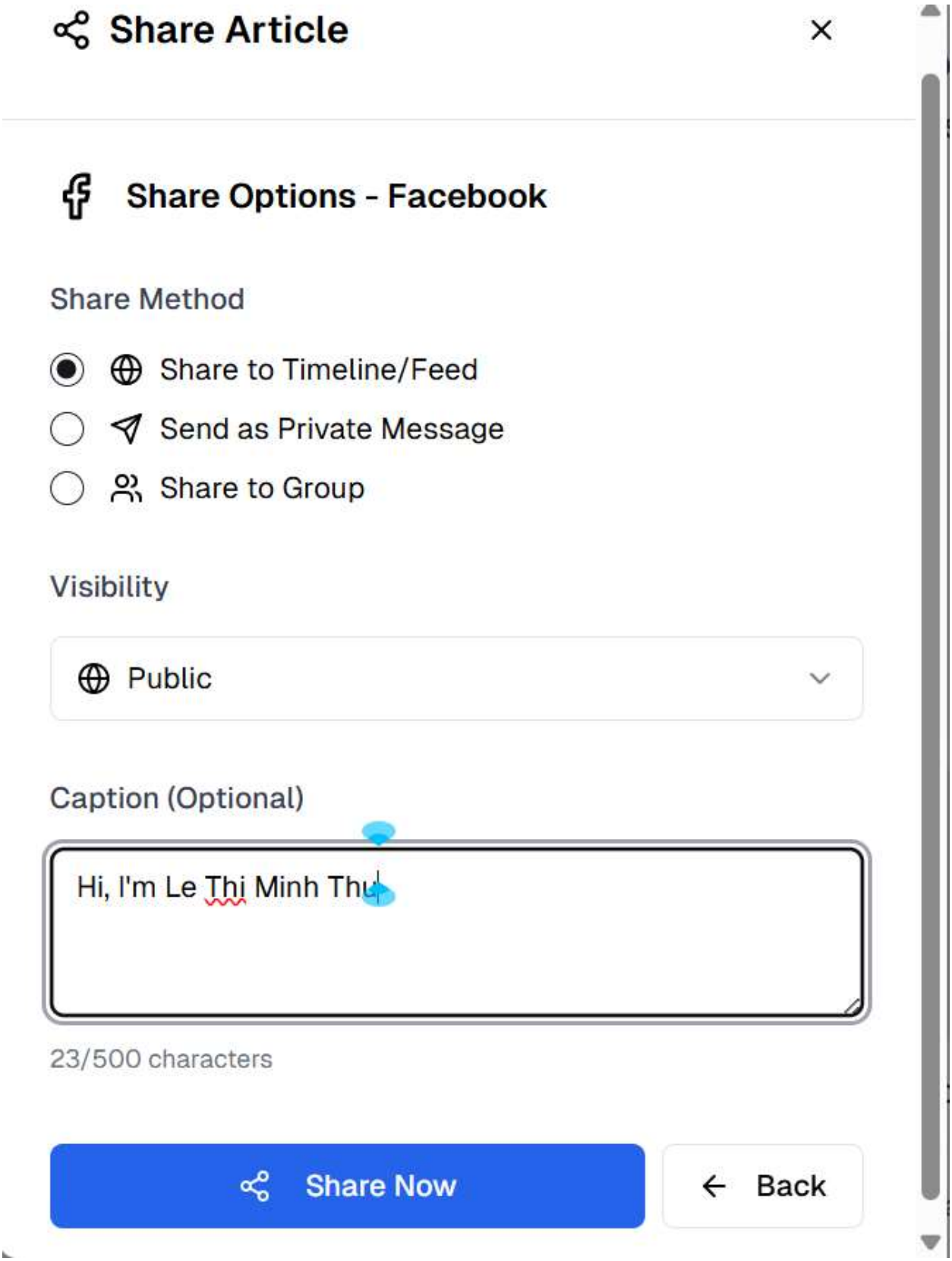
When the user fills all in the Log in screen. Login & Continue button is valid.



The screen shows as soon as the user has clicked the Login & Continue button.

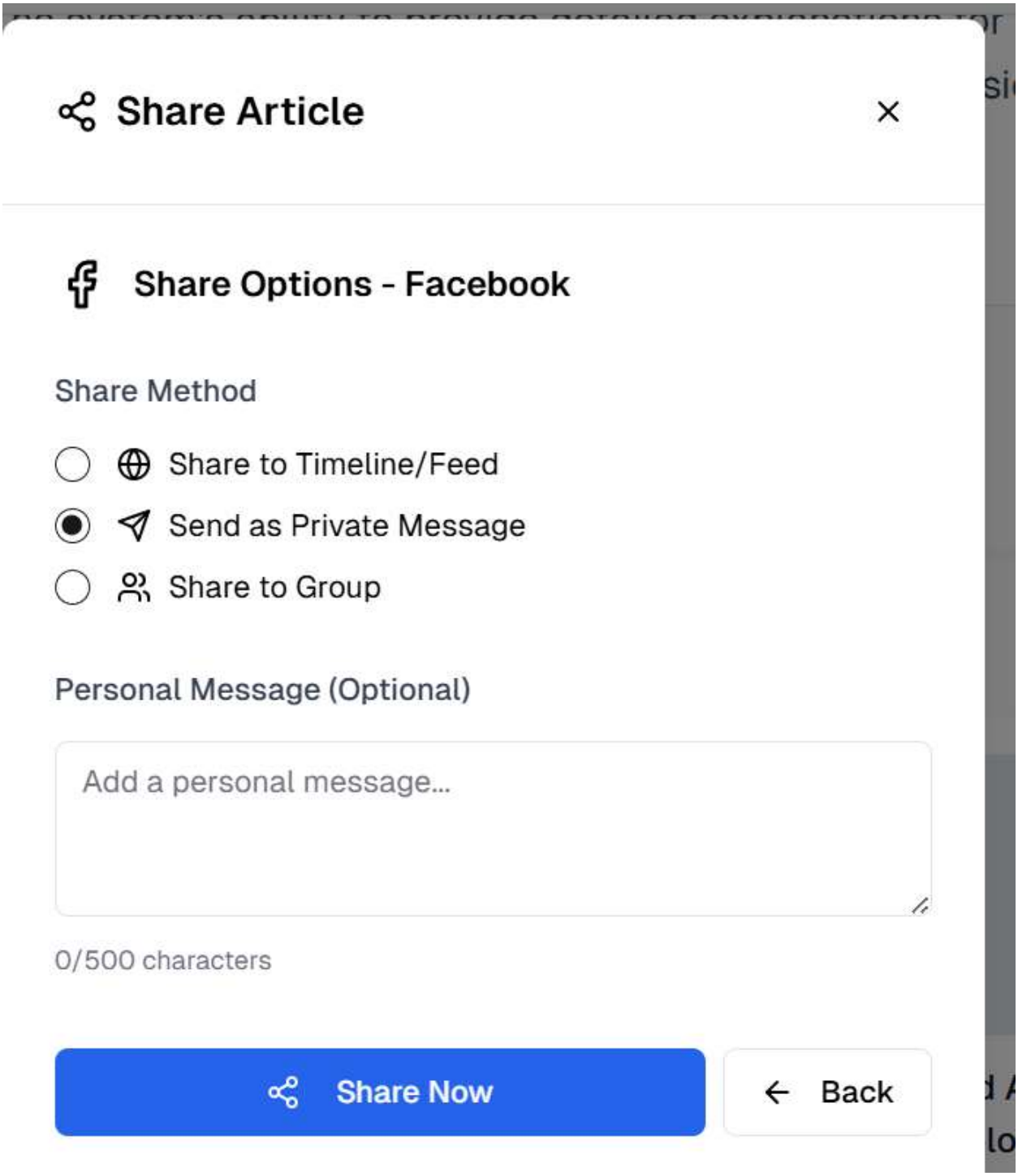

Share Article
Share Options - Facebook
Share Method
Share to Timeline/Feed
Send as Private Message
Share to Group
Visibility
Public
Caption (Optional)
Add a caption...
0/500 characters
Share Now
Back

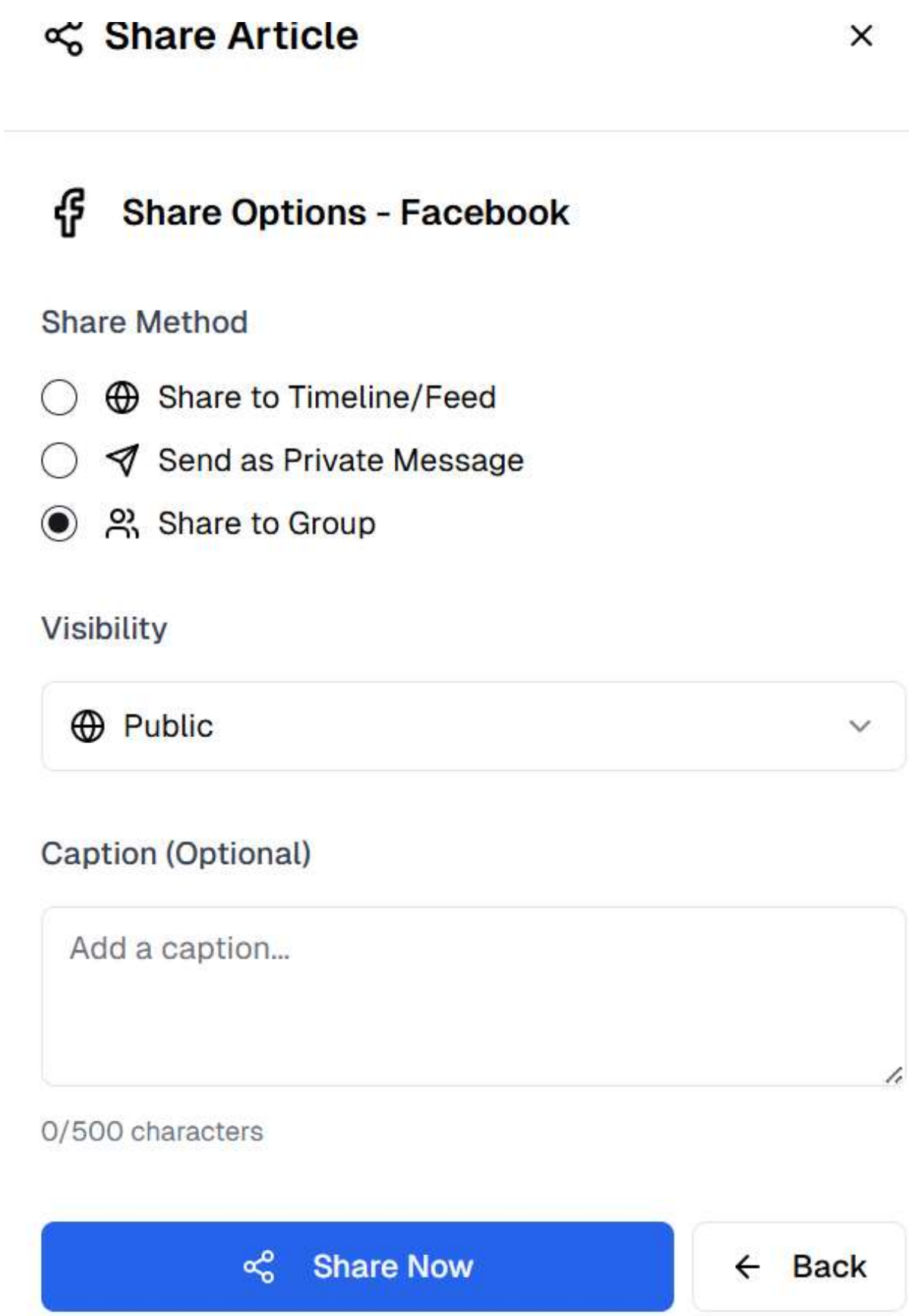
The screen after the system has just processed the Login processing. The default sharing method is Public and share to Feed/Timeline.



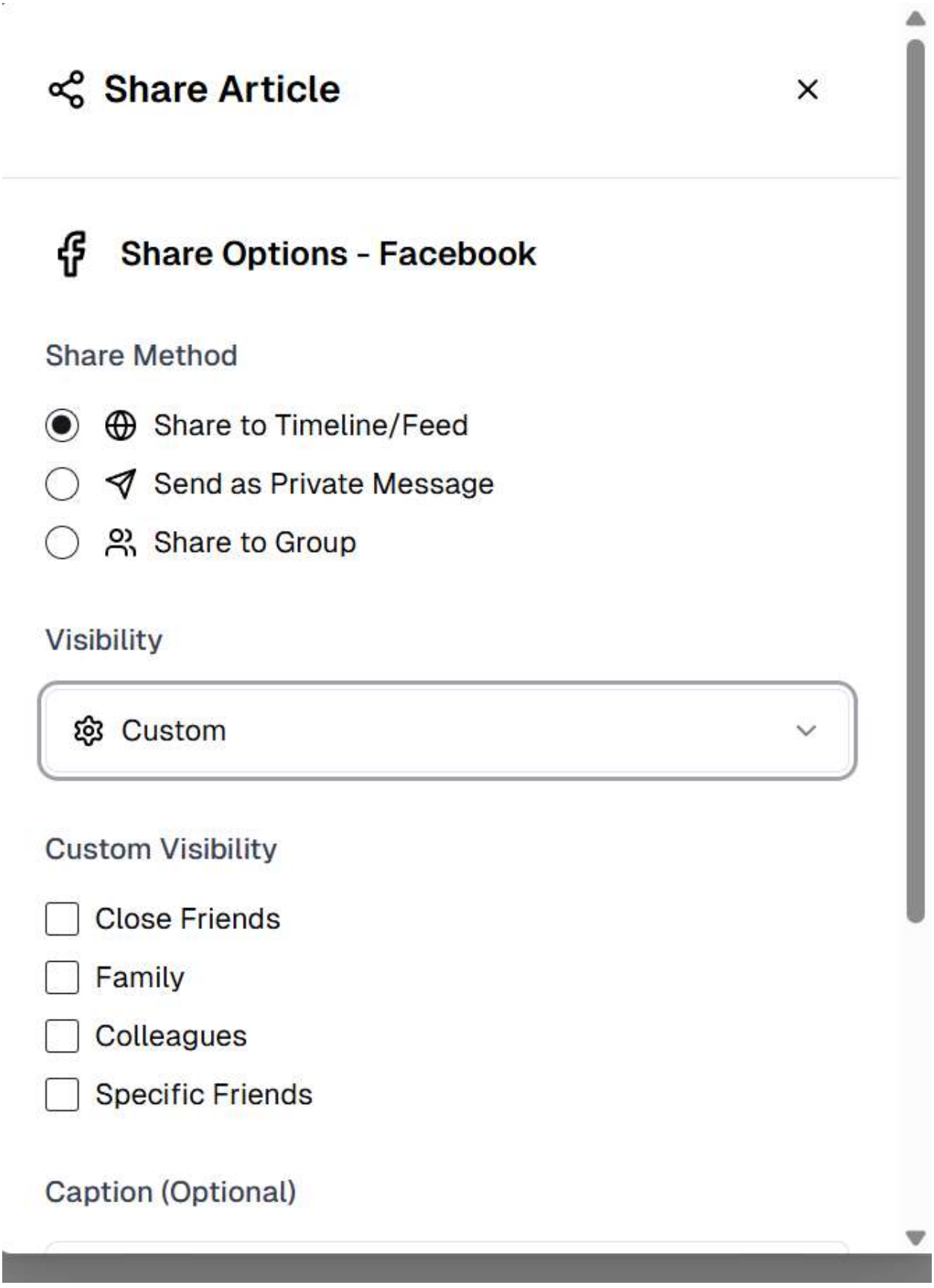
supplied., 2025

The users can add captions for their Sharing Arrticle posts.

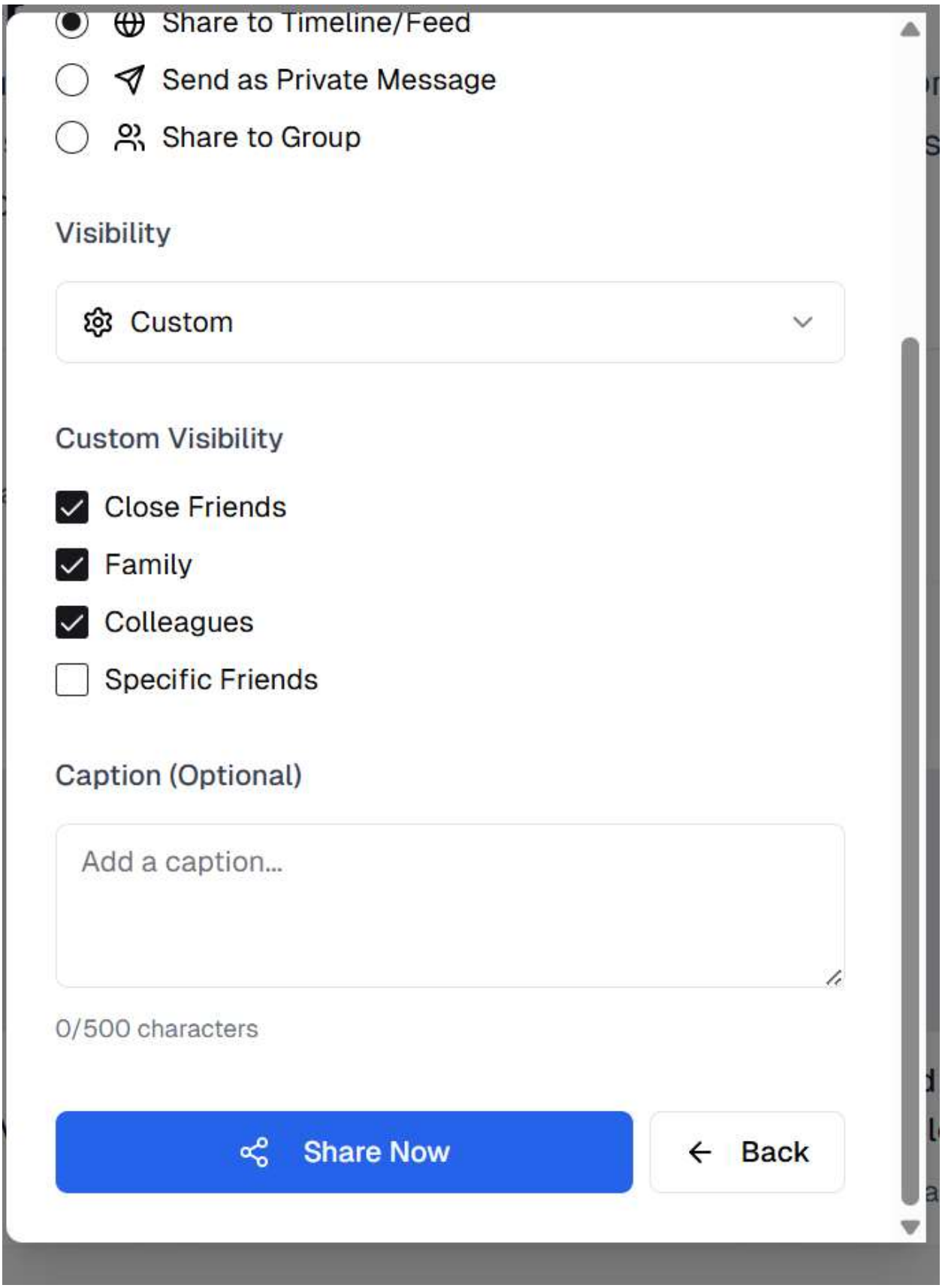


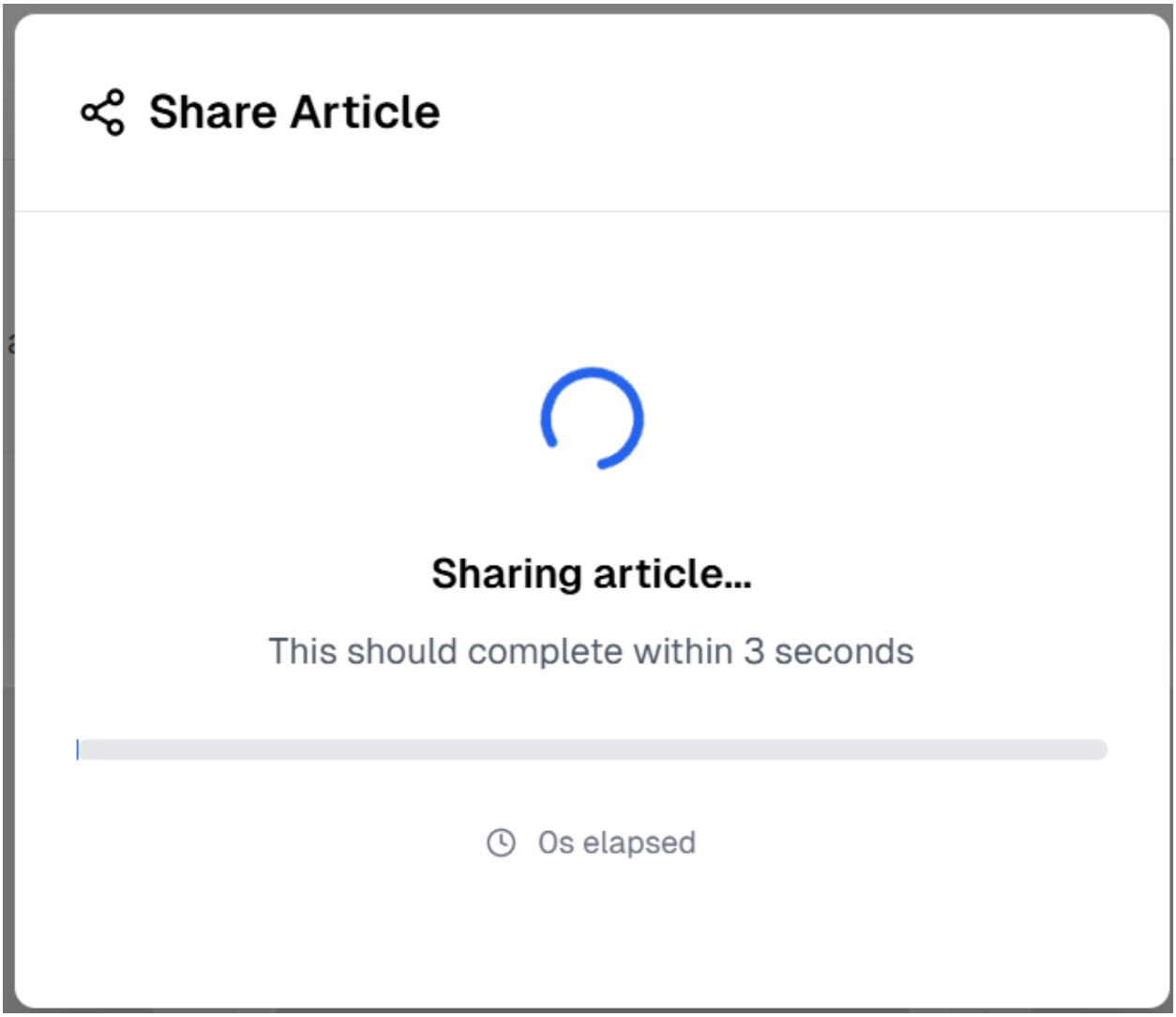

Share Article
Share Options - Facebook
Share Method
Share to Timeline/Feed
Send as Private Message
Share to Group
Visibility
Public
Caption (Optional)
Add a caption...
0/500 characters
Share Now
Back

Or the user can choose to share for a specific group/ Send as Private Message.

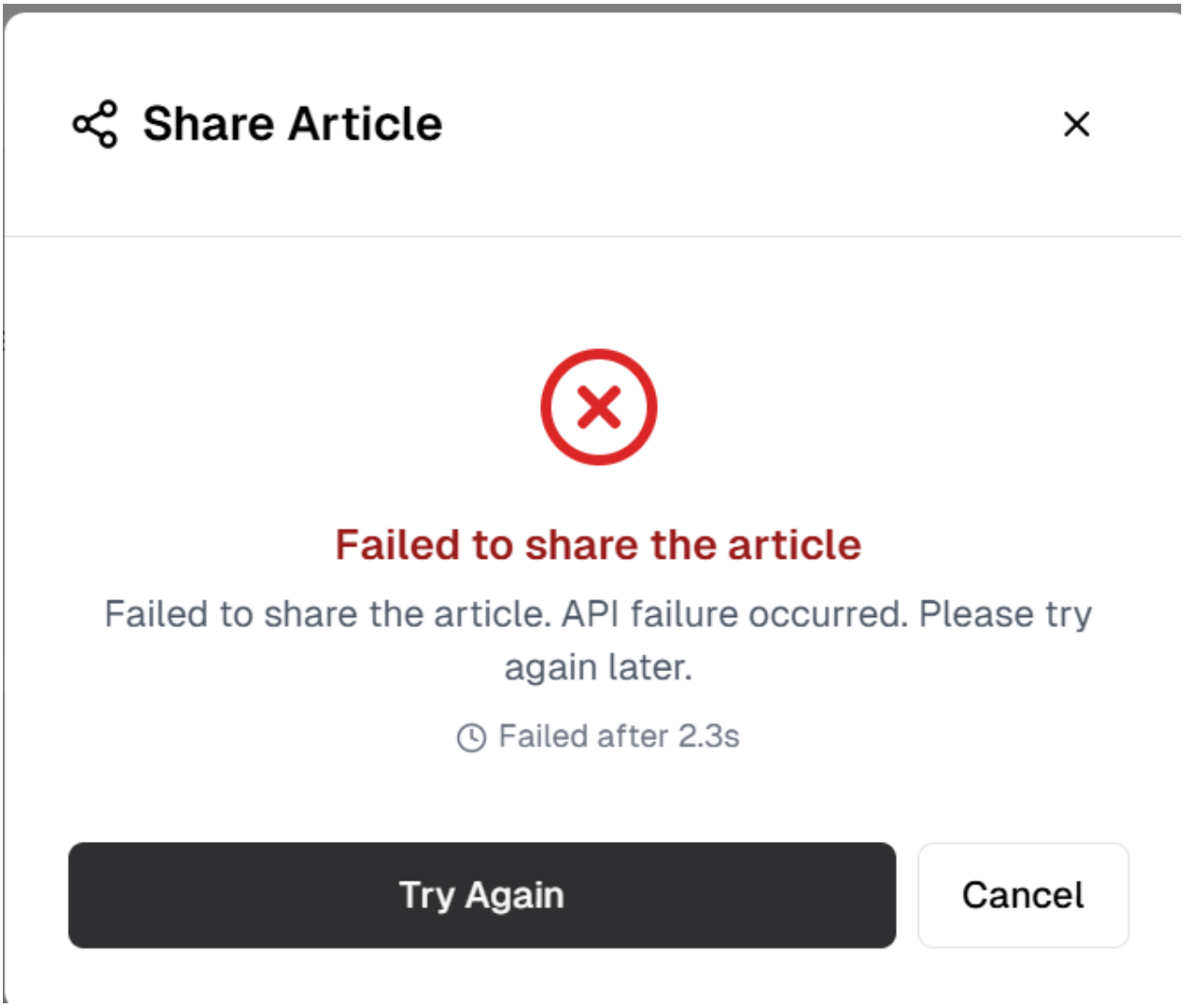

Share Article
Share Options - Facebook
Share Method
Share to Timeline/Feed
Send as Private Message
Share to Group
Visibility
Custom
Custom Visibility
Close Friends
Family
Colleagues
Specific Friends
Caption (Optional)

The users can Custom their Sharing Visibility

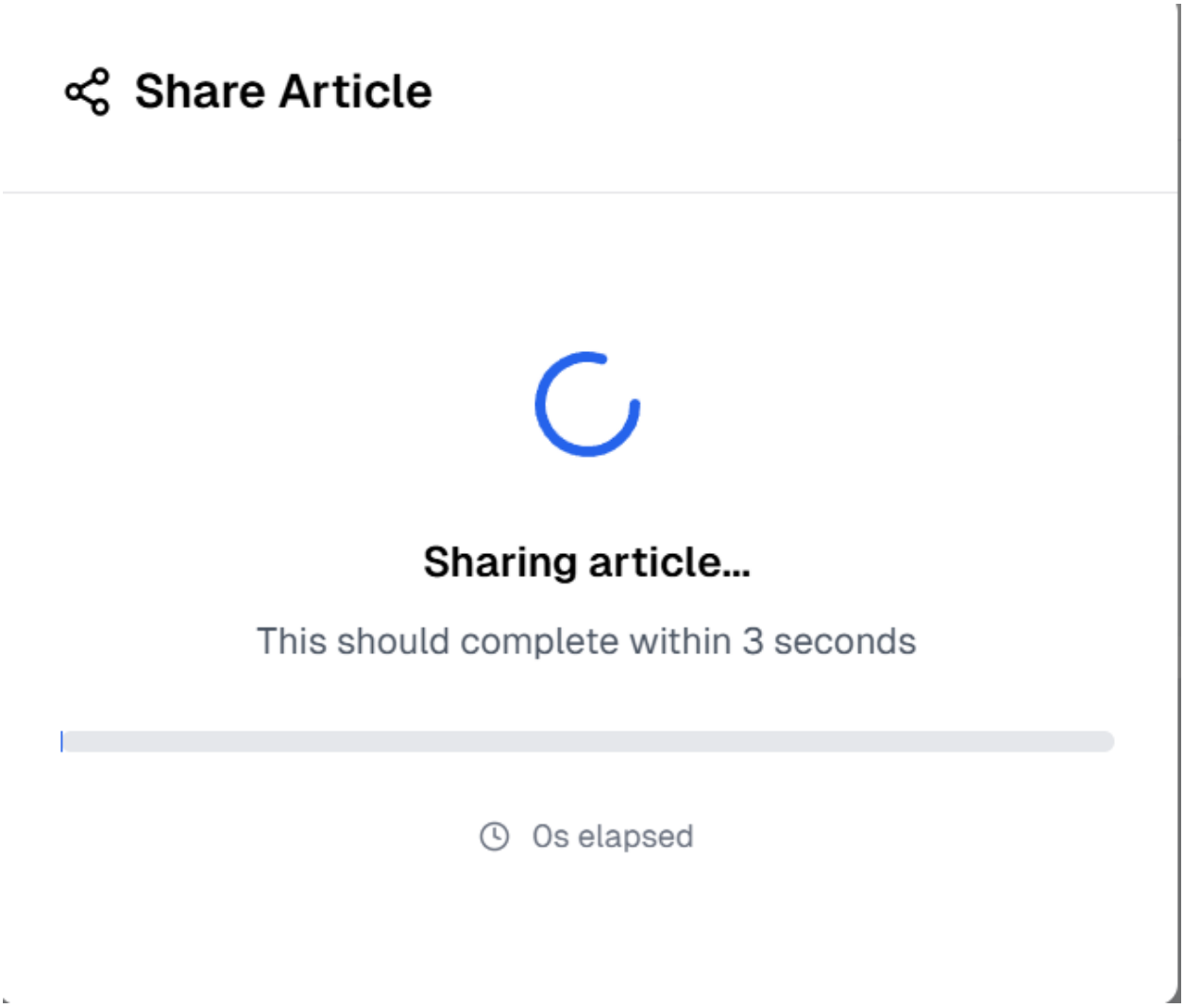

Share to Timeline/Feed
Send as Private Message
Share to Group
Visibility
Custom
Custom Visibility
Close Friends
Family
Colleagues
Specific Friends
Caption (Optional)
Add a caption...
0/500 characters
Share Now
Back



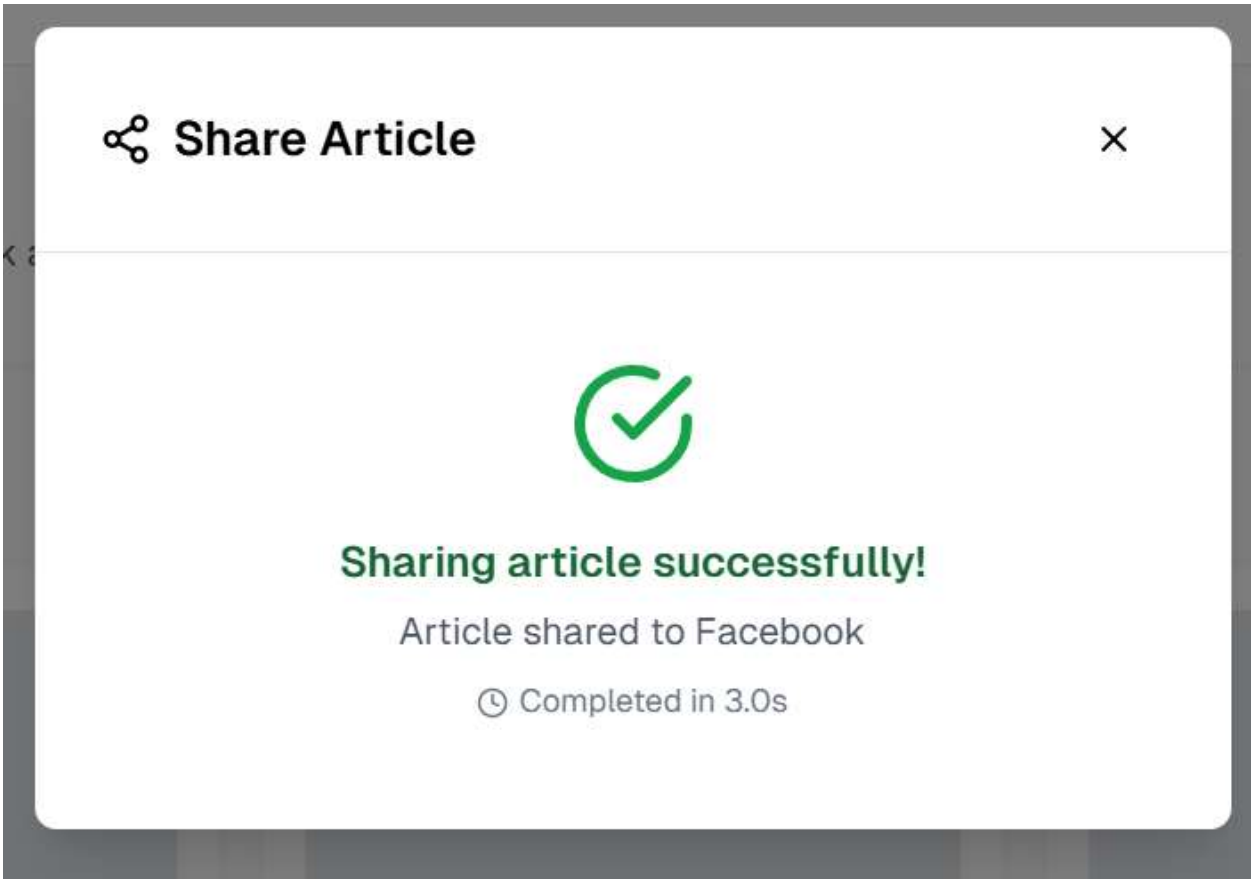
The screen shows when the system processes the Sharing processing.



The system sends an error message "Failed to share the article", and the mistake occurs (e.g API). The Cancel will lead the user go back to the Multiple Platform Selections screen (screen after clicking the Share button).



©Error! No property name supplied., 2025



The system messages "Sharing article successfully!" straight after the article has been shared.

404 Not Found

Digital Newspaper Website
Use-Case Specification: Save Article to Favourite

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
20/06/2025	1.0	Specific about the Save Article to Favourite use case	Thạch Ngọc Hân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Save Article to Favourites **4**

1.1 Brief Description 4

2. Flow of Events **4**

2.1 Basic Flow 4

2.2 Alternative Flows 4

2.2.1 Alternative Flow 1: User Not Logged In 4

3. Special Requirements **4**

3.1 Performance: The save operation should complete within 1 second. 4

3.2 Data Integrity: Prevent duplication of favourite records for the same article and user. 4

4. Preconditions **4**

5. Postconditions **5**

5.1 The article appears in the user's list of favourite articles. 5

6. Extension Points **5**

6.1 Allows the user to unfavourite a previously saved article. 5

©Error! No property name supplied., 2025

Use-Case Specification: Save Article to Favourite

1. Use-Case Name: Save Article to Favourites

1.1 Brief Description

This use case describes the process by which a logged-in user saves an article to their list of favourites. This enables the user to easily access preferred content later via their personal profile or favourites section.

2. Flow of Events

2.1 Basic Flow

1. The use case begins when the user views an article.
2. The user clicks the "Save to Favourite" button associated with the article.
3. The system checks whether the user is authenticated.
4. The system adds the article to the user's list of Favourite articles in the database.
5. The system confirms the save with a notification such as "Article saved to favourites"
6. The use case ends

2.2 Alternative Flows

2.2.1 Alternative Flow 1: User Not Logged In

At step 3 in the Basic Flow, if the user is not authenticated:

a. The system redirects the user to the login page or displays a login prompt.
b. After successful login, the system redirects the user back to the article.
c. The user may then proceed to save the article.

2.2.2 *Alternative Flow 2: Article Already in Favourites*

At step 4 in the Basic Flow, if the article is already in the user's favourites:

a. The system displays a message: "This article is already in your favourites."
b. The use case ends without duplicating the entry.

3. Special Requirements

3.1 Performance: The save operation should complete within 1 second.

3.2 Data Integrity: Prevent duplication of favourite records for the same article and user.

3.3 Security: Only authenticated users can add favourites; all actions must be logged for audit.

4. Preconditions

4.1 The article must be published and visible to the user.
4.2 The user must be authenticated (or be prompted to authenticate).

5. Postconditions

5.1 The article appears in the user's list of favourite articles.

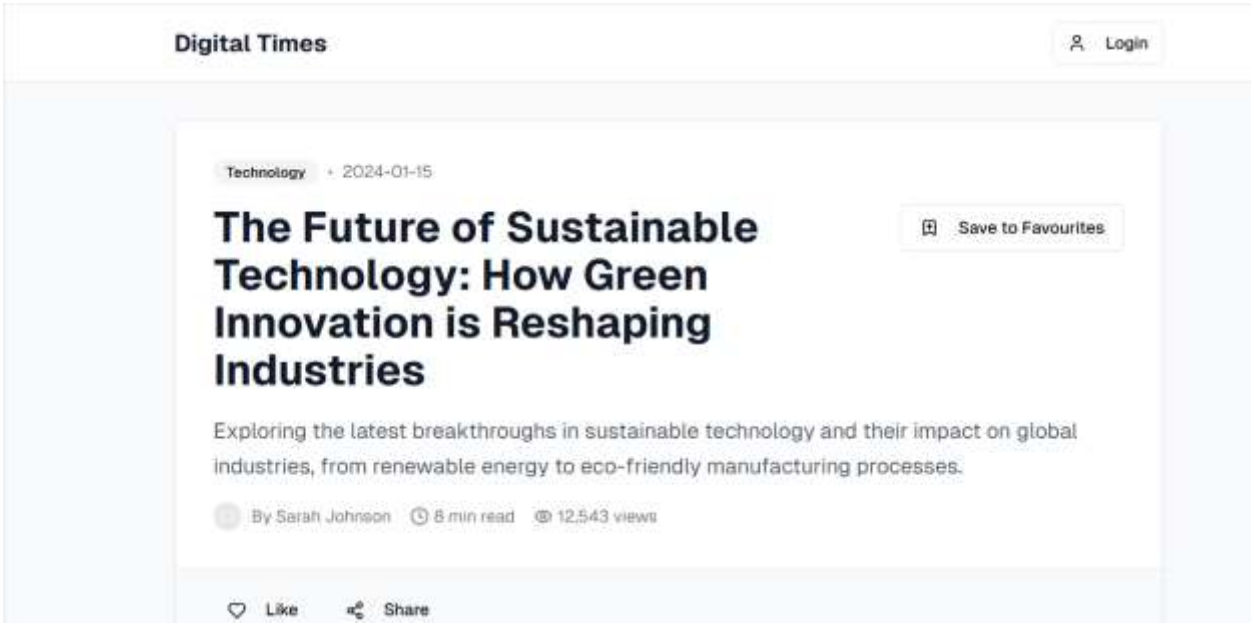
5.2 The system confirms success or notifies if the article is already saved.

©Error! No property name supplied., 2025

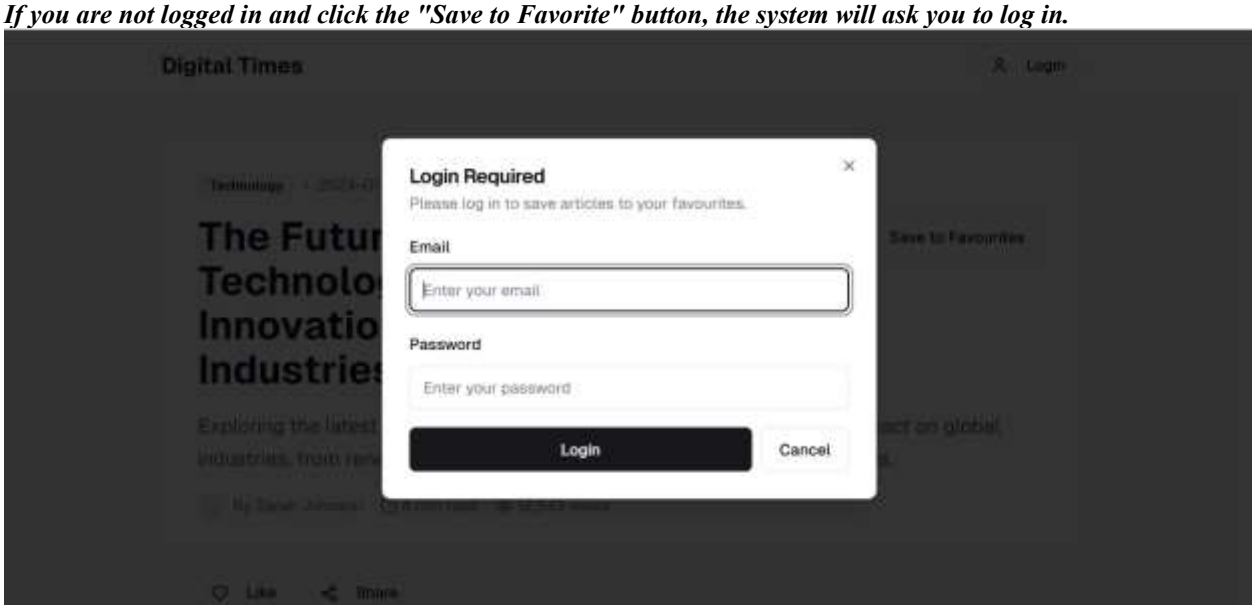
6. Extension Points

6.1 Allows the user to unfavourite a previously saved article.

Original interface



If you are not logged in and click the "Save to Favorite" button, the system will ask you to log in.



Interface after successful login

After saving successfully, a notification will appear. The notification will disappear after a few seconds.

