



## ***IT-314 Software Engineering***

### **Software Requirements Specifications Documentation**

#### **Group-15 News Aggregator**

##### **Team Members:**

**202201201 BOGHARA ZEELKUMAR GHANSHYAMBHAI**

**202201212 RATHOD HARSH DILIPBHAI**

**202201213 AKABARI KARTAVYA MANSHUKHBHAI**

**202201215 AKBARI KAMAL NITINBHAI**

**202201218 SARVAN MEET SUNILKUMAR**

**202201220 LATHIYA YASHVI VIPULBHAI**

**202201221 CHAUHAN FAJILMIYA MAHEMUDMIYA**

**202201249 VAGHANI PARTHIV VIJAYBHAI**

**202201254 VASA SAUMYA KALPESHBHAI**

**202201256 AYUSH PANDITA**

**Prof : Dr. Saurabh Tiwari**

**Mentor : Sarthak**



## Table of Contents

<b>1. Introduction.....</b>	<b>3</b>
<b>1.1 Purpose.....</b>	<b>3</b>
<b>1.2 Introduction to News Aggregation Platform.....</b>	<b>3</b>
<b>1.2.1 Description of Project.....</b>	<b>3</b>
<b>1.2.2 Main Features of the project.....</b>	<b>4</b>
<b>2. Specific Requirements.....</b>	<b>4</b>
<b>2.1 Functional Requirements.....</b>	<b>4</b>
<b>2.2 Non-Functional Requirements.....</b>	<b>7</b>
<b>3. User Stories.....</b>	<b>9</b>
<b>4. Product Backlog.....</b>	<b>31</b>
<b>5. System Behaviour.....</b>	<b>34</b>
<b>5.1 Use Case Diagram.....</b>	<b>34</b>
<b>5.2 Class Diagram.....</b>	<b>35</b>
<b>5.3 Sequence Diagram.....</b>	<b>38</b>
<b>5.4 System Design.....</b>	<b>46</b>
<b>5.5 Object Design.....</b>	<b>47</b>
<b>6. Technology Used.....</b>	<b>49</b>
<b>7. References.....</b>	<b>49</b>



# 1. Introduction

- The Software Requirements Specification (SRS) introduction explains the purpose, scope, and main features of the News Aggregator Platform. It describes the problem, the project's goals, and the needs of all users, including readers, news providers, and administrators. It also highlights what makes this platform special and useful in today's fast-moving digital news world. This SRS will guide every stage of the software development process and help keep developers, designers, and other team members on the same page

## 1.1 Purpose

- This document explains what the News Aggregation Platform needs to do and the features it should have by looking at the needs of users, news providers, and administrators. The main goal of this platform is to create one place where users can find news from different sources, interact with content, and enjoy a simpler news-reading experience. It aims to fix common problems like too much information, lack of interaction, and scattered news sources.

## 1.2 Introduction to News Aggregation Platform

### 1.2.1 Description of Project

- With more people reading news online, there is a huge amount of information available. Many news platforms have problems with too much content, no personalized recommendations, and difficulty managing user interactions. Our News Aggregation Platform will make it easy for users to find and read news from different sources in a way that feels personal and fun. Users can follow their favorite news providers, save articles, and interact by liking, commenting, and sharing, all while keeping their privacy and security safe. This project



aims to create a new, user-friendly news experience that fits the changing needs of today's online readers.

### **1.2.2 Main Features of the project**

- Article Viewing and Interaction (like, comment, share, bookmark)
- Search and Discovery
- Personalized News Feed
- User History and Activity Tracking
- Content Moderation
- News Provider Account Management
- Daily News Quiz
- User Profile Management
- Dark Mode and Accessibility

## **2. Specific Requirements**

### **2.1 Functional Requirements**

#### **1. User Account Management**

- Users, news providers, and admins must be able to register and log in.
- Users can update their profile (name, age, email, interests, etc.).

#### **2. News Access and Interaction**

- Non-logged-in users can read only headlines.
- Logged-in users can read full articles, like, share, comment, bookmark, and report news.
- Users can filter and search for news by categories and interests.

#### **3. News Provider Operations**



- News providers can create channels, post articles, and edit or delete posts.
- They can track analytics for posts and follower engagement.

#### **4. Quiz Feature**

- Logged-in users can play quizzes related to current news.
- Quiz results should be displayed upon completion.

#### **5. User Preferences**

- Users can toggle dark mode, follow/unfollow providers.

#### **6. Content Moderation**

- Admins and automated systems should flag or review inappropriate content.
- Admins have the final authority to approve, block, or remove flagged content.

#### **7. Registration and Authentication**

- Users must confirm registration via email verification.
- News providers should undergo additional verification for account approval.
- Password recovery through email should be available for all user types.

#### **8. News Categorization and Personalization**

- News articles must be tagged with categories (e.g., Science, Technology).
- Users can select and update their interests for personalized news feeds.
- Mute functionality allows users to hide specific news providers.

#### **9. Content Interaction**

- Users can report inappropriate content with specific reasons.
- Reported articles are flagged for admin review.



- Users can view all their interactions (likes, comments, bookmarks) in a dedicated "History" section.

## **10. Admin Management Features**

- Admins can suspend or restore user and provider accounts.
- Admins can create and modify quizzes for users.
- Admins can assign roles and privileges dynamically (e.g., promote a user to a moderator).

## **11. Provider Content Management**

- Providers can view detailed analytics (e.g., age, location, and engagement stats of their followers).
- Providers can create scheduled posts and set expiration dates for time-sensitive news.
- Providers can add multimedia content (images, videos, links) to articles.

## **12. Search and Discovery**

- A search bar should support filters like category, keyword, and date range.
- Suggested articles appear based on user reading history and preferences.

## **13. Dark Mode and Accessibility Features**

- Users can toggle dark mode from the navigation menu.
- Font sizes and color contrast should be adjustable for better readability.

## **14. Content Moderation Tools for Admins**

- Admins can review flagged articles in batches with pre-defined actions (e.g., warn, block, approve).
- Admins can set thresholds for automatic article flagging based on user reports.



## 2.2 Non-Functional Requirements

These are system qualities that define how the system should perform -

### 1. Performance

- The system must support a large number of concurrent users without significant delays.
- News feed updates and interactions (likes, shares, comments) should reflect in real time.

### 2. Scalability

- The platform should handle increasing numbers of users, news providers, and articles.
- Adding new categories or features should not disrupt the existing functionality.

### 3. Usability

- The interface should be user-friendly with intuitive navigation.
- The design must accommodate all device types (responsive design).

### 4. Security

- User data, especially passwords, must be encrypted.
- Role-based access controls should ensure users, news providers, and admins can only access authorized features.

### 5. Availability

- The platform should maintain 99.9% uptime.
- Downtime for maintenance must be scheduled during off-peak hours.

### 6. Accessibility.

- Features like dark mode enhance usability for diverse user needs.



## **7. Data Integrity**

- All user actions (like, comment, follow) must be accurately recorded.
- No data should be lost during system updates or crashes.

## **8. Analytics and Reporting**

- Admins and news providers require detailed analytics for decision-making.
- Analytics must be accurate and updated regularly.





### 3. User Stories

No .	Front of the Card	Back of the Card
1	As a user, I want to view news articles on the homepage so that I can stay informed without logging in.	<ul style="list-style-type: none"><li>○ The homepage displays a list of latest news articles.</li><li>○ Each article shows a headline, summary, and publication date.</li><li>○ Users can click on an article to read the full content.</li><li>○ No login or registration is required to view articles.</li></ul>
2	As a User, I want to register an account, so I can access more features of the news aggregator platform.	<ul style="list-style-type: none"><li>○ The registration form must include username, email, and password fields.</li><li>○ The system should validate the email address before account creation.</li><li>○ After registration, the user should receive a confirmation email.</li></ul>
3	As a Reader, I want to log into my account, so I can access full articles and other features like commenting.	<ul style="list-style-type: none"><li>○ The reader should log in with an email and password.</li><li>○ A “Remember Me” feature should be available.</li><li>○ The reader should be redirected to their personalized news feed after login.</li></ul>
4	As a Reader, I want to read full news articles, so I can stay informed about current events.	<ul style="list-style-type: none"><li>○ Readers should be able to click on any news thumbnail to view the full article.</li><li>○ The article page should include the title, full text, publication date, and related images.</li></ul>

5	As a Reader, I want to manage my profile, so I can update my personal information and preferences.	<ul style="list-style-type: none"> <li>○ The reader should be able to update their name, email, and password.</li> <li>○ Profile changes should be saved instantly and reflected in their account.</li> </ul>
6	As a Reader, I want to update my news feed preferences, so I can get articles tailored to my interests.	<ul style="list-style-type: none"> <li>○ The reader should be able to choose topics, categories, and regions.</li> <li>○ Changes to preferences should instantly affect the feed.</li> <li>○ The system should allow the reader to reset their feed to default settings</li> </ul>
7	As a Reader, I want to like articles, so I can express my appreciation for specific news content.	<ul style="list-style-type: none"> <li>○ The reader should be able to “like” any article from the article page.</li> <li>○ The system should display the total number of likes an article has received.</li> </ul>
8	As a Reader, I want to share articles on social media, so I can show my friends interesting news.	<ul style="list-style-type: none"> <li>○ Sharing options for Facebook, Twitter, and other platforms should be available.</li> <li>○ Clicking a share button should open the relevant platform with the article link pre-loaded.</li> </ul>
9	As a Reader, I want to comment on articles, so I can share my opinions and engage with other readers.	<ul style="list-style-type: none"> <li>○ A comment section should be available on each article page.</li> <li>○ Readers should be able to edit or delete their comments.</li> <li>○ Comments should be moderated for inappropriate content.</li> </ul>
10	As a Reader, I want to report articles, so I can flag content that violates guidelines (e.g., misinformation).	<ul style="list-style-type: none"> <li>○ A “Report” button should be available on all articles.</li> <li>○ The system should allow readers to choose the reason for reporting.</li> </ul>



		<ul style="list-style-type: none"><li>○ Admins and News providers should be notified of all reported articles.</li></ul>
11	As a Reader, I want to play quizzes related to news topics, so I can test my knowledge and have fun.	<ul style="list-style-type: none"><li>○ The platform should offer quizzes linked to specific news categories (e.g., sports, politics).</li><li>○ Quizzes should show results and scores at the end.</li></ul>
12	As a Reader, I want to bookmark articles, so I can save them for later reading.	<ul style="list-style-type: none"><li>○ Readers should be able to bookmark any article.</li><li>○ A “Bookmarks” section should be accessible from the reader's profile.</li></ul>
13	As a Reader, I want to view my reading history, so I can easily find articles I’ve previously read.	<ul style="list-style-type: none"><li>○ The platform should store the history of articles the reader has clicked on.</li><li>○ The history should be accessible from the profile page.</li></ul>
14	As a Reader, I want to mute specific news providers, so I don’t see content from sources I don’t trust or like.	<ul style="list-style-type: none"><li>○ Readers should be able to mute any news provider from the article page.</li><li>○ The system should prevent articles from muted sources from appearing in the feed.</li></ul>
15	As a News Provider, I want to register as a content provider, so I can post news articles on the platform.	<ul style="list-style-type: none"><li>○ Providers should fill out a registration form with details about their organization.</li><li>○ The system should send a verification email to confirm their account</li></ul>
16	As a News Provider, I want to log into my account, so I can post and manage my news articles.	<ul style="list-style-type: none"><li>○ Providers should log in using their email and password.</li><li>○ A dashboard should be available for content management.</li></ul>



17	As a News Provider, I want to post news articles, so I can share content with readers.	<ul style="list-style-type: none"><li>○ Providers should be able to upload articles, including title, content, images, and categories.</li><li>○ The article should be visible to readers once posted.</li></ul>
18	As a News Provider, I want to edit articles after they've been published, so I can update information or fix mistakes.	<ul style="list-style-type: none"><li>○ Providers should be able to edit any of their published articles.</li><li>○ Changes should reflect instantly after saving.</li></ul>
19	As a News Provider, I want to track the performance of my articles, so I can see how well they are being received.	<ul style="list-style-type: none"><li>○ Providers should see analytics such as views, likes, shares, and comments.</li></ul>
20	As a News Provider, I want to remove articles that are outdated or no longer relevant.	<ul style="list-style-type: none"><li>○ Providers should have an option to unpublish or delete articles.</li><li>○ Unpublished articles should no longer appear in the feed.</li></ul>
21	As an Admin, I want to manage user accounts, so I can ensure that only legitimate users are accessing the platform	<ul style="list-style-type: none"><li>○ Admin should have the ability to view, suspend, or delete user accounts.</li><li>○ Suspended users should receive an email explaining the suspension.</li></ul>
22	As an Admin, I want to manage news provider accounts, so I can verify and maintain the quality of content providers.	<ul style="list-style-type: none"><li>○ Admins should have a list of all registered news providers.</li><li>○ The admin should be able to suspend or delete provider accounts for violations of terms.</li></ul>
23	As an Admin, I want to review reported articles, so I can take action on content that violates guidelines.	<ul style="list-style-type: none"><li>○ Admin should have a dashboard listing all reported articles with reasons for reporting.</li></ul>

		<ul style="list-style-type: none"> <li>○ Admin should be able to mark reports as resolved, approve, or remove the reported articles.</li> </ul>
24	As a Reader, I want to report inappropriate comments, so I can help maintain a positive and respectful environment.	<ul style="list-style-type: none"> <li>○ Each comment should have a “Report” button for users to flag inappropriate content.</li> <li>○ Users should be prompted to select a reason for reporting.</li> </ul>
25	As a News Provider, I want to view if any of my articles have been reported, so I can understand any issues with my content.	<ul style="list-style-type: none"> <li>○ News providers should be able to see a list of their articles that have been reported by readers.</li> <li>○ The system should provide details on the nature of the report (e.g., misleading information, inappropriate content).</li> <li>○ Providers should be able to appeal or edit the reported articles before admins make a final decision.</li> </ul>
26	As a Reader, I want to follow specific news providers, so I can get updates directly from sources I trust.	<ul style="list-style-type: none"> <li>○ Readers should be able to follow or unfollow news providers from the article or provider profile pages.</li> <li>○ Articles from followed providers should be highlighted or prioritized in the reader’s feed.</li> <li>○ Readers should receive notifications when new articles from followed providers are posted.</li> </ul>
27	As an Admin, I want to view usage analytics across the platform, so I can monitor user engagement and content performance.	<ul style="list-style-type: none"> <li>○ The admin dashboard should display key metrics such as the number of active users, article views, and most popular categories.</li> </ul>



## **USE CASE DESCRIPTION:**

### **1) Use Case: Access and Read News Content**

#### **Description:**

This use case enables users to browse news articles with limited content. Users must sign up or log in to access the full articles.

#### **Actors:**

- General User

#### **Preconditions:**

1. The user has access to the platform.
2. Articles are available for viewing.

#### **Main Flow:**

1. The user accesses the news aggregator platform.
2. The system displays a list of available articles with limited content (e.g., titles, excerpts, or a preview).
3. The user selects an article to read.
4. If the user is not signed in, the system displays the limited content and prompts them to sign up or log in for full access.
5. If the user is signed in, the system displays the full article content.

**Alternative Flow:**

4.1. If the user chooses to sign up, the system presents the registration form.

4.2. If the user already has an account, they are redirected to the login form.

4.3. If login credentials are invalid, the system shows an error message and prompts for retry.

**Postconditions:**

1. The user can view full articles after successful authentication.
2. Failed login attempts are logged for security purposes.

**2) Use Case: News Provider Registration and Viewing Reported Content****Description:**

This use case allows news providers to register on the platform and log in to view articles that have been reported by either an admin or a reader.

**Actors:**

- News Provider

**Preconditions:**

1. The user must have access to the platform.
2. The user should have a valid registration or be eligible to register as a news provider.

**Main Flow:**



1. The user accesses the platform and chooses to register as a news provider.
2. The system presents a registration form specifically for news providers.
3. The user fills out and submits the registration form with the required details (e.g., credentials, organization information).
4. The system validates the information and creates a new provider account if all details are valid.
5. Once registered, the user can log in using their credentials.
6. Upon successful login, the system grants access to a dashboard where the provider can view articles that have been reported by admins or readers for review or further action.

**Alternative Flow:**

- 4.1. If registration details are invalid, the system prompts the user to correct and resubmit the form.
- 6.1. If login credentials are invalid, the system shows an error message and prompts for retry.

**Postconditions:**

1. The user is registered as a news provider and can log in to view reported articles.
2. Failed login attempts are logged for security purposes.

**3) Use Case: Read, Like, Share, Comment, Bookmark Articles and play Quizzes**



**Description:**

This use case allows registered readers to engage fully with articles on the platform by reading the entire content, liking, sharing, commenting, bookmarking articles, and participating in quizzes.

**Actors:**

- Reader

**Preconditions:**

1. The reader must be registered and logged into the platform.
2. Articles and quizzes must be available for engagement.

**Main Flow:****1. Read Full Articles:**

- The reader selects an article to read.
- The system displays the full content of the article for the logged-in reader.

**2. Like Articles:**

- The reader clicks on the "Like" button for an article.
- The system registers the "Like" and updates the count accordingly.

**3. Share Articles:**

- The reader clicks on the "Share" button.



- The system presents sharing options (e.g., via social media, email).
- The reader selects a preferred option, and the article link is shared.

#### **4. Comment on Articles:**

- The reader writes and submits a comment on an article.
- The system validates and displays the comment for the article if it adheres to platform guidelines.