404 Not Found

Digital Newspaper Website
Use-Case Specification: View Reading History

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
20/06/2025	1.0	Specific about View Reading History use case	Thạch Ngọc Hân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: View Reading History **4**

1.1 Brief Description 4

2. Flow of Events **4**

2.1 Basic Flow 4

2.2 Alternative Flows 4

2.2.1 No Reading History 4

3. Special Requirements **4**

3.1 Data Privacy 4

4. Preconditions **4**

4.1 The user must be logged in to access personal reading history. 4

5. Postconditions **4**

5.1 The reading history page is populated with previously accessed article records, or a notice is shown if none exist. 4

6. Extension Points **4**

6.1 Clear Reading History 4

©Error! No property name supplied., 2025

Use-Case Specification: View Reading History

1. Use-Case Name: View Reading History

1.1 Brief Description

This use case allows a logged-in user to view a list of articles they have previously read. The history provides links to past articles and may include metadata such as title, author and read date to improve user personalization.

2. Flow of Events

2.1 Basic Flow

1. The use case begins when the user navigates to the "Reading History" page from their profile or main menu.
2. The system checks if the user is authenticated.
3. The system retrieves the list of articles previously read by the user from the database, sorted by the most recent activity.
4. The system displays the reading history, including article titles, authors and timestamps.
5. The user may click on any listed article to view it again.
6. The use case ends.

2.2 Alternative Flows

2.2.1 No Reading History

At step 3 of the Basic Flow, if no reading data is found:

a. The system displays the message: “You haven’t read any articles yet.”
b. The use case ends.

3. Special Requirements

3.1 Data Privacy

1. Reading history must be stored securely and accessible only to the authenticated user.
2. The system must comply with data protection regulations
3. History data should be automatically removed after a defined retention period, if required by privacy policy.

4. Preconditions

4.1 The user must be logged in to access personal reading history.

5. Postconditions

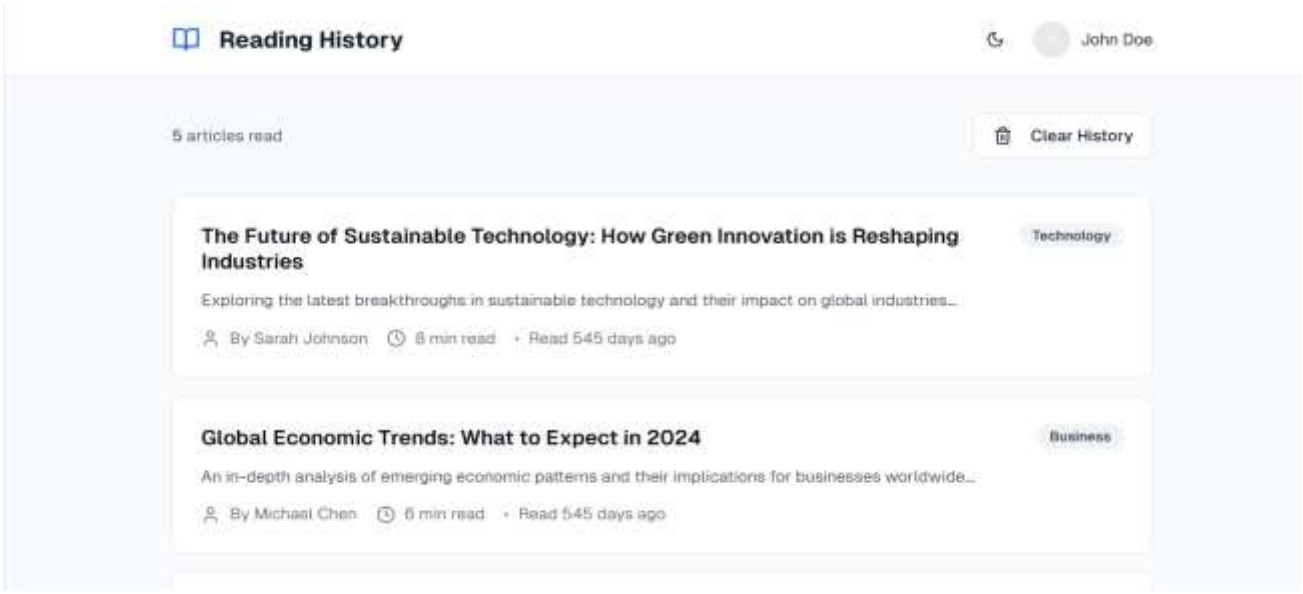
5.1 The reading history page is populated with previously accessed article records, or a notice is shown if none exist.

6. Extension Points

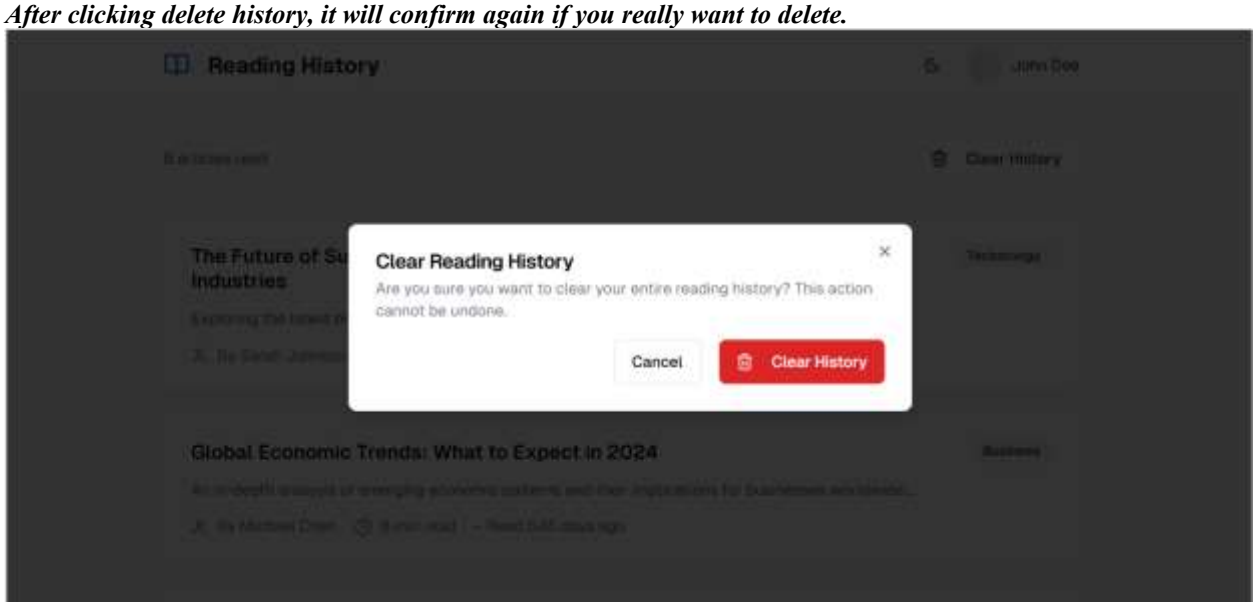
6.1 Clear Reading History

- Location: After step 4 of the Basic Flow.
- Description: The user may choose to delete their entire reading history via a "Clear History" button. The system confirms the action before deletion.

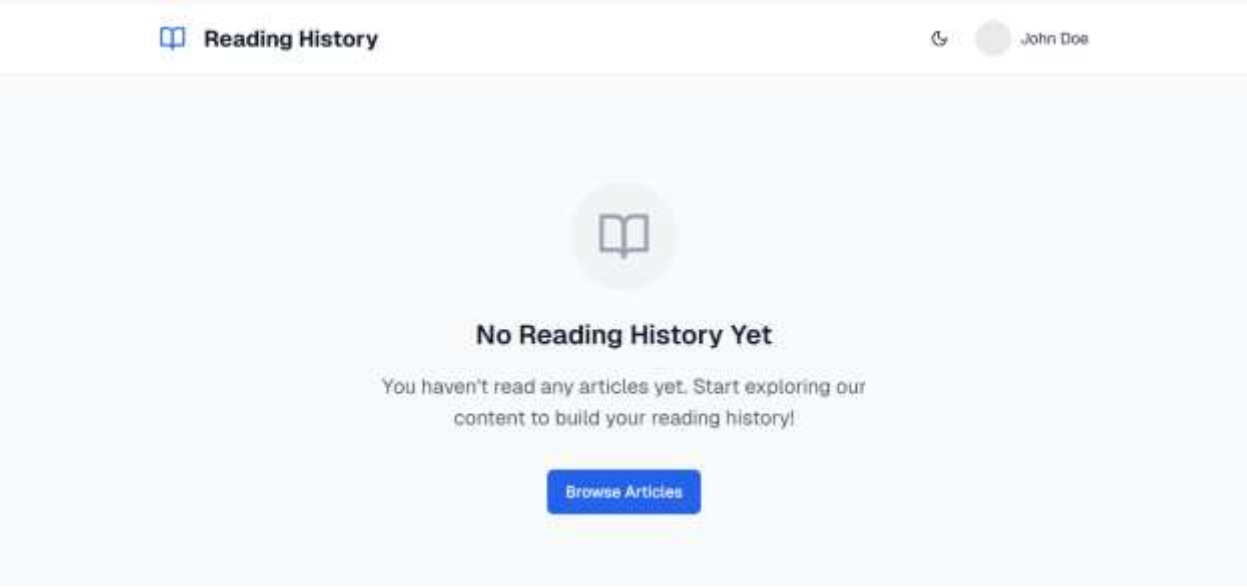
You must log in to view history



After clicking delete history, it will confirm again if you really want to delete.



Interface after deleting all history



©Error! No property name supplied., 2025

404 Not Found

Digital Newspaper System
Use-Case Specification: View Statistics

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Specific about View Statistics use case	Thạch Ngọc Hân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: View Statics **4**

1.1 Brief Description 4

2. Flow of Events **4**

2.1 Basic Flow 4

2.2 Alternative Flows 4

2.2.1 No Data Available 4

3. Special Requirements **4**

3.1 Performance and Visualization 4

4. Preconditions **4**

4.1 The user must be logged in with the role of Editor, Admin, or Author. 4

5. Postconditions **4**

6. Extension Points **4**

6.1 Edit/Delete Review 4

©Error! No property name supplied., 2025

Use-Case Specification: View Statics

1. Use-Case Name: View Statics

1.1 Brief Description

This use case allows an editor, admin, or writer to view analytics and statistics related to article performance, user engagement, or platform activity. These may include metrics such as article views, likes, shares and comments.

2. Flow of Events

2.1 Basic Flow

1. The use case begins when the user navigates to the Statistics Dashboard from their profile or admin panel.
2. The system verifies the User's role and permissions.
3. The user selects the type of statistics they want to view (e.g., by article, author, date range, or category).
4. The system retrieves relevant data from the analytics database.
5. The system generates and displays visualizations such as charts, tables, or summary cards.
6. The user may interact with filters or drill-down options for deeper analysis.
7. The use case ends

2.2 Alternative Flows

2.2.1 No Data Available

At step 4 of the Basic Flow, if no statistics are found for the selected criteria:

a. The system displays the message: "No statistics available for the selected range."
b. The user may choose a different date range or filter to continue.

3. Special Requirements

3.1 Performance and Visualization

1. Statistics must load within 3 seconds for standard date ranges.
2. Visuals should be responsive, accessible, and interactive
3. System must prevent access to restricted or sensitive data

4. Preconditions

4.1 The user must be logged in with the role of Editor, Admin, or Author.

5. Postconditions

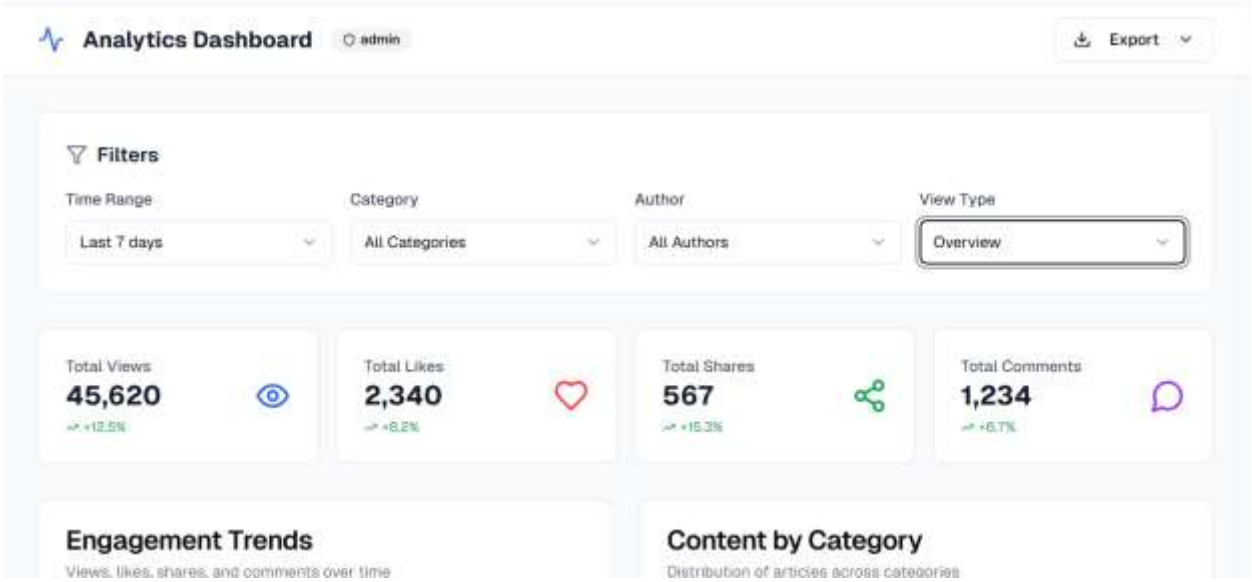
5.1 The requested statistics are shown in a clear, visual format, and the user can derive insights from them.

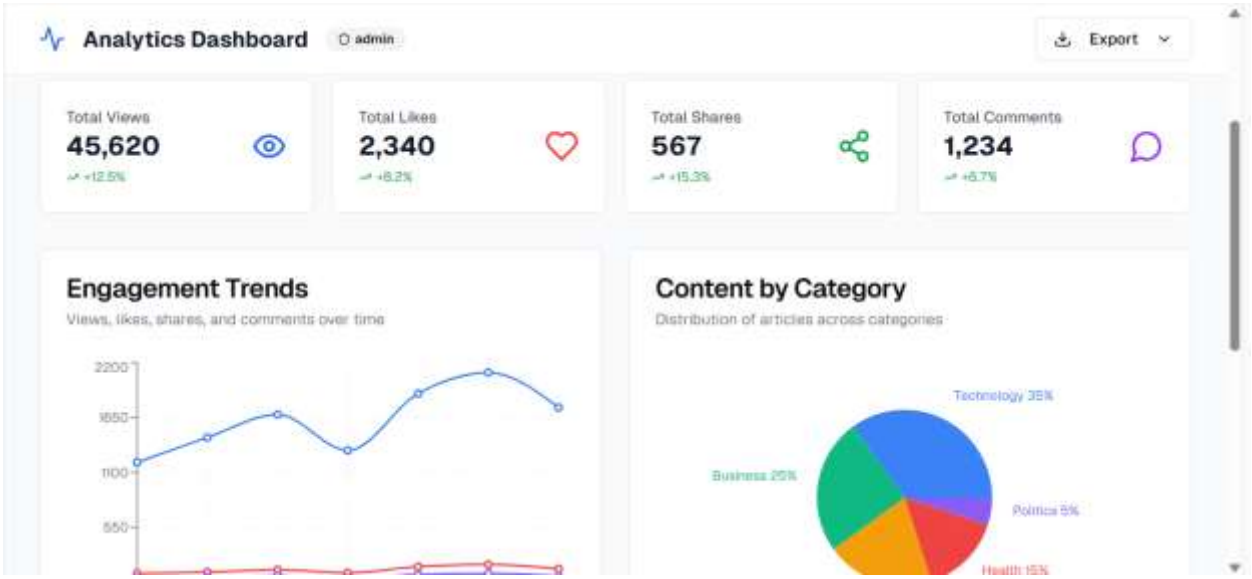
6. Extension Points

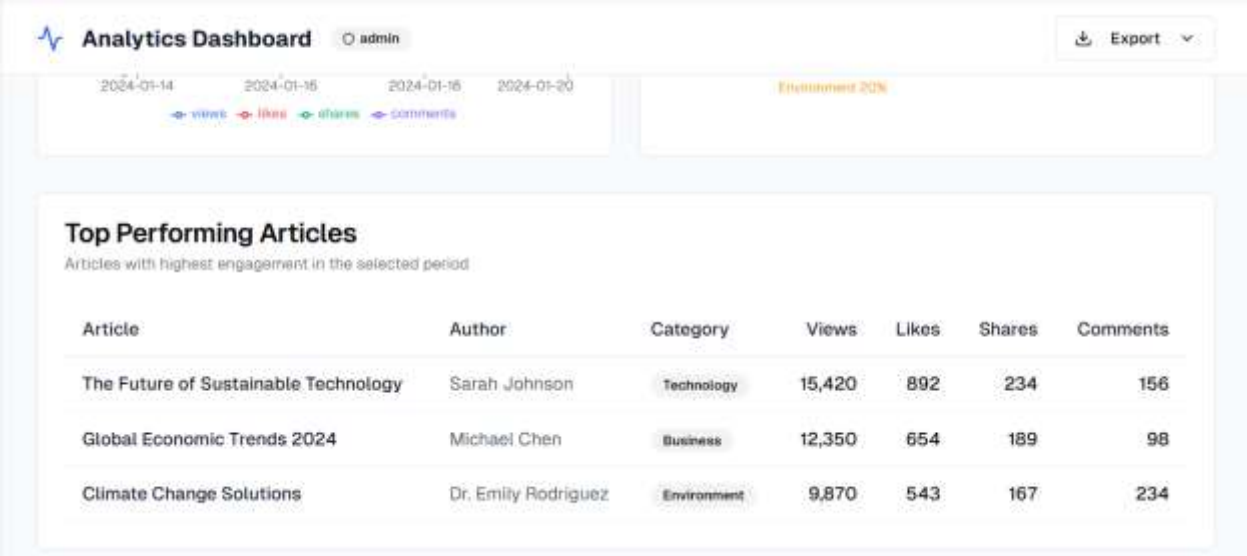
6.1 Edit/Delete Review

- Location: After step 5 of the Basic Flow.
- Description: The user may export current statistics in formats like CSV, PDF, or Excel for offline analysis or reporting.

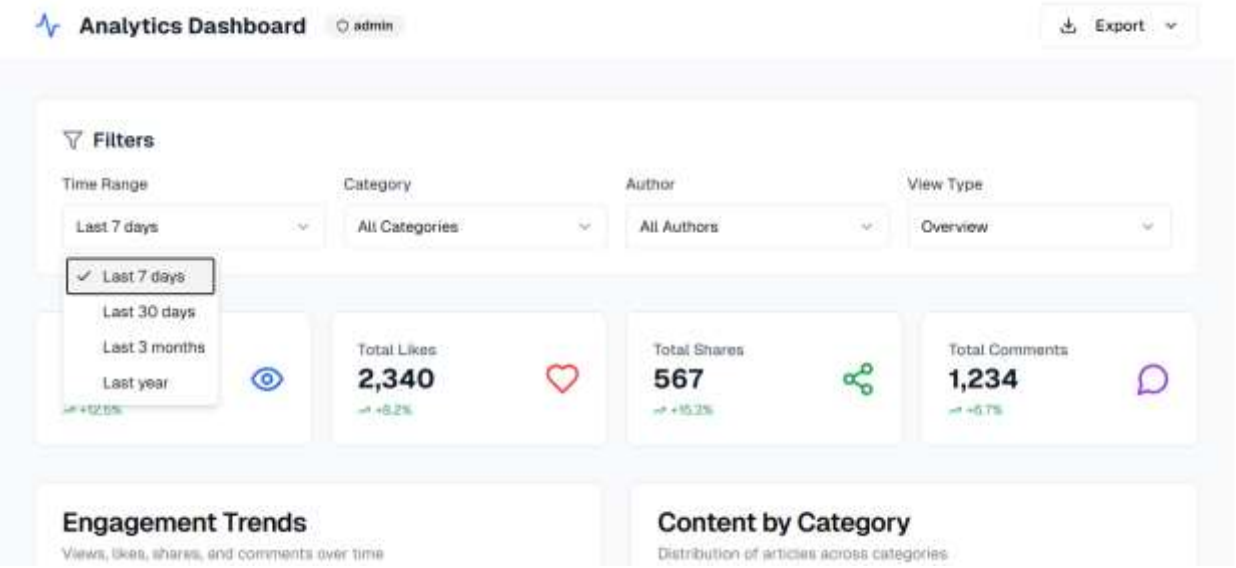
Only when logged in as admin can view

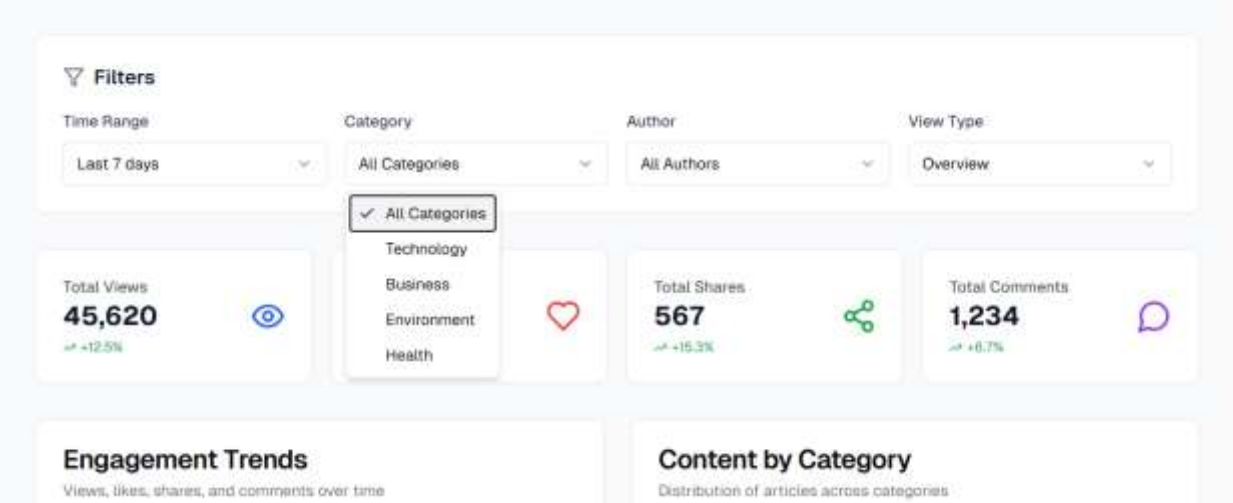

Analytics Dashboard
admin
Export
Filters
Time Range
Last 7 days
Category
All Categories
Author
All Authors
View Type
Overview
Total Views
45,620
Total Likes
2,340
Total Shares
567
Total Comments
1,234
Engagement Trends
Views, likes, shares, and comments over time
Content by Category
Distribution of articles across categories

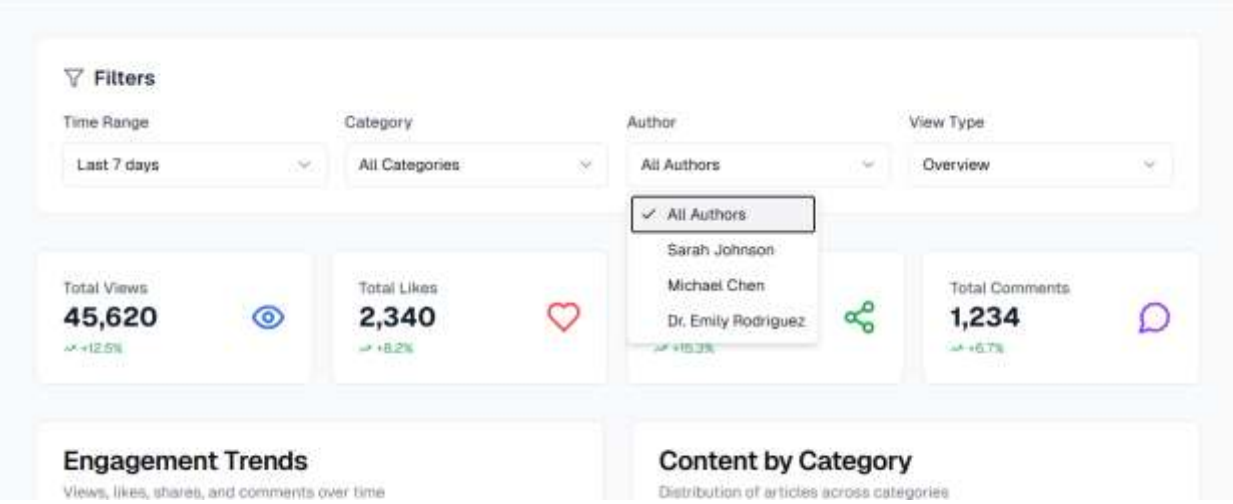

Analytics Dashboard
admin
Export
Total Views
45,620
Total Likes
2,340
Total Shares
567
Total Comments
1,234
Engagement Trends
Views, likes, shares, and comments over time
2200
1650
1100
550
Content by Category
Distribution of articles across categories
Technology 35%
Business 25%
Politics 8%
Health 15%

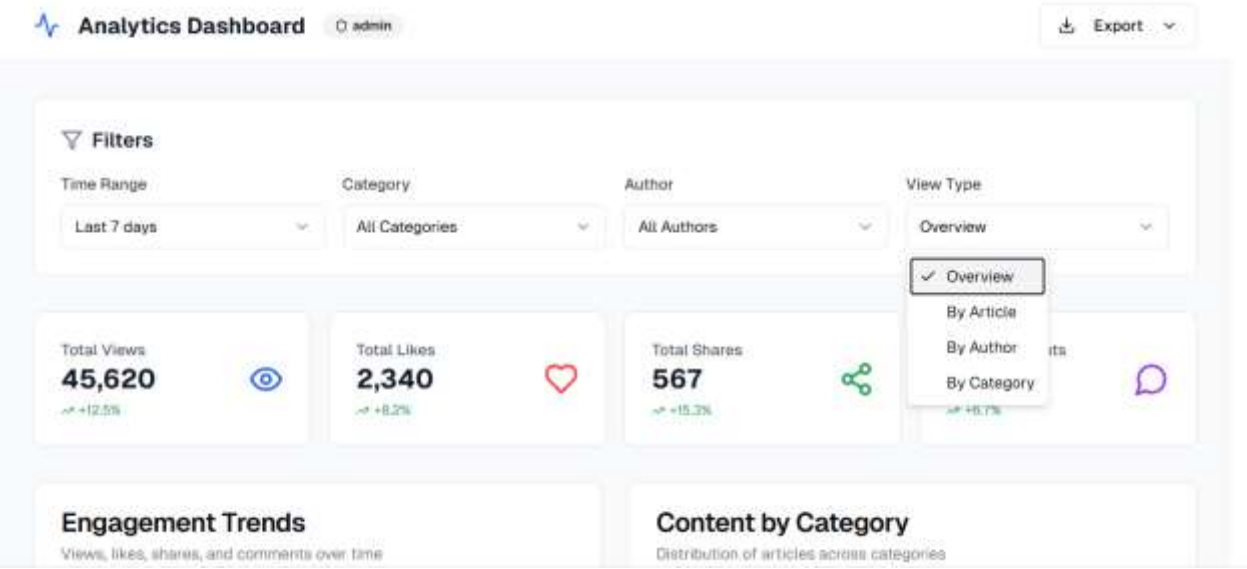


Filters to see more details

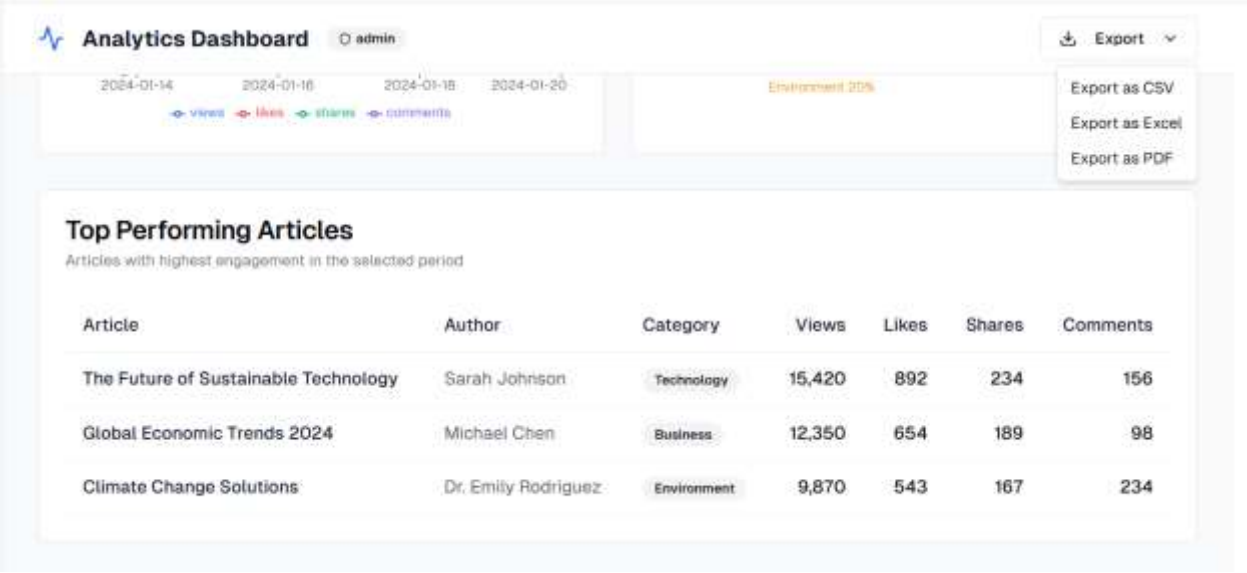



Filters
Time Range
Last 7 days
Category
All Categories
Author
All Authors
View Type
Overview
All Categories
Technology
Business
Environment
Health
Total Views
45,620
+12.5%
Total Shares
567
+15.3%
Total Comments
1,234
+6.7%
Engagement Trends
Views, likes, shares, and comments over time
Content by Category
Distribution of articles across categories


Filters
Time Range
Last 7 days
Category
All Categories
Author
All Authors
View Type
Overview
All Authors
Sarah Johnson
Michael Chen
Dr. Emily Rodriguez
Total Views
45,620
+12.5%
Total Likes
2,340
+8.2%
Total Comments
1,234
+6.7%
Engagement Trends
Views, likes, shares, and comments over time
Content by Category
Distribution of articles across categories



Different types of fillers can be downloaded.



404 Not Found

Digital Newspaper Website
Use-Case Specification: Search Articles

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
07/07/2025	1.0	Specific about Search Articles use case	Lê Thị Minh Thư

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Search Articles 4
    1.1 Brief Description 4
2. Flow of Events 4
    2.1 Basic Flow 4
    2.2 Alternative Flows 4
        2.2.1 User Submits Empty Search 4
        2.2.2 The Input Keyword Too Long 4
        2.2.3 No matching result 4
        2.2.4 Autocomplete or Suggestions. 4
        2.2.4.1 Autocomplete or Suggestions SubFlows 4
3. Special Requirements 5
    3.1 System Performance 5
    3.2 Scalability 5
4. Preconditions 5
    4.1 The article database is available. 5
5. Postconditions 5
    5.1 A list of relevant articles is displayed to the user. 5
    5.2 If no match is found, a "No results found" message is shown. 5
    5.3 The user can click on a result to view the full article. 5
6. Extension Points 5
    6.1 Autocomplete or Suggestions 5
7. Low Fidelity Prototype For Use Case 5
    7.1 Prompt Used for AI Tool 5
    7.2 Generated Interface 5

Use-Case Specification: Search Articles

1. Use-Case Name: Search Articles

1.1 Brief Description

This use case describes the functionality that allows users to search the articles by keywords, authors, tags or categories. It also allows users to combine multiple attributes for searching (e.g.Searching by keyword, tag and author, or searching by all attributes)

2. Flow of Events

2.1 Basic Flow

1. The user enters a keyword, author, tag or category into the search bar.
2. The user clicks the Search button or hits “Enter”.
3. The system validates the input.
4. The system queries the article database
5. The system displays a list of matching articles with default sorted by relevance or date.

2.2 Alternative Flows

2.2.1 User Submits Empty Search

- At step 2, if the user submits an empty search:
- The system displays an error message: "Please enter a search keyword." The current page remains unchanged.

2.2.2 The Input Keyword Too Long

- At step 1, if the user input a keyword exceeding the allowed character limit (more than 200 characters)
- The system displays an error message: “The keyword is too long”.
- It automatically ignores any characters beyond the 200th. Or if the search keyword is entered via **copy-paste**, the system only accepts the first 200 characters and truncates the rest..

2.2.3 No matching result

- At step 5, if no articles match:
- The system displays a message: “No articles found. Please try another keywords”

2.2.4 Autocomplete or Suggestions.

- At step 1, when the user is typing in the search bar:

2.2.4.1 Autocomplete or Suggestions SubFlows

1. The system suggests keywords or article titles.
2. The user can either:
    a. Select a suggestion from the dropdown, or
    b. Press “Tab” to choose the current suggestion (shown in gray in the search bar).
3. The selected suggestion replaces the current input.
4. The remaining steps in the Basic Flow continue as usual. (from Step2)

©Error! No property name supplied., 2025

3. Special Requirements

3.1 System Performance

- The system should return search results within 2 seconds for queries with up to 3 combined filters (e.g., keyword + tag + author).
- Autocomplete suggestions should appear in less than 500 milliseconds after the user types each character.

3.2 Scalability

- The system should handle at least 1,000 concurrent search requests

4. Preconditions

4.1 The article database is available.

5. Postconditions

5.1 A list of relevant articles is displayed to the user.

5.2 If no match is found, a "No results found" message is shown.

5.3 The user can click on a result to view the full article.

-

6. Extension Points

6.1 Autocomplete or Suggestions

- **Location in flow:** Step 1 (when the user is typing in the search bar)
- **Description**
  + As the user types into the search bar, the system provides real-time suggestions based on popular keywords, article titles, or search history..
  + Users can select a suggestion or press **Tab** to autocomplete.
  + The selected suggestion replaces the input and the flow resumes at **Step 2.**

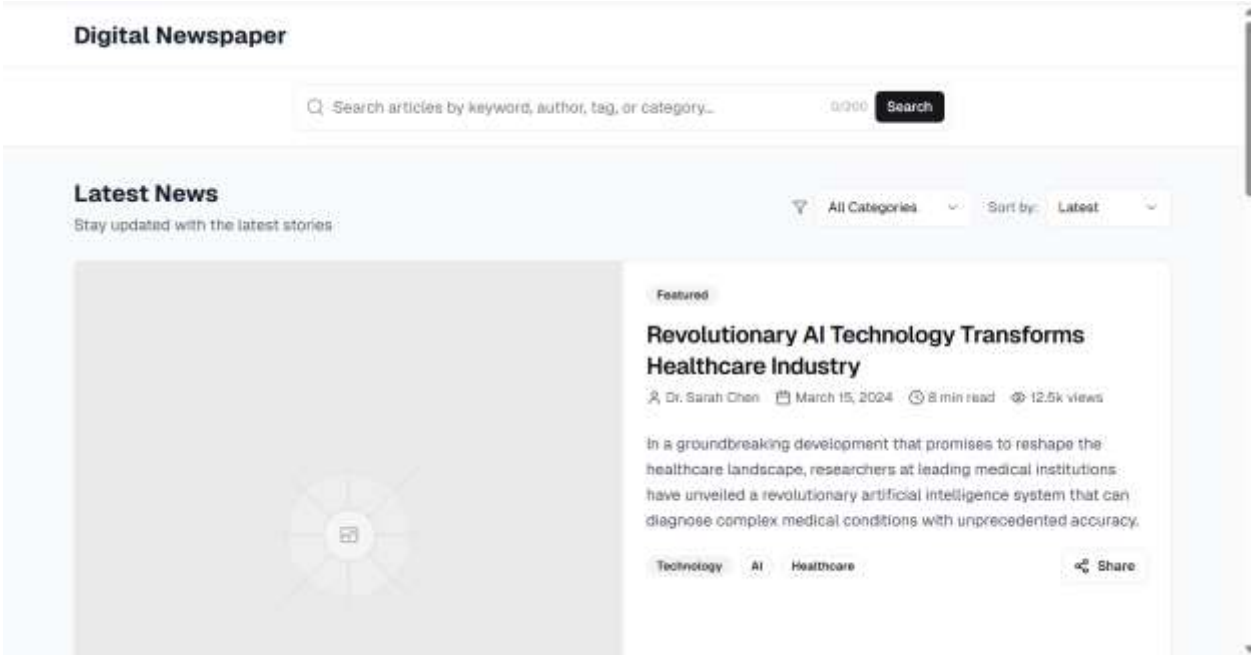
7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

- Create a responsive web UI for a digital newspaper article page with a “Search Article” feature.
- Requirements: Meets all that have been specified in the use-case specification.
  [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*

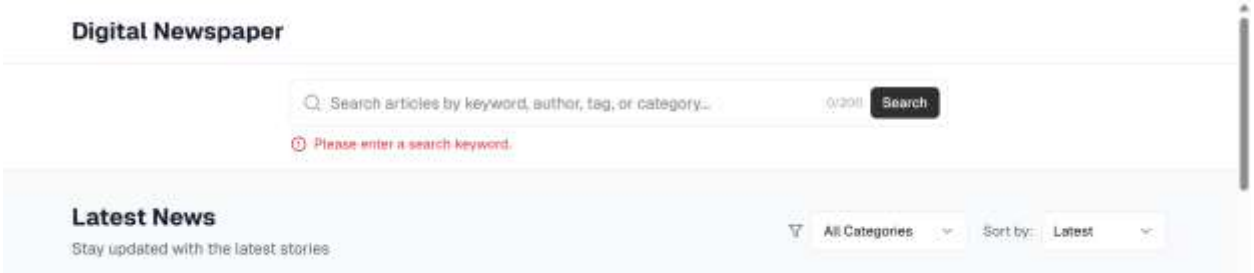
- The UI should be mobile-friendly and support modern browsers. Prioritize usability and simplicity.

7.2 Generated Interface



The main screen of the website, the Search bar is shown as above.

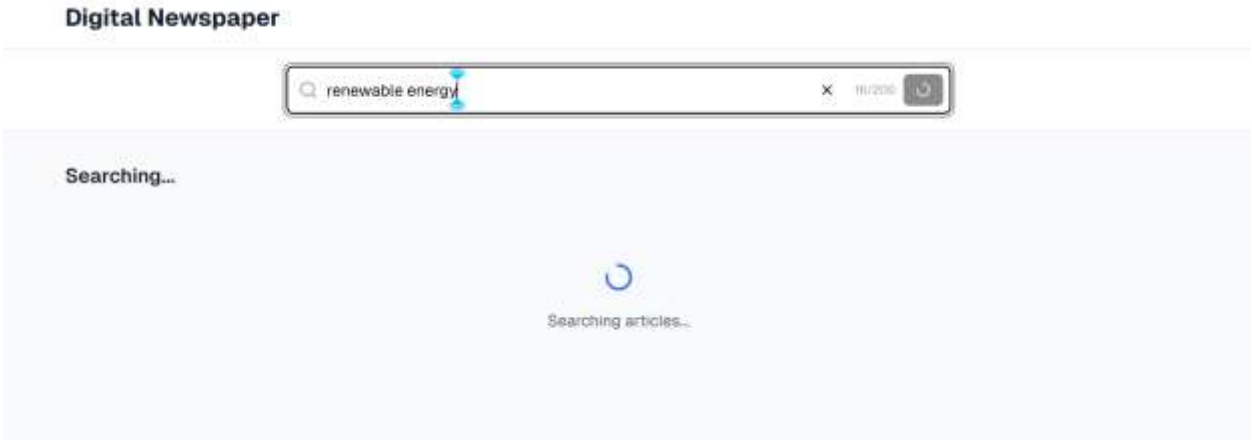
Typing space characters in the Search bar.



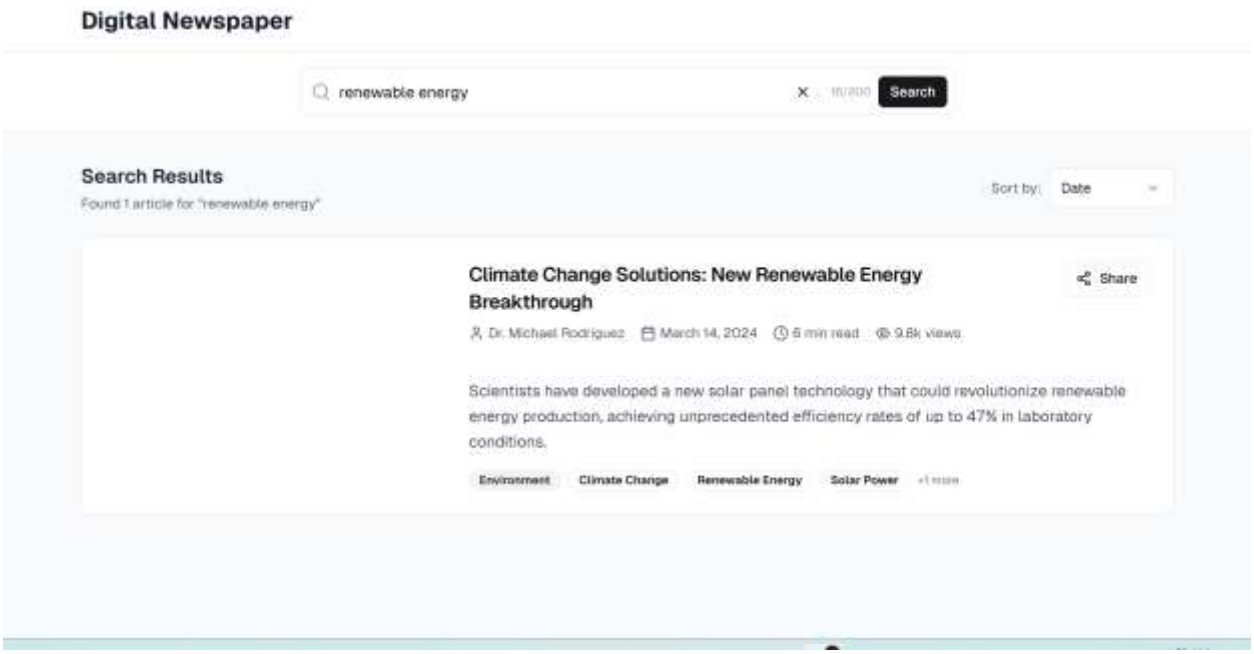
After clicking the Search button when the user has just input space characters. The system message error "Please enter a search keyword."

The user retypes the search keyword. The keyword suggestion (if exists) is displayed below.