#### **5. Play Quizzes:**

- The reader accesses a quiz linked to specific articles or topics.
- The system displays quiz questions, and the reader participates by selecting or inputting answers.
- Upon completion, the system shows the results and feedback.

#### **6. Bookmark Articles:**

- The reader clicks the "Bookmark" button for an article.
- The system adds the article to the reader's bookmarked list for future reference.

#### **Alternative Flow:**

2.1. If an error occurs while liking an article (e.g., network issue), the system shows an error message and prompts the reader to retry.

4.1. If a comment contains prohibited content, the system rejects the



comment and notifies the reader with a relevant message.

5.1. If there is an error during a quiz submission, the system prompts the reader to retry.

**Postconditions:**

1. The reader has engaged with the articles using the various available features (e.g., like, comment, bookmark, etc.).
2. User activity such as likes, comments, and bookmarks is recorded for future reference.

#### **4) Use Case: Reader Profile Management**

**Description:**

This use case enables readers to manage their profile information on the platform, including updating personal details, preferences, and profile photo.

**Actors:**

- Reader

**Preconditions:**

1. The reader must be logged into the platform.

**Main Flow:**

1. The reader accesses the profile management section.



2. The system displays the current profile details, including fields such as username, first name, last name, age, phone number, email, preferences, and profile photo.
3. The reader edits the desired fields (e.g., updating name, changing contact information, or setting preferences).
4. The reader can upload a new profile photo by selecting a file from their device.
  - The system displays the selected photo for preview before saving.
5. The reader saves the changes by clicking on the "Save Changes" button.
6. The system validates and updates the profile information and photo if the data is valid.
7. The system confirms successful profile updates.

**Alternative Flow:**

5.1. If the entered information is invalid (e.g., incorrect email format, missing required fields) or the uploaded photo is unsupported (e.g., wrong file format), the system shows an error message and prompts the reader to correct the input before resubmitting.

**Postconditions:**

1. The reader's profile details and photo are updated successfully.
2. Invalid or incomplete profile changes or unsupported photo uploads are not saved, and the reader is prompted for corrections.

## 5) Use Case: Reader Content and Feed Management

**Description:** This use case allows readers to update their news feed preferences, view their reading history, and manage the channels they follow and the articles they like on the platform.

**Actors:**

- Reader

**Preconditions:**

1. The reader must be logged into the platform.

**Main Flow:**

1. **Update News Feeds:**

- The reader accesses the news feed preferences section.
- The system displays current feed preferences (e.g., categories of interest, authors, or sources).
- The reader can update their news feed settings to follow new categories, authors, or sources.
- The reader saves the changes, and the system updates the feed accordingly.

2. **View Reading History:**

- The reader accesses the reading history section.
- The system displays a list of previously read articles, including titles and dates.

- The reader can view details of any article from the reading history.

### **3. View Followed Channels:**

- The reader accesses a section that lists all the channels or authors they are currently following.
- The system displays a list of followed channels or authors.
- The reader can choose to follow or unfollow channels from this list.

### **4. View Liked Articles:**

- The reader accesses a section that displays all the articles they have liked.
- The system displays a list of liked articles, including titles and dates.
- The reader can click on any article to read it again.

## **Alternative Flow:**

### **1. (Update News Feeds):**

- If the reader attempts to follow a channel or category that does not exist, the system shows an error message indicating no such category or author is available and prompts the reader to try again or search for available ones.

### **2. (View Reading History):**

- If the reader has not read any articles yet, the system displays a message: "You haven't read any articles yet."

- The reader is prompted to explore and read new articles.

### **3. (View Followed Channels):**

- If the reader is not following any channels or authors, the system displays a message: "You are not following any channels or authors."
- The reader is given an option to explore and follow new channels or authors.

### **4. (View Liked Articles):**

- If the reader has not liked any articles, the system displays a message: "You haven't liked any articles yet."
- The reader is prompted to like articles for future reference.

### **Postconditions:**

1. The reader's news feed preferences, followed channels, and liked articles are updated successfully.
2. The reader's reading history is displayed, allowing for easy access to previously read content.
3. Any errors encountered during feed updates or profile changes are resolved, and no incorrect data is saved.

## **6) Use Case: Mute News Provider and Report Articles**

### **Description:**



This use case allows readers to mute news providers, report comments, and report articles on the platform. These actions ensure that readers can control their content exposure and maintain the integrity of discussions and articles.

**Actors:**

- Reader

**Preconditions:**

1. The reader must be logged into the platform.
2. The reader has access to content (articles and comments) that can be reported.

**Main Flow:****1. Mute News Providers:**

- The reader accesses a news provider's profile or a specific article's source.
- The reader selects the option to mute the news provider.
- The system confirms the action and removes any future articles or updates from the muted provider from the reader's feed.
- The system displays a confirmation message: "You have muted this news provider."

**2. Report Comments:**

- The reader views an article and reads the comments section.



- If the reader finds a comment that violates the platform's guidelines (e.g., offensive language, spam), they can click on the "Report" button next to the comment.
- The reader selects a reason for the report (e.g., harassment, misinformation, inappropriate content).
- The system submits the report and displays a confirmation message: "Your report has been submitted."

### **3. Report Articles:**

- The reader views an article and identifies a potential issue (e.g., misinformation, offensive content).
- The reader clicks on the "Report Article" button.
- The reader selects a reason for reporting the article (e.g., fake news, inappropriate content).
- The system submits the report and displays a confirmation message: "Your report has been submitted."

### **Alternative Flow:**

#### **1. (Mute News Providers):**

- If the reader attempts to mute a provider that is already muted, the system shows a message: "This news provider is already muted."

#### **2. (Report Comments):**

- If the comment is not in violation of platform guidelines, the system shows a message: "This comment does not violate the platform guidelines."

### 3. (Report Articles):

- If the article does not meet the criteria for reporting, the system shows a message: "This article does not meet the reporting criteria."

### Postconditions:

1. The reader has successfully muted the chosen news provider.
2. Any reported comments or articles are submitted for review by the platform's moderation team.
3. The reader is notified of the successful submission of their reports.
4. Reports are logged for further investigation and moderation action, if necessary.

## 7) Use Case: Admin controls – Readers and News Providers

### Description:

This use case allows the admin to manage user accounts and news providers on the platform, ensuring the proper operation and content control of the system.

### Actors:

- Admin

**Preconditions:**

1. The admin is logged into the platform.
2. The admin has the necessary privileges to manage users and news providers.

**Main Flow:****1. Manage Users:**

- Admin accesses the user management section.
- The system displays a list of all registered users.
- The admin can view, edit, deactivate, or delete user accounts.
- The admin can also assign or change user roles (e.g., reader, news provider).
- Any changes made are confirmed, and the system updates user information accordingly.

**2. Manage News Providers:**

- Admin accesses the news provider management section.
- The system displays a list of all registered news providers.
- The admin can approve, reject, deactivate, or remove news providers.
- The admin can also modify the provider's access or privileges.



- Any changes made are confirmed, and the system updates the provider's status accordingly.

### **Alternative Flow:**

#### **1. (Manage Users):**

- If a user is already deactivated, the system shows "User is already inactive."

#### **2. (Manage News Providers):**

- If a news provider is already inactive, the system shows "News provider is already deactivated."

### **Postconditions:**

1. User and news provider information is updated, deactivated, or removed as needed.
2. Changes are reflected in the system, ensuring proper user and provider management.

## **8) Use Case: View System Usage Analytics**

### **Description:**

This use case allows the admin to manage reported articles for violations and access system usage analytics to monitor platform activity.

### **Actors:**

- Admin

### **Preconditions:**



1. Admin is logged into the platform with the necessary privileges.

## **Main Flow:**

### **1. Review Reported Articles:**

- Admin accesses the reported articles section.
- The system displays a list of flagged articles with details on why they were reported.
- Admin reviews each article and takes action (approve, delete, or flag for further review).
- System updates the article status based on the admin's decision.

### **2. View System Usage Analytics:**

- Admin accesses the system analytics dashboard.
- The system shows key metrics like user activity, article views, new sign-ups, and platform performance.
- Admin can filter the data by time period, user type, and other factors, and view visual representations (charts/graphs) of the data.

## **Alternative Flow:**

### **1. (Review Reported Articles):**

- If an article is found to be in compliance, the system will mark it as cleared, showing "No violation found."

### **2. (View System Usage Analytics):**



- If there is insufficient data (e.g., no recent activity), the system displays "No data available for the selected period."

**Postconditions:**

1. Reported articles are either cleared, deleted, or flagged based on admin's review.
2. System analytics data is updated and available for monitoring.