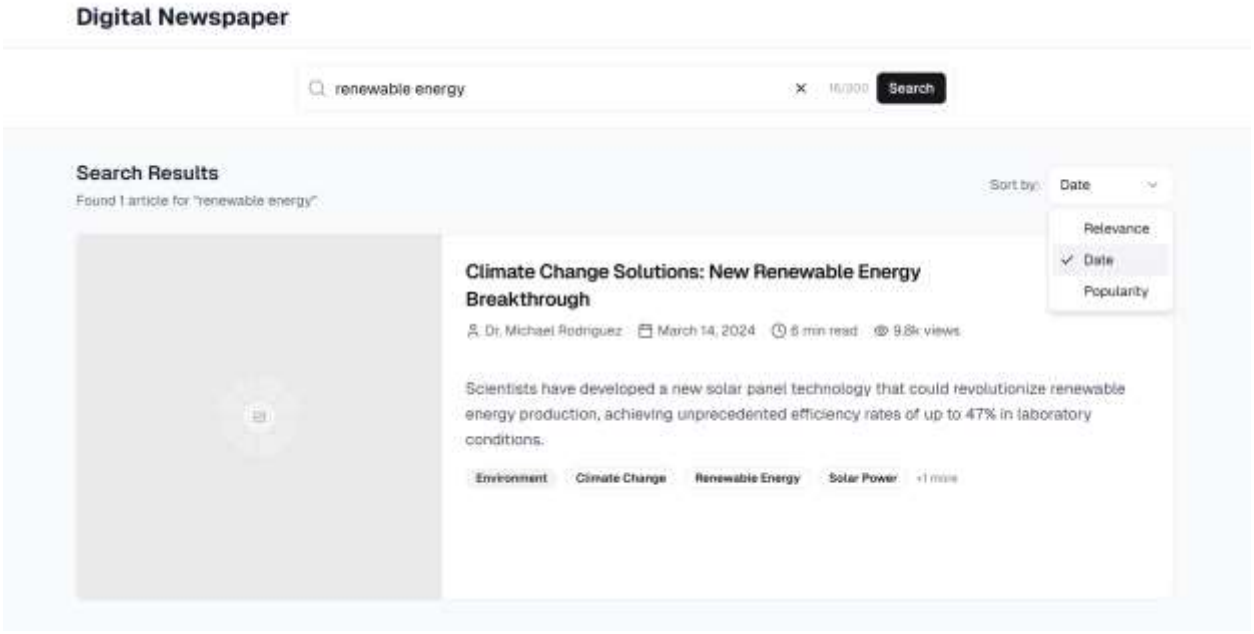
The user Tab to get the first keyword suggestion.



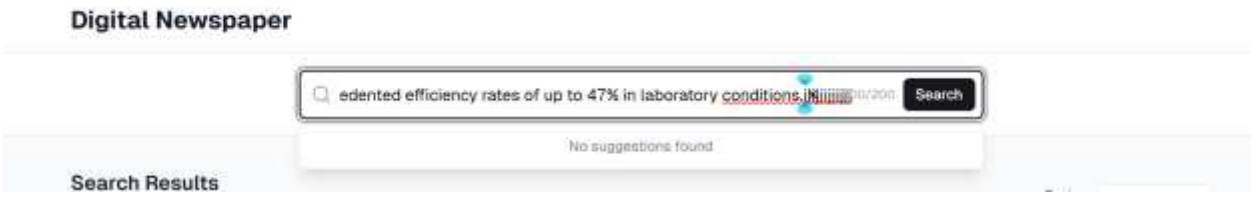
After clicking the Search button, the Search process is processing (the Running Circle)



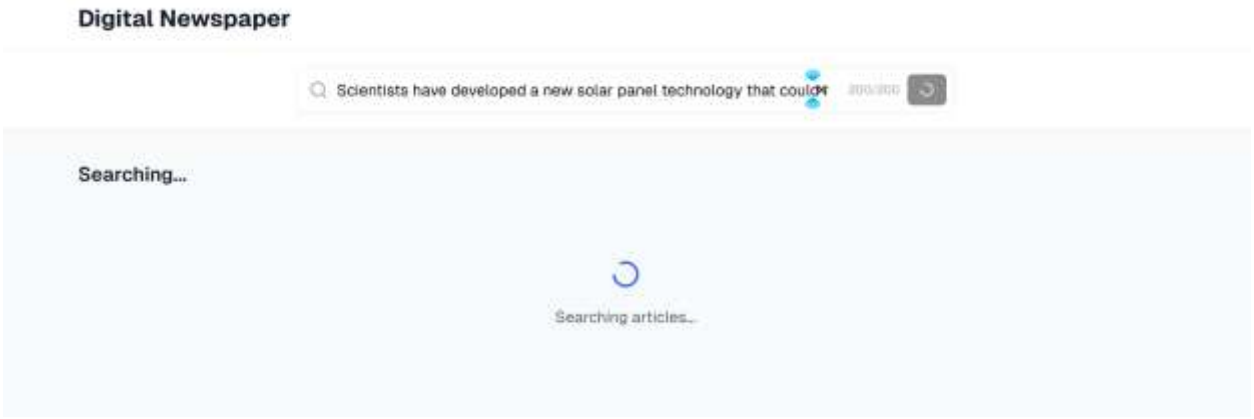
The results is shown after searching.



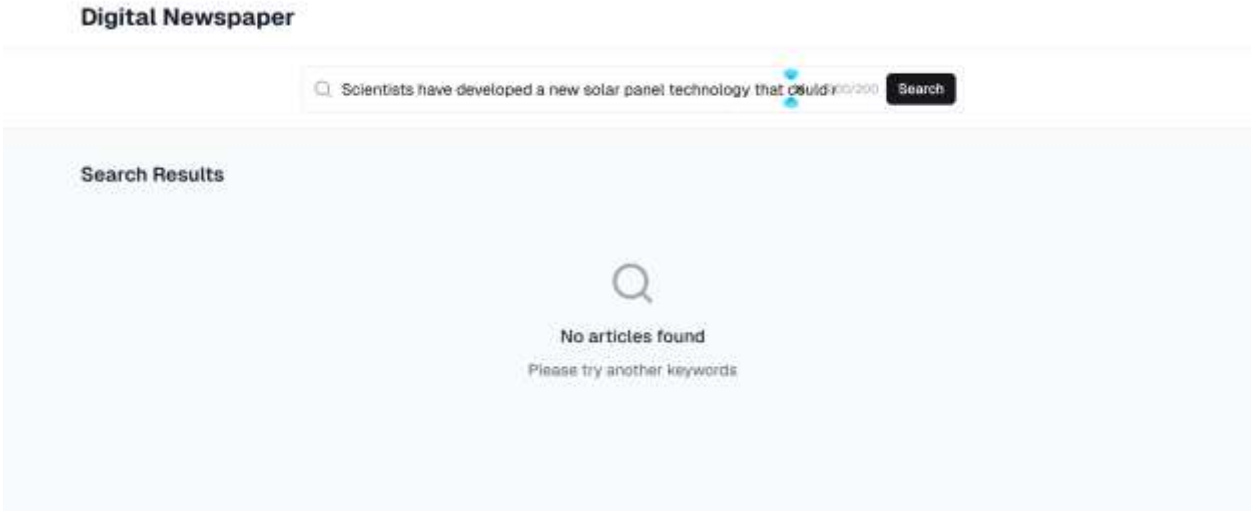
The user can select the index in the Sort by box to Filter the articles by Date, Relevance or Popularity.



In case that the Input Keywords access 200 characters, no character is added after that, even if the user types directly or uses the "Copy-Paste" method.



The system is processing the Search process.



In case no article is found with the Searching Keywords, the screen displayed "No article found" as above.

404 Not Found

Digital Newspaper Website
Use-Case Specification: Filter Articles

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Specific about Filter Articles use case	Thạch Ngọc Hân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Filter Articles **4**

1.1 Brief Description 4

2. Flow of Events **4**

2.1 Basic Flow 4

2.2 Alternative Flows 4

2.2.1 Filter with No Results 4

2.2.2 Invalid Filter Input 4

3. Special Requirements **4**

3.1 Performance 4

4. Preconditions **5**

4.1 The article listing page with filtering options must be fully loaded and interactive. 5

5. Postconditions **5**

5.1 The screen shows only those articles that meet the selected criteria, or a “no results” message if none match. 5

6. Extension Points **5**

6.1 Save Filter Preferences 5

©Error! No property name supplied., 2025

Use-Case Specification: Filter Articles

1. Use-Case Name: Filter Articles

1.1 Brief Description

This use case allows a user to filter the list of available articles based on selected criteria such as category, tag, author, publication date, or keyword. It enhances user experience by narrowing down content to match interests or search queries.

2. Flow of Events

2.1 Basic Flow

1. The use case begins when the User navigates to the article listing or search page.
2. The User selects one or more filter criteria.
3. The User clicks the “Apply Filters” button.
4. The System validates the selected filter criteria.
5. The System queries the article database based on the filter parameters.
6. The System returns a list of articles that match the filters.
7. The System displays the filtered list to the User
8. The use case ends.

2.2 Alternative Flows

2.2.1 Filter with No Results

At step 6 in the Basic Flow, if no articles match the criteria:

a. The system displays a message: “No articles found matching your filters.”
b. The user is given the option to reset filters or modify them.
c. The flow returns to step 2 of the Basic Flow.

2.2.2 Invalid Filter Input

At step 4 in the Basic Flow, if filter input is invalid (e.g., incorrect date format):

a. The system highlights the invalid fields and displays an error message.
b. The user corrects the input.
c. The flow resumes from step 3 of the Basic Flow.

3. Special Requirements

3.1 Performance

1. The system should return filtered results within 2 seconds.
2. Filtering must work on both desktop and mobile views responsively.

©Error! No property name supplied., 2025

4. Preconditions

4.1 The article listing page with filtering options must be fully loaded and interactive.

5. Postconditions

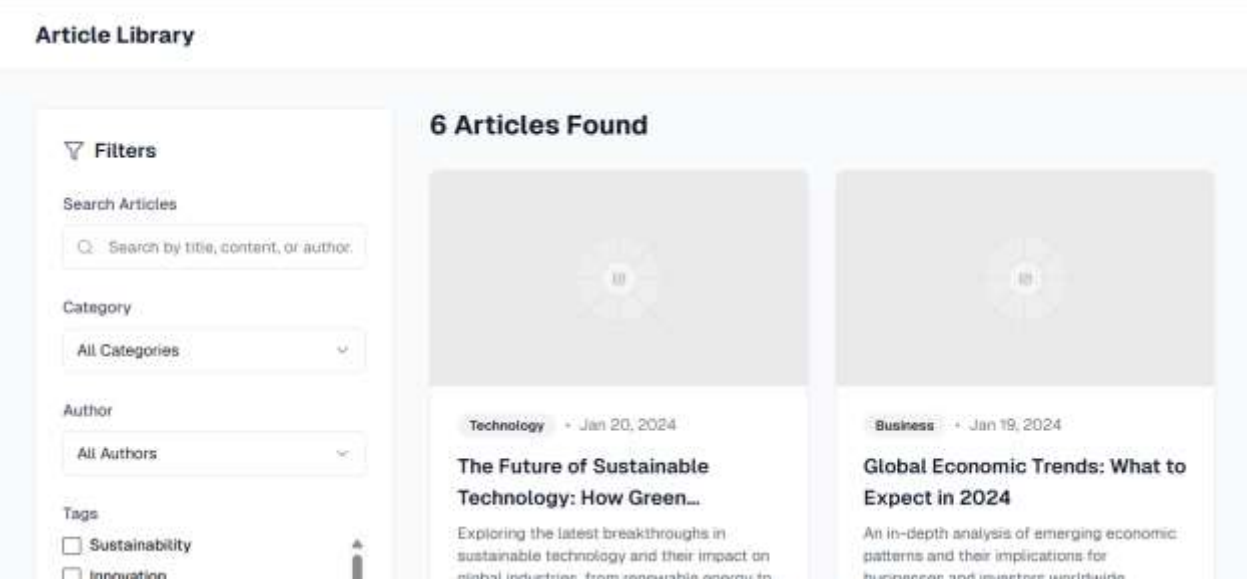
5.1 The screen shows only those articles that meet the selected criteria, or a “no results” message if none match.

6. Extension Points

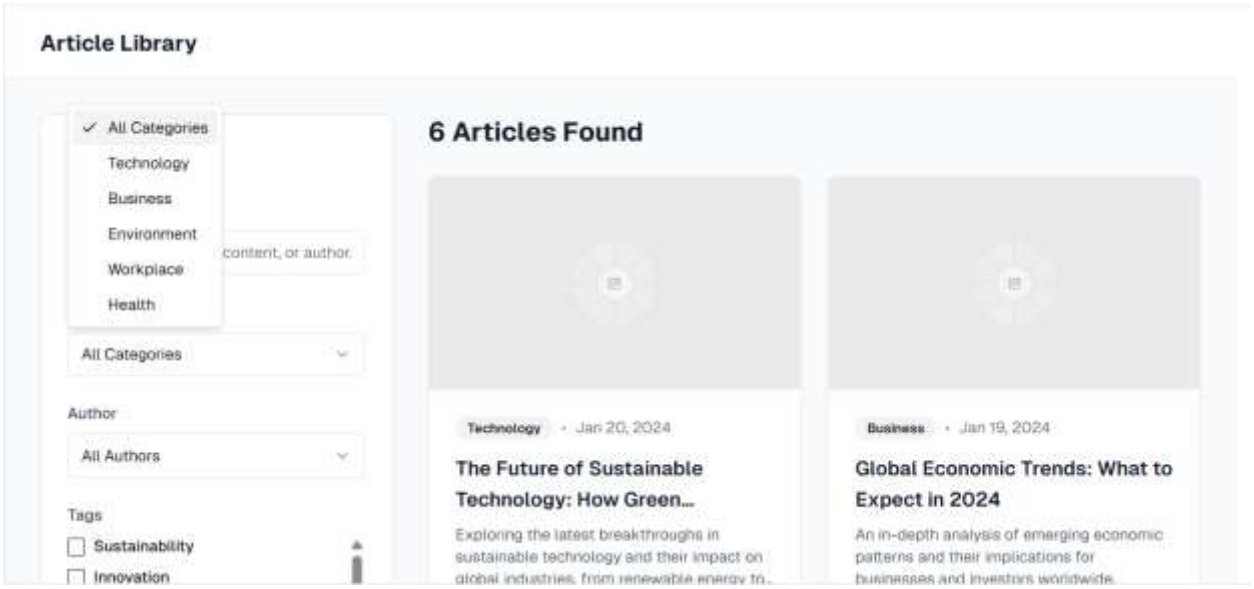
6.1 Save Filter Preferences

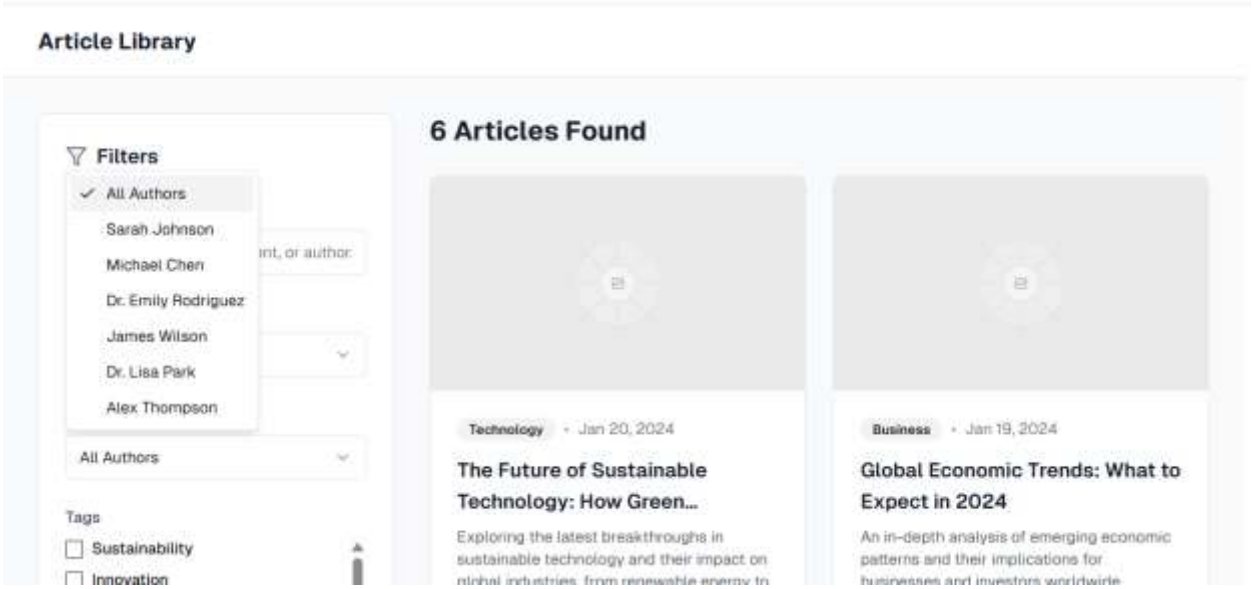
- Location: after step 7 of the Basic Flow
- Description: allows the user to save the current filter settings for reuse in future sessions.

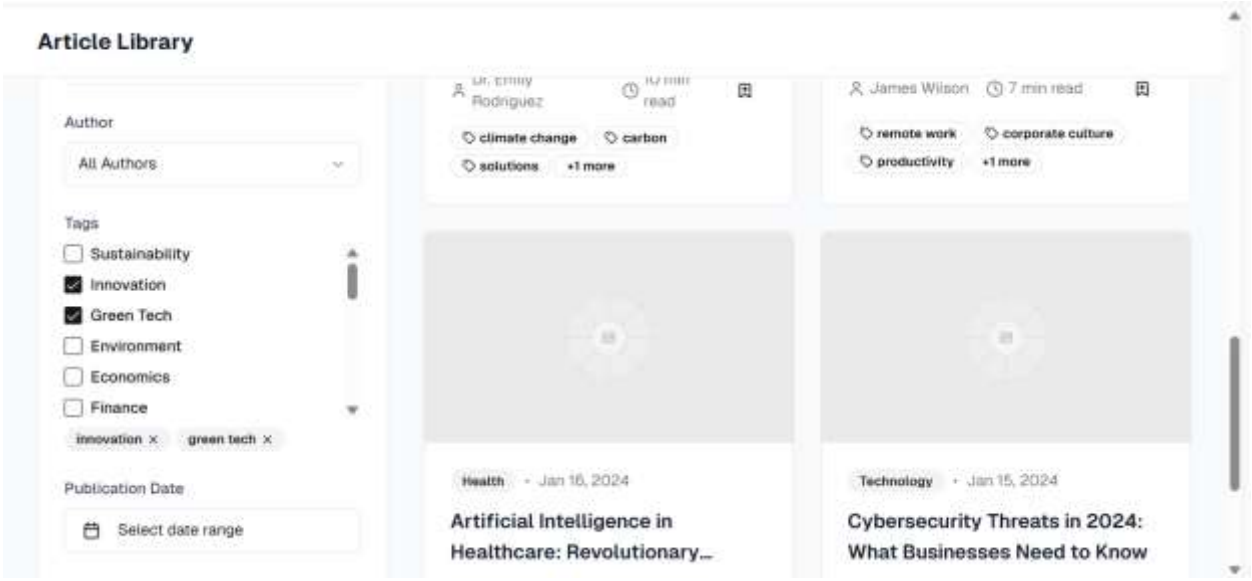
Original interface

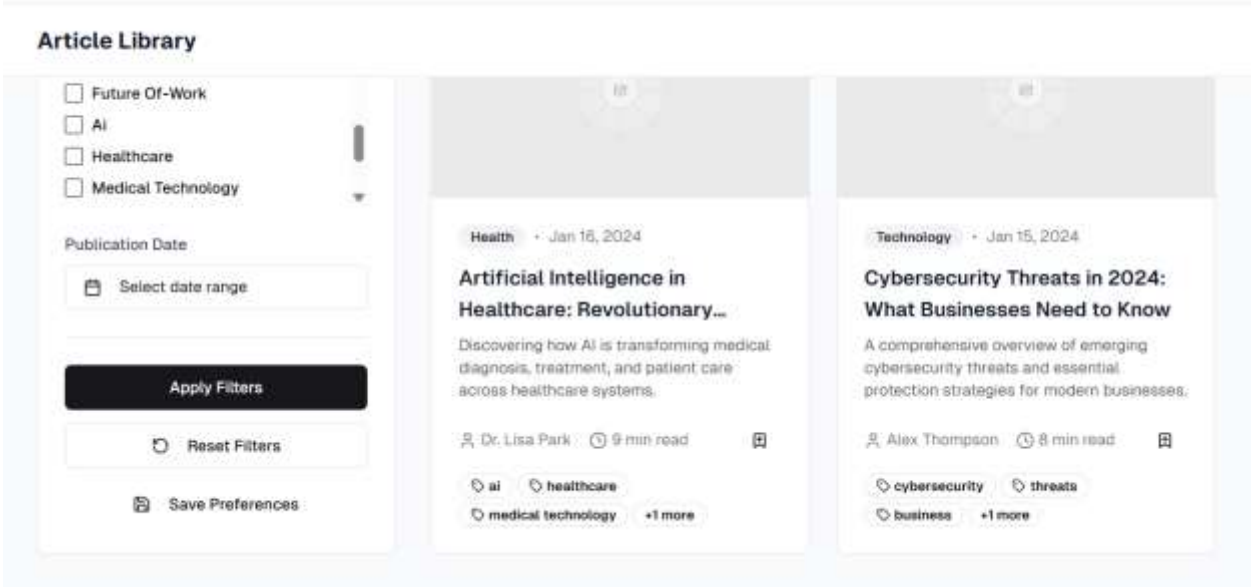


Filters available

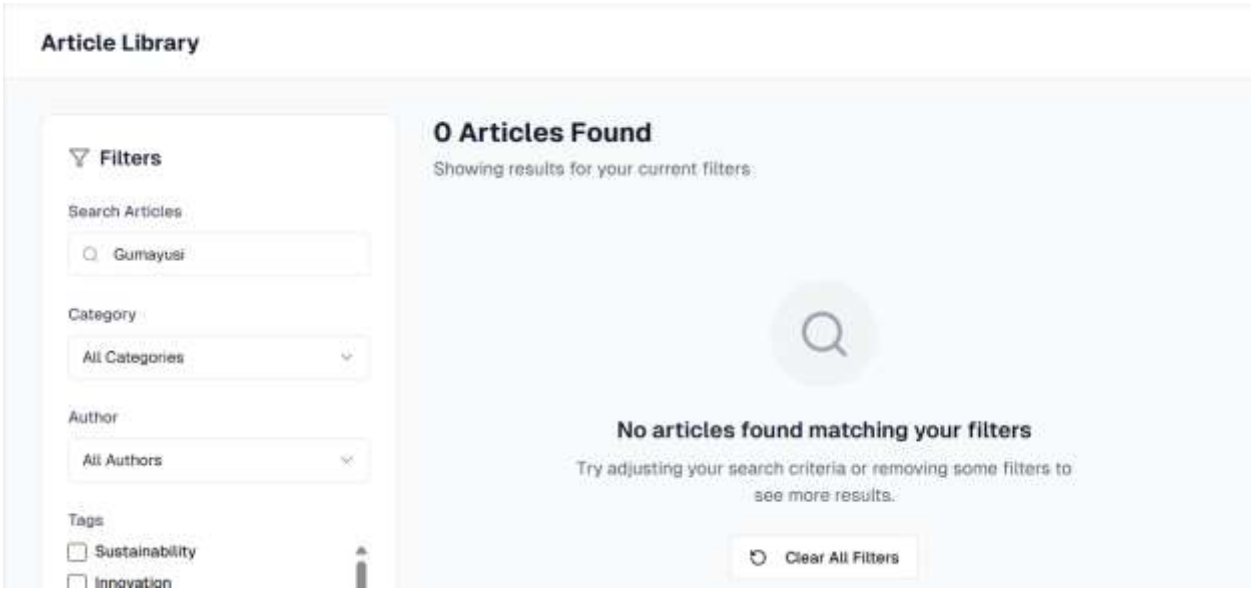

Article Library
All Categories
Technology
Business
Environment
Workplace
Health
content, or author.
All Categories
Author
All Authors
Tags
Sustainability
Innovation
6 Articles Found
Technology • Jan 20, 2024
The Future of Sustainable Technology: How Green...
Exploring the latest breakthroughs in sustainable technology and their impact on
Business • Jan 19, 2024
Global Economic Trends: What to Expect in 2024
An in-depth analysis of emerging economic patterns and their implications for


Article Library
Filters
All Authors
Sarah Johnson
Michael Chen
Dr. Emily Rodriguez
James Wilson
Dr. Lisa Park
Alex Thompson
All Authors
Tags
Sustainability
Innovation
6 Articles Found
Technology • Jan 20, 2024
The Future of Sustainable Technology: How Green...
Exploring the latest breakthroughs in sustainable technology and their impact on
Business • Jan 19, 2024
Global Economic Trends: What to Expect in 2024
An in-depth analysis of emerging economic patterns and their implications for

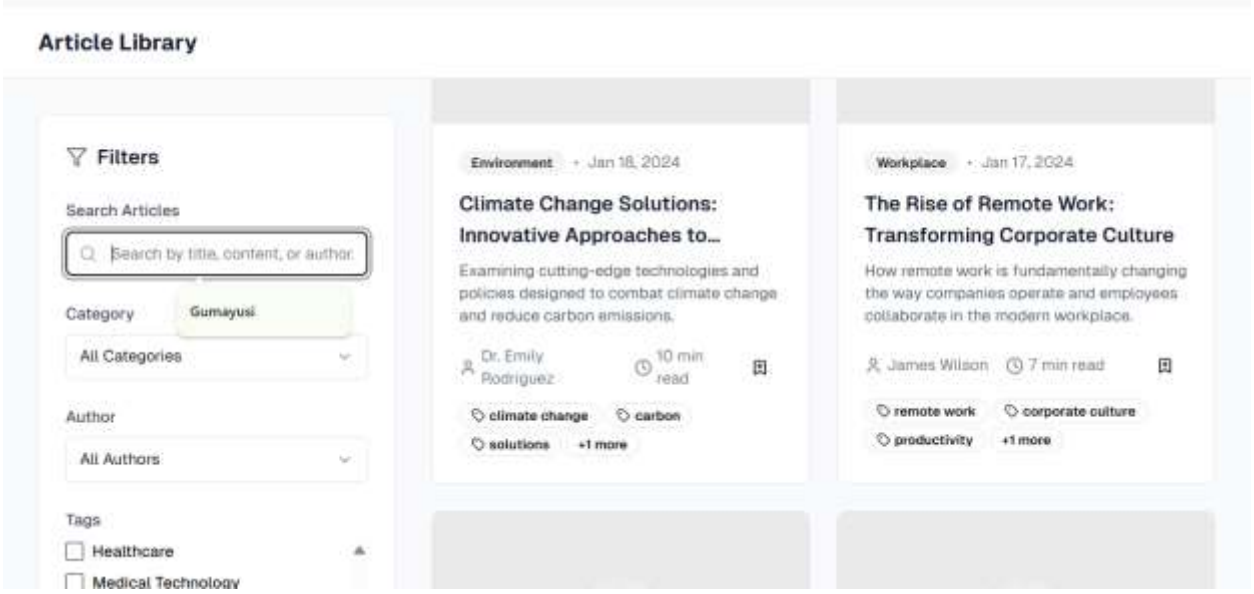




Interface after applying filter does not exist



After saving the filter it will suggest again.



404 Not Found

Digital Newspaper Website
Use-Case Specification: Change Password

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Specific about Change Password use case	Lê Thị Minh Thư

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Change Password 4
    1.1 Brief Description 4
2. Flow of Events 4
    2.1 Basic Flow 4
    2.2 Alternative Flows 4
        2.2.1 Invalid Current Password 4
        2.2.2 The New Password does not meet format/ security requirements. 4
        2.2.3 The Confirmation Password and New Password are not the same. 5
        2.2.4 User Cancels the Operation 5
        2.2.5 The New and Old Passwords are the Same. 5
        2.2.6 MFA Verification Failed 5
        2.2.6.1 MFA Verification Failed Subflows 5
        2.2.7 MFA Code Expired 5
        2.2.7.1 MFA Code Expired Subflows 5
3. Special Requirements 5
    3.1 Security Requirement 5
    3.2 System Performance 5
4. Preconditions 6
    4.1 UserAuthentication 6
    4.2 Setting Access 6
5. Postconditions 6
    5.1 Password Successfully Updated 6
    5.2 Confirmation Displayed 6
    5.3 Optional Device Logout 6
6. Extension Points 6
    6.1 Validate Password Format 6
    6.2 Perform Multi-Factor Authentication MFA 6
    6.3 Prompt Device Logout 6
7. Low Fidelity Prototype For Use Case 6
    7.1 Prompt Used for AI Tool 6
    7.2 Generated Interface 7

©Error! No property name supplied., 2025

Use-Case Specification: Change Password

1. Use-Case Name: Change Password

1.1 Brief Description

This use case describes the functionality that allows users to change their current password to a new one in the system.

2. Flow of Events

2.1 Basic Flow

1. The user navigates to the Change Password section through access to the **Setting & Privacy** via their **Profile Information.**
2. The system checks whether **Multi-Factor Authentication (MFA)** is required:
    a. If yes, the user is prompted to enter a verification code (via SMS, email, or authenticator).
    b. The user enters the code.
    c. The system verifies the code before proceeding.
    d. If the code is valid and not expired, the flow proceeds.
3. The system displays a form requiring:
    a. Current password
    b. New password
    c. Confirm new password
4. The user fills in all fields and submits the form.
5. The system verifies:
    a. The current password is correct.
    b. The new password meets format/ security requirements:
        i. At least 8 characters.
        ii. Contains at least one number.
        iii. Includes at least one special notation.
        iv. Includes both uppercase and lowercase letters.
    c. The new password matches the confirmation.
    d. The new password is NOT the same as the old password.
6. If all checks pass, the system updates the password.
7. The system displays a success message: “Your password has been changed.”
8. After a successful password change, the system prompts the user: "Do you want to log out from other devices?"
    a. If the user selects "Yes", the system logs the user out from all other active sessions except the current one.
    b. If the user selects "No", the current session continues as usual.

2.2 Alternative Flows

2.2.1 Invalid Current Password

- At step 4.a, if the current password entered is incorrect:
- The system displays an error: “Incorrect password”.
- The system will go to the step 3 of the **Basic flows** to allow retry.

2.2.2 The New Password does not meet format/ security requirements.

- At step 4.b, if the new password fails format/security rules:

- Depending on the inputting mistakes, the system displays an error:
  + The password must contain at least one special notation (#, @, $, ..)
  + The password must contain at least one number (0,1, ..9)
  + The password must contain at least one uppercase letter (A, B, C,..)
  + The password must contain at least one lowercase letter (a, b, c,..)
  + The password must have at least 8 characters.
- The flow returns to **Step 3**.

2.2.3 The Confirmation Password and New Password are not the same.

- At step 4.c, if the new password and confirmation do not match:
- The system displays an error: “Password confirmation does not match.”
- The flow returns to **Step 3**.

2.2.4 User Cancels the Operation

- At any point before submission, if the user cancels the operation:
- The system closes the form and returns to the **Settings & Privacy** page, and no changes are made.

2.2.5 The New and Old Passwords are the Same.

- If the new password is identical to the current one:
- The system displays an error: “New password must be different from the current password.”
- The flow returns to **Step 3**.

2.2.6 MFA Verification Failed

- At Step 2.b, if the entered OTP is invalid:

2.2.6.1 MFA Verification Failed Subflows

1. Show error: “Invalid verification code.”
2. Allow the user to retry (max 3 attempts).
3. After 3 failed attempts, temporarily lock the password change process for 5 minutes.
4. The system returns to step 3 of the basic flows.

2.2.7 MFA Code Expired

- At Step 2.b, is the OTP has expired:

2.2.7.1 MFA Code Expired Subflows

- Show error: “Verification code has expired.”
- Provide a button **Resend Code**
- Return to the OTP entry step after resending.

3. Special Requirements

3.1 Security Requirement

- Passwords must be stored using secure hashing algorithms (e.g: **SHA-256 with salt**) to ensure data protection and prevent password breaches.
- The system must prevent password reuse and brute-force attacks.

3.2 System Performance

- Password update operation should complete within 2 seconds under normal load.
- Error messages must appear within 500 milliseconds of validation failure.
- The system should handle simultaneous password change requests for up to 500 users without performance

degradation.

4. Preconditions

4.1 UserAuthentication

- The user must be logged in to their account.

4.2 Setting Access

- **The Settings & Privacy** section is accessible from the user's profile.

5. Postconditions

5.1 Password Successfully Updated

- The user's password is securely updated and stored.

5.2 Confirmation Displayed

- A success message is shown: **"Your password has been changed."**

5.3 Optional Device Logout

- If the user chooses to log out from other devices, all other sessions are terminated except the current one.

6. Extension Points

6.1 Validate Password Format

- **Location in flow:** Step 5 of the Basic Flow
- **Description:** The system checks whether the new password meets the format requirements and security criteria.

6.2 Perform Multi-Factor Authentication MFA

- **Location in flow:** After Step 1 (user accesses Change Password section)
- **Description:** The system may require additional verification such as OTP before proceeding to display the form, depending on user settings or risk evaluation.

6.3 Prompt Device Logout

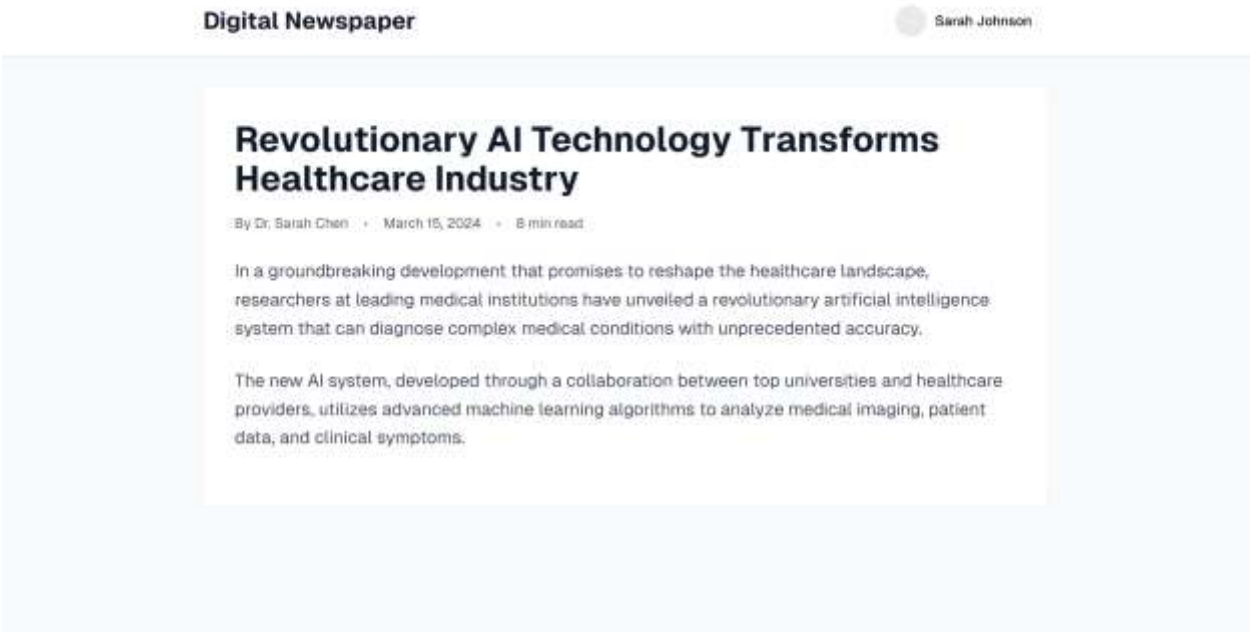
- **Location in flow**: After step 6 of Basic Flow
- **Description**: The system asks whether the user wants to log out of other devices. If yes, the system ends all other active sessions.

7. Low Fidelity Prototype For Use Case

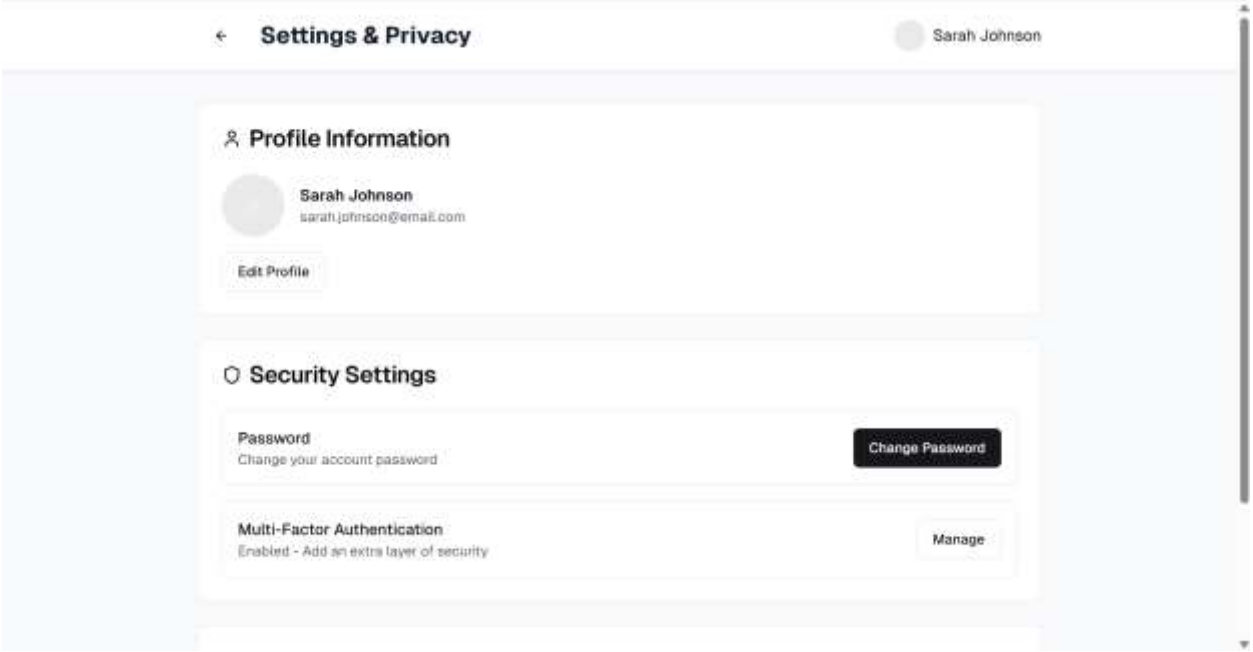
7.1 Prompt Used for AI Tool

- Create a responsive web UI for a digital newspaper article page with a "Change Password." feature.
- Requirements: Meets all that have been specified in the use-case specification.
  [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*
- The UI should be mobile-friendly and support modern browsers. Prioritize usability and simplicity.
- The system must show OTP code verification, show expired time, message error and allow it to try again.

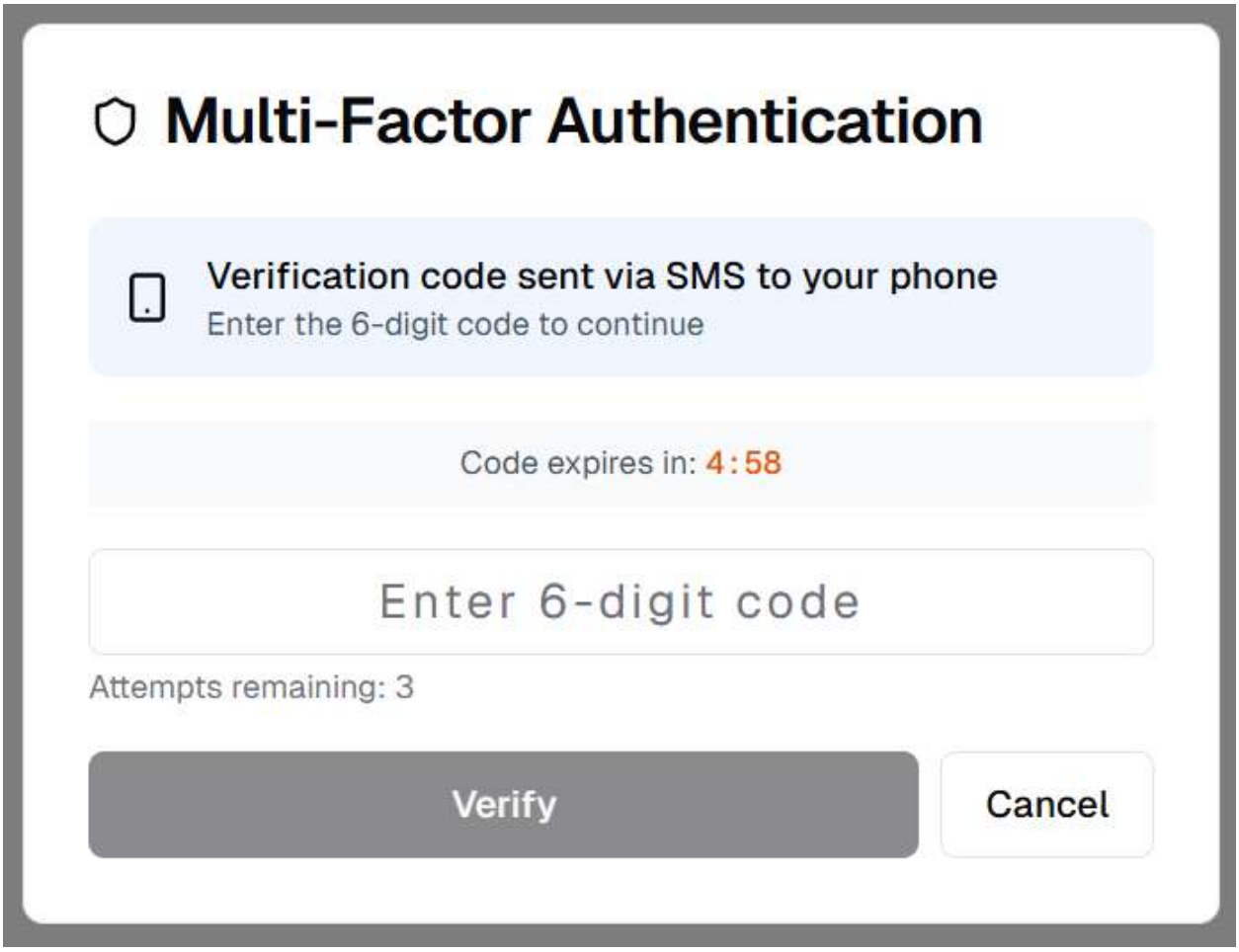
7.2 Generated Interface



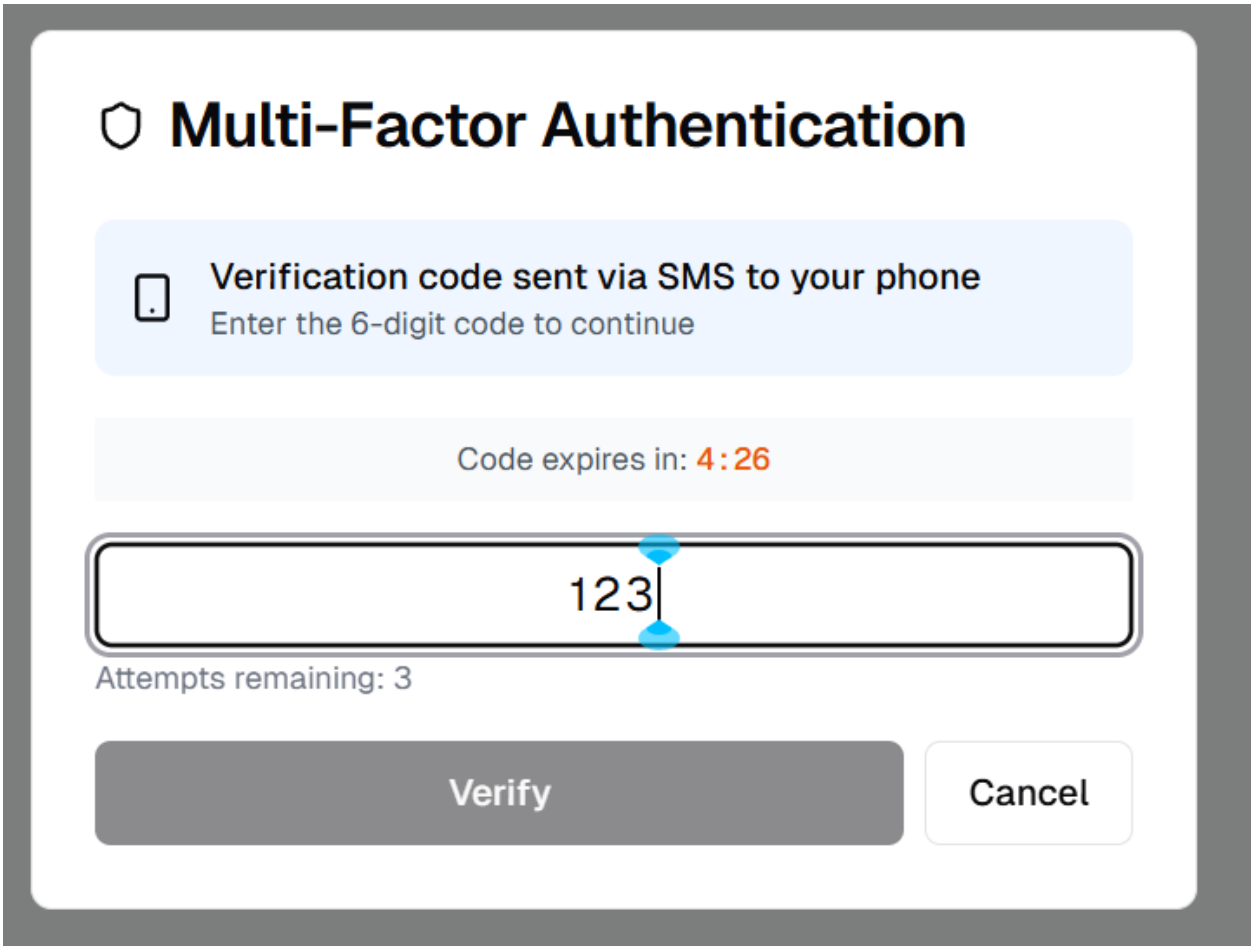
The main screen of the main website



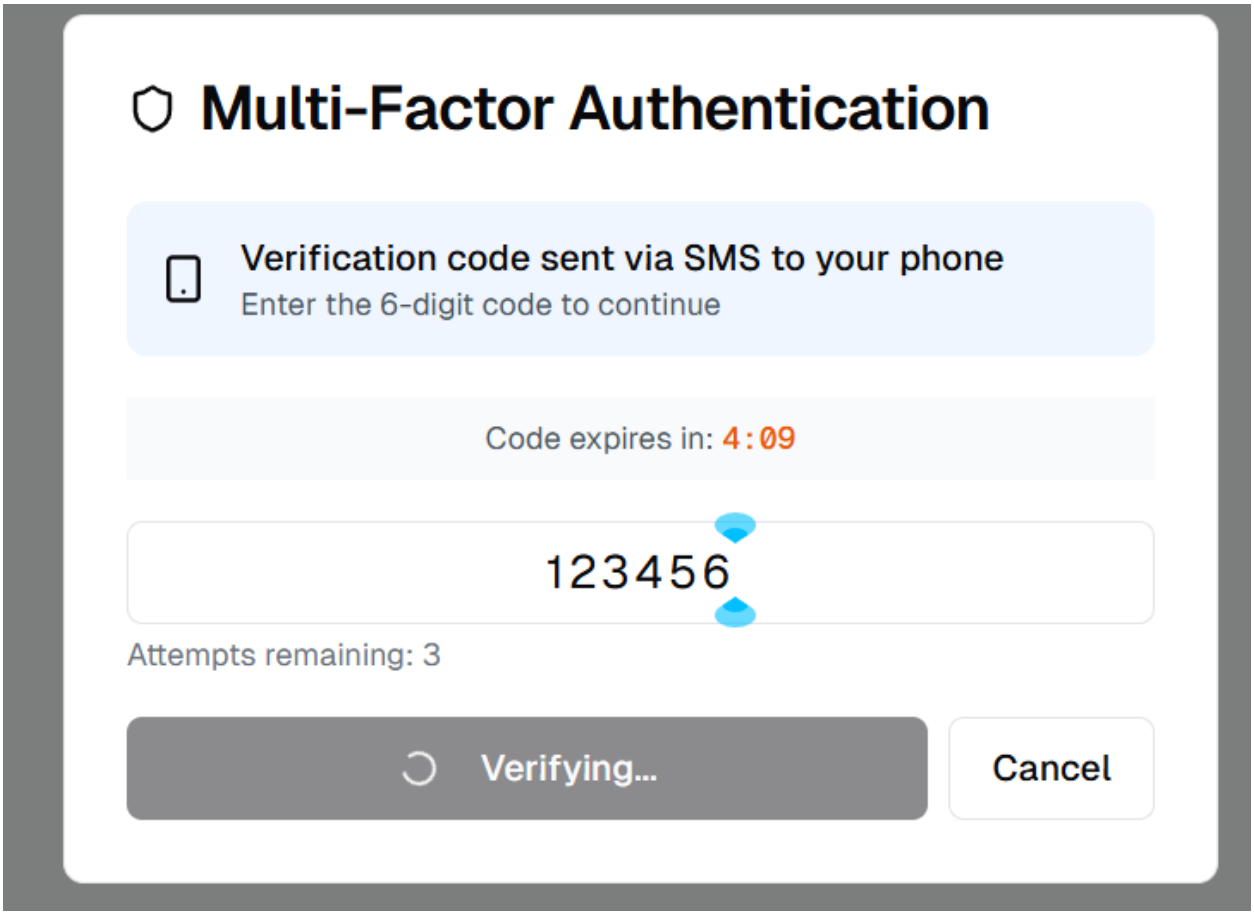
Here are the Profile Information after the user click to their profile symbol. At the Security Setting has a Change Password button.



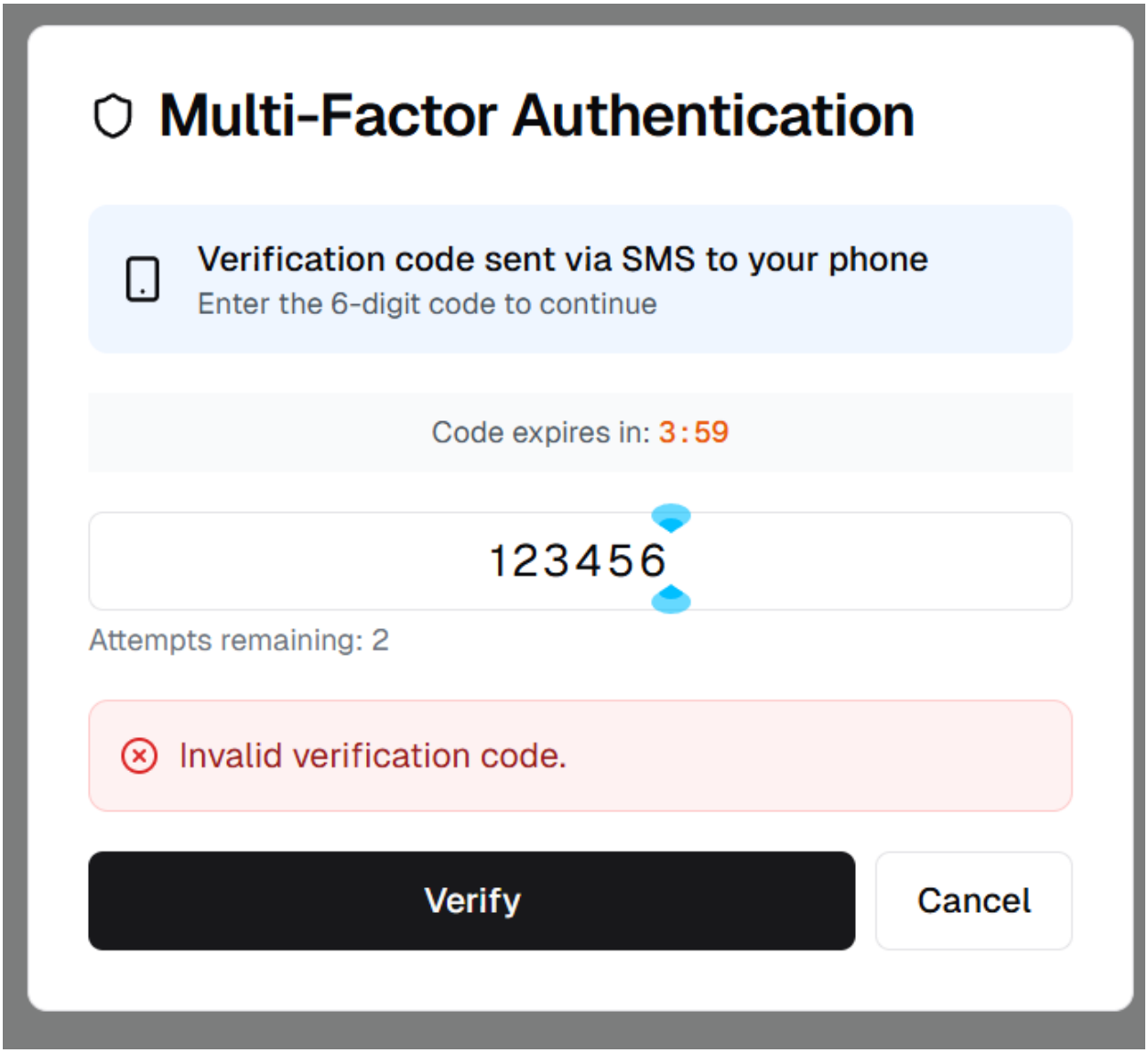
Here is the main screen after clicking the Change Password button. The OTP code hasn't input yet, so the Verify button is invalid



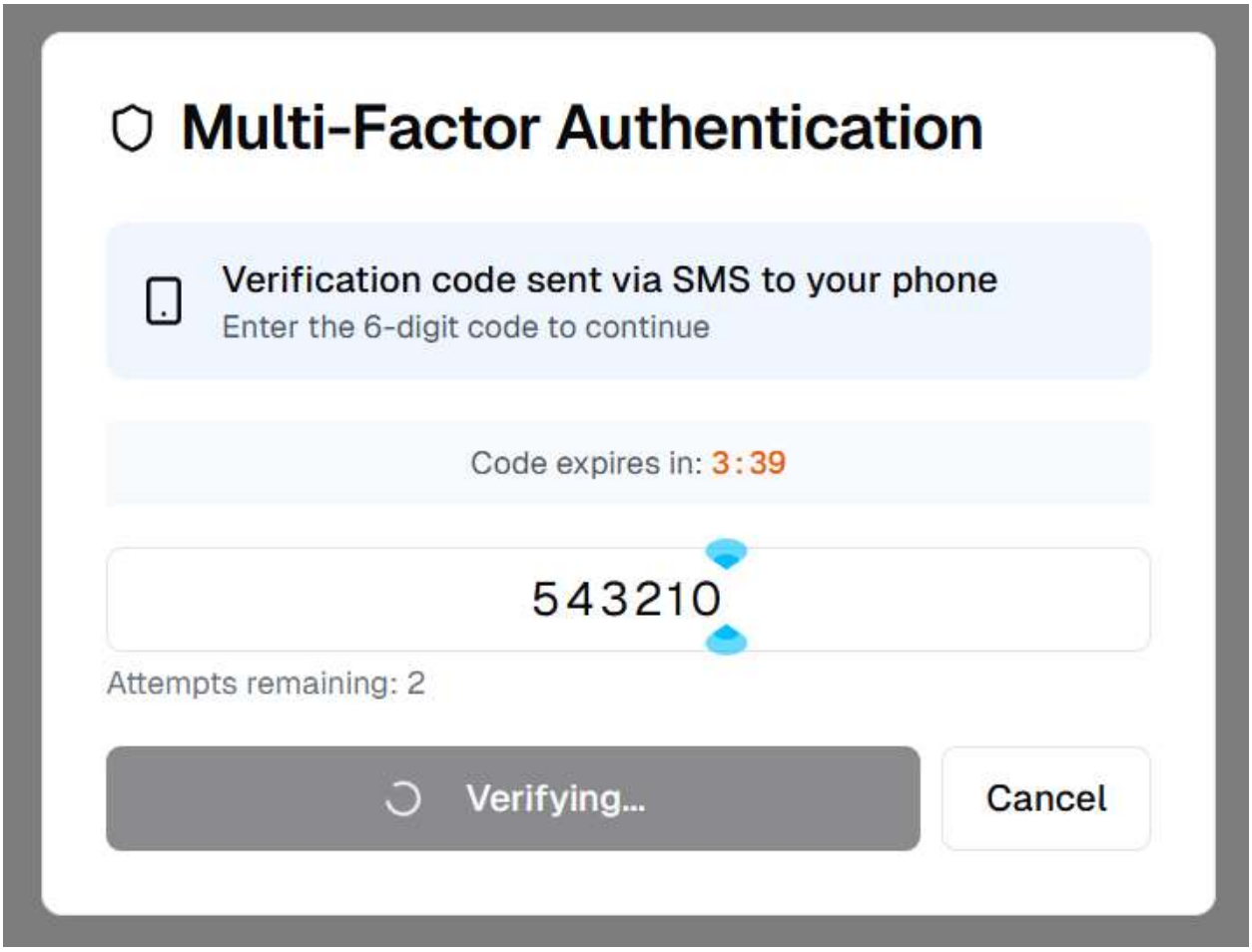
The OTP code is input, but the length is not equal to 6, so the Verify button is invalid



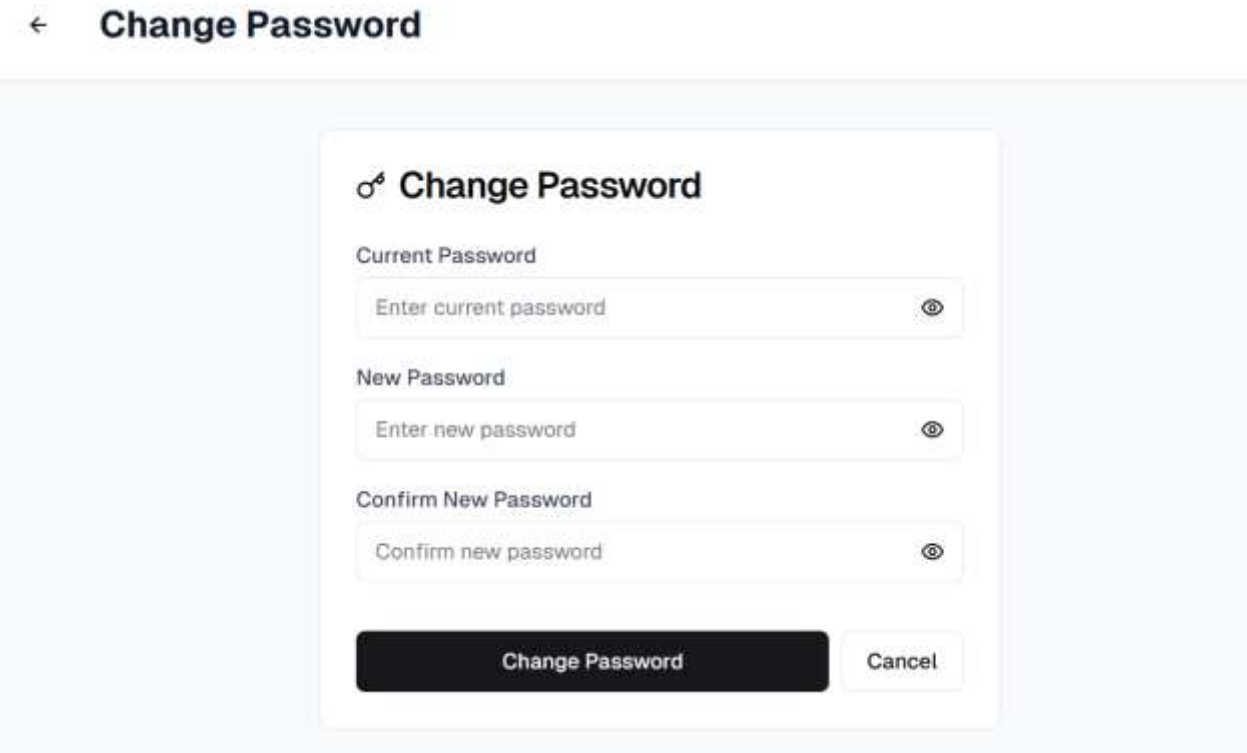
The OTP code is input, the Verify button is Valid. Here is the processing screen after clicking the Verify button.



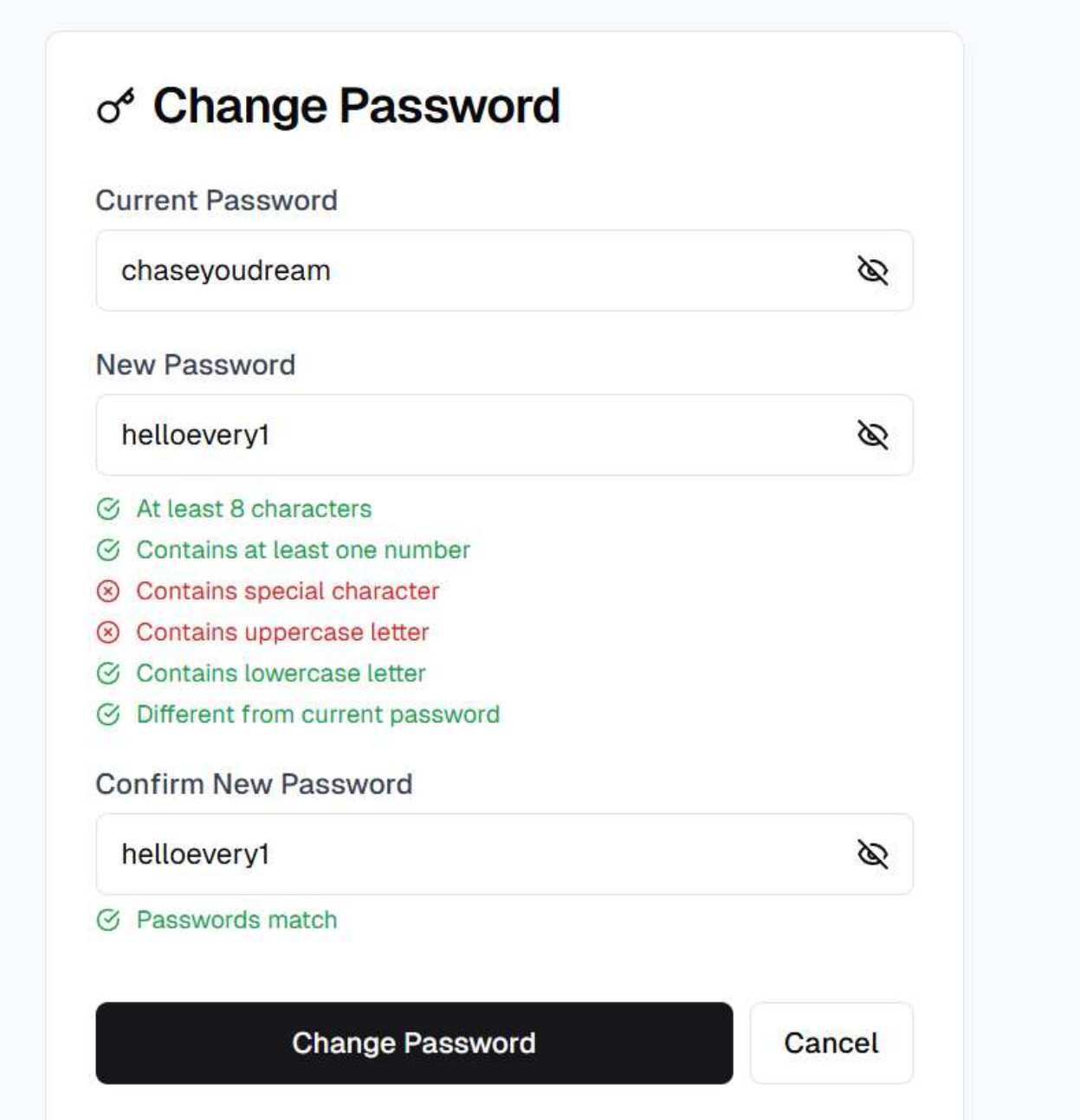
In case the OTP code is incorrect, or time out, the system will message "Invalid verification code". The cancel will go back to the main screen of the Profile Information.



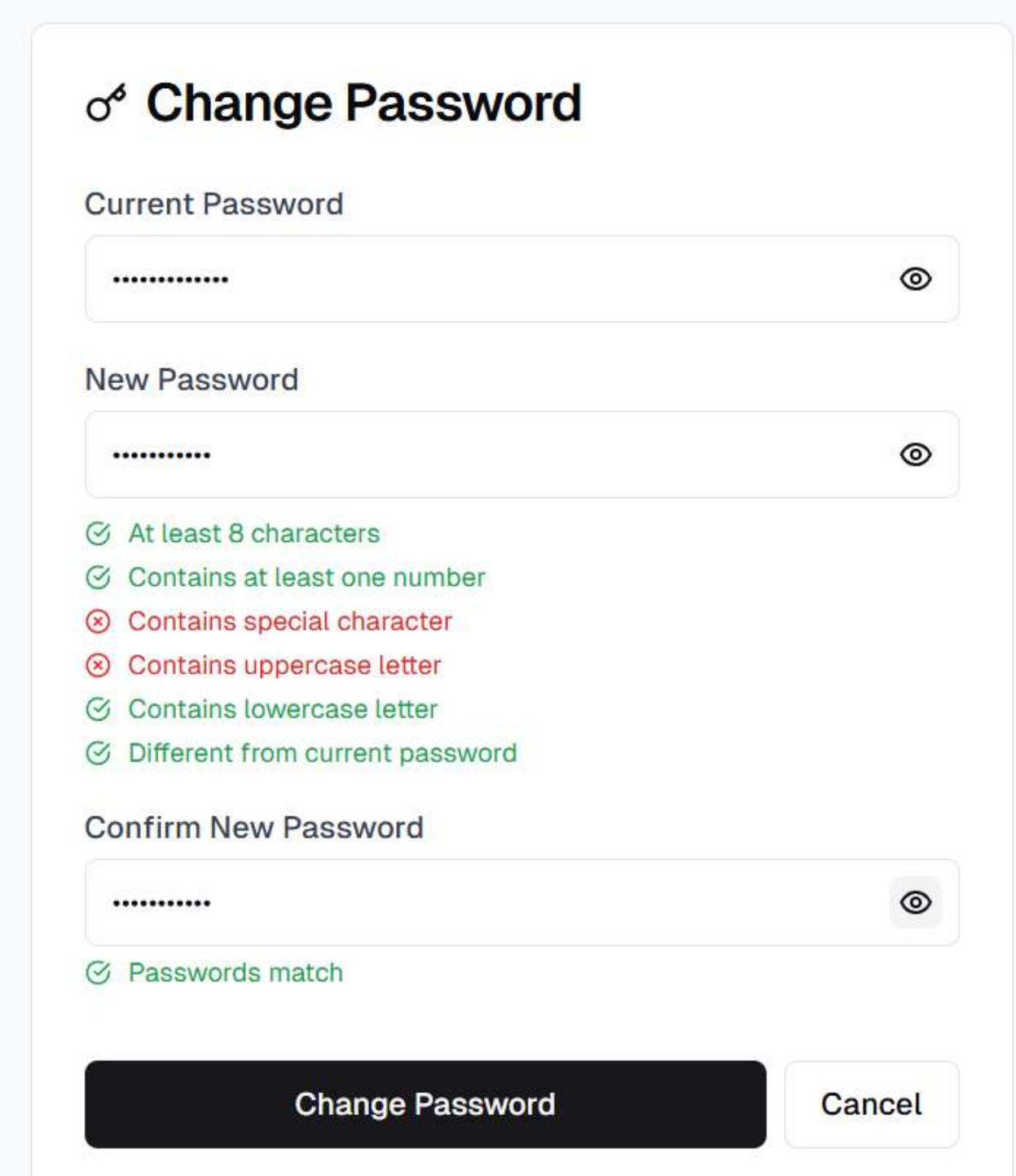
The previous OTP code is incorrect, so that the Attempts remaining decrease from 3 to 2. After retyping the OTP code and clicking the Verify button. The verification is processing.



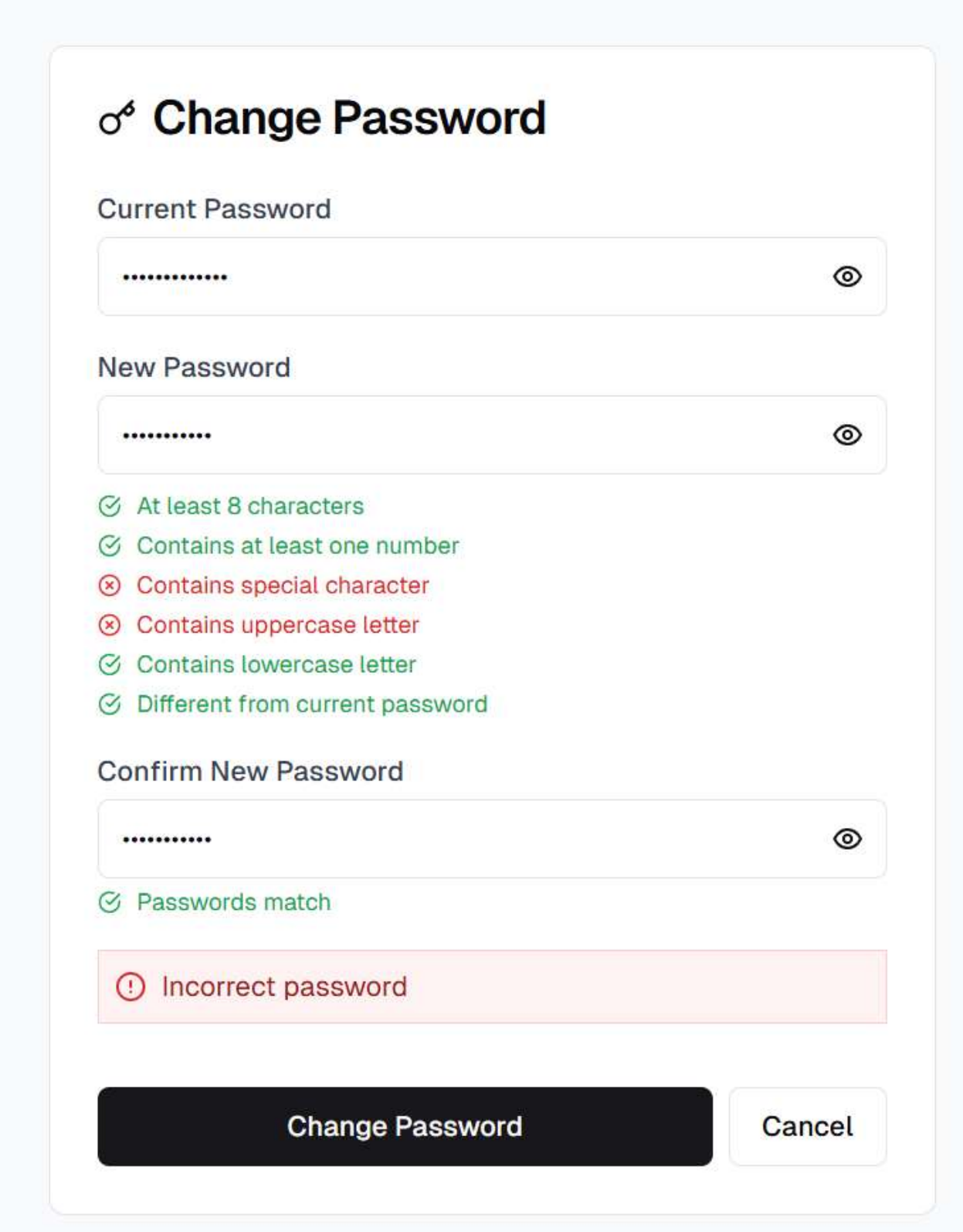
After the OTP code is successfully verified, the Change Password screen is displayed.



The new password does not meets the format and security criteria for the password. Depends on which mistakes, the system shows to the user to adjust the right format password. In this case, if the user chooses the Cancel button, it will go to the Profile Information screen.

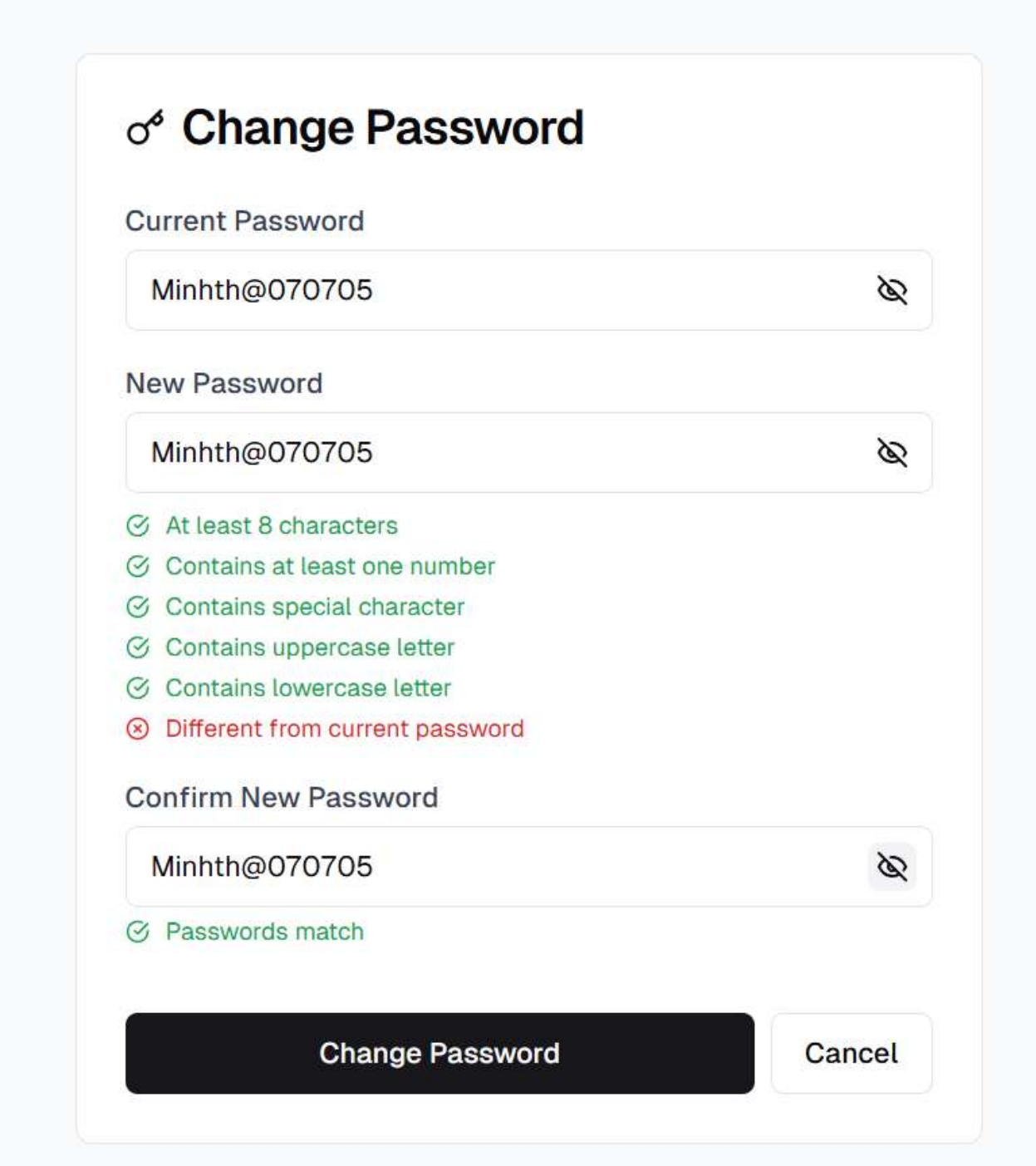


The same as the picture above. One difference is that the user can click to the Eye Symbol to show or hide the password.

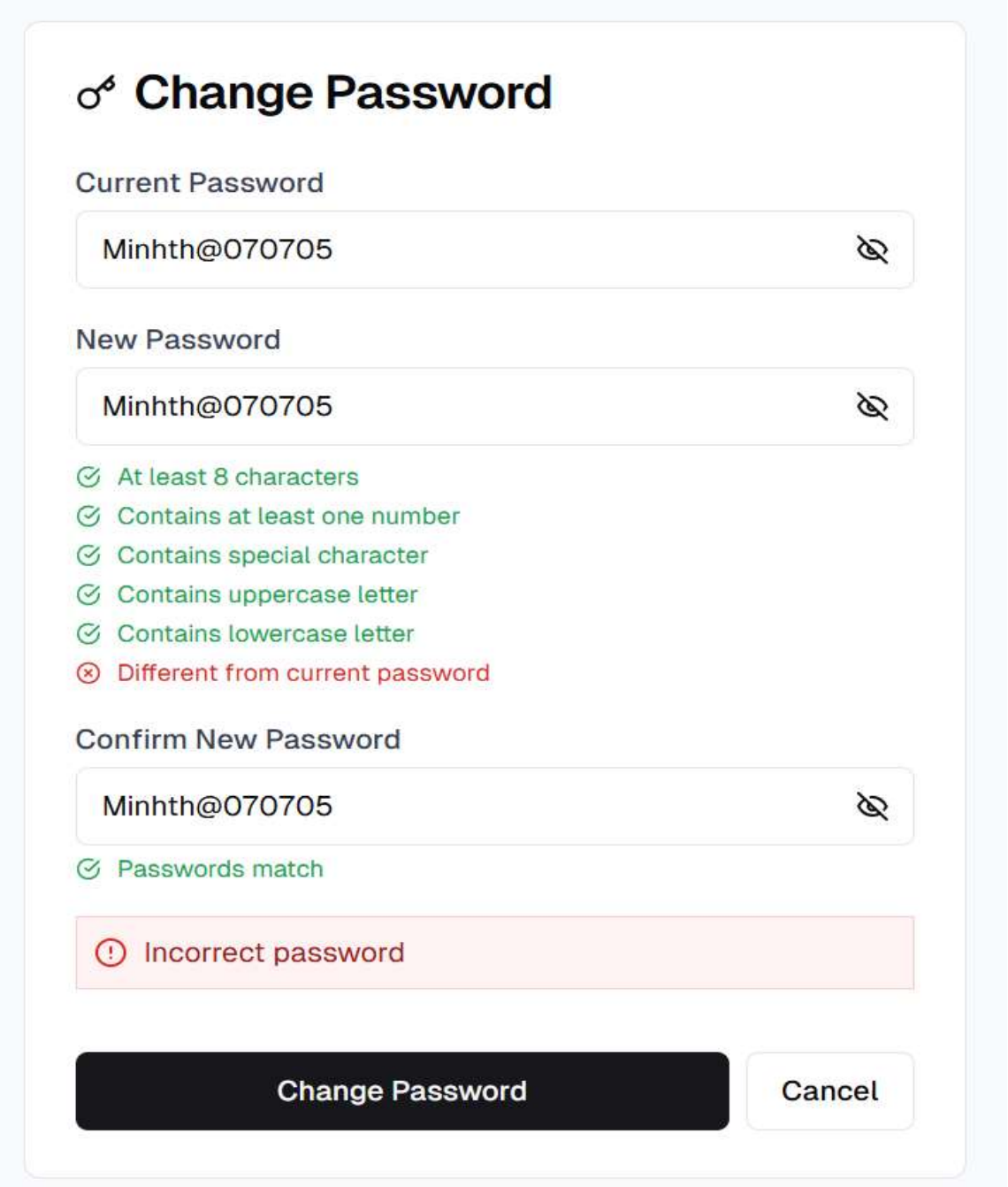


©Error! No property name supplied., 2025

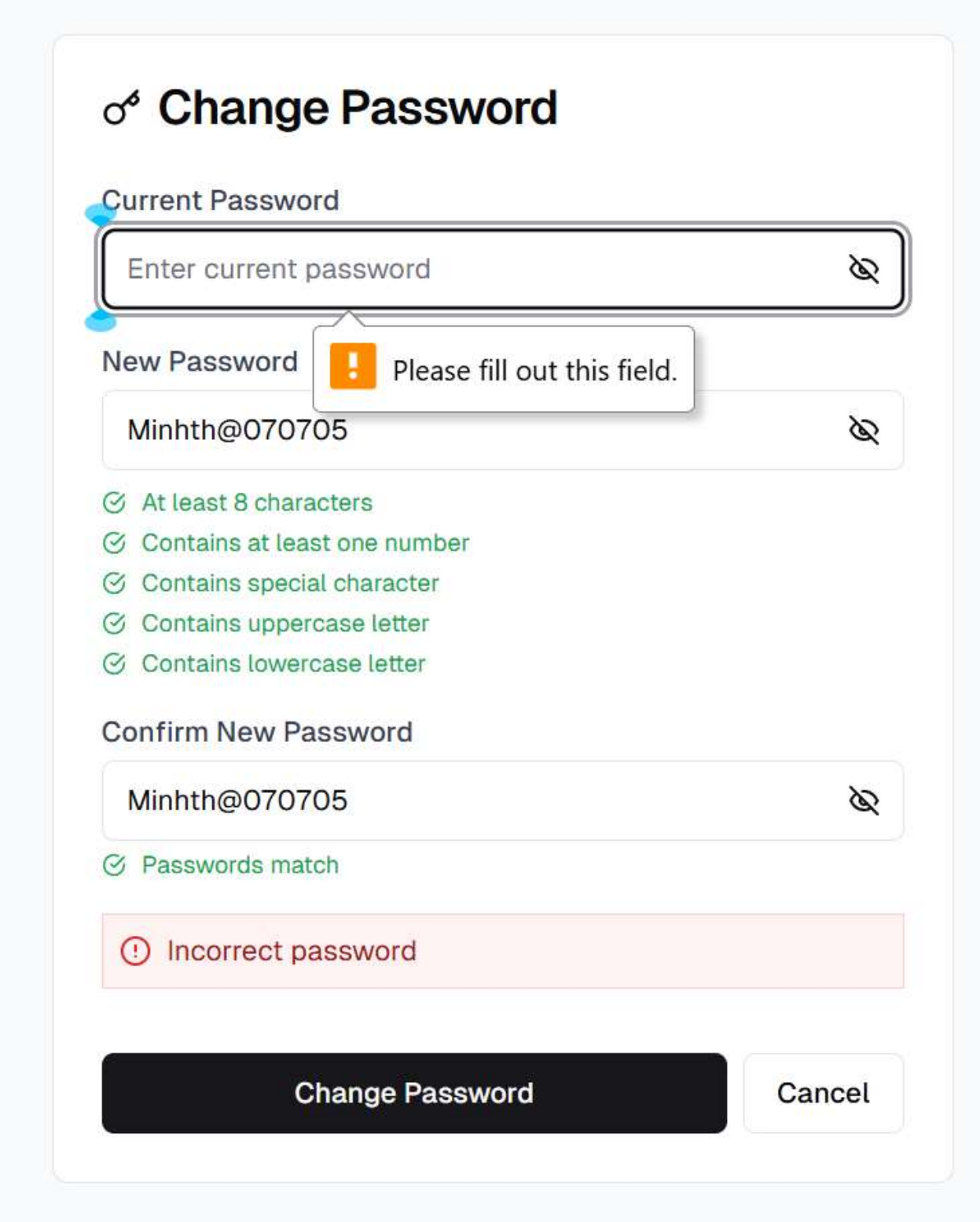
After the user clicks the Change Password button, the system will check the information. In this case, the password does not meet the format and security criteria for password requirements, so that the system messages "Incorrect password"



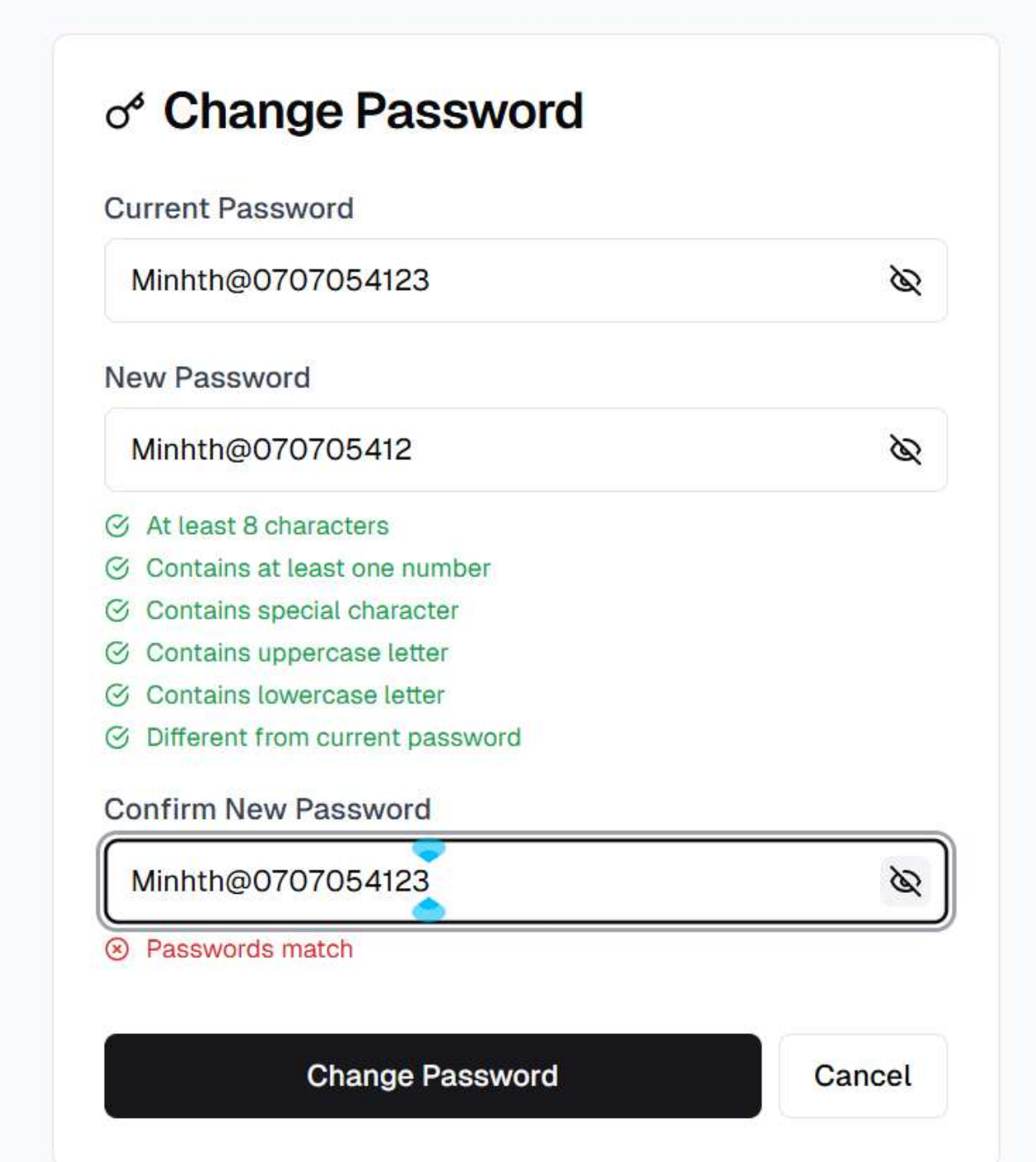
The systems show the mistakes directly when the user typing the wrong password.