## **4. Product Backlog**

### **Sprint 1**

#### **User Registration & Basic Profile Management**

##### **USER STORIES:**

- User: View News Articles
- User: Register an Account
- Reader: Login to Read News
- News Provider: Register as a Provider
- News Provider: Login to Manage Content

### **Sprint 2**

#### **Article Viewing & Basic Interactions**

##### **USER STORIES:**

- Reader: Read Full Articles
- Reader: Like Articles
- Reader: Share Articles
- Reader: Comment on Articles
- Reader: Play Quizzes
- Reader: Bookmark Articles



## **Sprint 3**

### **News Provider Management**

#### **USER STORIES:**

- News Provider: Post News Articles
- News Provider: Edit Published Articles
- News Provider: Track Article Performance
- News Provider: Remove Articles
- News Provider: View Reporting on Their Articles

## **Sprint 4**

### **Reader Profile Management**

#### **USER STORIES:**

- Reader: Manage Profile
- Reader: Update News Feeds
- Reader: View Reading History
- Reader: Mute News Providers
- Reader: Report Comments
- Reader: Report Articles
- Reader: Follow Specific News Providers





## **Sprint 5**

### **Admin Controls & Analytics**

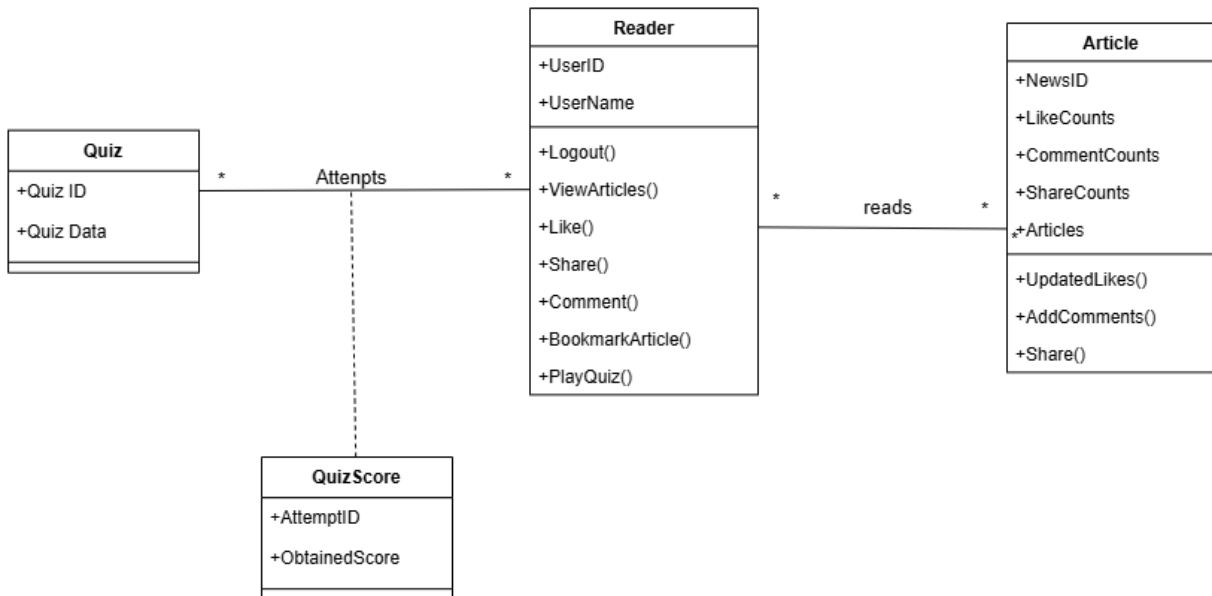
#### **USER STORIES:**

- Admin: Manage Users
- Admin: Manage News Providers
- Admin: Review Reported Articles
- Admin: View System Usage Analytics

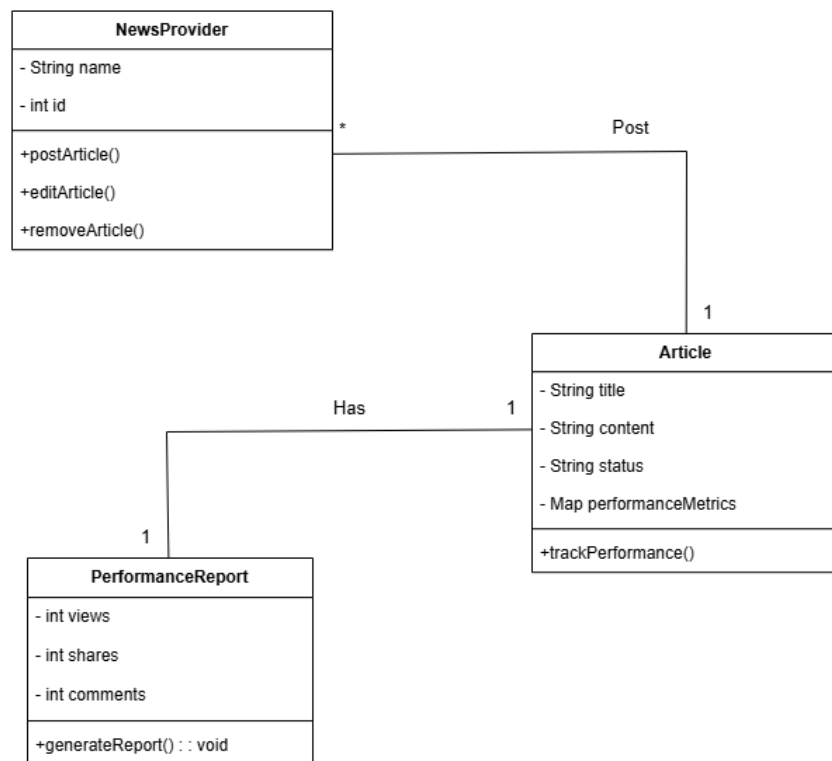


## 5.2 Class Diagram

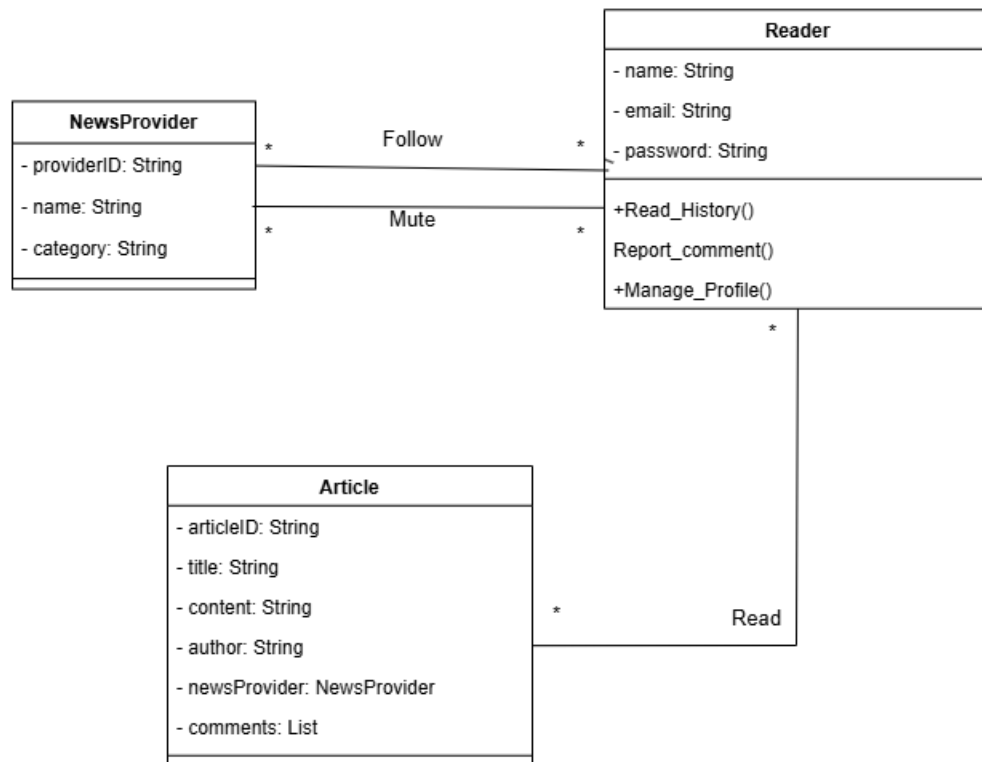
### Sprint 2



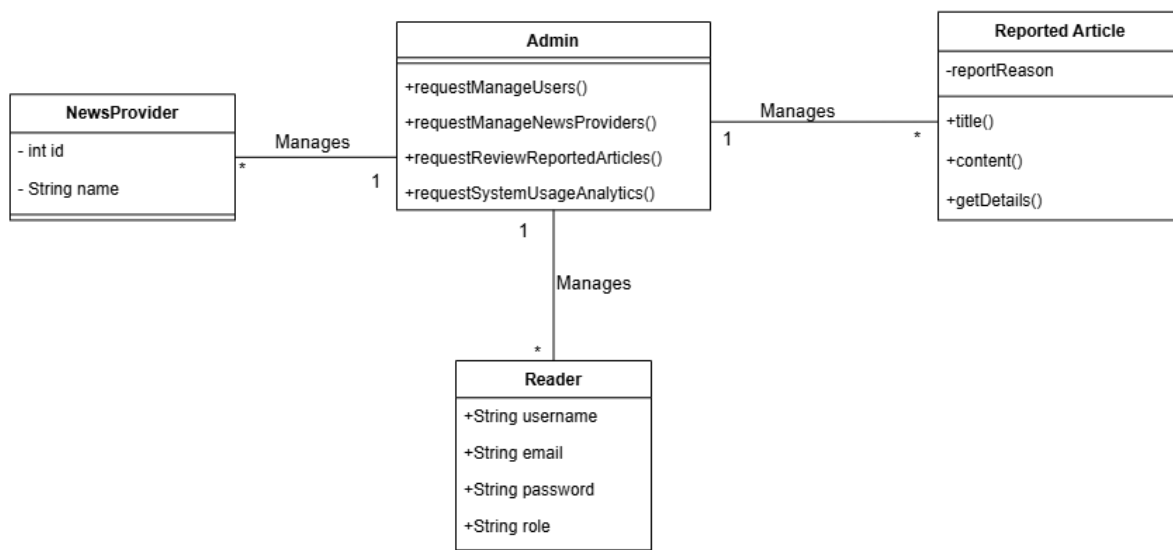