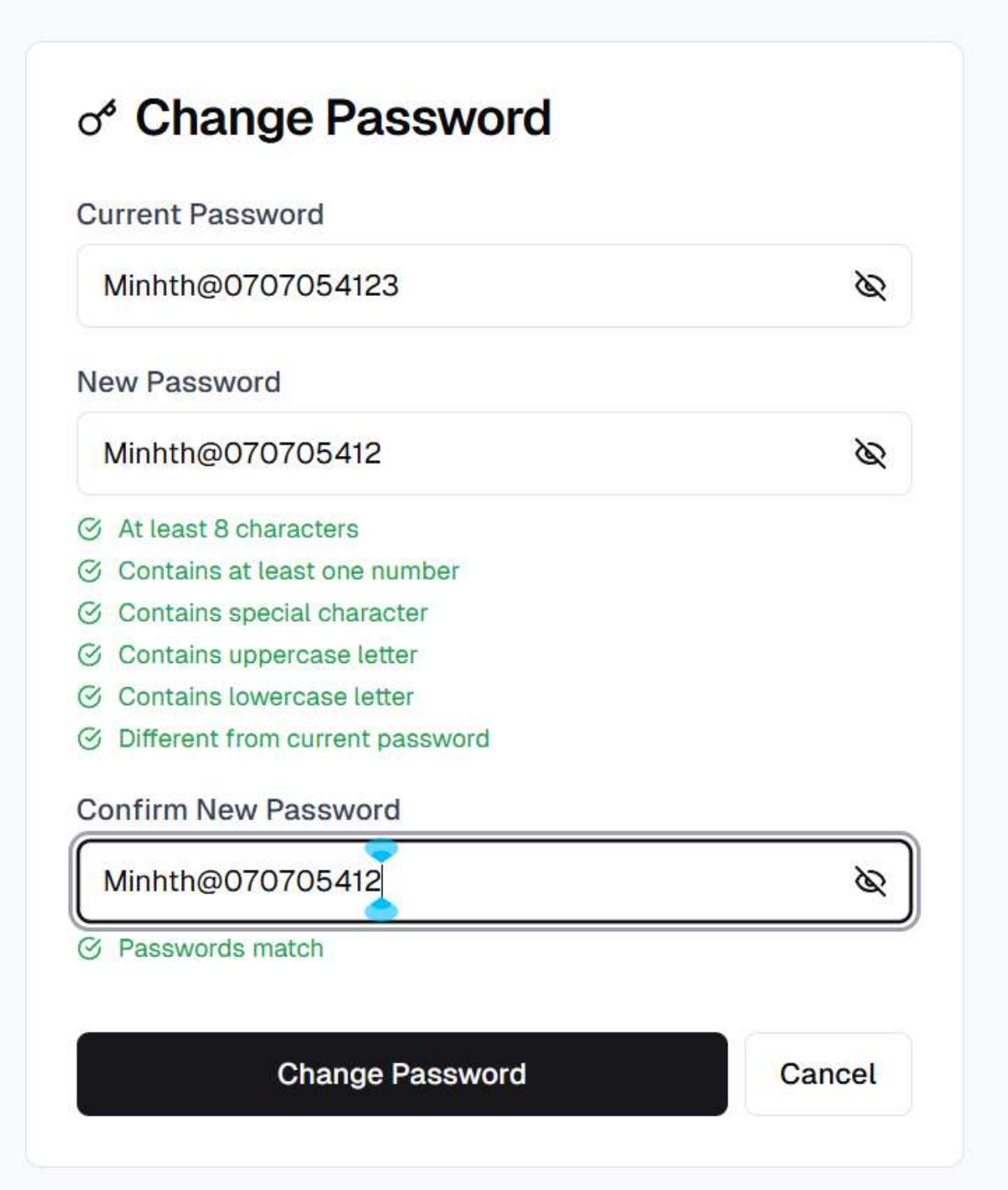


The system still check the Confirm New Password with New Password regardless of it is the same or not the same with Old Password.


Change Password
Current Password
Enter current password
Please fill out this field.
New Password
Minhth@070705
At least 8 characters
Contains at least one number
Contains special character
Contains uppercase letter
Contains lowercase letter
Confirm New Password
Minhth@070705
Passwords match
Incorrect password
Change Password
Cancel

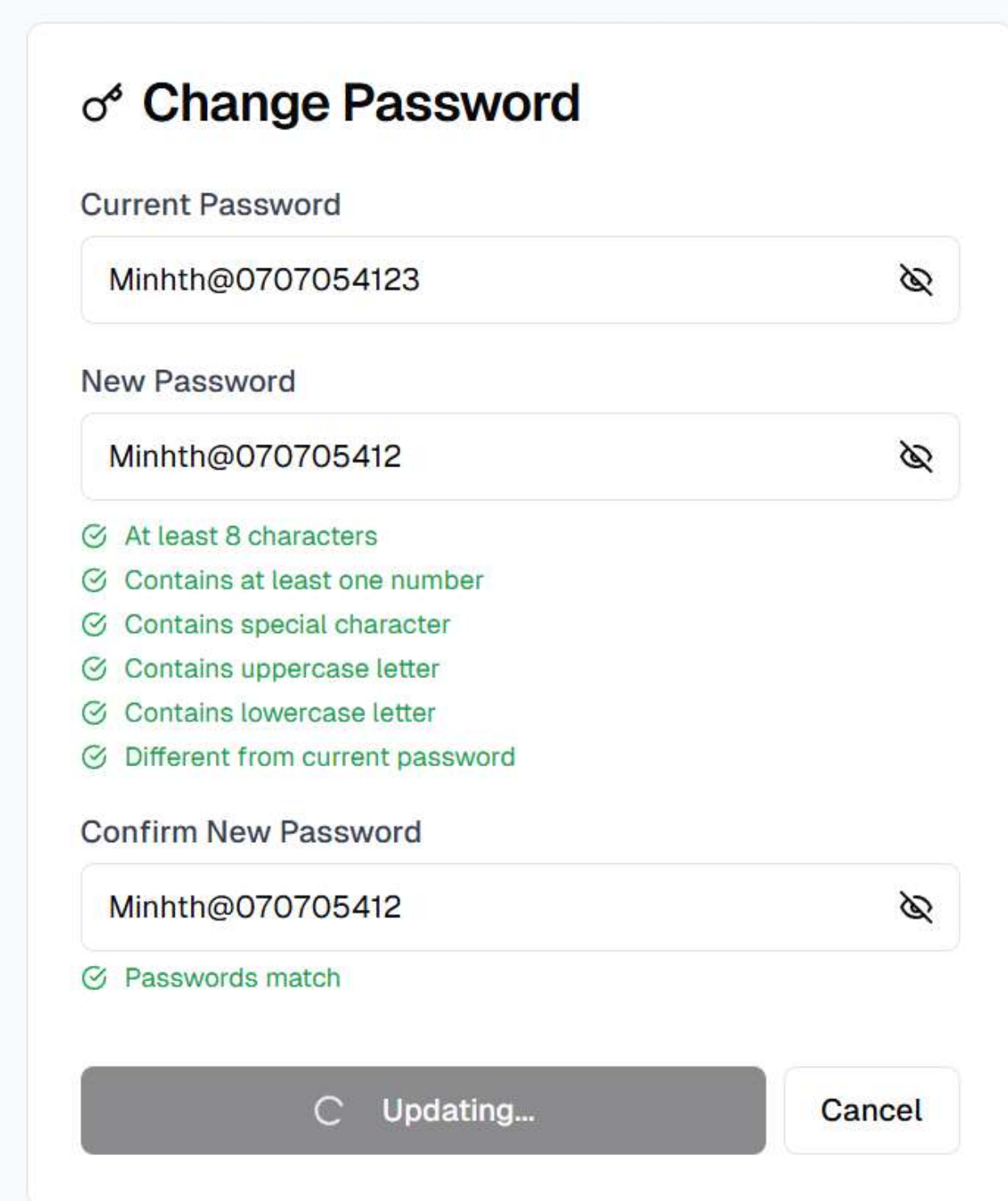
The system messages "Please fill out this field" at the box that the user leave it empty.



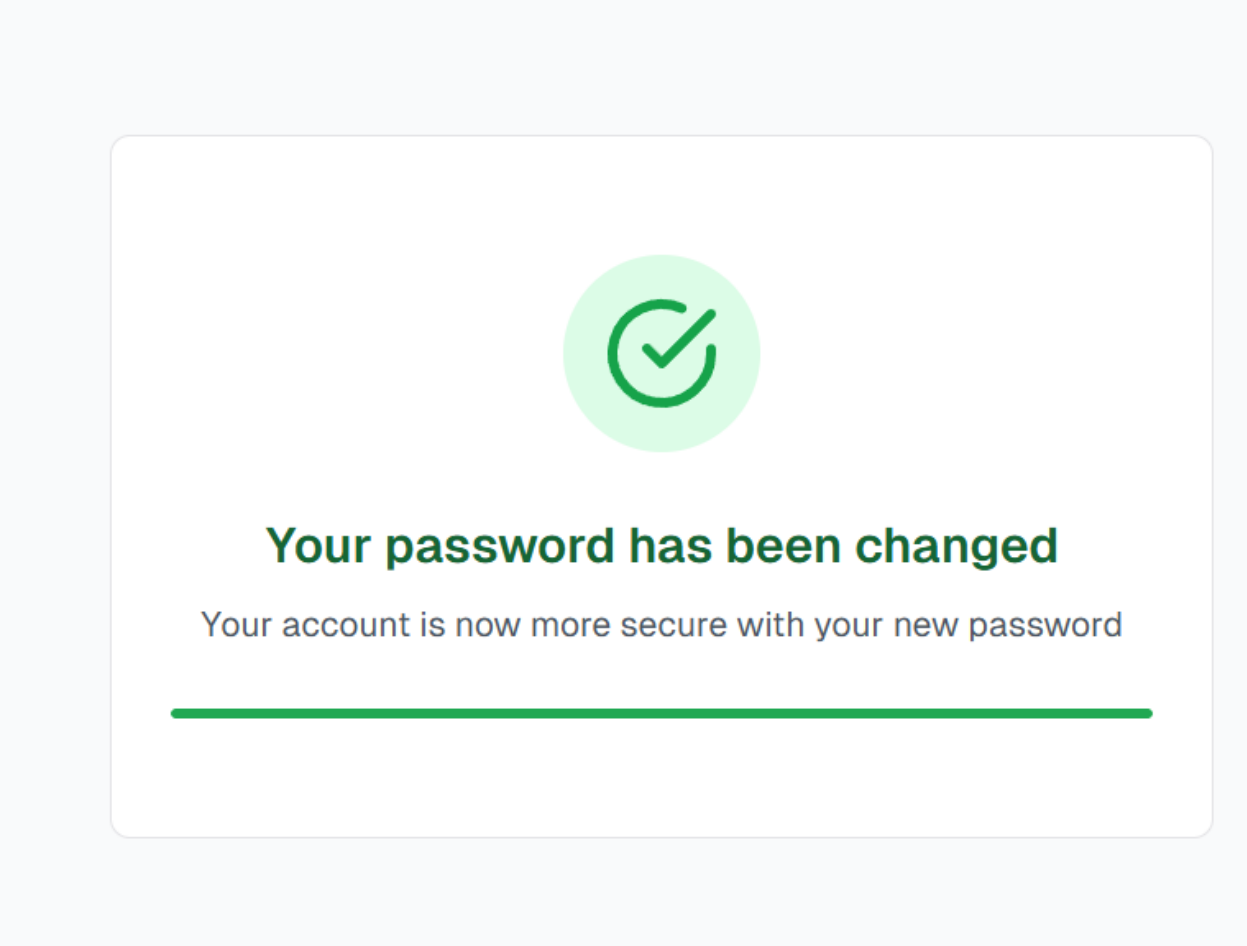


No red color check, that means the password is correct with the format and security criteria for password

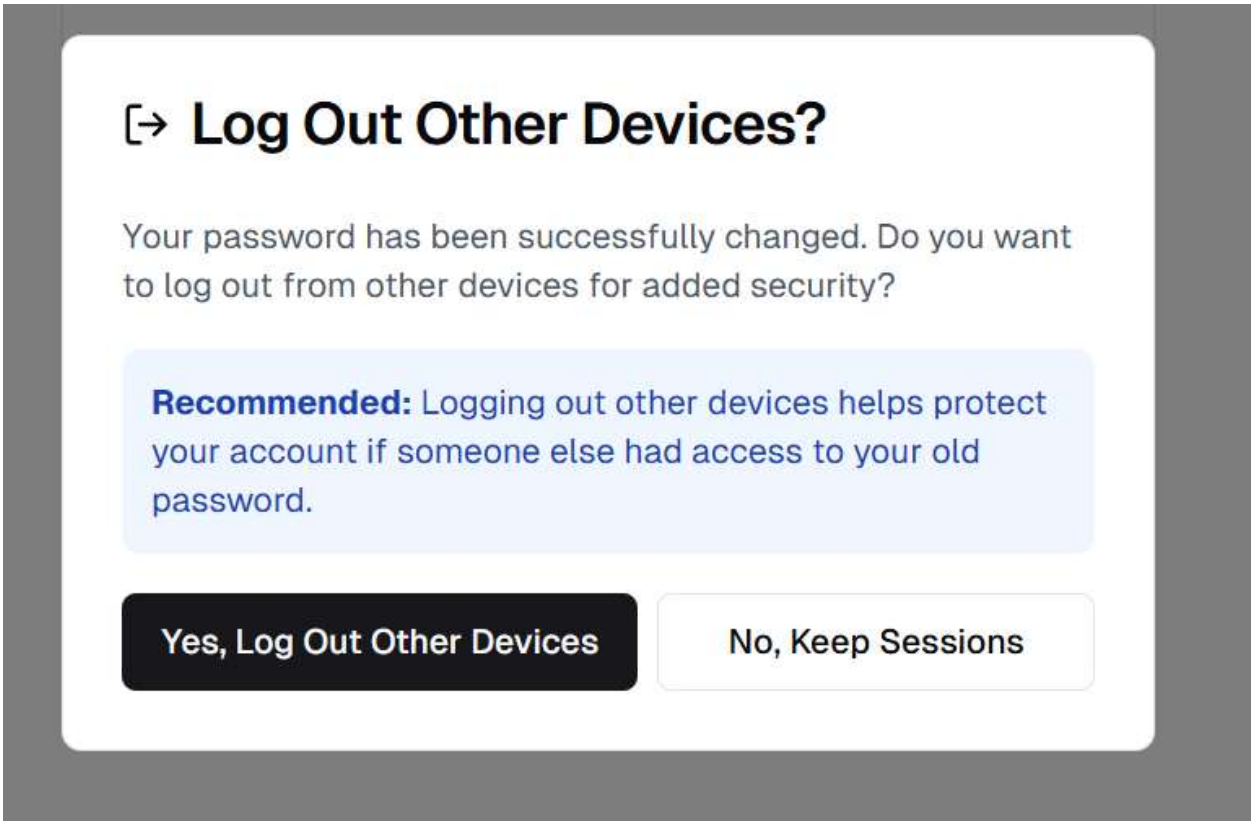
requirements.



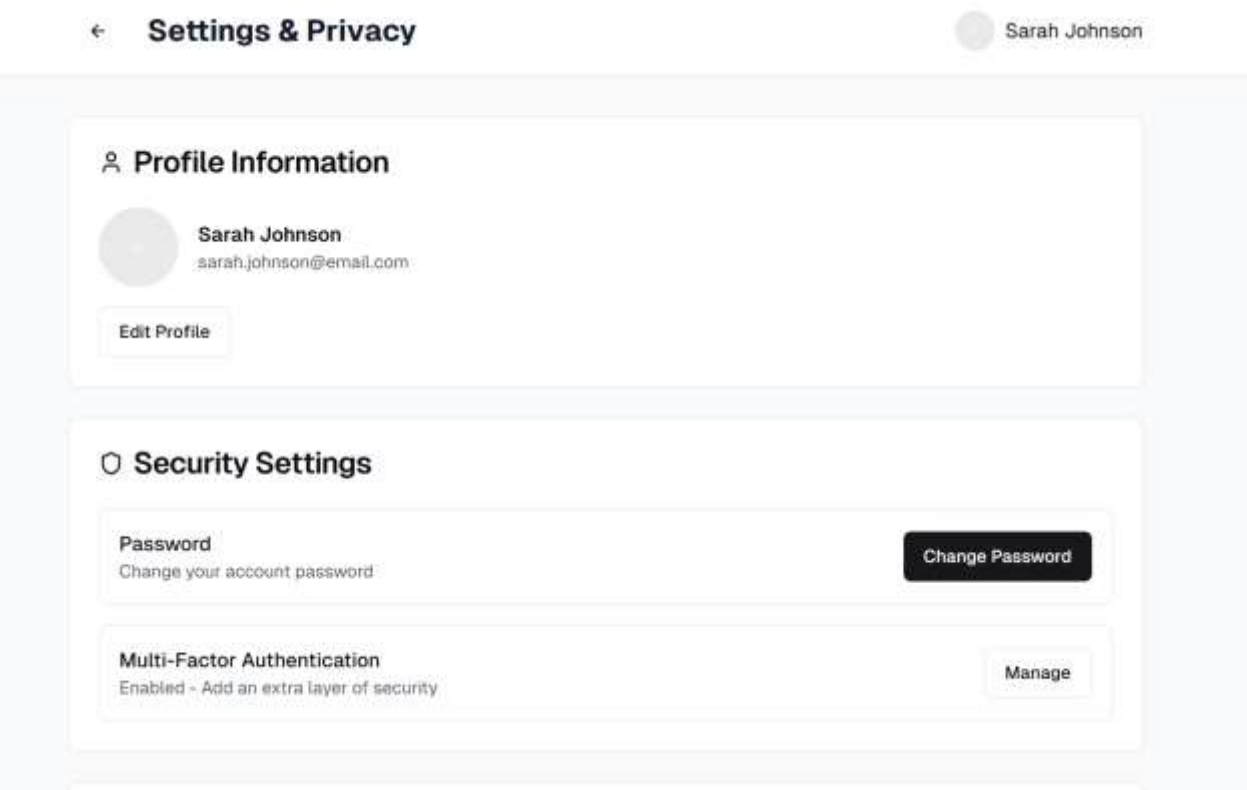
After clicking the Change Password button, because the Old Password is corrected and the New Password and Confirm New Password meet and concurrency. So that the system Updating the New Password.



©Error! No property name supplied., 2025



After Change Password successfully, the system asks the user is they want to log out other devices or not



After choosing "Yes, Log Out Other Devices" or "No, Keep Sections", the system back to the Profile Information scree

404 Not Found

Digital Newspaper Website
Use-Case Specification: Recover Password

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Specific about the Recover Password use case	Thạch Ngọc Hân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Recover Password **4**

1.1 Brief Description 4

2. Flow of Events **4**

2.1 Basic Flow 4

2.2 Alternative Flows 4

2.2.1 Invalid Email Address 4

2.2.2 Expired or Invalid Token 4

3. Special Requirements **4**

3.1 Security & Privacy 4

4. Preconditions **4**

4.1 The user is on the login page and selects the option to recover their password. 4

5. Postconditions **5**

5.1 The user’s password has been securely changed. 5

6. Extension Points **5**

6.1 Redirect to Login 5

©Error! No property name supplied., 2025

Use-Case Specification: Recover Password

1. Use-Case Name: Recover Password

1.1 Brief Description

This use case describes how a user can recover access to their account in case they forget their password. It involves requesting a password reset link via email and setting a new password using a secure recovery token.

2. Flow of Events

2.1 Basic Flow

1. The use case starts when the user clicks on the “Forgot Password?” link on the login page.
2. The user is directed to the Password Recovery page.
3. The user enters their registered email address and submits the form.
4. The System validates the email format and checks if it exists in the database.
5. If the email is valid, the system generates a secure, time-limited password reset token
6. The system sends a recovery email with a reset link to the user’s email address.
7. The user clicks the link and is redirected to the Reset Password page.
8. The user enters a new password and confirms it.
9. The system validates the token and the new password, updates the password in the database, and notifies the user of success.
10. The use case ends.

2.2 Alternative Flows

2.2.1 Invalid Email Address

At step 4 of the Basic Flow, if the email address is not associated with any account:

a. The system shows an error message: “This email is not registered.”
b. The user may try a different email or return to the login page.

2.2.2 Expired or Invalid Token

At step 9 of the Basic Flow, if the token is expired or tampered with:

a. The system displays a message: “This reset link is invalid or has expired.”
b. The user is prompted to request a new password reset link.
c. The flow returns to step 1 of the Basic Flow.

3. Special Requirements

3.1 Security & Privacy

1. The reset token must be encrypted, unique, and expire within 15 minutes.
2. The new password must meet system-defined complexity requirements.

4. Preconditions

4.1 The user is on the login page and selects the option to recover their password.

4.2 The system is online and email services are operational.

©Error! No property name supplied., 2025

5. Postconditions

5.1 The user's password has been securely changed.

5.2 The reset token is invalidated.

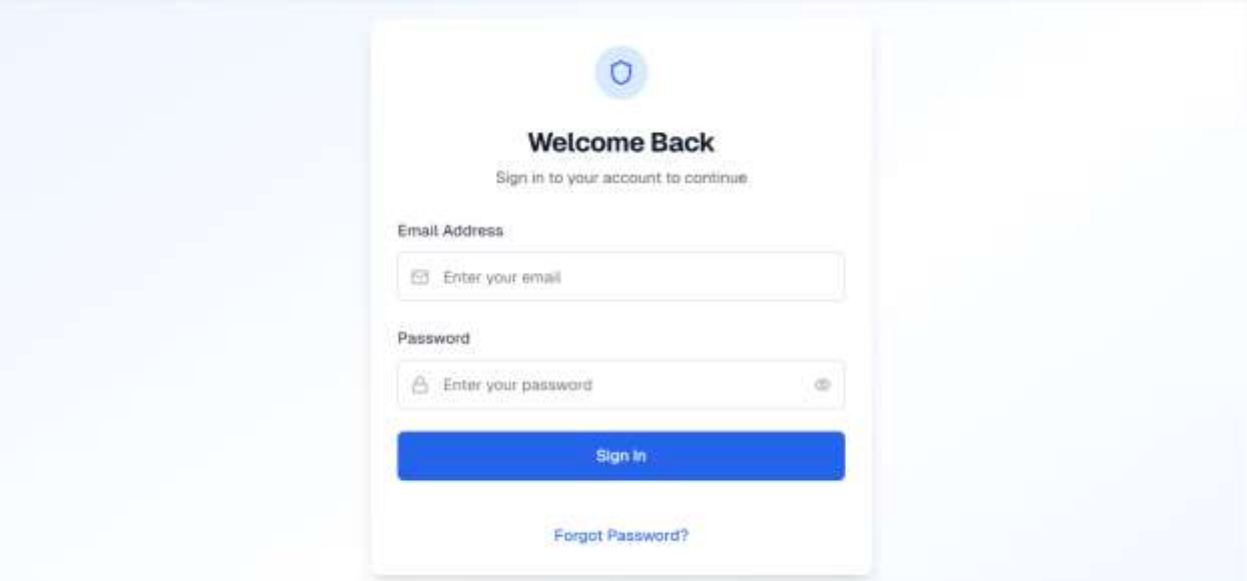
5.3 The user may now log in with the new password.

6. Extension Points

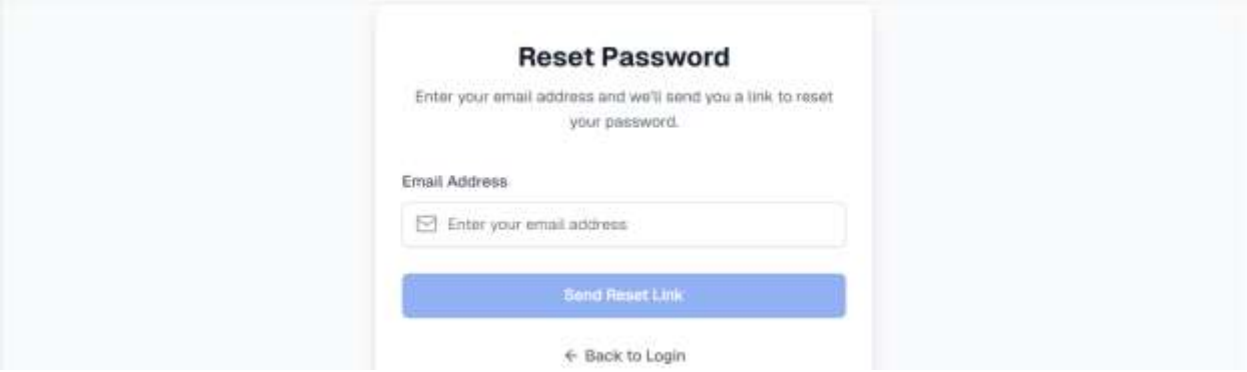
6.1 Redirect to Login

- Location: After step 9 of the Basic Flow.
- Description: The system redirects the user to the login page with a success message: "Your password has been updated. Please log in."

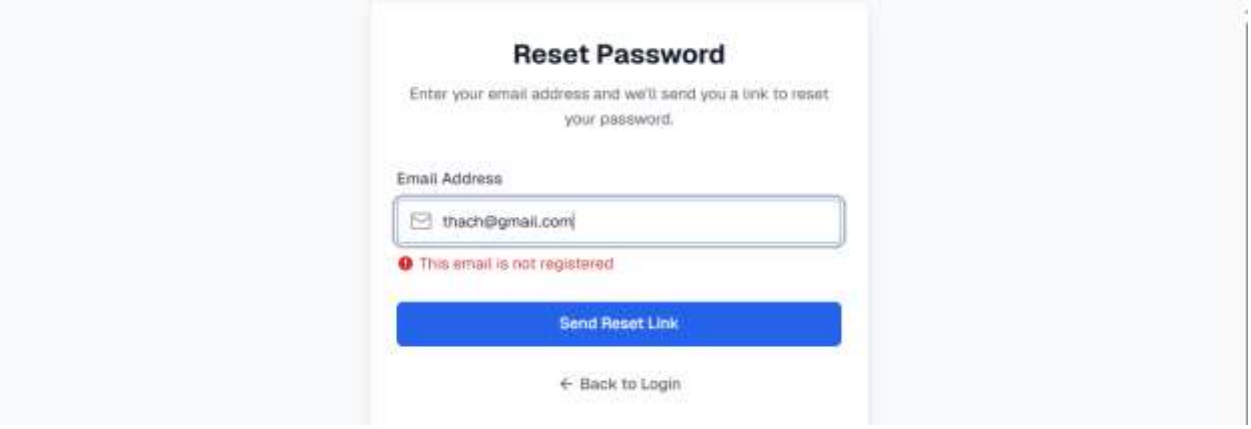
Original interface



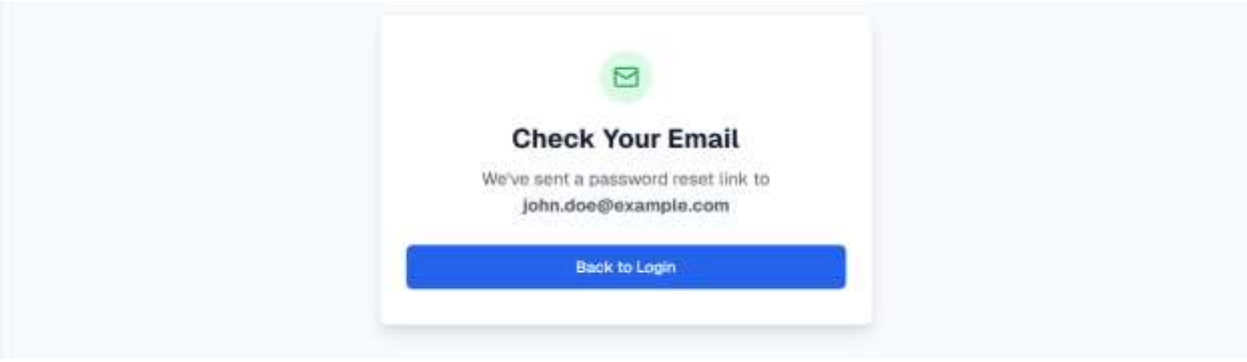
When you click "Forgot Password?"



When you enter an invalid email



When the email is valid



404 Not Found

Digital Newspaper Website
Use-Case Specification: View Recommended Articles

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
14/07/2025	1.0	Complete the document	Trần Hữu Khang

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: View Recommended Articles 4
1.1 Brief Description 4
2. Flow of Events 4
2.1 Basic Flow 4
2.2 Alternative Flows 4
2.2.1 No Recommendations Available 4
2.2.2 Reader Shares Article 4
3. Special Requirements 4
3.1 Recommendation Algorithm 4
3.2 Performance 4
3.3 Accessibility 4
3.4 Browser Compatibility 5
3.5 Data Privacy 5
4. Preconditions 5
4.1 System Accessibility 5
4.2 Article Database 5
4.3 Reader Profile 5
5. Postconditions 5
5.1 Article List Display 5
5.2 Article Content Access 5
5.3 Browsing History Update 5
6. Extension Points 5
6.1 Customize Recommendations 5
7. Low Fidelity Prototype For Use Case 5
7.1 Prompt Used for AI Tool 5
7.2 Generated Interface 5

©Error! No property name supplied., 2025

Use-Case Specification: View Recommended Articles

1. Use-Case Name: View Recommended Articles

1.1 Brief Description

This use case describes the process by which a Reader views a list of recommended articles on the Digital Newspaper Website. The system uses the Reader's browsing history, preferences, or trending topics to generate and display a personalised or curated list of articles.

2. Flow of Events

2.1 Basic Flow

This use case starts when the Reader accesses the Digital Newspaper Website.

1. The Reader navigates to the homepage or a dedicated "Recommended Articles" section.
2. The system retrieves the Reader's profile data (if logged in) or session-based browsing history (if not logged in) to generate a list of recommended articles/
3. The system displays a list of recommended articles, including each article's title, a thumbnail image (if available), a brief excerpt (first 50 words of the article), and publication date.
4. The Reader scrolls through the list and selects an article by clicking on its title or thumbnail.
5. The system displays the full article content, including text, images, and any embedded media.
6. The Reader may choose to return to the recommended articles list or continue browsing other sections of the website.

2.2 Alternative Flows

2.2.1 No Recommendations Available

- If the system cannot generate recommendations (e.g., no browsing history for a new user or system error), it displays a default list of trending articles based on the most viewed or shared articles in the past 24 hours.
- The flow continues from Basic Flow step 3.

2.2.2 Reader Shares Article

- At Basic Flow step 5, the Reader selects a "Share" option (e.g., via social media, email, or link copy).
- The system provides sharing options (e.g., Twitter, Facebook, email) and, upon selection, facilitates the sharing process (e.g., opens a pre-filled tweet or copies the article URL to the clipboard).
- The flow continues from Basic Flow step 6.

3. Special Requirements

3.1 Recommendation Algorithm

The recommendation algorithm must prioritise articles based on relevance to the Reader's preferences or browsing history, with a fallback to trending articles.

3.2 Performance

The system must load the recommended articles list within 2 seconds under normal server load.

3.3 Accessibility

The system must support accessibility standards (e.g., WCAG 2.1) for the article list and content display, including screen reader compatibility.

©Error! No property name supplied., 2025

3.4 Browser Compatibility

The system must ensure compatibility with common browsers (e.g., Chrome, Firefox, Safari) and mobile devices.

3.5 Data Privacy

The system must comply with GDPR for handling Reader data used in generating recommendations.

4. Preconditions

4.1 System Accessibility

The Digital Newspaper Website must be accessible and operational.

4.2 Article Database

The system must have access to a database of published articles.

4.3 Reader Profile

If personalised recommendations are used, the Reader must have a profile with stored preferences or a browsing history.

5. Postconditions

5.1 Article List Display

The Reader views a list of recommended or trending articles.

5.2 Article Content Access

The Reader may view the full content of a selected article.

5.3 Browsing History Update

The system updates the Reader's browsing history with the viewed article (if tracking is enabled).

6. Extension Points

6.1 Customize Recommendations

- Location: Basic Flow, step 2.
- The Reader can update their preferences (e.g., preferred categories or topics) via a settings page, which the system uses to refine the recommendation algorithm.

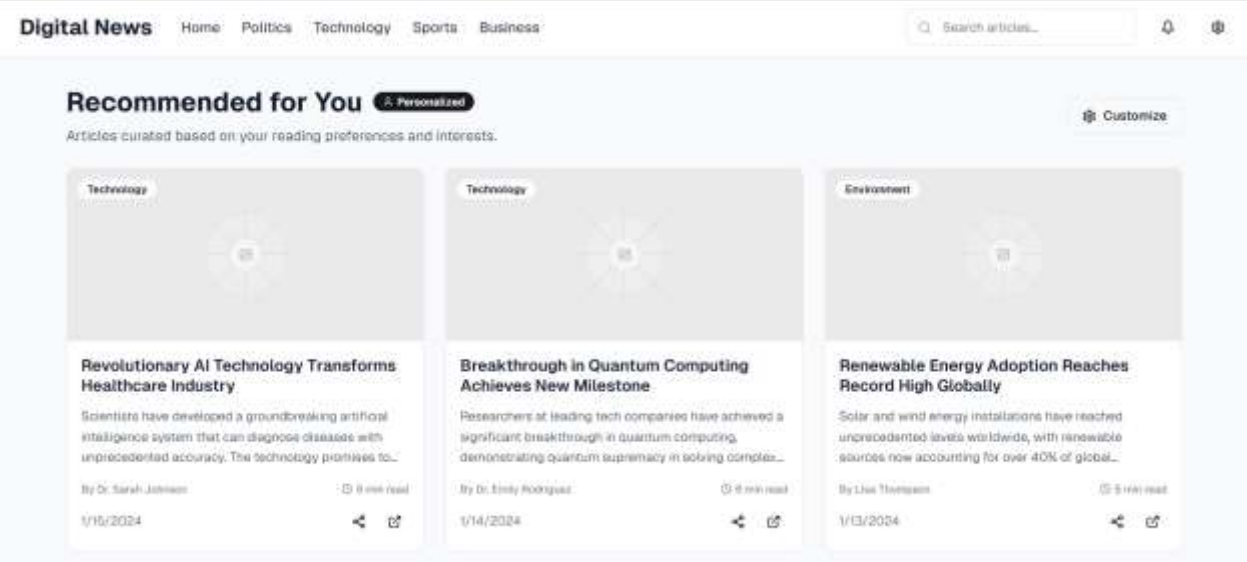
7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

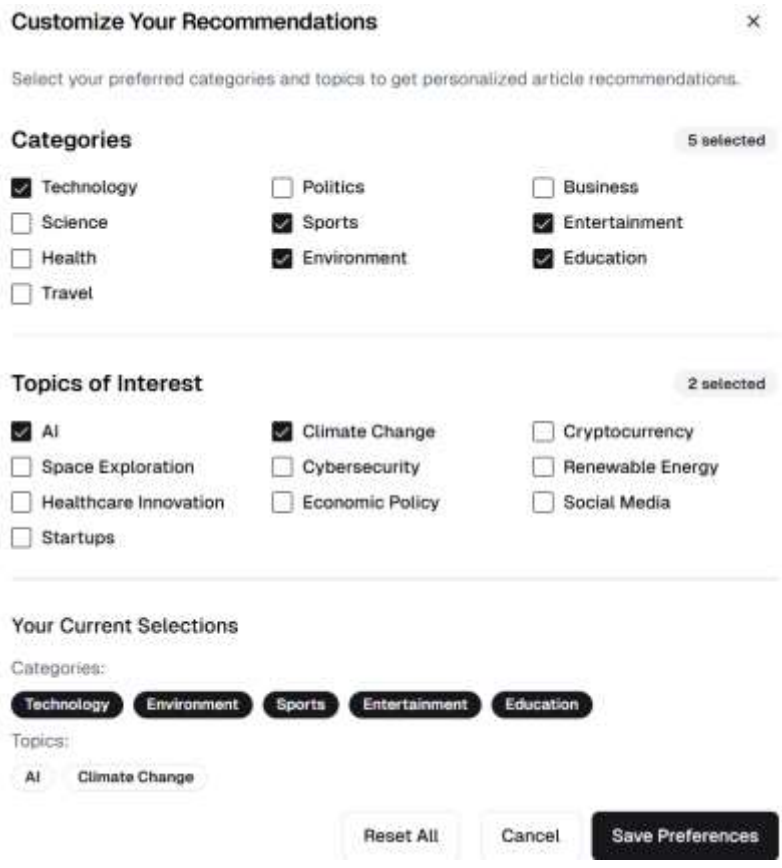
- Design a responsive web interface for a Digital Newspaper Website that implements the "View Recommended Articles" use case.
- Functional Requirements: [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*
- Optional: Sidebar or settings page for updating user preferences. / Visual indicator for whether recommendations are based on history or trending.

7.2 Generated Interface

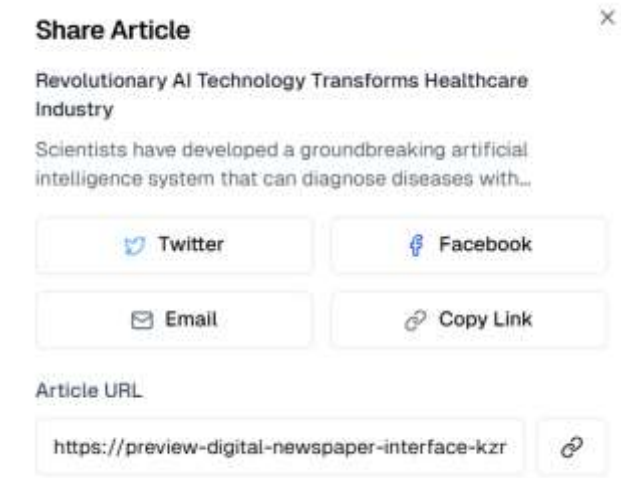
The home page:



Customisation Menu:



Sharing post:



404 Not Found

Digital Newspaper Website
Use-Case Specification: Generate Reports

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
14/07/2025	1.0	Complete the document	Trần Hữu Khang

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Generate Reports **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows **4**
2.2.1 Custom Report Filters 4
2.2.2 Export Failure **4**
3. Special Requirements **4**
3.1 Supported Metrics 4
3.2 Performance 4
3.3 Export Formats 5
3.4 Browser Compatibility 5
3.5 Data Privacy 5
4. Preconditions **5**
4.1 Admin Authentication 5
4.2 Admin Permissions 5
4.3 System Accessibility 5
4.4 Data Availability 5
5. Postconditions **5**
5.1 Report Generation 5
5.2 Confirmation 5
5.3 Report Delivery 5
6. Extension Points **5**
6.1 Schedule Report Generation 5
7. Low Fidelity Prototype For Use Case **5**
7.1 Prompt Used for AI Tool 5
7.2 Generated Interface 6

Use-Case Specification: Generate Reports

1. Use-Case Name: Generate Reports

1.1 Brief Description

This use case describes the process by which an Admin generates reports on the Digital Newspaper Website's performance, user activity, or content metrics using the content management system (CMS). The Admin selects report parameters, generates the report, and exports or views the results.

2. Flow of Events

2.1 Basic Flow

This use case starts when the Admin logs into the Digital Newspaper Website's CMS and accesses the reporting dashboard.

1. The Admin navigates to the "Reports" section in the CMS.
2. The system displays a list of available report types.
3. The Admin selects a report type and specifies parameters.
4. The system validates the input parameters.
5. The Admin selects the output format.
6. The system generates the report based on the selected type and parameters, retrieving relevant data from the CMS database.
7. The system displays the report on-screen or provides a download link for the selected format.
8. The Admin reviews the report and may choose to save, print, or share it via email.
9. The system confirms the successful generation and delivery of the report to the Admin.

2.2 Alternative Flows

2.2.1 Custom Report Filters

- At Basic Flow step 3, the Admin applies advanced filters (e.g., specific Writers, article categories, or geographic regions for traffic).
- The system updates the report generation options to reflect the custom filters and proceeds with validation (returns to Basic Flow step 4).

2.2.2 Export Failure

- At Basic Flow step 7, if the report export fails (e.g., due to file format issues or server timeout), the system displays an error message (e.g., "Export failed. Please try again or select a different format").
- The Admin can retry the export or choose a different format (returns to Basic Flow step 5).

3. Special Requirements

3.1 Supported Metrics

The CMS must support report generation for at least the following metrics: article views, user registrations, comment activity, and website traffic.

3.2 Performance

The system must generate and display or export reports within 10 seconds under normal server load for datasets up to 10,000 records.

©Error! No property name supplied., 2025

3.3 Export Formats

The system must support report exports in PDF and CSV formats, with PDF reports adhering to a professional layout.

3.4 Browser Compatibility

The system must ensure compatibility with common browsers (e.g., Chrome, Firefox, Safari) for the reporting interface.

3.5 Data Privacy

The system must comply with GDPR for handling personal data included in reports.

4. Preconditions

4.1 Admin Authentication

The Admin must be registered and logged into the CMS with valid credentials.

4.2 Admin Permissions

The Admin must have the role of "Admin" with permissions to generate reports.

4.3 System Accessibility

The CMS must be accessible and operational.

4.4 Data Availability

The CMS database must contain relevant data (e.g., articles, user activity, or traffic metrics) for report generation.

5. Postconditions

5.1 Report Generation

The report is generated and either displayed on-screen or exported as a PDF or CSV file.

5.2 Confirmation

The Admin receives confirmation of the report generation or export.

5.3 Report Delivery

If exported, the report is saved or sent to the Admin’s specified destination (e.g., downloaded file or email).