### Sprint 3



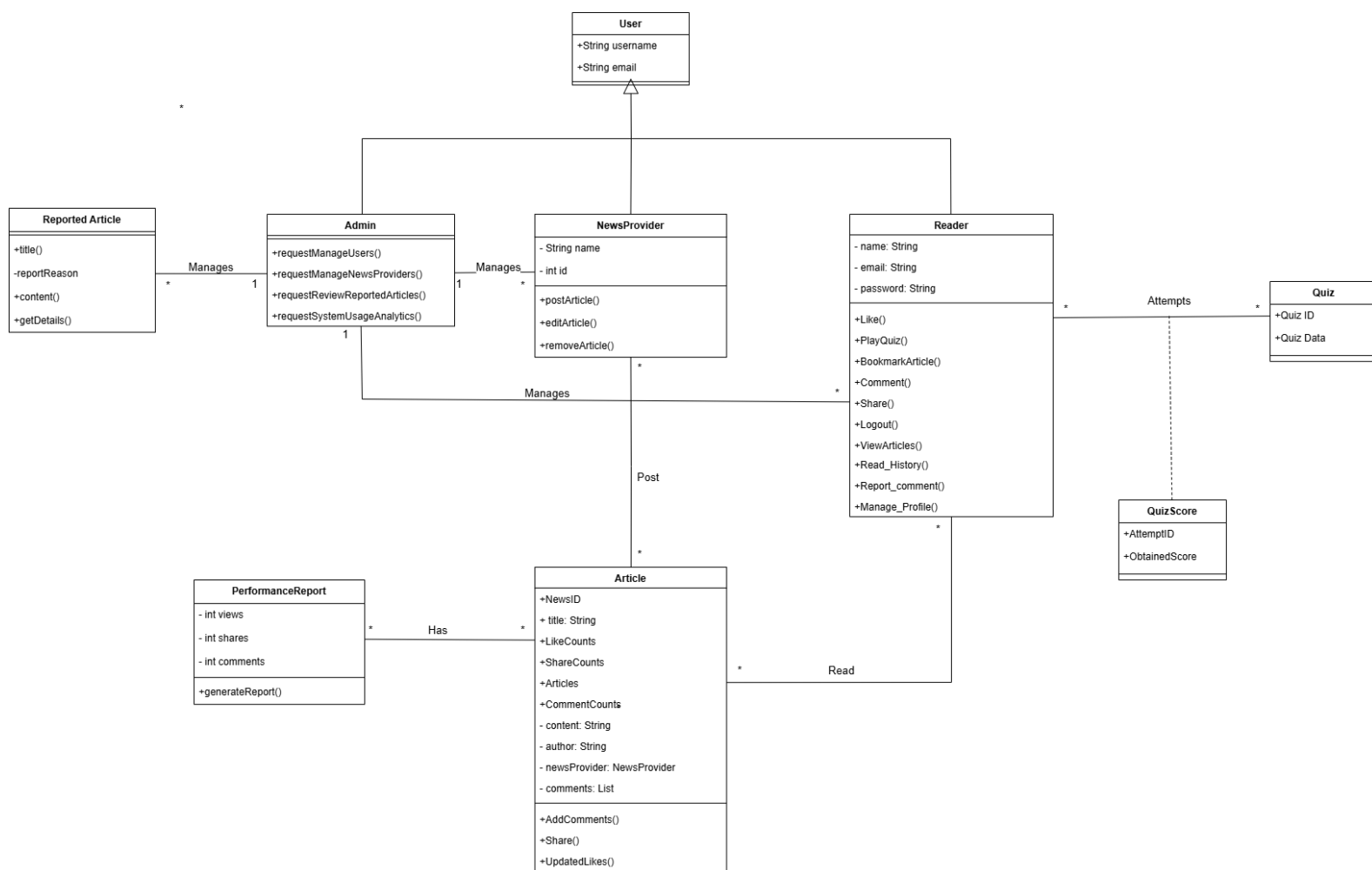
## Sprint 4



## Sprint 5

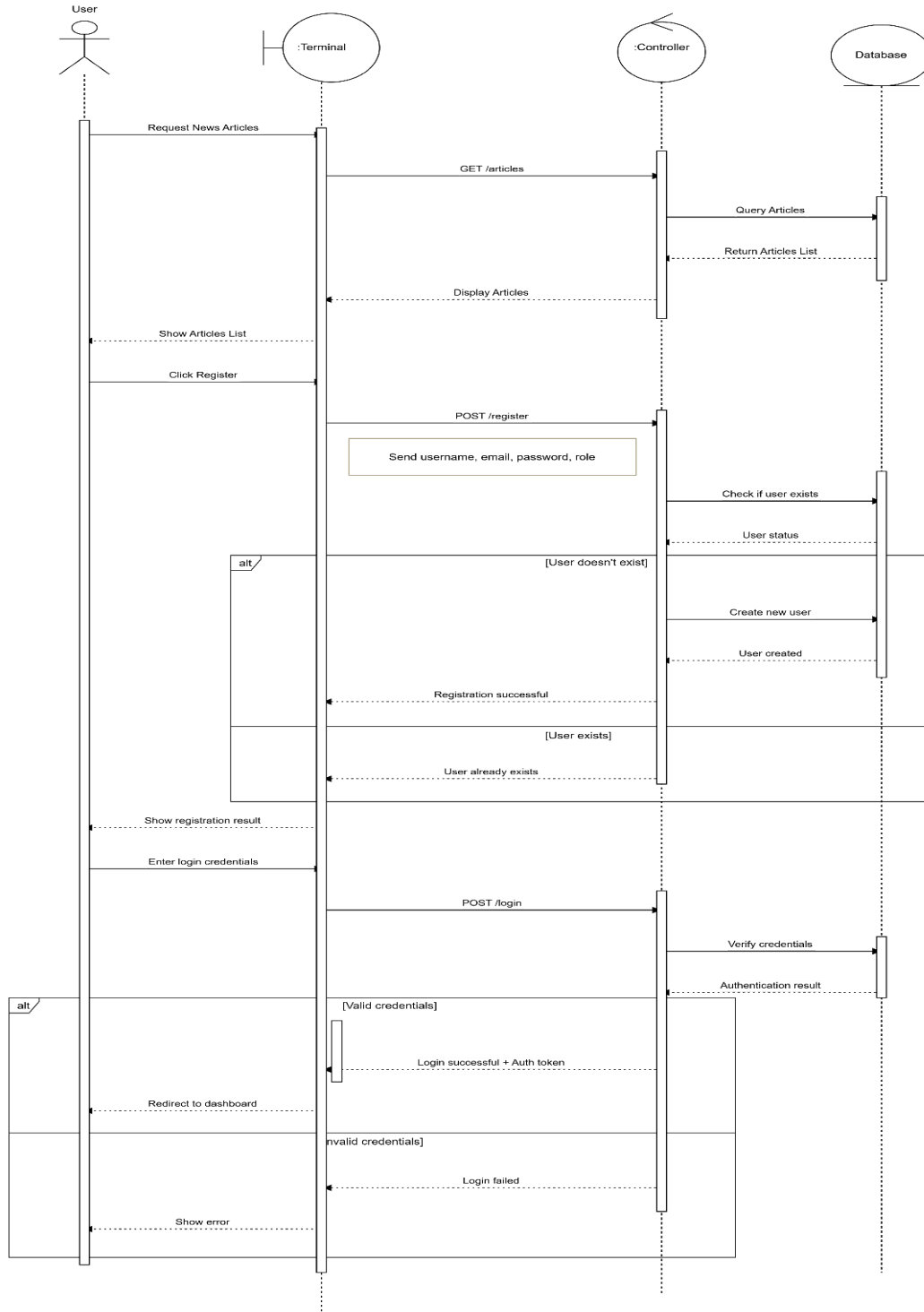


## Merged class diagram

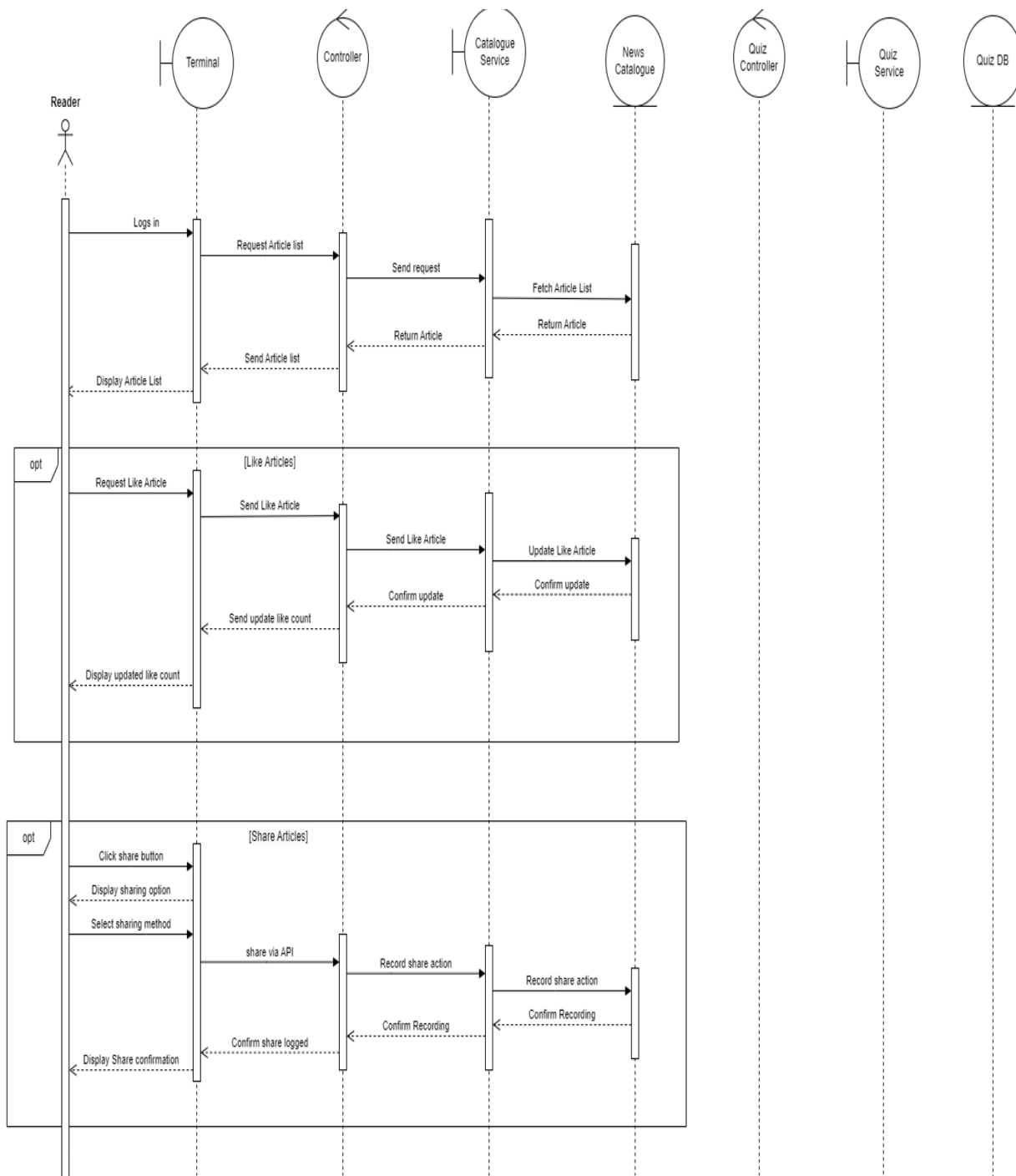