6. Extension Points

6.1 Schedule Report Generation

- Location: Basic Flow, step 3.
- The Admin can schedule recurring reports (e.g., weekly or monthly) by specifying a schedule and delivery method (e.g., email). The system generates and sends the report automatically at the specified intervals.

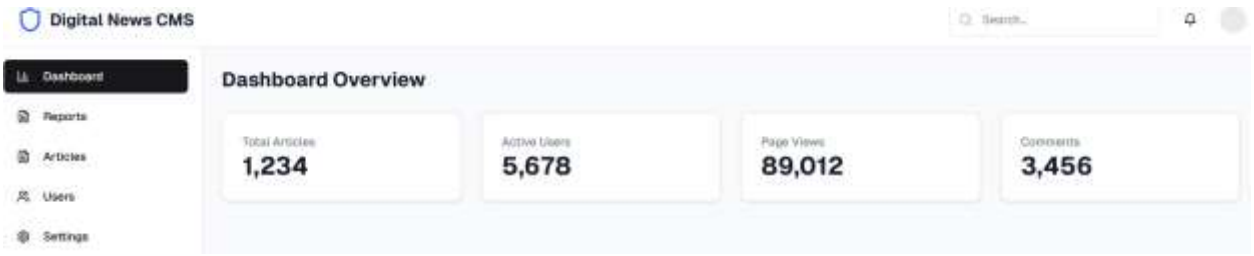
7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

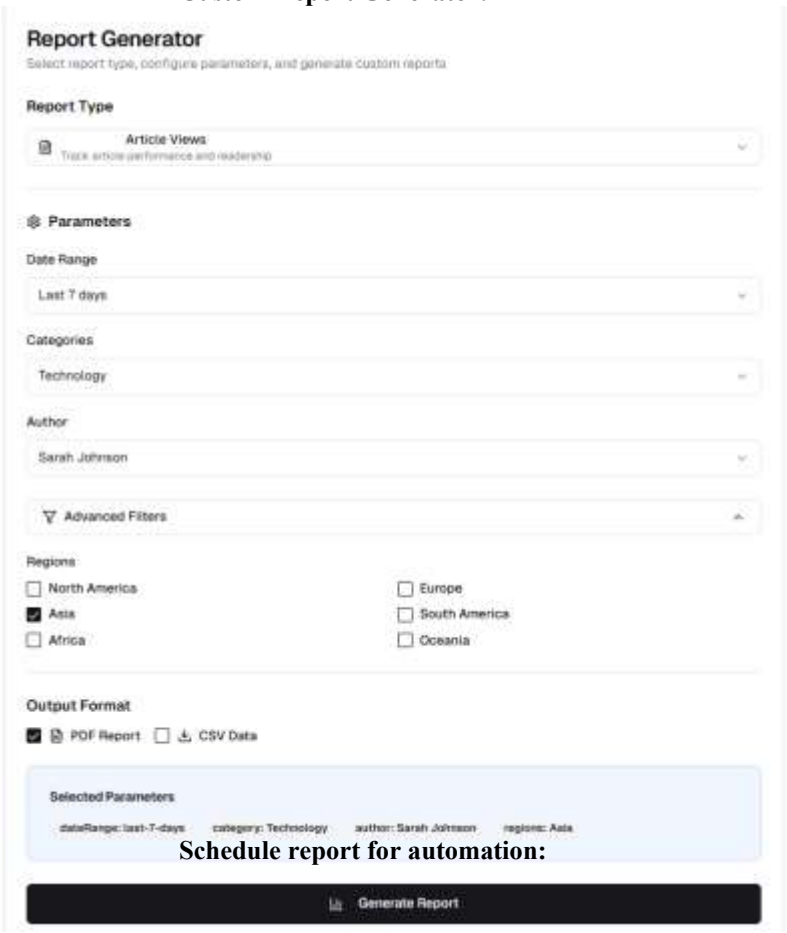
- Design an Admin Dashboard interface for a Digital Newspaper Website CMS that supports the "Generate Reports" use case.
- Functional Requirements: [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*
- Assume the admin is already authenticated and viewing the CMS dashboard.

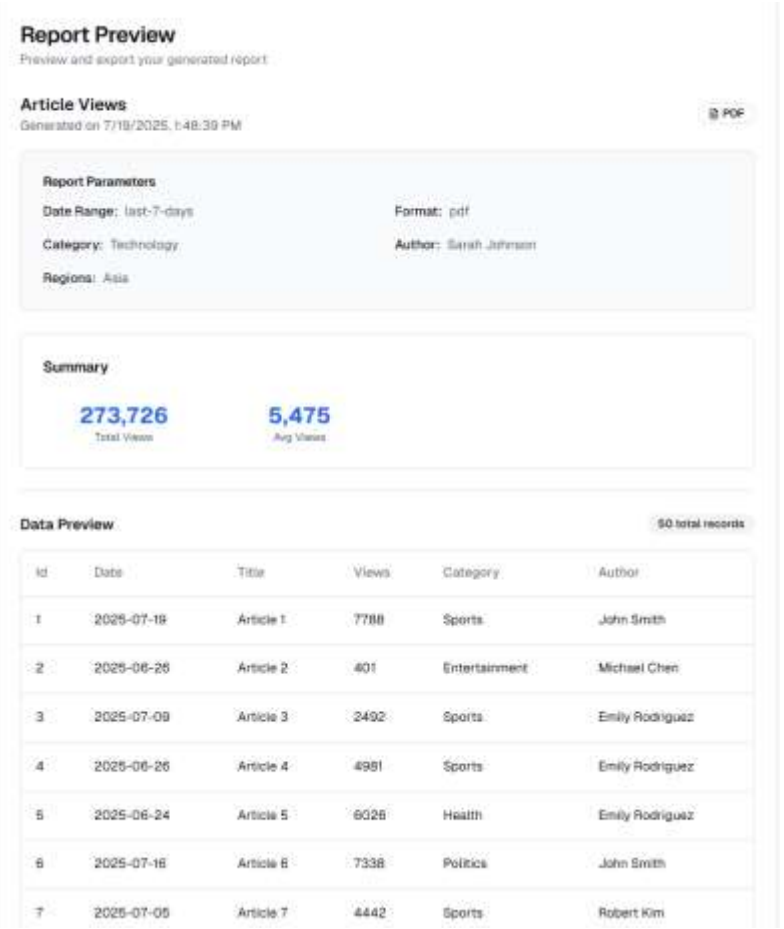
7.2 Generated Interface

The overall dashboard from the Admin perspective:

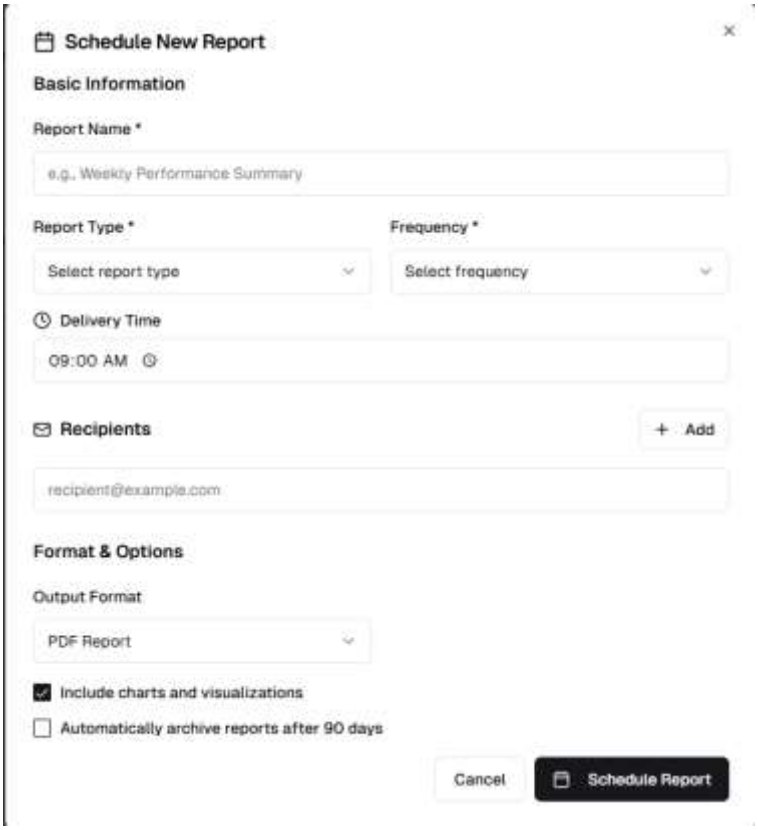


Custom Report Generator:

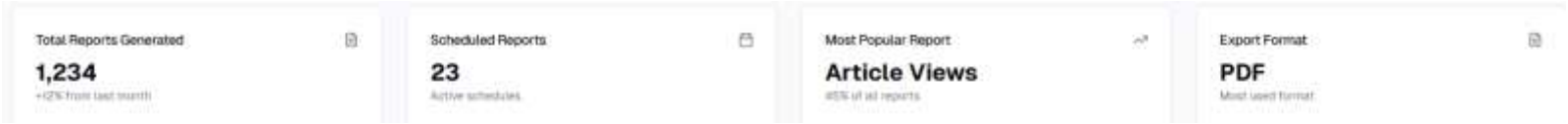




Schedule report for automation:



Analytics on reports:



404 Not Found

Digital Newspaper Website
Use-Case Specification: Moderate Comments

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
14/07/2025	1.0	Complete the document	Trần Hữu Khang

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Moderate Comments **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows **4**
2.2.1 Bulk Moderation 4
2.2.2 Flagged Comment Review **4**
3. Special Requirements **5**
3.1 Comment Filtering 5
3.2 Performance 5
3.3 Bulk Moderation 5
3.4 Browser Compatibility 5
3.5 Data Privacy 5
4. Preconditions **5**
4.1 Editor Authentication 5
4.2 Editor Permissions 5
4.3 System Accessibility 5
4.4 Comment Availability 5
5. Postconditions **5**
5.1 Comment Status Update 5
5.2 Commenter Notification 5
5.3 Comment Visibility 5
5.4 Action Confirmation 5
6. Extension Points **6**
6.1 Ban Commenter 6
7. Low Fidelity Prototype For Use Case **6**
7.1 Prompt Used for AI Tool 6
7.2 Generated Interface 6

©Error! No property name supplied., 2025

Use-Case Specification: Moderate Comments

1. Use-Case Name: Moderate Comments

1.1 Brief Description

This use case describes the process by which an Editor moderates user comments on articles published on the Digital Newspaper Website. The Editor reviews, approves, rejects, or deletes comments using the content management system (CMS) to ensure they comply with community guidelines.

2. Flow of Events

2.1 Basic Flow

This use case starts when the Editor logs into the Digital Newspaper Website's CMS and accesses the comment moderation dashboard.

1. The Editor navigates to the "Comment Moderation" section in the CMS.
2. The system displays a list of comments, including their status (e.g., Pending, Approved, Rejected), the associated article title, the commenter's username or email, the comment text, and the submission date.
3. The Editor selects a comment marked "Pending" to review its content.
4. The system displays the full comment text, the associated article, and available actions (e.g., Approve, Reject, Delete, Edit).
5. The Editor reviews the comment for compliance with community guidelines and chooses one of the following actions:
    a. **Approve**: The Editor marks the comment as approved for public display.
    b. **Reject**: The Editor rejects the comment and provides optional feedback.
    c. **Delete**: The Editor deletes the comment without notifying the commenter.
    d. **Edit**: The Editor modifies the comment (e.g., removes offensive words) and approves or resubmits it for review.
6. If the Editor selects "Edit," the system saves the modified comment and updates its status based on the Editor's choice (e.g., Approved or Pending).
7. If the Editor selects "Approve," the system marks the comment as "Approved," and it becomes visible on the article's public comment section.
8. If the Editor selects "Reject," the system notifies the commenter via email or CMS notification with the provided feedback (if any).
9. If the Editor selects "Delete," the system removes the comment from the CMS without notification.
10. The system confirms the action to the Editor.

2.2 Alternative Flows

2.2.1 Bulk Moderation

- At Basic Flow step 2, the Editor selects multiple comments and applies a bulk action (e.g., approve, reject, or delete).
- The system prompts the Editor to confirm the bulk action.
- Upon confirmation, the system applies the action to all selected comments and notifies commenters (if applicable, e.g., for rejections).
- The flow continues from Basic Flow step 10.

2.2.2 Flagged Comment Review

- At Basic Flow step 2, the system highlights comments flagged by users or automated filters The Editor prioritises reviewing flagged comments and proceeds with the actions in Basic Flow step 5.

©Error! No property name supplied., 2025

- The flow continues from Basic Flow step 6.

3. Special Requirements

3.1 Comment Filtering

The CMS must support filtering comments by status (Pending, Approved, Rejected) and by article or submission date.

3.2 Performance

The system must load the comment moderation dashboard within 3 seconds under normal server load.

3.3 Bulk Moderation

The system must support bulk moderation for up to 100 comments at once without performance degradation.

3.4 Browser Compatibility

The system must ensure compatibility with common browsers (e.g., Chrome, Firefox, Safari) for the CMS interface.

3.5 Data Privacy

The system must comply with GDPR for handling personal data in comments and notifications.

4. Preconditions

4.1 Editor Authentication

The Editor must be registered and logged into the CMS with valid credentials.

4.2 Editor Permissions

The Editor must have the role of "Editor" with permissions to moderate comments.

4.3 System Accessibility

The CMS must be accessible and operational.

4.4 Comment Availability

At least one comment must be available in the CMS for moderation.

5. Postconditions

5.1 Comment Status Update

The comment’s status is updated to Approved, Rejected, or Deleted in the CMS.

5.2 Commenter Notification

The commenter is notified if the comment is rejected with feedback provided.

5.3 Comment Visibility

Approved comments are visible in the article’s public comment section.

5.4 Action Confirmation

The Editor receives confirmation of the moderation action taken.

6. Extension Points

6.1 Ban Commenter

- Location: Basic Flow, step 5.
- The Editor can ban the commenter from posting further comments by selecting a “Ban User” option. The system updates the commenter’s account status to “Banned” and prevents future comment submissions.

7. Low Fidelity Prototype For Use Case

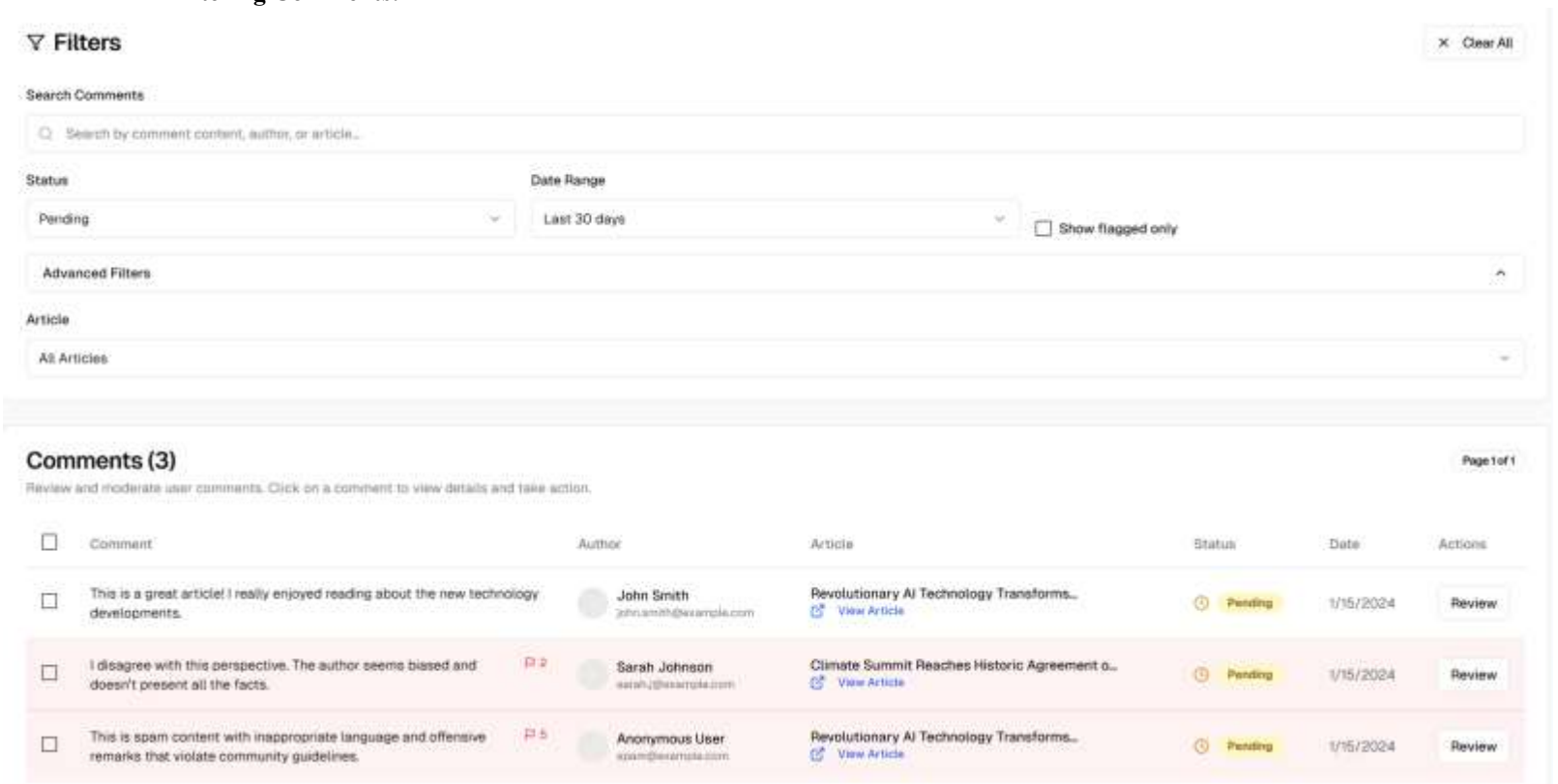
7.1 Prompt Used for AI Tool

- Design a CMS interface for a Digital Newspaper Website that supports the "Moderate Comments" use case. The interface is used by Editors to manage user comments submitted on articles.
- Functional Requirements: [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*
- Optional: A search bar and filter controls (status, date, article). / Pagination for long comment lists.

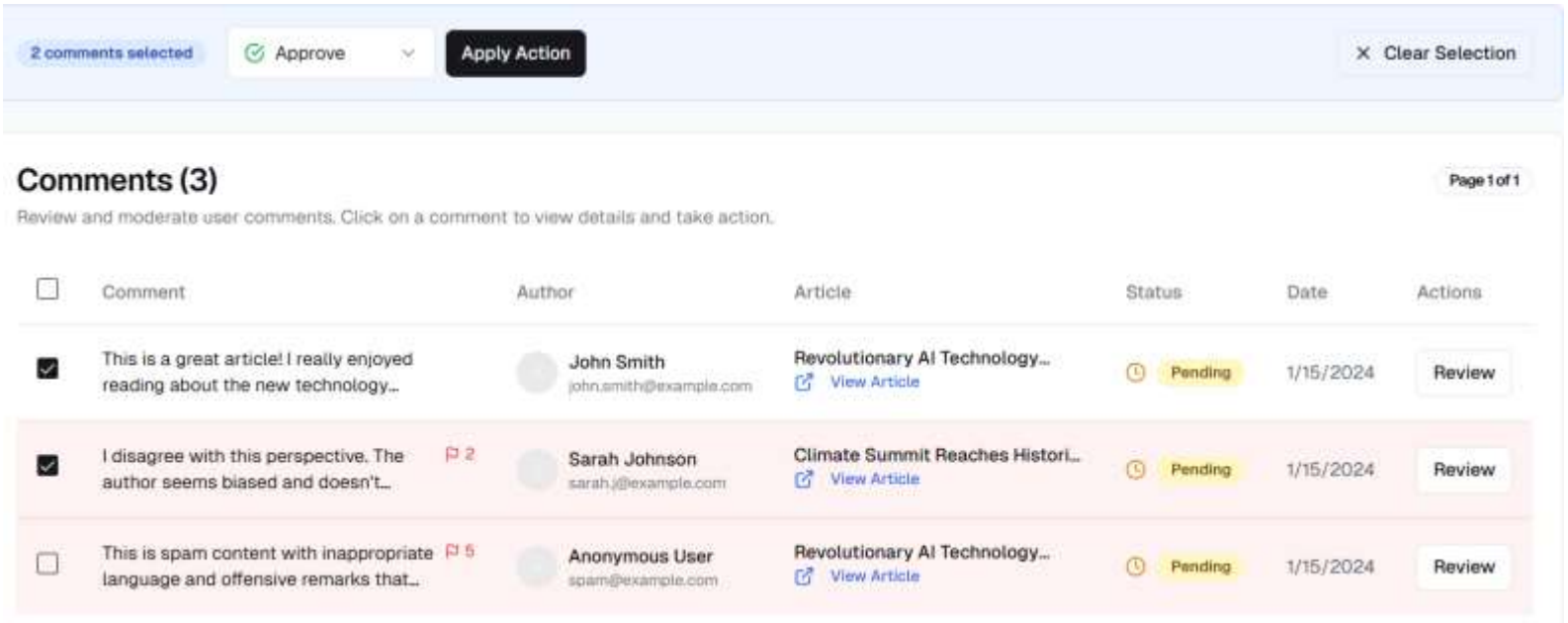
7.2 Generated Interface

Overall Data:

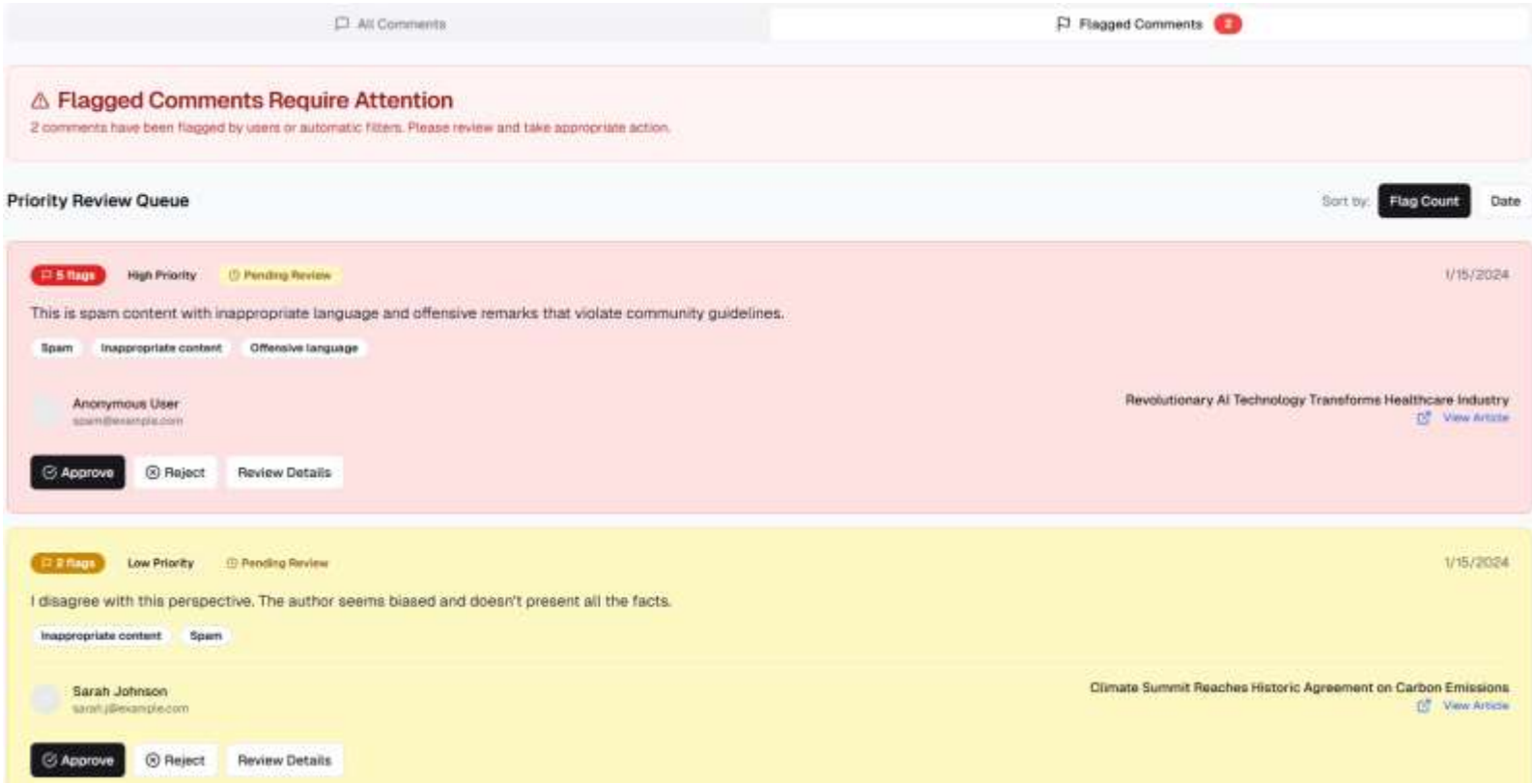
Filtering Comments:



Apply action simultaneously:



Focus on all flagged comments:



404 Not Found

Digital Newspaper Website
Use-Case Specification: Manage User Roles

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Complete the document	Trần Hữu Khang

Table of Contents

1. Use-Case Name: Manage User Roles **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows **4**
2.2.1 No Users Available 4
2.2.2 Invalid Role Assignment **4**
3. Special Requirements **5**
3.1 Admin Role Protection 5
3.2 Performance 5
3.3 Bulk Role Management 5
3.4 Browser Compatibility 5
3.5 Data Privacy 5
4. Preconditions **5**
4.1 Admin Authentication 5
4.2 Admin Permissions 5
4.3 System Accessibility 5
4.4 User Availability 5
5. Postconditions **5**
5.1 Role Update 5
5.2 User Notification 5
5.3 Admin Role Assurance 5
5.4 Action Confirmation 5
6. Extension Points **6**
6.1 Invite New User 6
7. Low Fidelity Prototype For Use Case **6**
7.1 Prompt Used for AI Tool 6
7.2 Generated Interface 6

©Error! No property name supplied., 2025

Use-Case Specification: Manage User Roles

1. Use-Case Name: Manage User Roles

1.1 Brief Description

This use case describes the process by which an Admin manages user roles (e.g., Reader, Writer, Editor) on the Digital Newspaper Website. The Admin uses the content management system (CMS) to assign, modify, or revoke user roles to control access and permissions.

2. Flow of Events

2.1 Basic Flow

This use case starts when the Admin logs into the Digital Newspaper Website's CMS and accesses the user management dashboard.

1. The Admin navigates to the "User Management" section in the CMS.
2. The system displays a list of registered users, including their names, email addresses, current roles (e.g., Reader, Writer, Editor, Admin), and account status (e.g., Active, Inactive).
3. The Admin selects a user from the list to manage their role.
4. The system displays the user's profile, including their current role and available actions (e.g., Assign Role, Modify Role, Revoke Role).
5. The Admin selects an action:
    a. **Assign Role**: The Admin assigns a new role (e.g., Writer, Editor) to a user with no role or an additional role.
    b. **Modify Role**: The Admin changes the user's existing role (e.g., from Writer to Editor).
    c. **Revoke Role**: The Admin removes a specific role from the user, reverting them to a default role (e.g., Reader).
6. The system validates the action (e.g., ensures at least one Admin remains, checks for role conflicts).
7. The Admin confirms the action.
8. The system updates the user's role in the CMS and sends a notification to the user via email or CMS notification, informing them of the role change.
9. The system confirms to the Admin that the user's role has been successfully updated.

2.2 Alternative Flows

2.2.1 No Users Available

- If no users are registered in the CMS, the system displays a message.
- The Admin may choose to invite a new user by entering their email address, triggering an invitation process.
- The use case ends.

2.2.2 Invalid Role Assignment

- At Basic Flow step 6, if the Admin attempts an invalid action (e.g., revoking the last Admin role or assigning conflicting roles), the system displays an error message (e.g., "At least one Admin role must remain" or "User cannot have both Writer and Editor roles").
- The Admin corrects the action and resubmits (returns to Basic Flow step 5).

3. Special Requirements

3.1 Admin Role Protection

The CMS must ensure that at least one user retains the Admin role at all times.

3.2 Performance

The system must load the user management dashboard within 3 seconds under normal server load.

3.3 Bulk Role Management

The system must support bulk role management for up to 100 users without performance degradation.

3.4 Browser Compatibility

The system must ensure compatibility with common browsers (e.g., Chrome, Firefox, Safari) for the CMS interface.

3.5 Data Privacy

The system must comply with GDPR for handling personal user data during role management and notifications.

4. Preconditions

4.1 Admin Authentication

The Admin must be registered and logged into the CMS with valid credentials.

4.2 Admin Permissions

The Admin must have the role of "Admin" with permissions to manage user roles.

4.3 System Accessibility

The CMS must be accessible and operational.

4.4 User Availability

At least one user must be registered in the CMS.

5. Postconditions

5.1 Role Update

The selected user’s role is updated (assigned, modified, or revoked) in the CMS.

5.2 User Notification

The user is notified of the role change via email or CMS notification.

5.3 Admin Role Assurance

The system ensures that at least one Admin role remains active.

5.4 Action Confirmation

The Admin receives confirmation of the role management action.

6. Extension Points

6.1 Invite New User

- Location: Basic Flow, step 2.
- The Admin can invite a new user by entering their email address and assigning an initial role. The system sends an invitation link to the user, allowing them to register and activate their account with the assigned role.

7. Low Fidelity Prototype For Use Case

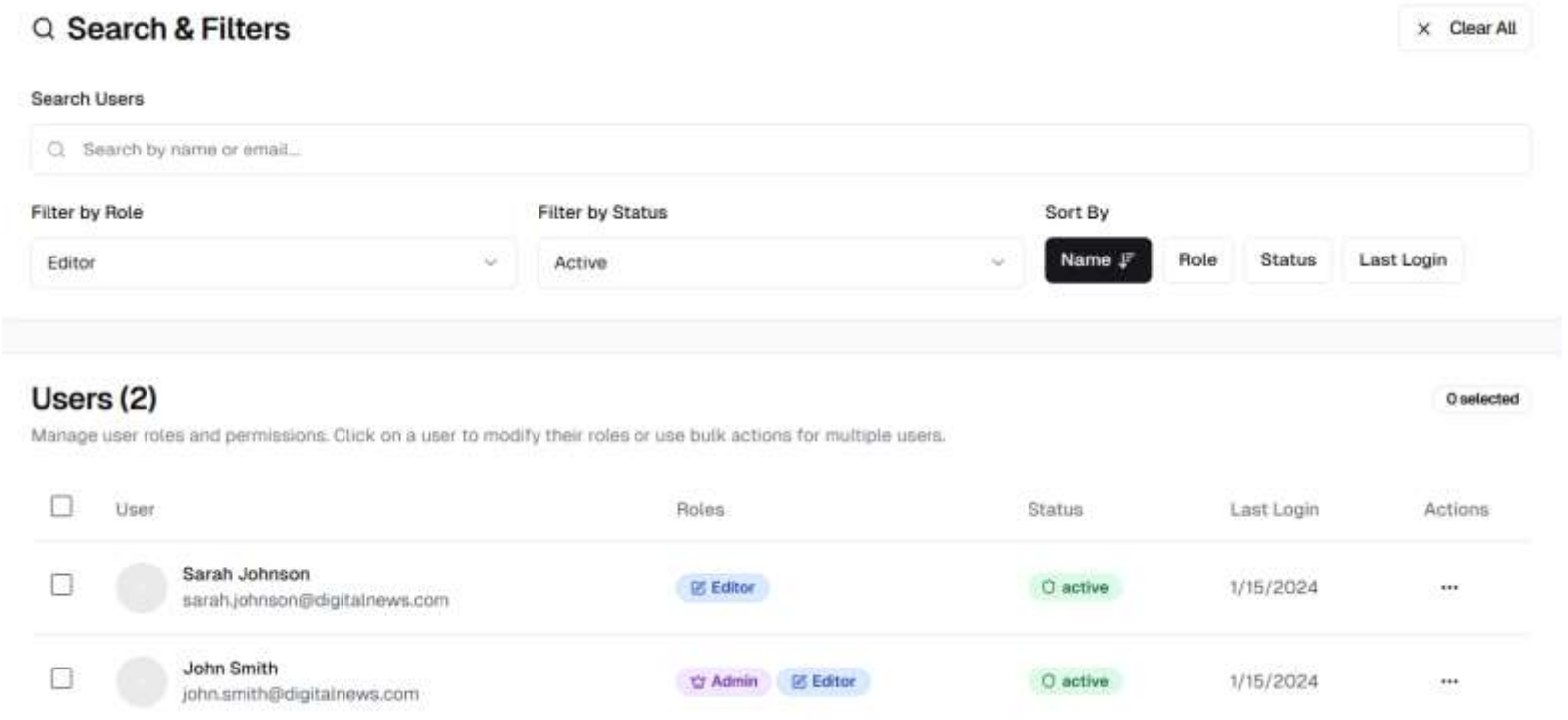
7.1 Prompt Used for AI Tool

- Design an Admin Interface for a CMS of a Digital Newspaper Website, supporting the "Manage User Roles" use case.
- Functional Requirements: [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*
- Optional: Search bar and filters (by role/status). / Sorting by name, role, or status. / Tags for user roles with color coding.

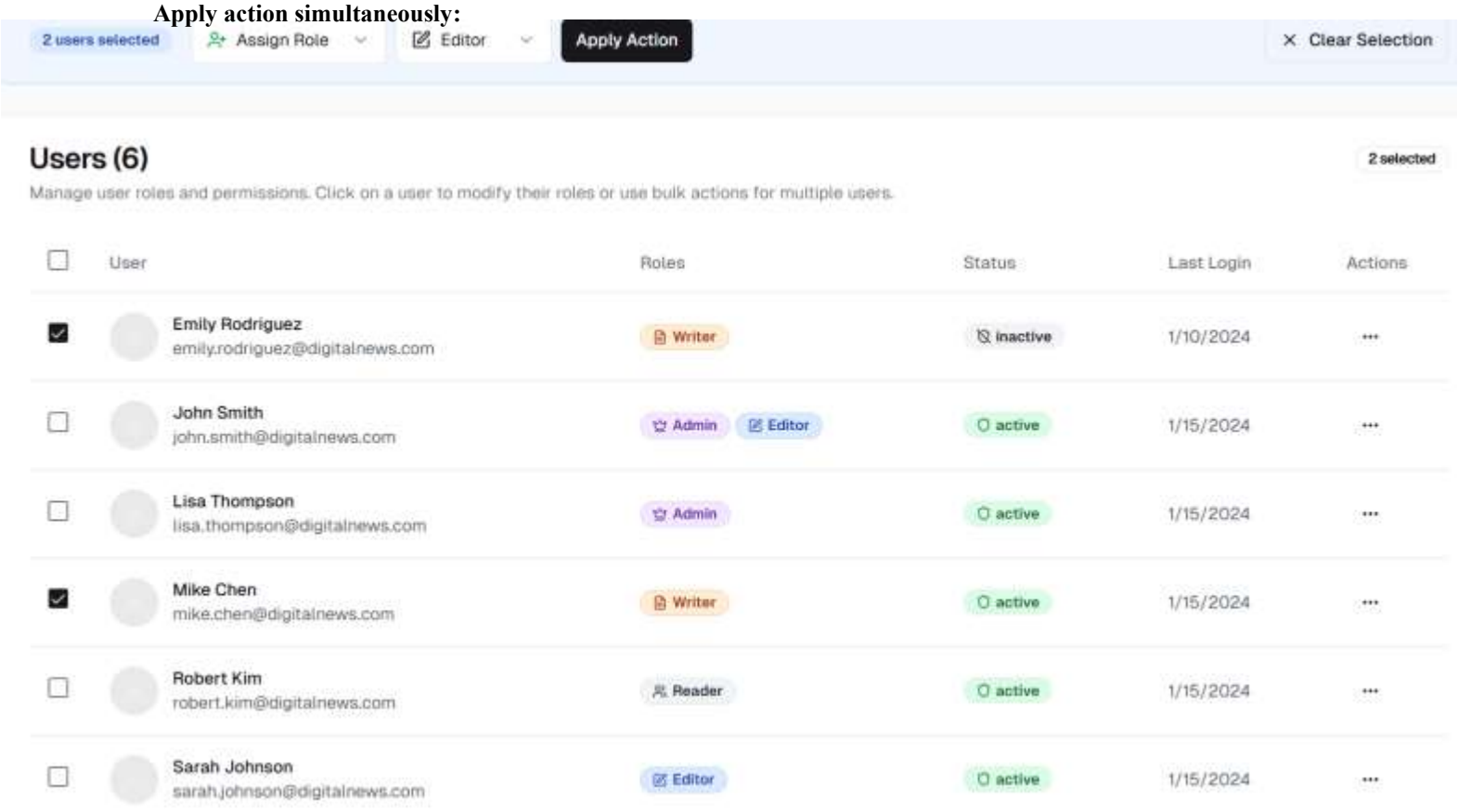
7.2 Generated Interface

Overall Data:

Filter User:



Apply action simultaneously:



404 Not Found

Digital Newspaper Website
Use-Case Specification: Manage Articles

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Complete the document	Trần Hữu Khang

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Manage Articles **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows **4**
2.2.1 No Articles Pending Review 4
2.2.2 Bulk Actions **4**
3. Special Requirements **5**
3.1 Rich Text Editing 5
3.2 Performance 5
3.3 Bulk Actions 5
3.4 Browser Compatibility 5
3.5 Data Privacy 5
4. Preconditions **5**
4.1 Editor Authentication 5
4.2 Editor Permissions 5
4.3 System Accessibility 5
4.4 Article Availability 5
5. Postconditions **5**
5.1 Article Status Update 5
5.2 Writer Notification 5
5.3 Publication Queue 5
5.4 Action Confirmation 5
6. Extension Points **6**
6.1 Assign Reviewer 6
7. Low Fidelity Prototype For Use Case **6**
7.1 Prompt Used for AI Tool 6
7.2 Generated Interface 6

©Error! No property name supplied., 2025

Use-Case Specification: Manage Articles

1. Use-Case Name: Manage Articles

1.1 Brief Description

This use case describes the process by which an Editor manages articles submitted by Writers on the Digital Newspaper Website. The Editor reviews, edits, approves, rejects, or schedules articles for publication using the content management system (CMS).

2. Flow of Events

2.1 Basic Flow

This use case starts when the Editor logs into the Digital Newspaper Website's CMS and accesses the article management dashboard.

1. The Editor navigates to the "Article Management" section in the CMS.
2. The system displays a list of articles with their status, including each article's title, Writer's name, submission date, and category.
3. The Editor selects an article marked "Pending Review" to review its content.
4. The system displays the article's full content, including title, body, category, tags, and any attached media (e.g., images, videos).
5. The Editor reviews the article and chooses one of the following actions.
    a. **Edit**: The Editor modifies the article's title, body, tags, or media, then saves changes.
    b. **Approve**: The Editor marks the article as approved for publication.
    c. **Reject**: The Editor rejects the article and provides feedback.
    d. **Schedule**: The Editor sets a publication date and time for an approved article.
6. If the Editor selects "Edit," the system saves the changes and updates the article status to "Draft" or "Pending Review" (based on the Editor's choice).
7. If the Editor selects "Approve," the system marks the article as "Approved" and makes it available for publication or scheduling.
8. If the Editor selects "Reject," the system notifies the Writer via email or CMS notification with the provided feedback.
9. If the Editor selects "Schedule," the system saves the publication date and time, and the article is queued for automatic publishing.
10. The system confirms the action to the Editor.

2.2 Alternative Flows

2.2.1 No Articles Pending Review

- No Articles Pending Review
- If no articles are marked "Pending Review," the system displays a message (e.g., "No articles awaiting review") and shows a list of articles in other statuses (e.g., Draft, Approved, Rejected).
- The Editor may select an article in another status to review or edit (returns to Basic Flow step 4).

2.2.2 Bulk Actions

- At Basic Flow step 2, the Editor selects multiple articles and applies a bulk action (e.g., approve, reject, or delete).
- The system prompts the Editor to confirm the action.
- Upon confirmation, the system applies the action to all selected articles and notifies the respective Writers (if applicable).

©Error! No property name supplied., 2025

- The flow continues from Basic Flow step 10.

3. Special Requirements

3.1 Rich Text Editing

The CMS must support rich text editing for article content (e.g., bold, italics, hyperlinks, bullet lists).

3.2 Performance

The system must load the article management dashboard within 3 seconds under normal server load.

3.3 Bulk Actions

The system must support bulk actions for at least 50 articles at once without performance degradation.

3.4 Browser Compatibility

The system must ensure compatibility with common browsers (e.g., Chrome, Firefox, Safari) for the CMS interface.

3.5 Data Privacy

The system must comply with GDPR for handling any personal data included in articles or Writer notifications.

4. Preconditions

4.1 Editor Authentication

The Editor must be registered and logged into the CMS with valid credentials.

4.2 Editor Permissions

The Editor must have the role of "Editor" with permissions to manage articles.

4.3 System Accessibility

The CMS must be accessible and operational.

4.4 Article Availability

At least one article must be available in the CMS for management.

5. Postconditions

5.1 Article Status Update

The article’s status is updated to Draft, Approved, Rejected, or Scheduled in the CMS.

5.2 Writer Notification

The Writer is notified if the article is rejected or requires further edits.

5.3 Publication Queue

If scheduled, the article is queued for publication at the specified date and time.

5.4 Action Confirmation

The Editor receives confirmation of the action taken.

6. Extension Points

6.1 Assign Reviewer

- Location: Basic Flow, step 3.
- The Editor can assign another Editor or Reviewer to review the article by selecting their name or email from a list of registered CMS users. The system notifies the assigned Reviewer and grants them access to the article.