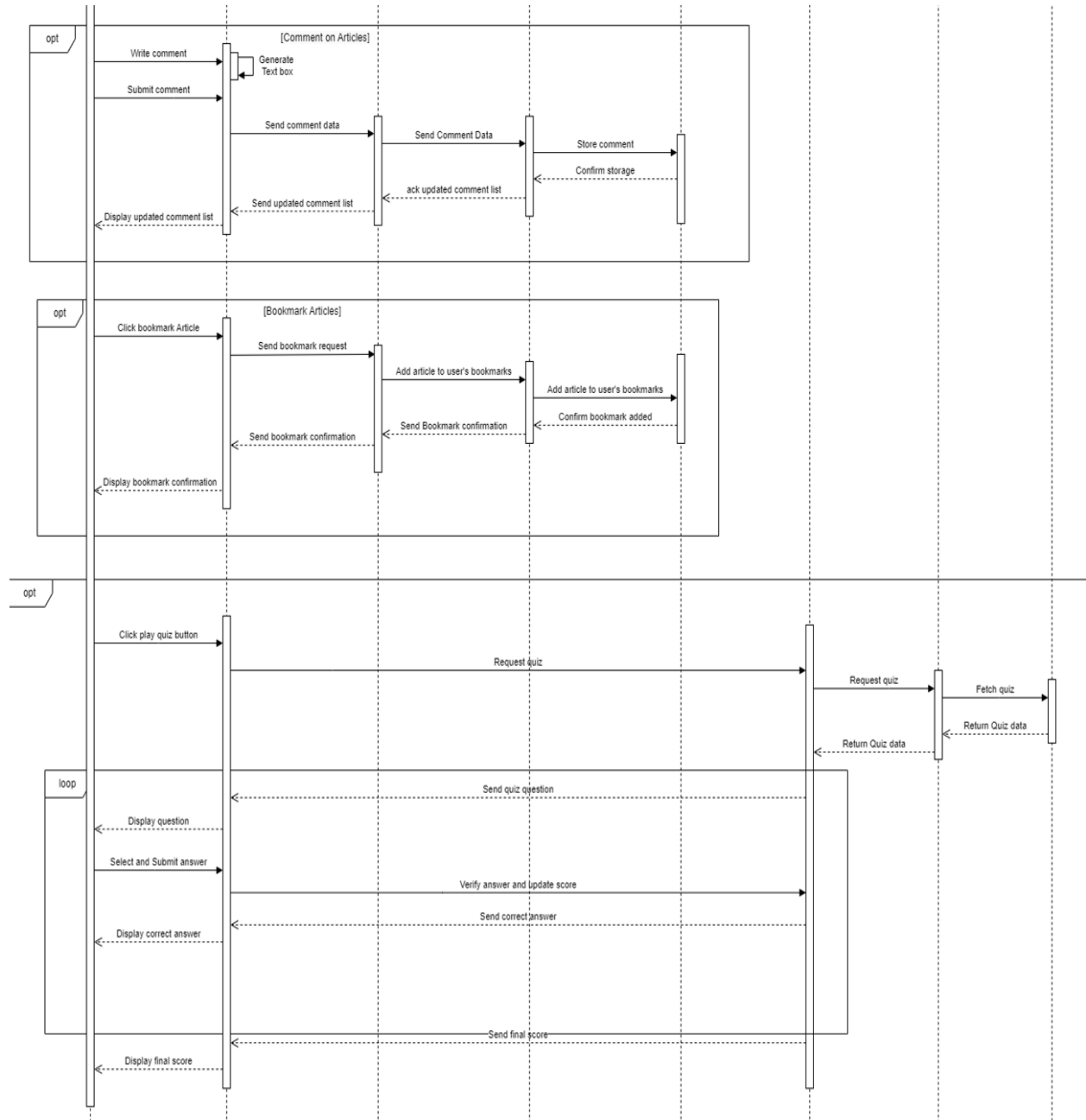
## 5.3 Sequence Diagram

### Sprint 1



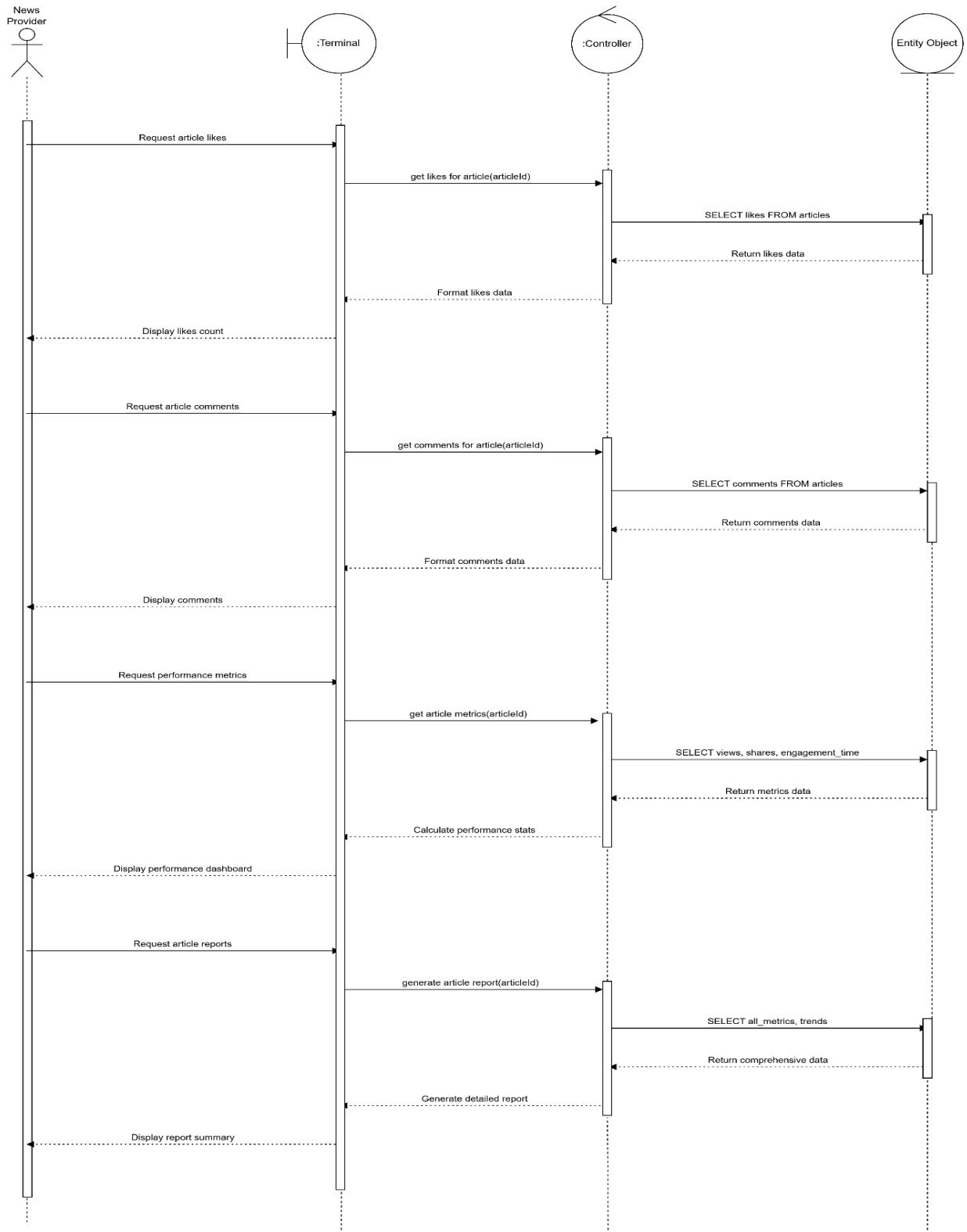
## Sprint 2



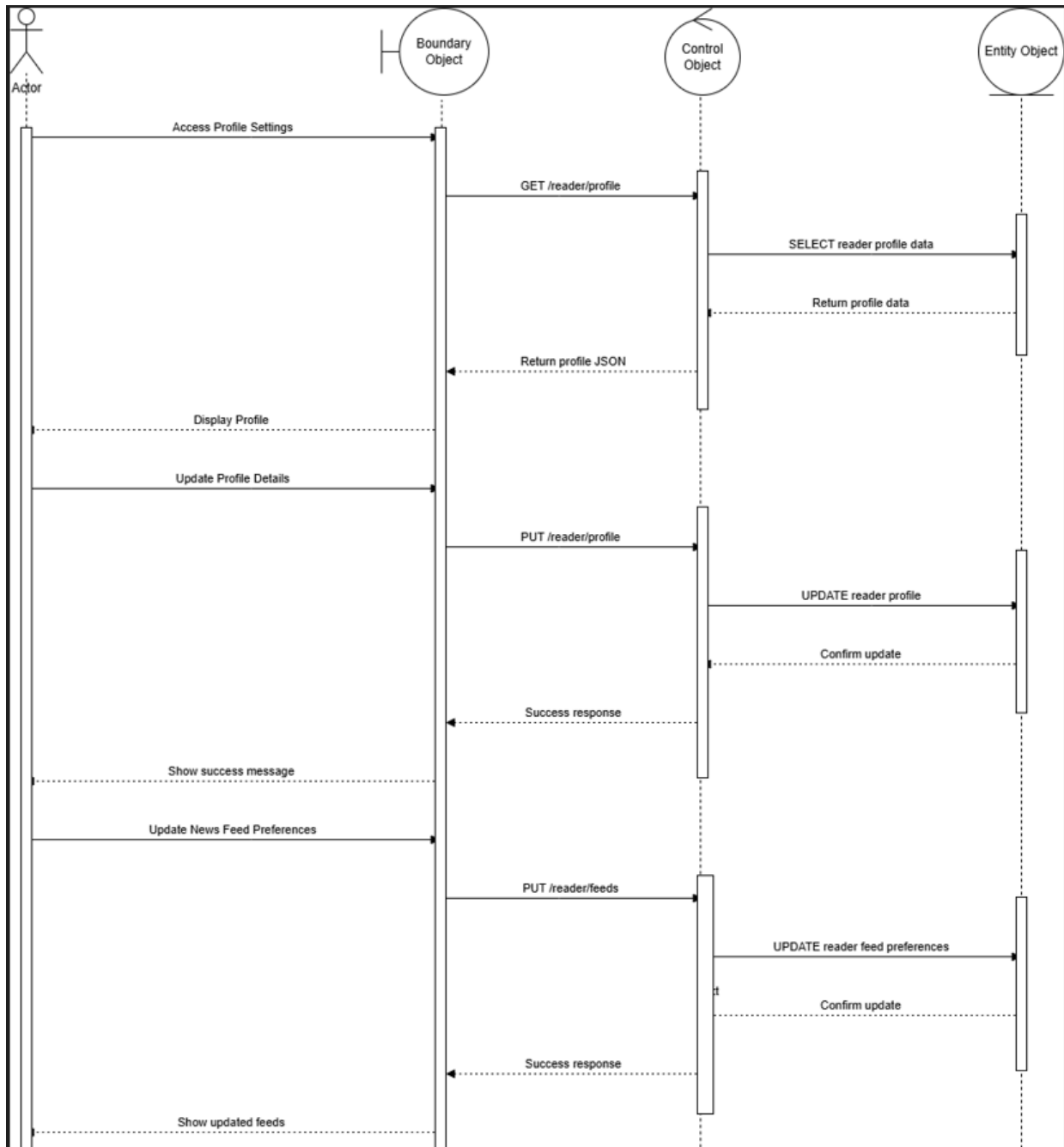