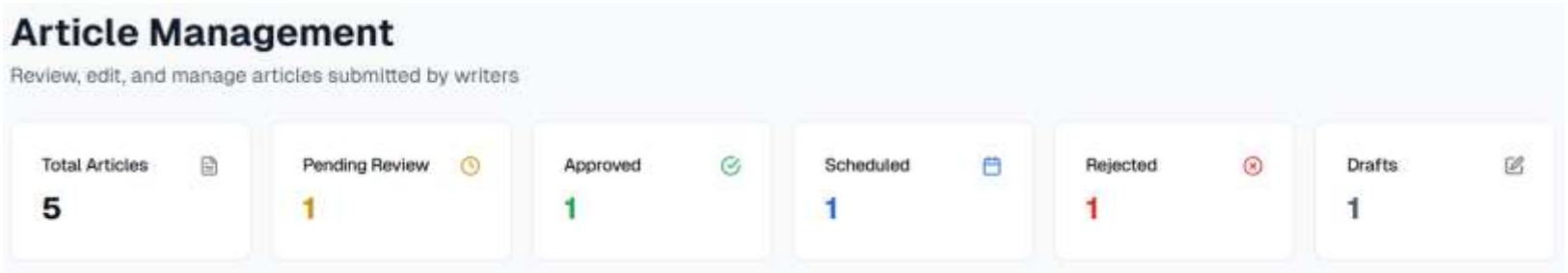
7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

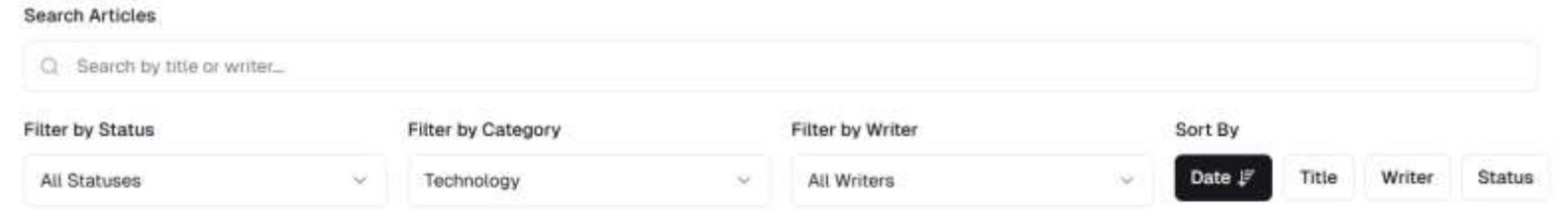
- Design a CMS interface for Editors on a Digital Newspaper Website that supports the "Manage Articles" use case.
- Functional Requirements: [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*
- Optional: Search bar or filters by status or category. /Tags for article status (color-coded). / Responsive layout for tablet and desktop.

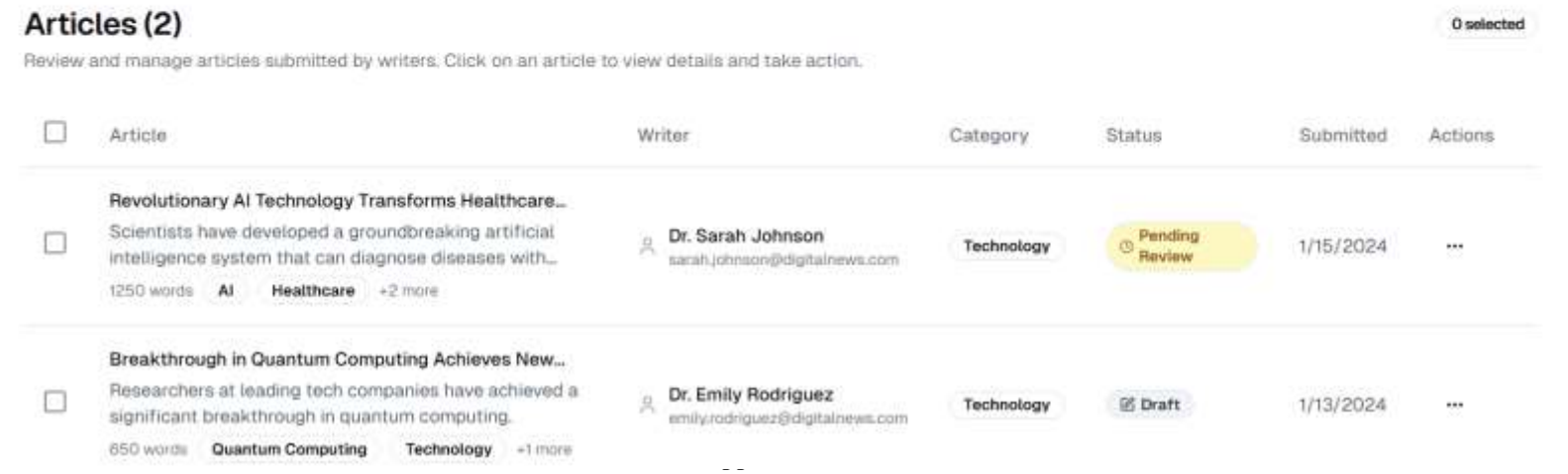
7.2 Generated Interface

Overall Data:

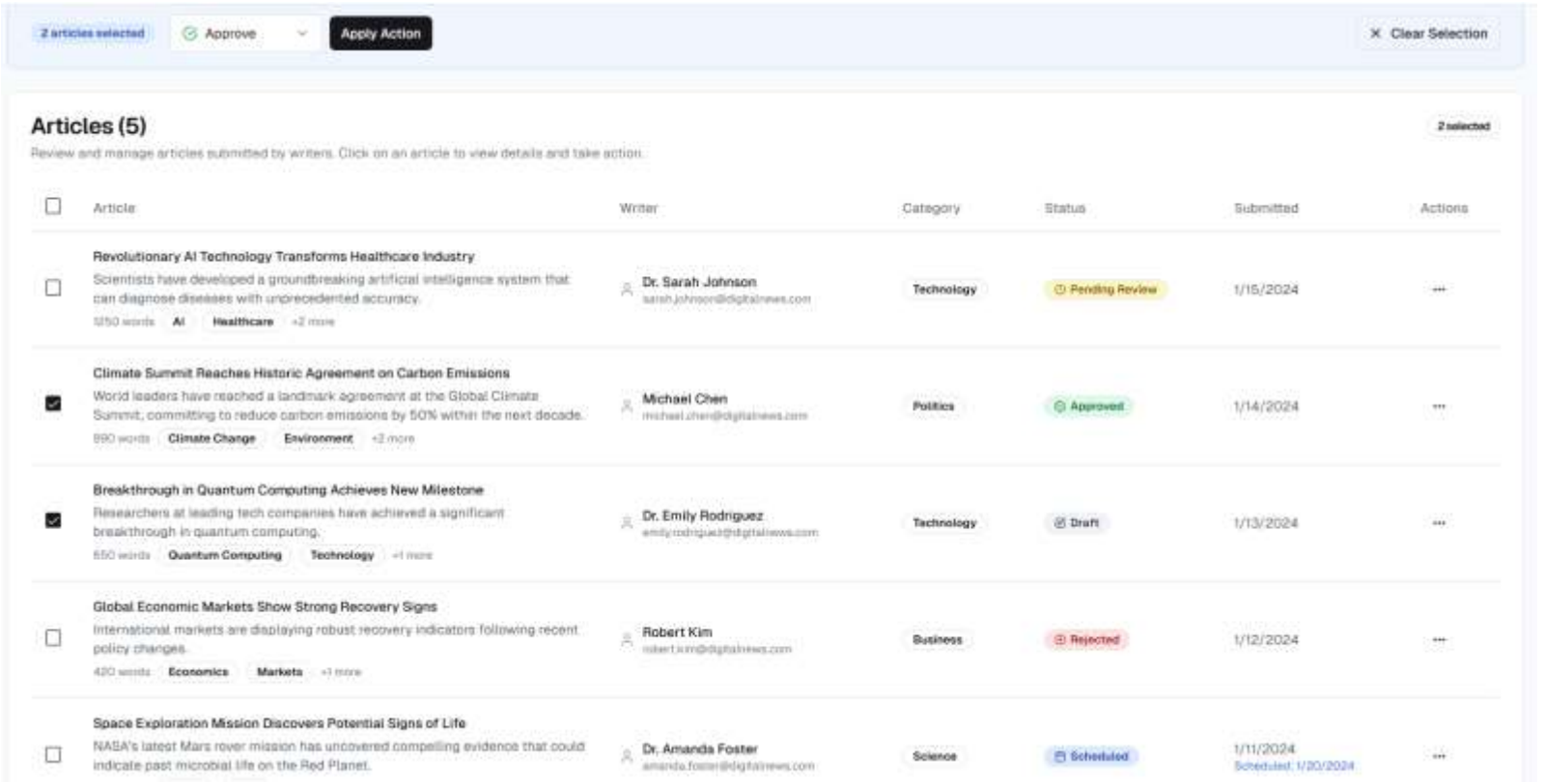


Filter User:





Apply actions simultaneously:



404 Not Found

Digital Newspaper Website
Use-Case Specification: Create Article

Version 1.0

©404 Not Found, 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Complete the document	Trần Hữu Khang

©404 Not Found, 2025

Table of Contents

1. Use-Case Name: Create Article **4**
1.1 Brief Description 4
2. Flow of Events **4**
2.1 Basic Flow 4
2.2 Alternative Flows **4**
2.2.1 Invalid Input 4
2.2.2 Save as Draft **4**
3. Special Requirements **5**
3.1 Rich Text Formatting 5
3.2 Browser Compatibility 5
3.3 Media Upload Handling 5
3.4 Performance 5
3.5 Data Privacy 5
4. Preconditions **5**
4.1 Writer Authentication 5
4.2 Writer Permissions 5
4.3 System Accessibility 5
5. Postconditions **5**
5.1 Article Storage 5
5.2 Editor Notification 5
5.3 Writer Confirmation 5
6. Extension Points **5**
6.1 Add Co-Author 5
7. Low Fidelity Prototype For Use Case **6**
7.1 Prompt Used for AI Tool 6
7.2 Generated Interface 6

©404 Not Found, 2025

Use-Case Specification: Create Article

1. Use-Case Name: Create Article

1.1 Brief Description

This use case describes the process by which a Writer creates and submits an article for publication on the Digital Newspaper Website. The Writer uses the content management system (CMS) to input article details, including title, content, and media, and submits the article for editorial review.

2. Flow of Events

2.1 Basic Flow

This use case starts when the actor does something. An actor always initiates use cases. The use case starts when the Writer logs into the Digital Newspaper Website's CMS and selects the option to create a new article.

1. The Writer navigates to the "Create Article" section in the CMS dashboard.
2. The system displays a form with fields for the article's title, body content, category, tags, and optional media uploads (e.g., images or videos).
3. The Writer enters the article title, body content, selects a category (e.g., News, Opinion, Sports), and adds relevant tags (e.g., politics, technology).
4. The Writer optionally uploads media files to accompany the article.
5. The system validates the input to ensure required fields (title, body content, category) are filled and meet formatting requirements.
6. The Writer selects the "Preview" option to review the article's appearance.
7. The system renders a preview of the article, showing the formatted content and media as they would appear on the website.
8. The Writer either edits the article (returning to step 3) or selects "Submit for Review."
9. The system saves the article as a draft and notifies the Editor via email or CMS notification that a new article is pending review.
10. The system confirms to the Writer that the article has been successfully su*bmitted for review.*

2.2 Alternative Flows

2.2.1 Invalid Input

- If the Writer submits the article with incomplete or invalid data, the system displays an error message specifying the issue.
- The Writer corrects the errors and resubmits the form (returns to Basic Flow step 3).

2.2.2 Save as Draft

- At any point after entering some data, the Writer selects "Save as Draft" instead of submitting.
- The system saves the current state of the article in the CMS and allows the Writer to return to it later.
- The use case ends, and no notification is sent to the Editor.

3. Special Requirements

3.1 Rich Text Formatting

The CMS must support rich text formatting (e.g., bold, italics, hyperlinks, bullet lists) for the article body.

©404 Not Found, 2025

3.2 Browser Compatibility

The system must ensure compatibility with common browsers (e.g., Chrome, Firefox, Safari) for the article creation interface.

3.3 Media Upload Handling

The system must handle media uploads up to 10 MB per file and support JPEG, PNG, and MP4 formats.

3.4 Performance

The article submission process must be completed within 5 seconds under normal server load.

3.5 Data Privacy

The system must comply with GDPR for handling any personal data included in articles or media.

4. Preconditions

4.1 Writer Authentication

The Writer must be registered and logged into the CMS with valid credentials.

4.2 Writer Permissions

The Writer must have the role of "Writer" with permissions to create articles.

4.3 System Accessibility

The CMS must be accessible and operational.

5. Postconditions

5.1 Article Storage

The article is saved in the CMS as a draft or submitted for editorial review.

5.2 Editor Notification

If submitted, the Editor is notified of the pending article.

5.3 Writer Confirmation

The Writer receives confirmation of successful submission or draft save.

6. Extension Points

6.1 Add Co-Author

- Location: Basic Flow, step 3.
- The Writer can add a co-author by entering their registered email or selecting from a list of Writers in the CMS. The system notifies the co-author and grants them edit access to the article.

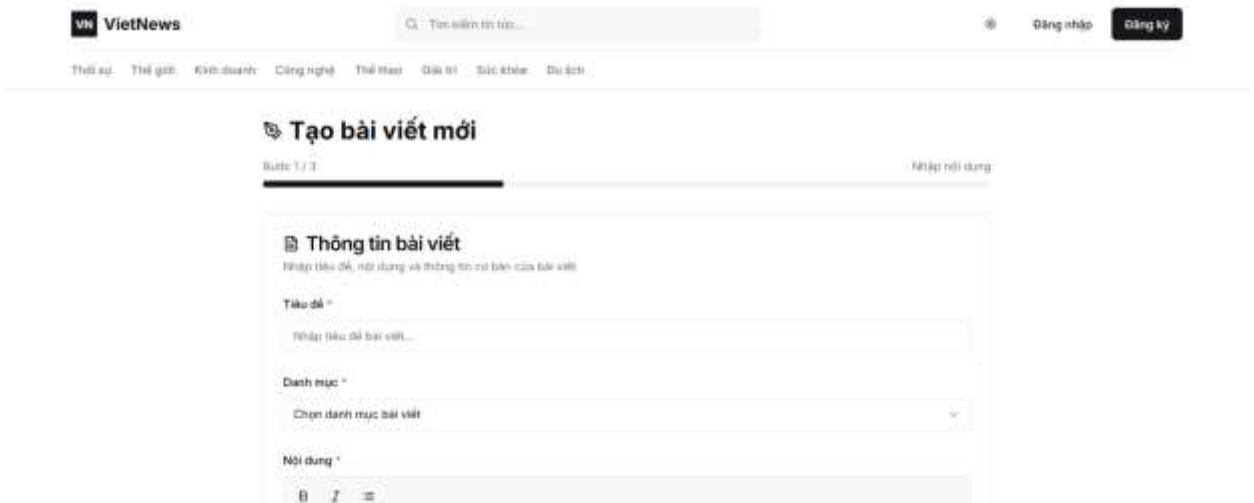
7. Low Fidelity Prototype For Use Case

7.1 Prompt Used for AI Tool

- Create a test page at /test/create-article for my Digital Newspaper Website that simulates the full 'Create Article' use case.
- Functional Requirements: [2. Flow of Event …… 6.Extension Points] (*Paste the use case specification from section 2 to 6 here)*

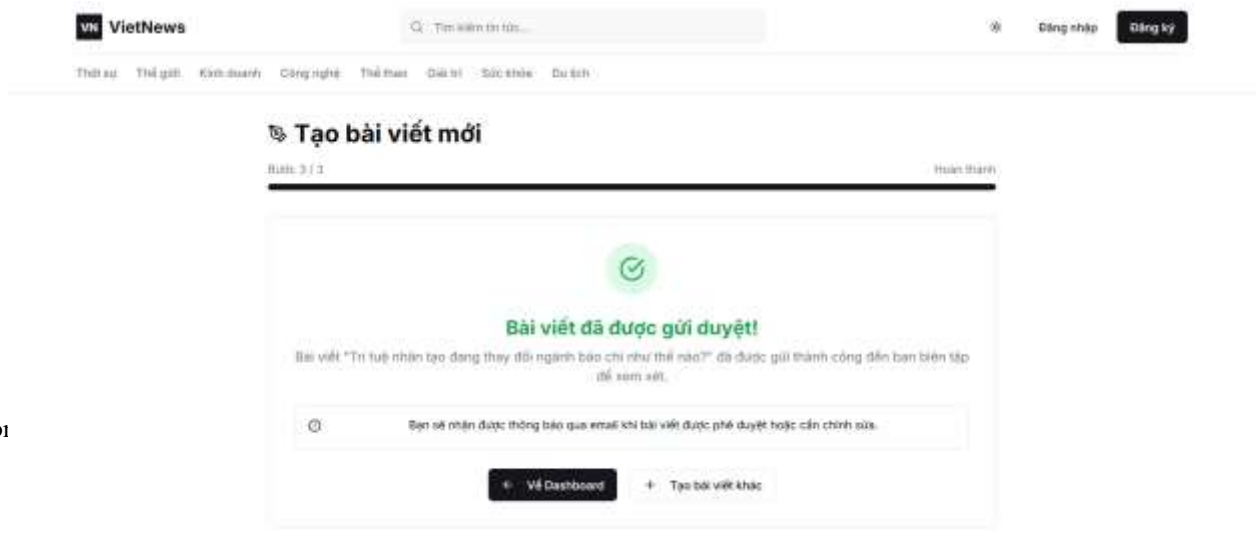
©404 Not Found, 2025

7.2 Generated Interface



VietNews
Đăng nhập
Đăng ký
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Sức khỏe
Du lịch
Tạo bài viết mới
Xem trước bài viết
Công nghệ
Trí tuệ nhân tạo đang thay đổi ngành báo chí như thế nào?
Nguyễn Văn A

Digital Newspaper Website	Version: 1.0
Use-Case Specification: Create Article	Date: 31/7/2025
create-article	



Co1

404 Not Found

Digital Newspaper Website
Use-Case Specification: Log Out

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
1/7/2025	1.0	Complete the document	Lê Đình Minh Quân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Log Out 5

1.1 Brief Description 5

2. Flow of Events 5

2.1 Basic Flow 5

2.2 Alternative Flows 6

2.2.1 Network Failure During Logout 6

2.2.2 Session Already Expired

2.2.3 Global (All Devices) Logout Selected

2.2.4 Unsaved Changes Warning 6

2.2.5 Forced Security Logout 7

2.2.6 Accessibility / Keyboard Invocation 7

3. Special Requirements **Error! No bookmark name given.**

4. Preconditions 7

5. Postconditions **7**

6. Extension Points **8**

7. Low Fidelity Prototype For Use Case **9**

©Error! No property name supplied., 2025

Use-Case Specification: Log Out

1. Use-Case Name: Log Out

1.1 Brief Description

The *Log Out* use case describes how an authenticated User (Writer, Editor, Admin, or any signed-in role) securely ends an active session in the Digital Newspaper Website. The system invalidates session tokens, clears sensitive client-side data, and returns the User to a public (unauthenticated) state.

2. Flow of Events

2.1 Basic Flow

Actor: User (any authenticated role)
Precondition: User is currently logged in.
Trigger: User chooses to log out.

1. User clicks **Log Out** (e.g., in header menu, profile dropdown, or side navigation).
2. The system optionally shows a brief *"Signing out…"* indicator (omit if the operation completes almost instantly).
3. System invalidates the active server session (destroys server session object and/or revokes refresh token).
4. The system instructs clients to clear all authentication artifacts: access token, refresh token (if locally stored), cached role/permission data, user profile cache.
5. System clears or overwrites any *"remember me"* cookie if the user did **not** opt into persistent login; if persistent login was enabled, only the minimal secure refresh mechanism (no active access token) remains.
6. System records an audit log event with details (userId, timestamp, IP, userAgent, action=LOG_OUT, scope=single).
7. System redirects the user to the public landing page (or a dedicated "You have signed out" page).
8. System displays a confirmation message (e.g., "You have successfully logged out.") or transient toast.
9. User may continue browsing public content or initiate a new login (use case ends).

2.2 Alternative Flows

2.2.1 Network Failure During Logout

Point of deviation: After Step 1, before Step 7.

1. *Session invalidation request times out or network error occurs.*
2. *Client prompts: "Network issue — unable to complete log out. Retry?"*
3. *If the user selects **Retry**, resume at Basic Flow Step 2.*
4. *If the user selects **Cancel**, the session remains active; the user stays in an authenticated view with a clear status message.*

2.2.2 Session Already Expired

Point of deviation: Step 3.

1. *The system detects the token already expired/invalid.*
2. *The system still clears local auth artifacts (equivalent to Steps 4–5).*
3. *System logs audit event with reason=ALREADY_INVALID.*
4. *Flow rejoins Basic Flow at Step 7.*

2.2.3 Global (All Devices) Logout Selected

Point of deviation: Step 3 (user chose "Log out all devices").

1. *The system invalidates all active sessions across devices.*
2. *System logs audit event with scope=GLOBAL.*
3. *Flow rejoins Basic Flow at Step 7.*

2.2.4 Unsaved Changes Warning

Point of deviation: Step 1.

1. *The user has unsaved draft edits.*
2. *System prompts: "You have unsaved changes. Log out anyway?"*
3. *If **Cancel**, use case ends (no logout).*
4. *If **Log Out Anyway**, continue at Basic Flow Step 2.*

2.2.5 Forced Security Logout

Point of deviation: Before Step 1 (system-initiated).

1. *Security rules (password reset, role revoked, suspicious activity) forces termination.*

2. *On the next request, the system invalidates the session and redirects to login with message: "Your session has ended. Please sign in again."*

3. *Use case terminates (no user-initiated trigger).*

2.2.6 Accessibility / Keyboard Invocation

Point of deviation: Step 1.

1. *The user activates the logout entirely via keyboard / assistive tech.*

2. *Flow proceeds equivalently from Basic Flow Step 2.*

3. Special Requirements

- **Security:** Revoke server session & refresh tokens; clear client tokens and cached identity data.
- **CSRF Protection:** Logout endpoint protected (anti-CSRF token or SameSite=strict, POST/DELETE).
- **Privacy:** No user identity/role info remains client-side post Step 4.
- **Performance:** `Perceived logout ≤ 2 s; if > 1 s show progress indicator.`
- **Auditability:** Log every logout (normal, expired, forced, global) with scope & reason.
- **Resilience:** Network failure path (2.2.1) offers retry to avoid ambiguous state.
- **Accessibility:** Control is focusable, labeled (e.g., `aria-label="Log Out"`), keyboard operable.
- **Multi-Device:** `Global logout (2.2.3) completes ≤ 5 s for ≤ 20 active sessions.`
- **Localization:** All messages localizable.
- **Compliance:** Conforms to internal security policy & OWASP ASVS session management.

4. Preconditions

- Valid authenticated sessions exist.
- Auth / session infrastructure operational.
- Logout control visible or discoverable (accessible).
- (If global logout) User role permitted to invalidate all sessions.

5. Postconditions

- **Success:** Session (and optionally all sessions) invalidated; client auth data cleared; user in public state.

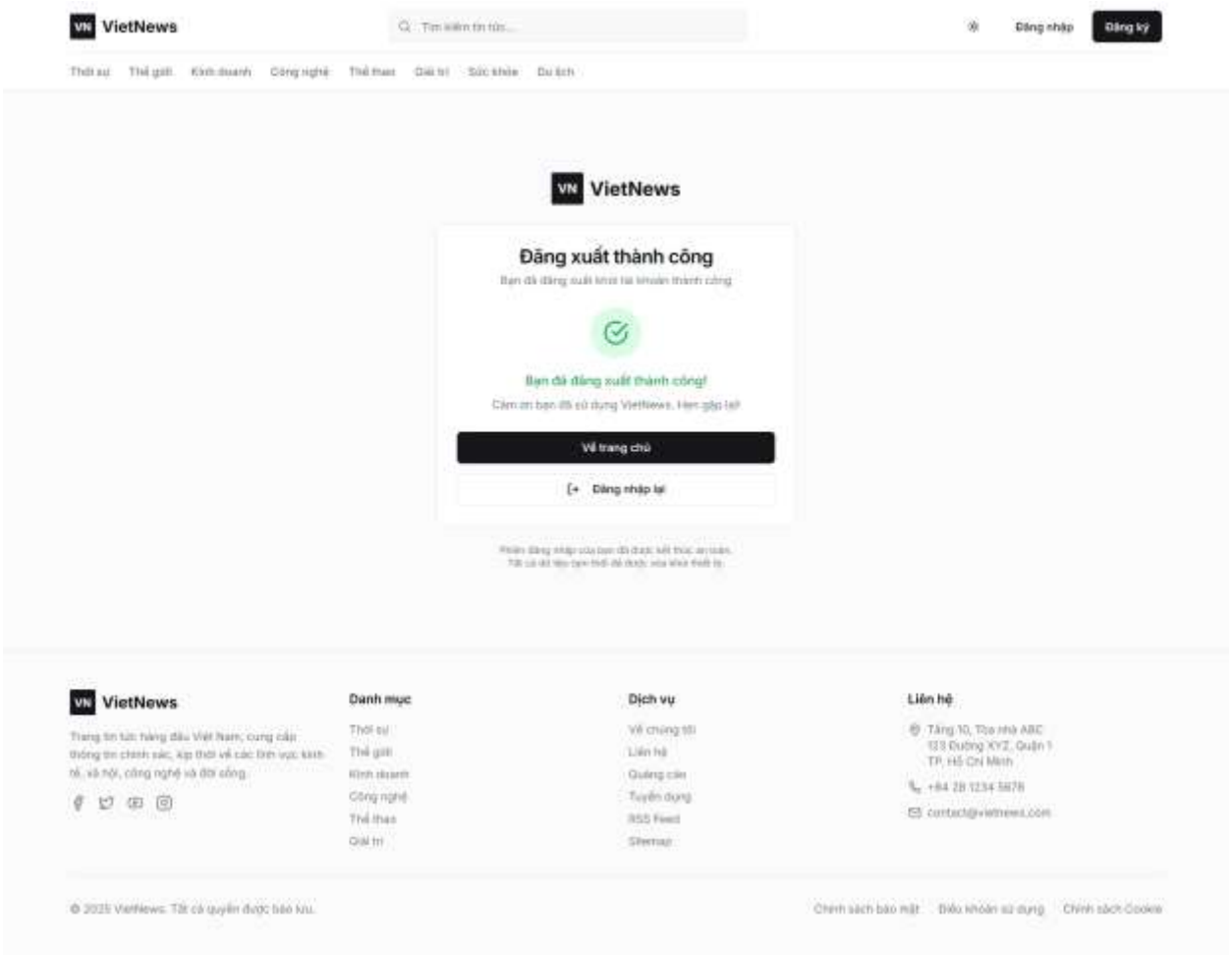
- **Partial Retry State:** On network failure + user cancel, session remains valid and user informed.
- **Forced Logout:** Session invalidated by system; user redirected to login with message.

6. Extension Points

- **Survey Prompt:** After Basic Flow Step 7 show optional feedback modal.
- **Single Sign-On Propagation:** Extend Step 3 to notify Identity Provider.
- **Idle Timeout Warning:** Precedes 2.2.5 offering “Stay signed in”.
- **Clear Client Drafts:** Hook in Step 4 to purge local unsaved drafts.

7. Low Fidelity Prototype For Use Case

Prompt for AI tools: Create 3 new React/Next.js pages for my existing Digital Newspaper Website. Use the same visual style, layout components (e.g., header, footer, sidebar), fonts, and responsive breakpoints as the rest of the site. Each page should simulate the following UI use cases for testing:Log Out Page (/test/logout), Simulate a logout confirmation view. Show a brief 'Signing out...' indicator. Then display: “You have successfully logged out.” Include a button to return to the homepage. Match existing site’s colors, font, and spacing. All pages should be self-contained and follow the overall site design using existing components (e.g., Layout, Button, Card, etc.)."


VietNews
Đăng nhập
Đăng ký
VietNews
Đăng xuất thành công
Bạn đã đăng xuất thành công!
Về trang chủ
Đăng nhập lại
VietNews
Danh mục
Dịch vụ
Liên hệ

404 Not Found

Digital Newspaper Website
Use-Case Specification: Verify MFA code

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Complete the document	Lê Đình Minh Quân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name: Verify MFA Code 4

1.1 Brief Description 4

2. Flow of Events 4

2.1 Basic Flow 4

2.2 Alternative Flows 5

2.2.1 Invalid Code (Single Failure) 5

2.2.2 Code Expired / Time Window Drift 5

2.2.3 Maximum Attempts Reached (Temporary Lockout) 5

2.2.4 Factor Not Available (User Lost Device) 5

2.2.5 Using a Backup Recovery Code 6

2.2.6 Push Notification Factor (Approval Flow) 6

2.2.7 Hardware Security Key (WebAuthn) Factor 6

2.2.8 Fallback to Primary Re-Authentication (Session Stale) 6

2.2.9 Forced Step-Up During Active Session 7

2.2.10 Suspicious Attempt (Risk Engine Intervention) 7

2.2.11 Session Continuation After Partial Failures 7

2.2.12 User Cancels Verification 7

7

3. Special Requirements 8

4. Preconditions **Error! No bookmark name given.**

5. Postconditions 8

6. Extension Points 9

7. Low Fidelity Prototype For Use Case 9

©Error! No property name supplied., 2025

Use-Case Specification: Verify MFA Code

1. Use-Case Name: Verify MFA Code

1.1 Brief Description

The *Verify MFA Code* use case describes how a partially authenticated User (after successful primary credential entry) completes multi-factor authentication (MFA) by submitting a one-time code (OTP) delivered via an approved second factor (e.g., authenticator app TOTP, SMS, email fallback, hardware key challenge, or push approval). The system validates the code or challenge response, establishes a fully authenticated session with appropriate assurance level, and enforces retry, lockout, and security policies.

2. Flow of Events

2.1 Basic Flow

Actor: User (partially authenticated; primary login successful)
Precondition: Primary credentials accepted; system requires MFA; a valid MFA factor is enrolled and a current one-time code (or challenge) is available to the user.
Trigger: System presents MFA verification prompt.

1. The system displays the **MFA verification screen** indicating the active factor (e.g., “Enter the 6-digit code from your authenticator app”).
2. The user enters the MFA code (or responds to the presented factor, e.g., types TOTP, approves push, taps hardware key).
3. System normalizes and validates input format (trims spaces, rejects non-digits if TOTP).
4. The system verifies the code/challenge against the factor source (e.g., TOTP time window, push approval status, WebAuthn signature).
5. System checks **attempt counters** (e.g., max consecutive failures) and **token freshness / time drift**.
6. On success, the system elevates the session from *partial* to *fully authenticated*, sets MFA assurance attributes (e.g., mfaLevel = strong, authn_time = timestamp).
7. System clears transient MFA state (e.g., pending challenge IDs, temporary cookies).
8. System redirects users to the originally requested protected resource (or default post-login landing page).
9. System logs an audit event { userId, factorType, outcome = SUCCESS, ip, userAgent, timestamp}.
10. The user continues with full privileges (use case ends).

2.2 Alternative Flows

2.2.1 Invalid Code (Single Failure)

Point of deviation: Step 4.

1. *Code does not match expected value (wrong digits, incorrect WebAuthn signature, push denied).*
2. *System increments failure counter and logs failure event.*
3. *System informs user: "Invalid code. Please try again." and re-prompts (remaining attempts indicated).*
4. *Flow returns to Basic Flow Step 2 (unless lockout reached).*

2.2.2 Code Expired / Time Window Drift

Point of deviation: Step 4.

1. *Code structure correct but outside valid time window (TOTP drift, expired SMS).*
2. *System shows: "Code expired. Request or generate a new code."*
3. *(If drift is small and policy allows) system suggests user resync device time.*
4. *Flow returns to Basic Flow Step 1 (or Step 2 after new code entry).*

2.2.3 Maximum Attempts Reached (Temporary Lockout)

Point of deviation: After repeated executions of 2.2.1.

1. *Failure counter hits configured threshold (e.g., 5 attempts).*
2. *The system locks further attempts for a cooldown period (e.g., 5 minutes) and logs outcome {outcome = LOCKOUT}*
3. *The system displays a lockout message with remaining wait time.*
4. *Use case ends (user may retry after cooldown → re-enter from Basic Flow Step 1).*

2.2.4 Factor Not Available (User Lost Device)

Point of deviation: Step 1 (user cannot supply code).

1. *The user selects the "I can't access my authenticator" link.*
2. *The system offers enrolled **backup factors** (e.g., backup codes, SMS, email, hardware key) per policy hierarchy.*

3. *The user selects a backup factor → system sends / prompts for that factor's challenge.*

4. *Flow resumes at Basic Flow Step 2 for the new factor.*

5. *System marks event with **factorFallback = true**.*

2.2.5 Using a Backup Recovery Code

Point of deviation: Step 2.

1. *The user enters one of their one-time backup recovery codes instead of standard factor code.*

2. *System matches hash; if valid, marks code as **consumed** (cannot reuse).*

3. *The system proceeds at Basic Flow Step 6 (successful elevation) and logs **factorType = RECOVERY_CODE.***

4. *The system warns the user if remaining recovery codes are low (e.g., ≤2).*

2.2.6 Push Notification Factor (Approval Flow)

Point of deviation: Step 2 (factor = push).

1. *Instead of entering digits, the user receives a push prompt on the device and taps **Approve**.*

2. *System polls or gets callback; on approval continues at Basic Flow Step 6.*

3. *If the user taps **Deny** (or times out), treat it as failure → go to 2.2.1 logic.*

2.2.7 Hardware Security Key (WebAuthn) Factor

Point of deviation: Step 2 (factor = hardware key).

1. *System initiates WebAuthn challenge in browser; user taps key or provides biometric.*

2. *System verifies signature (credential ID, origin, counter).*

3. *On success resume at Basic Flow Step 6; on failure resume at 2.2.1.*

2.2.8 Fallback to Primary Re-Authentication (Session Stale)

Point of deviation: Step 5.

1. *The system detects partial sessions have exceeded allowed pre-MFA age.*

2. *System discards partial session and redirects user back to primary login (restarts overall authentication).*

3. *After successful primary login again, flow returns to Basic Flow Step 1.*

2.2.9 Forced Step-Up During Active Session

Point of deviation: External protected action requests higher assurance while the user is already logged in at a lower level.

1. *The user tries a sensitive operation (e.g., change password, view PII).*

2. *System interrupts with MFA verification (step-up).*

3. *On success proceed to protected action (outside this use case's scope).*

4. *On failure follow alternative flows 2.2.1–2.2.3; if abandoned, action is canceled.*

2.2.10 Suspicious Attempt (Risk Engine Intervention)

Point of deviation: Step 4.

1. *Risk engine flags anomalies (geo-velocity, device mismatch).*

2. *The system requires additional factors (step-up inside step-up) or denies attempt.*

3. *If additional factor pass, resume at Basic Flow Step 6; if denied, record **outcome = RISK_DENY** and end use case.*

2.2.11 Session Continuation After Partial Failures

Point of deviation: Repeated Step 2 with intermittent failures below lockout threshold.

1. *The user fails once or twice but later provides a correct code within allowed attempts/time.*

2. *The system resets consecutive failure counters after success.*

3. *Flow rejoins Basic Flow Step 6.*

2.2.12 User Cancels Verification

Point of deviation: Step 2.

1. *User clicks **Cancel** / closes dialog.*

2. *System discards partial session or returns to primary login screen with message "MFA required to continue."*

©Error! No property name supplied., 2025

3. *Use case ends (no elevation).*

3. Special Requirements

- **Security:** Support multiple factors (TOTP, WebAuthn, push, backup codes) per enrolled configuration; verification must resist replay (nonce/time-based).
- **Time Drift Handling:** Accept TOTP codes within configurable ± time window (e.g., ±30s) while logging drift if > expected.
- **Rate Limiting:** Enforce per-user and per-IP attempt throttling (e.g., exponential backoff after repeated failures).
- **Lockout Policy:** Configurable thresholds; lockout events auditable.
- **Data Protection:** Never log raw codes; only success/failure metadata.
- **Accessibility:** All form controls keyboard-navigable; labels tied to inputs; announce errors via ARIA live region.
- **Internationalization:** Error and instruction messages localizable.
- **Device Binding:** WebAuthn requires origin and RP ID match; counters monitored to detect cloned keys.
- **Privacy:** Minimize PII in audit logs; no factor secrets stored in plain text.
- **Performance:** Verification round trip < 1 second 95th percentile (excluding user entry time).
- **Resilience:** Retry guidance offered after transient network errors.
- **Compliance:** Align with MFA policy (e.g., NIST 800-63B AAL2+) for enrolled factors.
- **Telemetry:** Emit structured events for success, failure, lockout, fallback use.

4. Preconditions

- The user completed primary credential authentication (username/password or SSO) and is in partial session state.
- At least one MFA factor is enrolled and active.
- MFA enforcement policy requires verification for this login or step-up action.
- Verification service (TOTP validator, WebAuthn API, push service) is operational.

5. Postconditions

- **Success:** Session promoted to fully authenticated with recorded assurance level; failure counters reset.

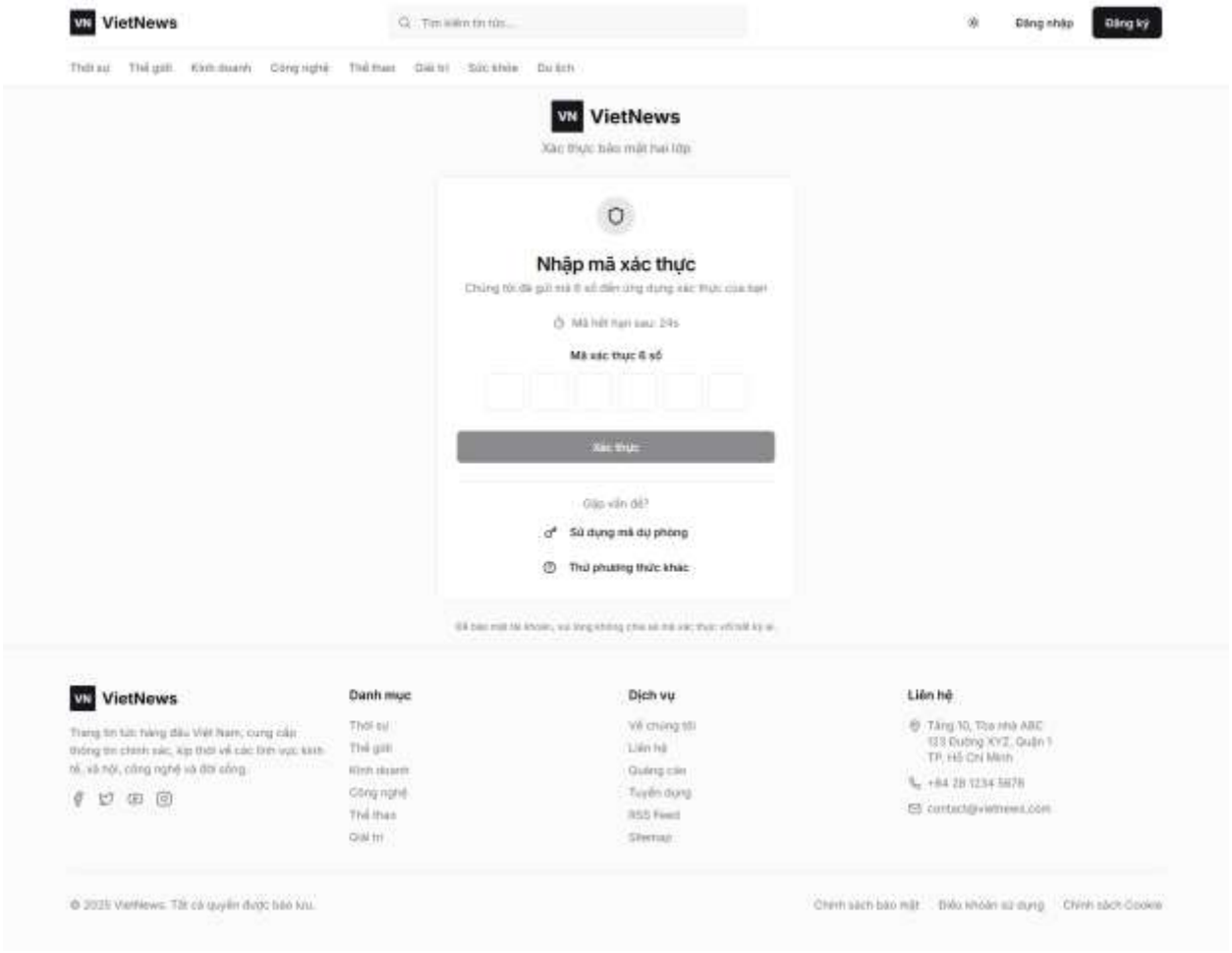
- **Lockout:** User blocked from further verification attempts until cooldown ends; partial session may be invalidated.
- **Abandon / Cancel:** Partial session terminated or left unusable for protected resources.
- **Fallback Success:** Backup or alternative factor verified; session elevated with **factorFallback = True.**
- **Risk Denied:** Attempt blocked; security alert optionally raised.

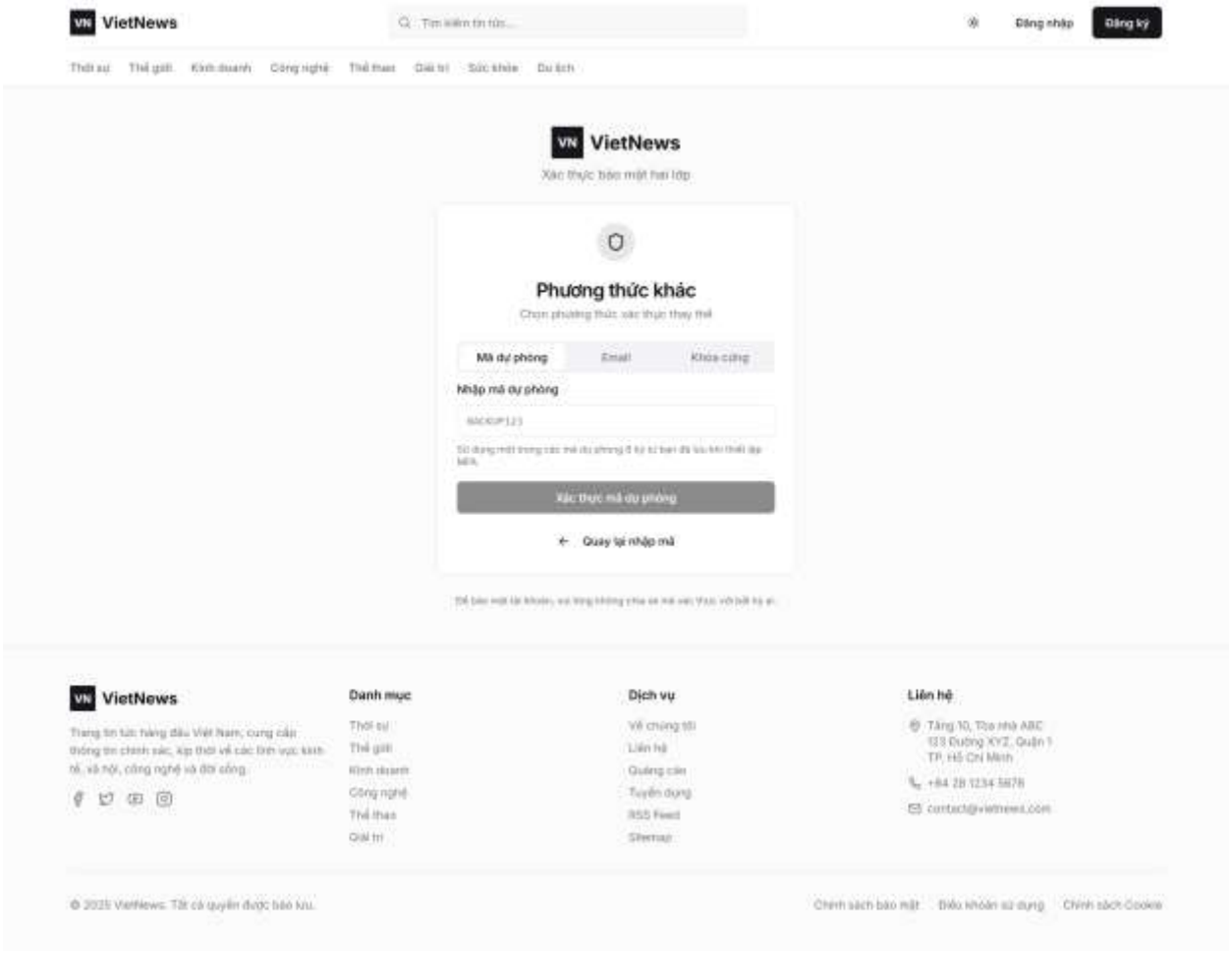
6. Extension Points

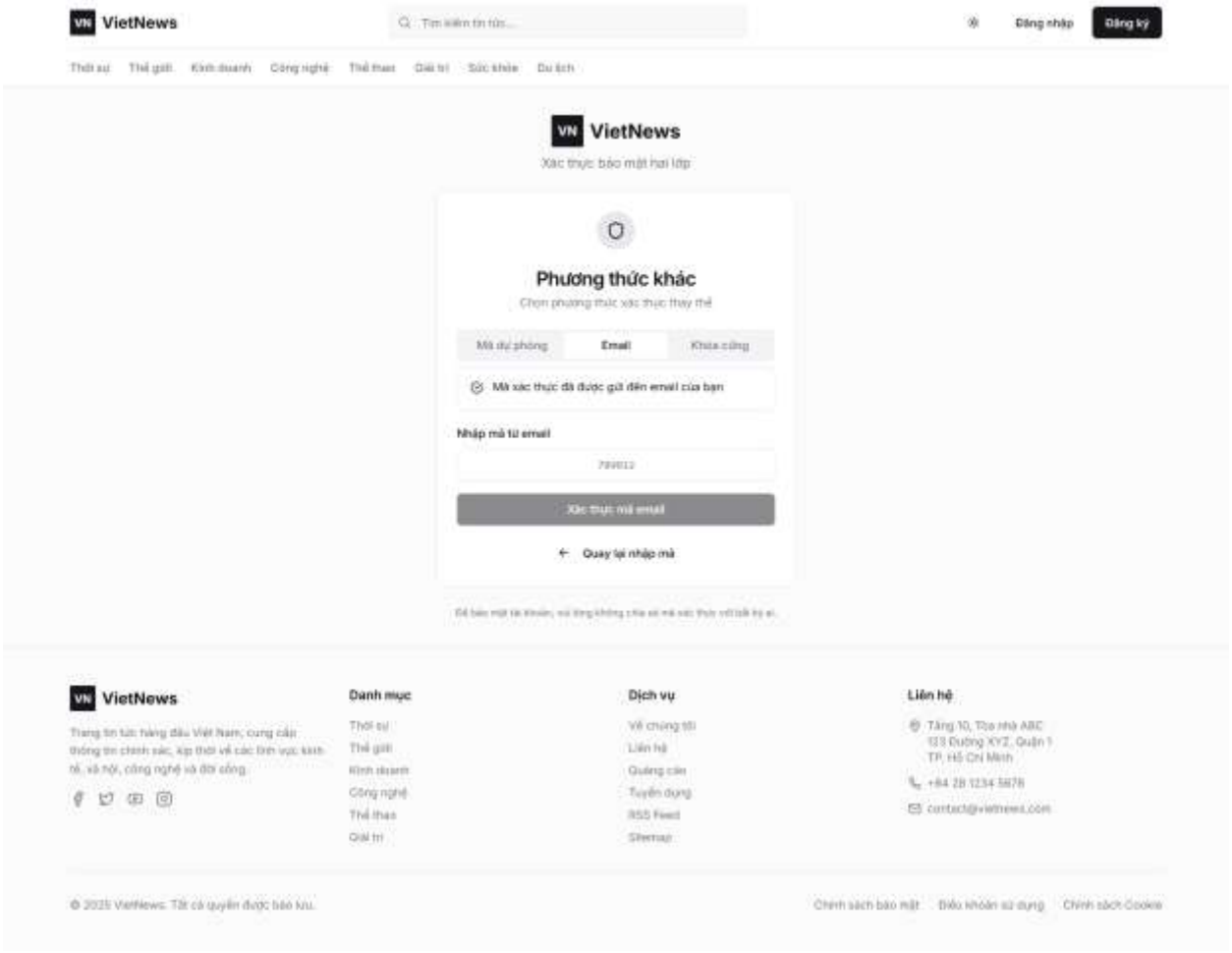
- **Adaptive Factor Selection: Insert between Basic Flow Steps 1–2 to dynamically choose factor based on risk scoring.**
- **Remember Device (Trusted Device Cookie): After Step 6, optionally set a signed "trusted device" token to skip MFA for low-risk future logins.**
- **Device Time Sync Suggestion: During 2.2.2 offer a one-click "Sync Time" helper or instructions.**
- **Additional Consent Prompt: After Step 6 for sensitive administrative consoles.**

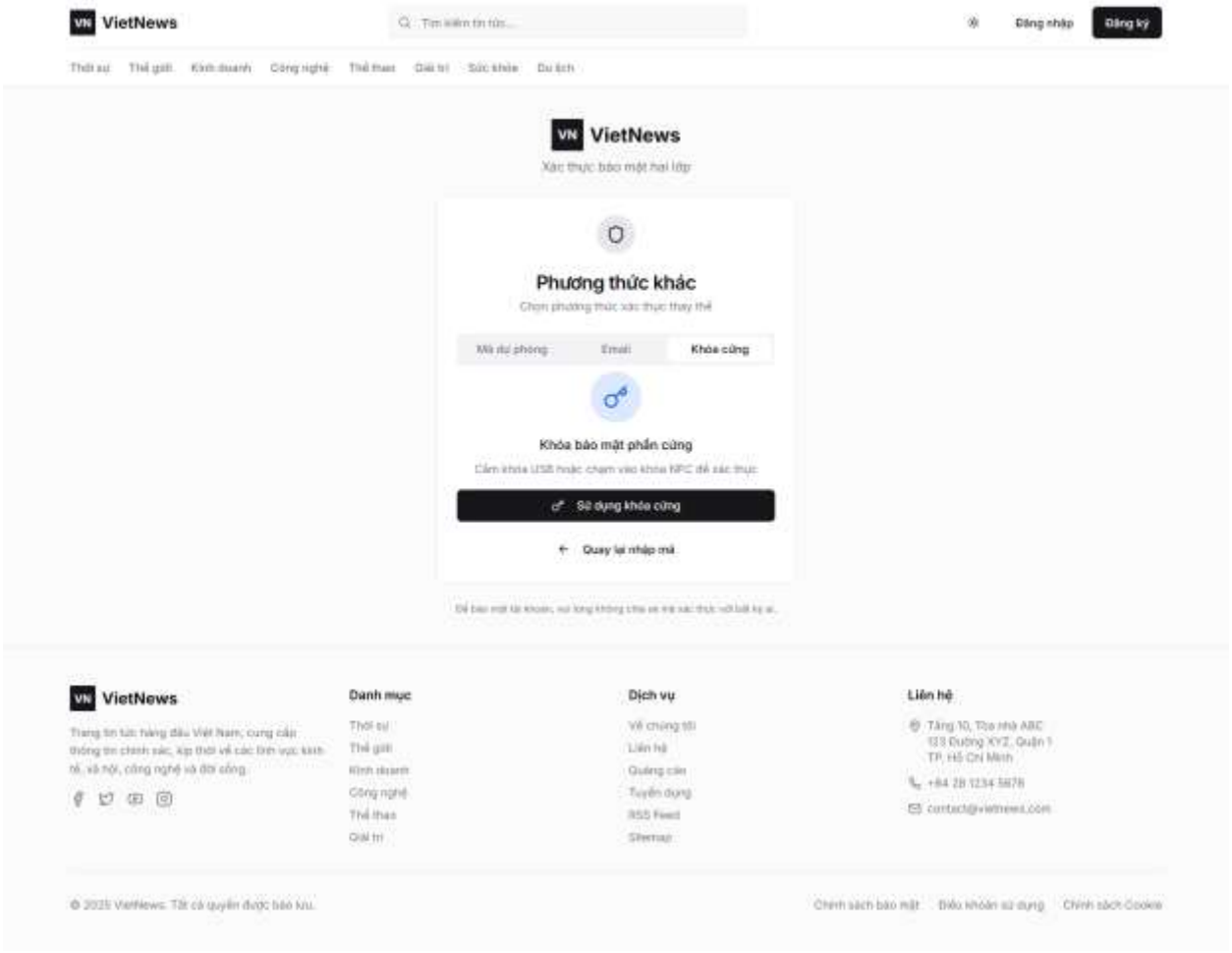
7. Low Fidelity Prototype For Use Case

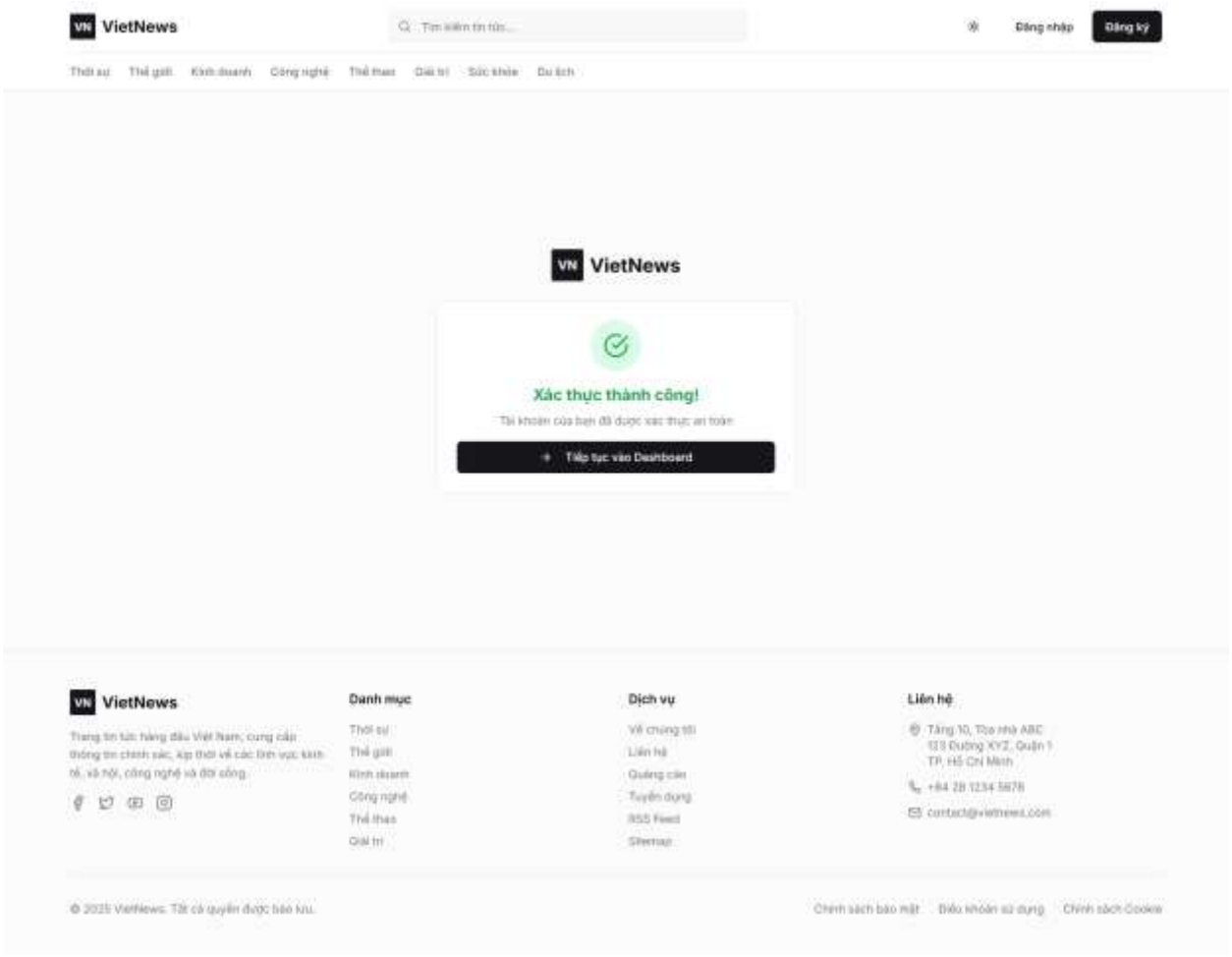
Prompt for AI tools: Enhance the /test/verify-mfa pages of my Digital Newspaper Website to simulate realistic, step-by-step processes. Use the same layout, design system, and UX style as the rest of the website. Include all typical stages described in the use-case specifications. 🔐 /test/verify-mfa – Multi-Factor Authentication FlowImplement a multi-step, interactive MFA verification process:Step 1 – Prompt for Code:Ask the user to enter a 6-digit code from an authenticator app (TOTP).Show fallback links: "Use backup code", "Try a different method".Include a live countdown (e.g., "Code expires in 30s").Step 2 – Handle Invalid Code:Show error on incorrect input with retry counter (e.g., "2 attempts left").After 3 failed attempts, lock verification for 5 minutes.Step 3 – Fallback Options:Let the user choose alternative methods: backup code, email, hardware key.Simulate each with mock inputs and success/failure.Step 4 – Success:Show confirmation: "Verification successful."Auto-redirect to dashboard or show "Continue" button.


VietNews
Đăng nhập
Đăng ký
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Sức khỏe
Du lịch
VietNews
Xác thực bảo mật hai lớp
Nhập mã xác thực
Mã hết hạn sau: 24s
Mã xác thực 6 số
Xác thực
Gặp vấn đề?
Sử dụng mã dự phòng
Thử phương thức khác
VietNews
Danh mục
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Dịch vụ
Về chúng tôi
Liên hệ
Quảng cáo
Tuyển dụng
RSS Feed
Sitemap
Liên hệ
Tầng 10, Tòa nhà ABC
123 Đường XYZ, Quận 1
TP. Hồ Chí Minh
+84 28 1234 5678
contact@vietnews.com
© 2025 VietNews. Tất cả quyền được bảo lưu.
Chính sách bảo mật
Điều khoản sử dụng
Chính sách Cookie


VietNews
Đăng nhập
Đăng ký
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Sức khỏe
Du lịch
VietNews
Xác thực bảo mật hai lớp
Phương thức khác
Chọn phương thức xác thực thay thế
Mã dự phòng
Email
Khóa cứng
Nhập mã dự phòng
Xác thực mã dự phòng
Quay lại nhập mã
VietNews
Danh mục
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Dịch vụ
Về chúng tôi
Liên hệ
Quảng cáo
Tuyển dụng
RSS Feed
Sitemap
Liên hệ
Tầng 10, Tòa nhà ABC
123 Đường XYZ, Quận 1
TP. Hồ Chí Minh
+84 28 1234 5678
contact@vietnews.com
© 2025 VietNews. Tất cả quyền được bảo lưu.
Chính sách bảo mật
Điều khoản sử dụng
Chính sách Cookie


VietNews
Đăng nhập
Đăng ký
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Sức khỏe
Du lịch
VietNews
Xác thực bảo mật hai lớp
Phương thức khác
Chọn phương thức xác thực thay thế
Mã dự phòng
Email
Khóa cứng
Mã xác thực đã được gửi đến email của bạn
Nhập mã từ email
Xác thực mã email
Quay lại nhập mã
VietNews
Danh mục
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Dịch vụ
Về chúng tôi
Liên hệ
Quảng cáo
Tuyển dụng
RSS Feed
Sitemap
Liên hệ
Tầng 10, Tòa nhà ABC
123 Đường XYZ, Quận 1
TP. Hồ Chí Minh
+84 28 1234 5678
contact@vietnews.com
Chính sách bảo mật
Điều khoản sử dụng
Chính sách Cookie


VietNews
Đăng nhập
Đăng ký
VietNews
Phương thức khác
Email
Khóa cứng
Khóa bảo mật phần cứng
Sử dụng khóa cứng
Quay lại nhập mã
VietNews
Danh mục
Dịch vụ
Liên hệ


VietNews
Đăng nhập
Đăng ký
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Sức khỏe
Du lịch
VietNews
Xác thực thành công!
Tiếp tục vào Dashboard
VietNews
Danh mục
Thời sự
Thế giới
Kinh doanh
Công nghệ
Thể thao
Giải trí
Dịch vụ
Về chúng tôi
Liên hệ
Quảng cáo
Tuyển dụng
RSS Feed
Sitemap
Liên hệ
Tầng 10, Tòa nhà ABC
123 Đường XYZ, Quận 1
TP. Hồ Chí Minh
+84 28 1234 5678
contact@vietnews.com
Chính sách bảo mật
Điều khoản sử dụng
Chính sách Cookie

404 Not Found

Digital Newspaper Website
Use-Case Specification: Check Plagiarism

Version 1.0

©Error! No property name supplied., 2025

Revision History

Date	Version	Description	Author
18/07/2025	1.0	Complete the document	Lê Đình Minh Quân

©Error! No property name supplied., 2025

Table of Contents

1. Use-Case Name 4

1.1 Brief Description 4

2. Flow of Events 4

2.1 Basic Flow 4

2.2 Alternative Flows 5

3. Special Requirements 8

4. Preconditions 9

5. Postconditions **9**

6. Extension Points **9**

7. Low Fidelity Prototype For Use Case **10**

©Error! No property name supplied., 2025

Use-Case Specification: Check Plagiarism

1. Use-Case Name

1.1 Brief Description

The *Check Plagiarism* use case describes how an authorized User (e.g., Writer, Editor, or Admin) submits an article draft (or selected text) to the plagiarism detection service. The system analyzes the content against internal repositories, previously published articles, and external sources (via integrated plagiarism API / search indexes) and returns a similarity report with matched segments, sources, and an overall similarity percentage. The outcome guides editorial decisions (revise, cite sources, reject, or approve).

2. Flow of Events

2.1 Basic Flow

Actor: Writer or Editor (authenticated; has permission to run plagiarism checks)
Preconditions: Draft article exists in the system and is accessible to the Actor; plagiarism service is operational; content length within allowed limits.
Trigger: Actor initiates a plagiarism check (e.g., clicks **"Check Plagiarism"** or saves draft with auto-check enabled).

1. Actor opens the draft and selects **Check Plagiarism**.
2. System validates prerequisites: user permission, draft state (not empty), size limits.
3. System normalizes content (strip markup, unify line endings, lowercase as needed, remove stop sections like boilerplate disclaimers per policy).
4. System packages normalized text with metadata (document ID, language, author ID, timestamp) and submits request to the plagiarism engine.
5. System displays a *"Scanning for similarity…"* progress indicator (with optional estimated time if > few seconds).
6. Plagiarism engine tokenizes and compares content against: (a) internal published corpus, (b) internal draft archive (other users' drafts if permissible), (c) cached external index results / third-party API.
7. Engine produces a similarity report (overall percentage, list of matched passages, source identifiers, match confidence, match types).
8. System stores the raw report and derives a summarized view (highlight spans, aggregated percentages by source).
9. System displays the plagiarism results to the Actor: overall similarity %, critical matches flagged above configured threshold(s), color-highlighted overlapping segments.
10. System logs an audit event `{userId, docId, action=PLAGIARISM_CHECK,`

`similarity=<value>, timestamp}`.

11. Actor reviews results and decides next action (revise, cite, request exemption, or mark as clean). Use case ends on decision.

2.2 Alternative Flows

2.2.1 Large Document – Asynchronous Processing

Point of deviation: Step 4.

1. If content size exceeds synchronous threshold, system queues job and returns a "Scan queued" status.
2. Actor sees queued indicator and may continue editing.
3. Background worker completes analysis and updates report.
4. System notifies Actor (in-app notification or email) → Actor opens report (resume at Basic Flow Step 9).

2.2.2 External Service Timeout / Unavailable

Point of deviation: Step 6.

1. External API times out or returns 5xx.
2. System retries per backoff policy (e.g., 2 more attempts).
3. If still failing, system returns partial internal corpus results + message "External sources temporarily unavailable."
4. Actor may retry manually later (return to Step 1) without losing draft state.

2.2.3 Insufficient Permissions

Point of deviation: Step 2.

1. User lacks required role/permission.
2. System denies request, shows "You are not authorized to perform plagiarism checks."
3. Audit event logged with outcome=DENIED.
4. Use case ends (no scan performed).

2.2.4 Content Too Short / Too Long

Point of deviation: Step 2 or 3.

1. Content length below minimum tokens (e.g., < 50 words) or above maximum (e.g., > 20,000 words).
2. System displays validation message specifying limit and prevents submission.
3. Actor adjusts content length and re-initiates (return to Step 1).

2.2.5 Duplicate Pending Scan
Point of deviation: Step 4.

1. An identical hash for this draft is already processing.
2. System merges request or informs Actor "A scan is already in progress."
3. Actor waits or views pending status (resume at Step 9 once complete).

2.2.6 High Similarity Threshold Breach (Critical Alert)
Point of deviation: Step 9.

1. Overall similarity or single-source match exceeds critical threshold (e.g., > 40%).
2. System flags report as **Critical** and (if configured) notifies Editor / Admin.
3. Actor must choose between revising or requesting formal review before publication (publication workflow blocked until resolved).

2.2.7 Exact Self-Match Only
Point of deviation: Step 7.

1. All matches are the author's previously published articles (self-plagiarism).
2. System categorizes these separately (self-matches) and may reduce their weighting per policy.
3. Actor views adjusted similarity (rejoins Step 9).

2.2.8 Significant Quoted / Properly Cited Material
Point of deviation: Step 8 / 9.

1. Large portions flagged but enclosed in quotes or have citations (e.g., references or block quotes).
2. System applies citation detection rules to discount properly cited spans from final critical percentage (but still shows them).
3. Revised similarity displayed to Actor (Step 9).

©Error! No property name supplied., 2025

2.2.9 Language Detection Mismatch
Point of deviation: Step 3.

1. Detected language differs from expected or mixed language detected.
2. System switches or augments comparison indexes accordingly.
3. Flow resumes at Step 4 without user interruption.

2.2.10 Policy Update During Scan
Point of deviation: Step 6.

1. Internal similarity thresholds changed mid-scan (e.g., Admin updates config).
2. System re-applies new thresholds to results before presenting Step 9.
3. Actor sees badge “Threshold updated during scan” for transparency.

2.2.11 Manual Cancellation by Actor
Point of deviation: Step 5 (asynchronous or long scan).

1. Actor clicks **Cancel Scan**.
2. System attempts to abort queued job; if already running, marks result stale on completion.
3. System logs cancellation outcome and informs Actor.
4. Use case ends (no report delivered unless job already finished early).

2.2.12 Integrity Check Failure (Tampered Content)
Point of deviation: Step 4 or 6.

1. Content hash on completion does not match original submission hash (draft changed mid processing without snapshot).
2. System flags integrity error, discards report, and requests a new scan.
3. Actor re-initiates (Step 1) after confirming stable draft.

2.2.13 Risk Engine Anomaly (Unusual Source Pattern)
Point of deviation: Step 7 / 8.

1. Many short matches from diverse unrelated sources trigger risk heuristic.

2. System elevates severity, optionally prompts for manual editorial review.

3. Actor cannot mark as “Clean” until review resolution (then resumes Step 11 decision).

2.2.14 Backup Source Index Offline
Point of deviation: Step 6.

1. Secondary (supplemental) index offline; primary still available.

2. System returns report with note “Some secondary sources unavailable.”

3. Actor may accept partial report or schedule re-scan.

2.2.15 Automatic Re-Scan on Revision
Point of deviation: After Step 11 (post decision).

1. Actor modifies content above configured change threshold (e.g., >15% new tokens) within a freshness window.

2. System triggers auto re-scan (notify Actor) and updates report (returns to Step 9).

3. Special Requirements

- **Accuracy:** Matching must minimize false positives via token n-gram overlap + fuzzy matching (stemmed / case-insensitive); properly cited/quoted sections discountable.
- **Performance:** Synchronous scans ≤ 8 seconds 95th percentile for average article length (e.g., ≤1500 words); async path handles larger documents.
- **Scalability:** Horizontal scaling for parallel scans; queue throughput target (e.g., 50 scans/minute baseline).
- **Security:** Submitted text transmitted over TLS; stored reports encrypted at rest; external API keys secured.
- **Privacy:** Do not expose private drafts from other authors unless access policy permits; anonymize user IDs in external requests if required.
- **Auditability:** Log each scan with docId, userId, similarity percentage, source counts, outcome (SUCCESS / PARTIAL / FAILED / CANCELED).
- **Configurability:** Thresholds (warning %, critical %, self-match discount) adjustable without redeploy.
- **Internationalization:** Multi-language tokenization & stemming; UI messages localizable.
- **Accessibility:** Results highlighting navigable via keyboard; color distinctions paired with text labels.

- **Reliability:** Retry logic for transient external failures; gracefully degrade to internal-only results.
- **Integrity:** Snapshot hash ensures the report matches the analyzed content version.
- **Legal / Compliance:** Retain reports for defined retention period (e.g., 1 year) then purge per policy.

4. Preconditions

- User authenticated and authorized for plagiarism checks.
- Draft content saved (not empty) and within permissible size range.
- Plagiarism engine and required indices reachable.
- Configured thresholds and policies loaded successfully.

5. Postconditions

- **Success:** Similarity report stored; Actor informed; decision pending editorial action.
- **Partial:** External sources missing; internal results delivered with notice.
- **Failure:** Scan aborted (timeout, integrity error); no final report; Actor prompted to retry.
- **Canceled:** User aborted; any late result marked stale.
- **Critical:** High similarity flagged; publication workflow gated pending remediation.
- **Auto Re-Scan Scheduled:** Content change triggers queued new scan.

6. Extension Points

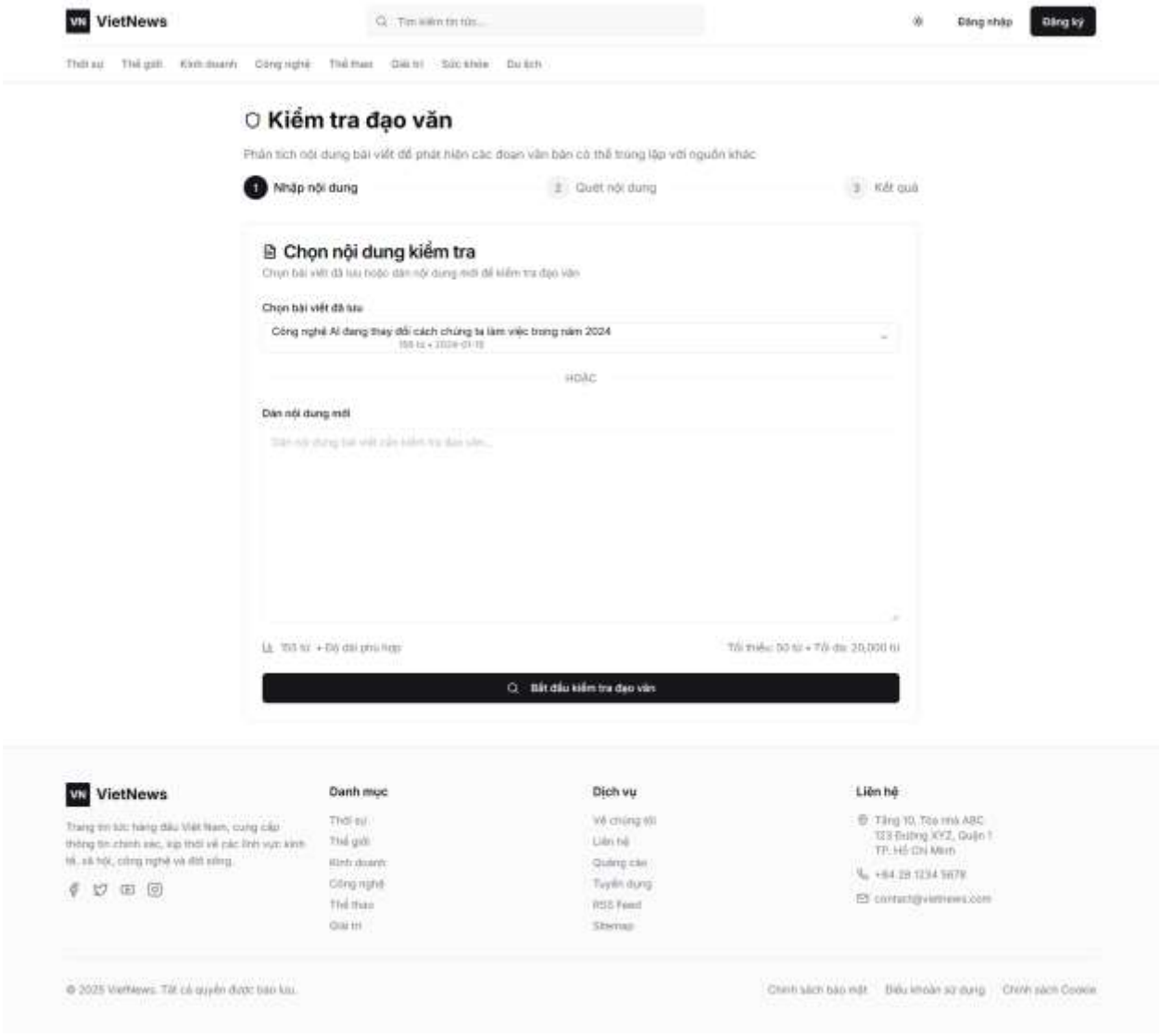
- **Auto Citation Suggestions: After Step 9, propose citation formatting for unmatched raw quotations.**
- **Machine Learning False Positive Filter: Between Steps 7–8 reclassify borderline matches.**
- **Similarity Trend Analytics: After Step 10, aggregate metrics for dashboards (average similarity per author).**
- **Inline Rewrite Assistant: Post Step 9 launches a guided paraphrasing assistant for flagged passages.**

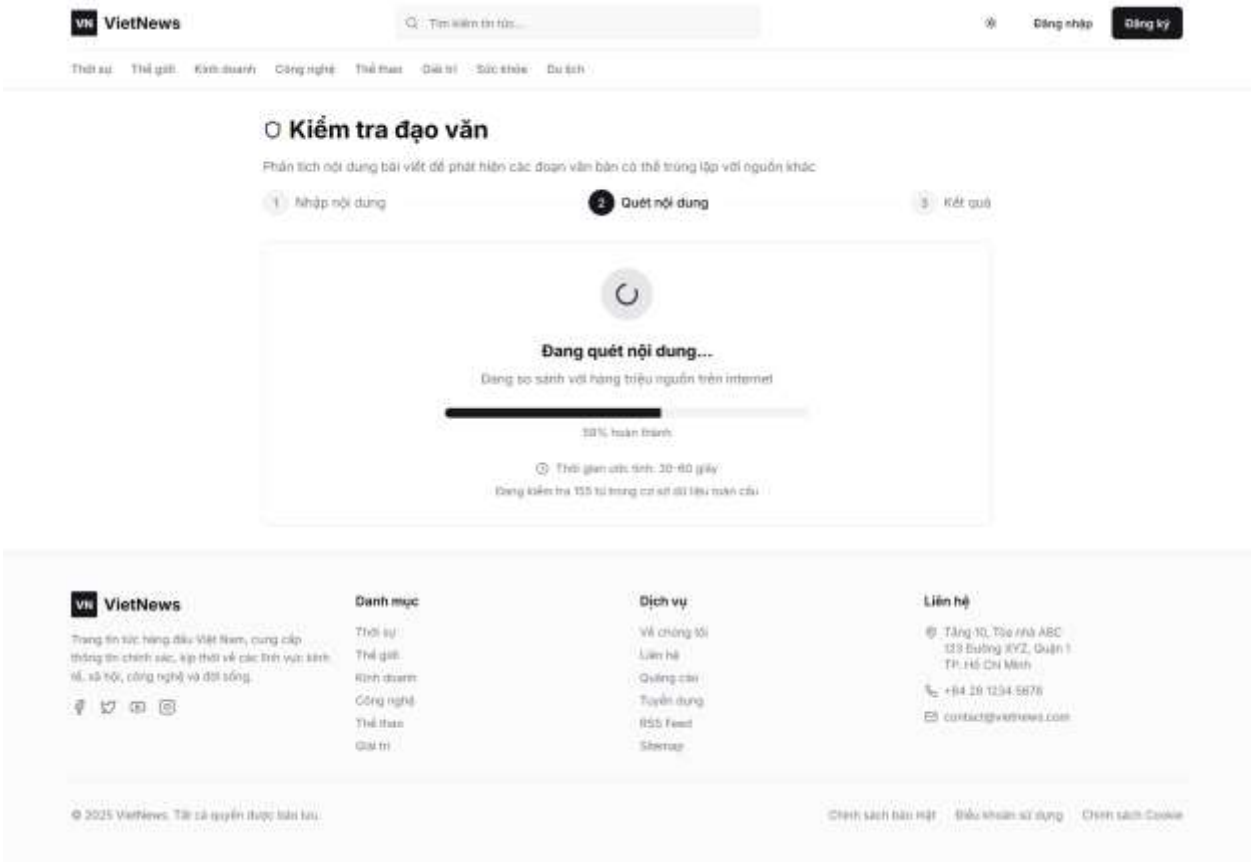
7. Low Fidelity Prototype For Use Case

Prompt for AI tools: Enhance the /test/plagiarism-report pages of my Digital Newspaper Website to simulate realistic, step-by-step processes. Use the same layout, design system, and UX style as the rest of

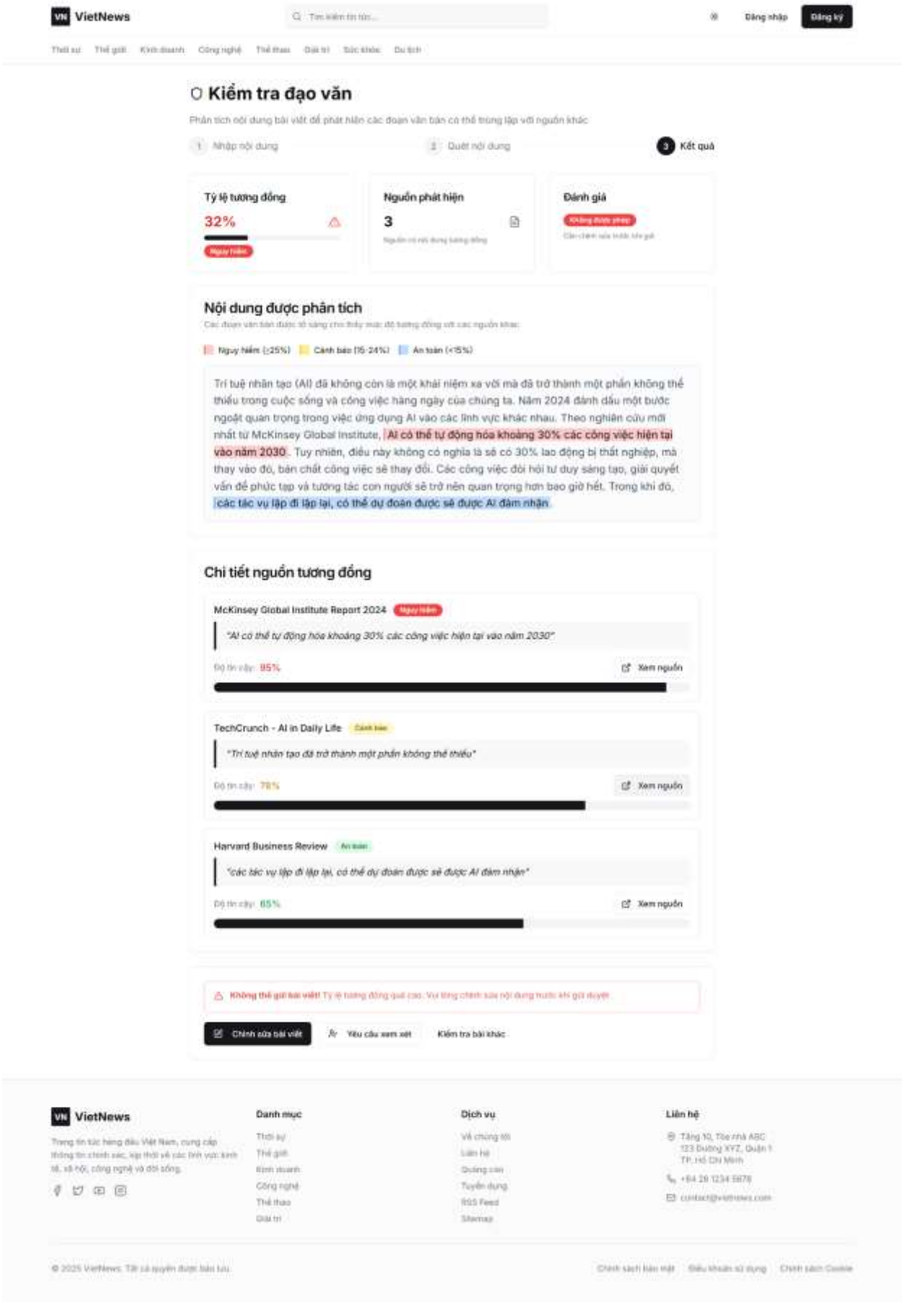
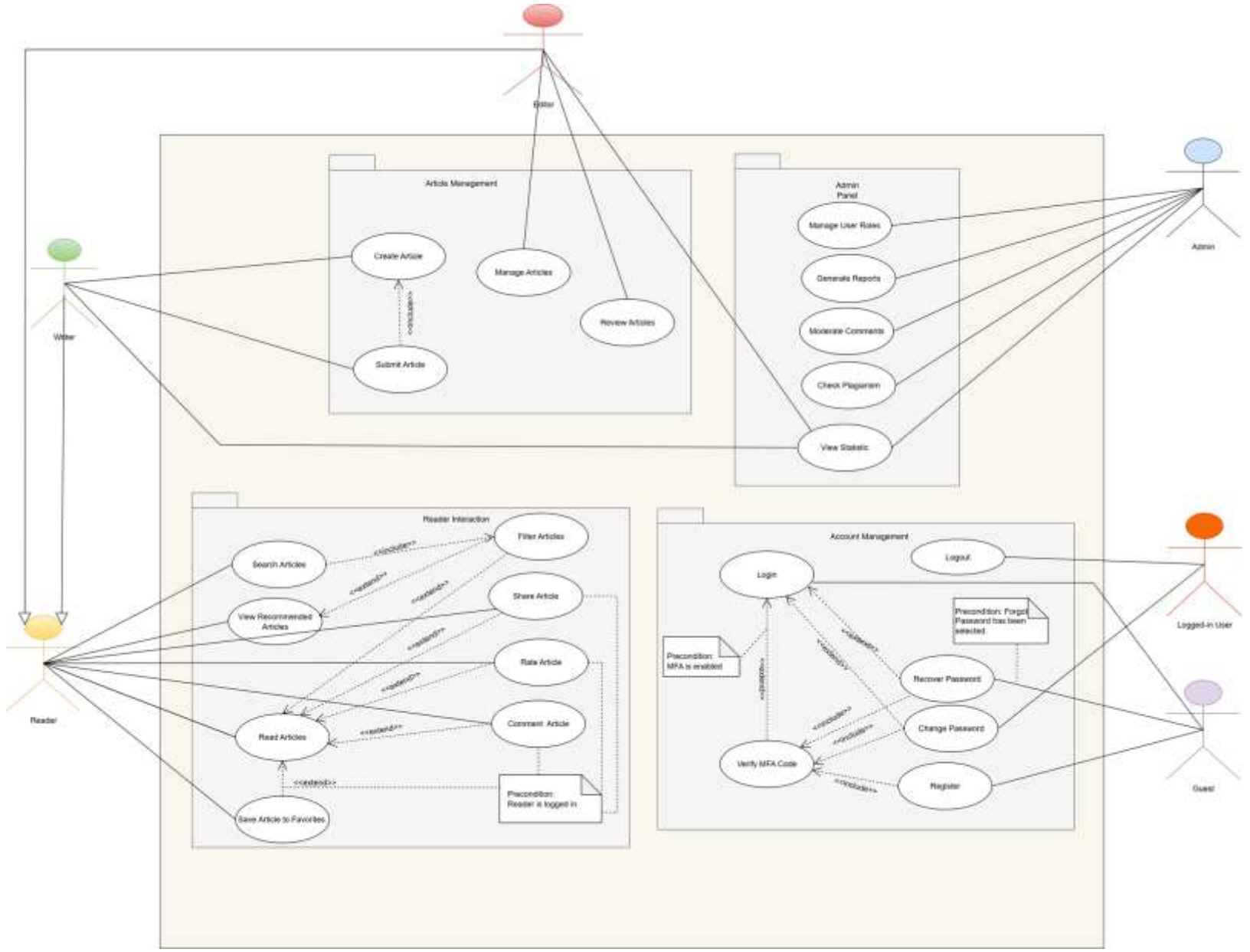
the website. Include all typical stages described in the use-case specifications. /test/plagiarism-report – Plagiarism Checking SimulationCreate a full plagiarism detection workflow:Step 1 – Draft Input:Allow selecting a saved draft or pasting article content.Validate length: warn if too short (<50 words) or too long (>20,000).Include “Run Check” button.Step 2 – Scanning:Show loading screen: “Scanning for similarity…” with animated progress.Simulate delay or mock API response.Step 3 – Results Display:Show:Overall similarity score (e.g., 32%)List of matching sources with confidence barsHighlighted matched segments in article contentUse different colors for critical (red), warning (orange), and clean (green) matches.Step 4 – Decision:If score is high, show alert and block submission.Buttons: “Revise”, “Request Review”, “Mark as Clean”.On acceptably low score, show: “Check passed. You may submit to Editor.”

©Error! No property name supplied., 2025


VietNews
Đăng nhập
Đăng ký
Kiểm tra đạo văn
Phân tích nội dung bài viết để phát hiện các đoạn văn bản có thể trùng lặp với nguồn khác
1 Nhập nội dung
2 Quét nội dung
3 Kết quả
Chọn nội dung kiểm tra
Chọn bài viết đã lưu hoặc dán nội dung mới để kiểm tra đạo văn
Chọn bài viết đã lưu
Công nghệ AI đang thay đổi cách chúng ta làm việc trong năm 2024
HOẶC
Dán nội dung mới
Bắt đầu kiểm tra đạo văn
VietNews
Danh mục
Dịch vụ
Liên hệ


VietNews
Đăng nhập
Đăng ký
Kiểm tra đạo văn
Phân tích nội dung bài viết để phát hiện các đoạn văn bản có thể trùng lặp với nguồn khác
1 Nhập nội dung
2 Quét nội dung
3 Kết quả
Đang quét nội dung...
Đang so sánh với hàng triệu nguồn trên internet
Danh mục
Dịch vụ
Liên hệ

Digital Newspaper Website	Version: 1.0
Use-Case Specification: Check Plagiarism	Date: 31/7/2025
check-plagiarism	



SƠ ĐỒ USE CASE