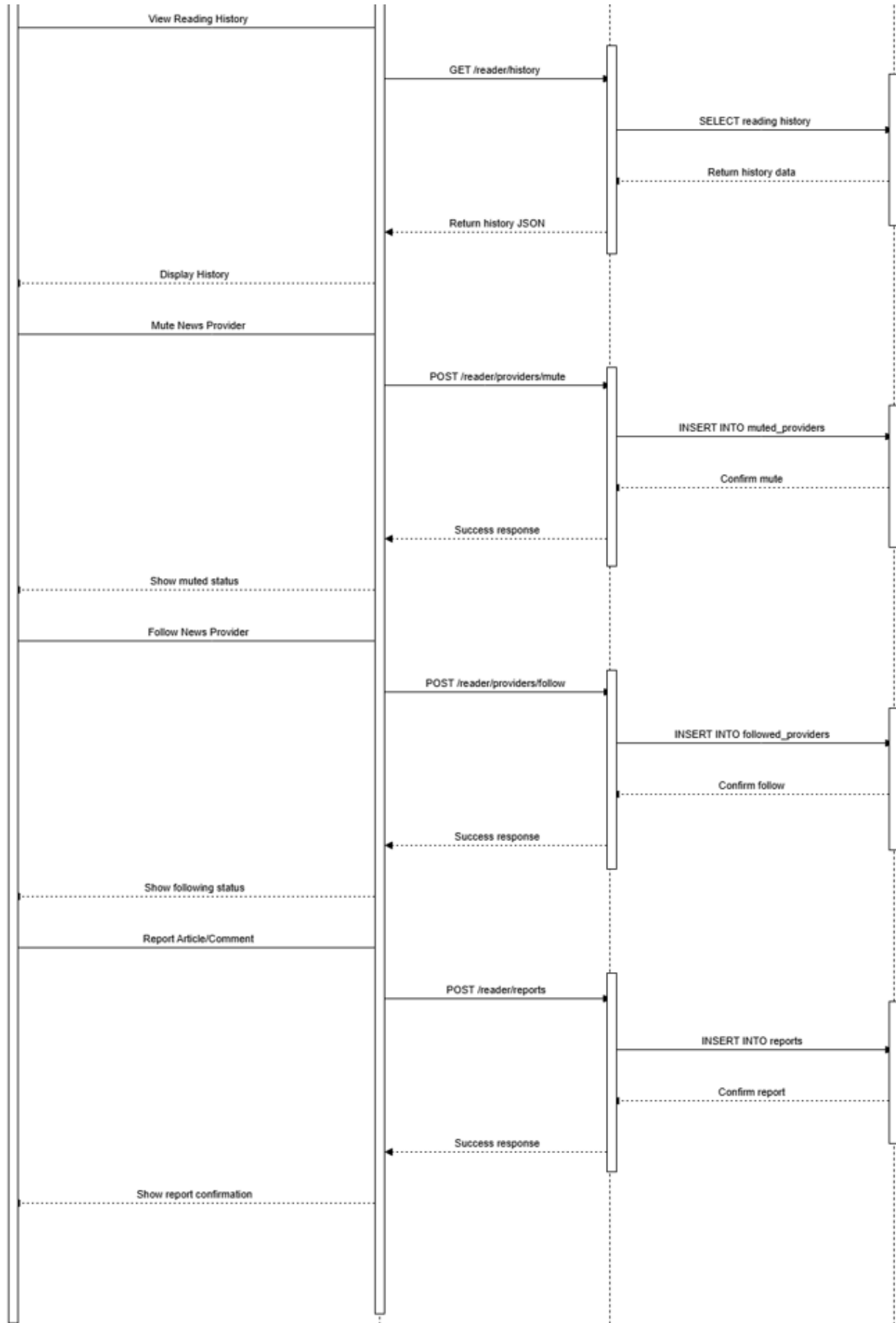


## Sprint 3

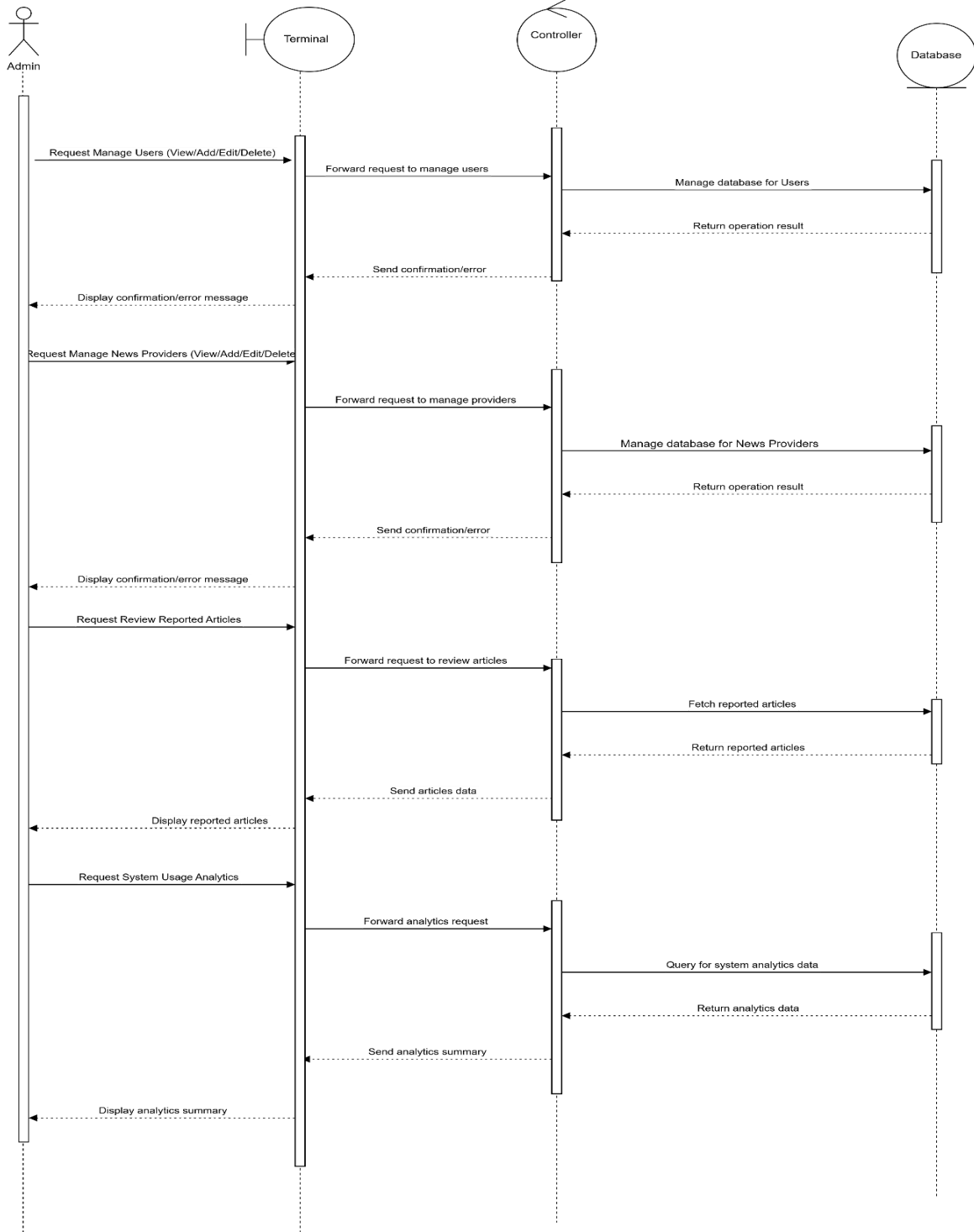


## Sprint 4

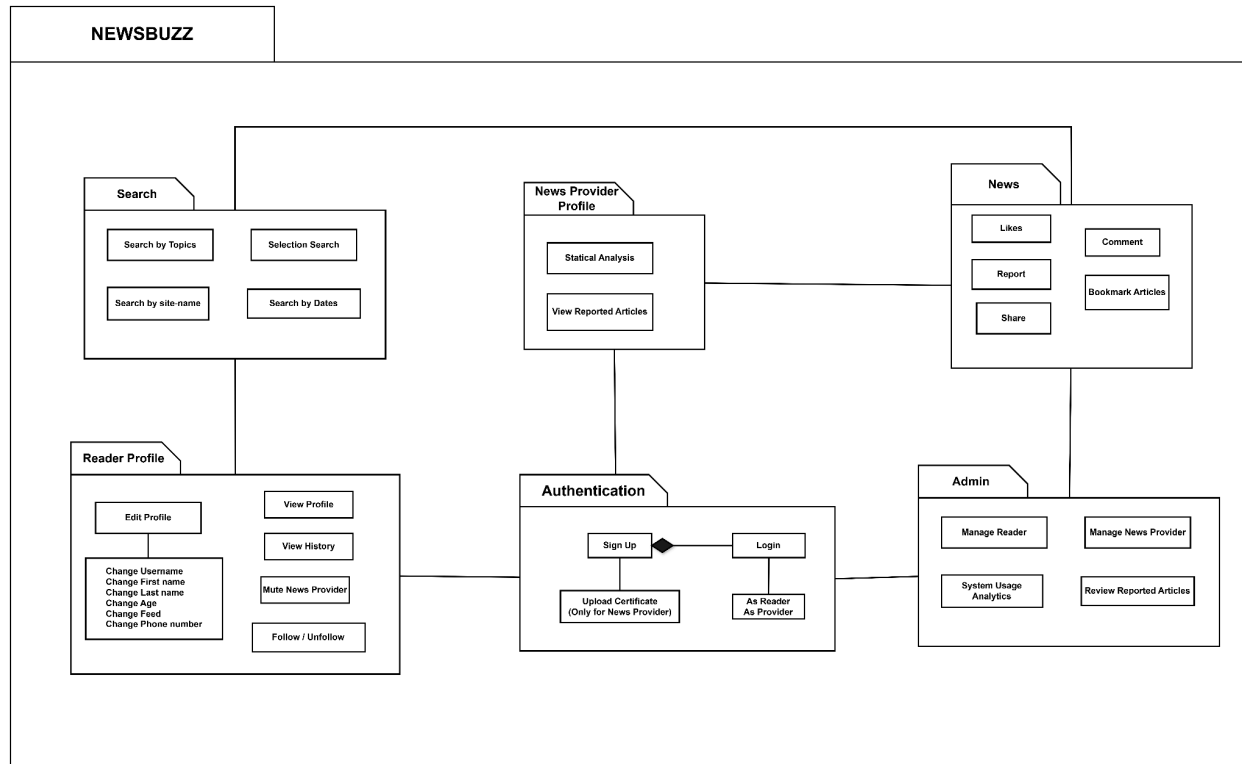




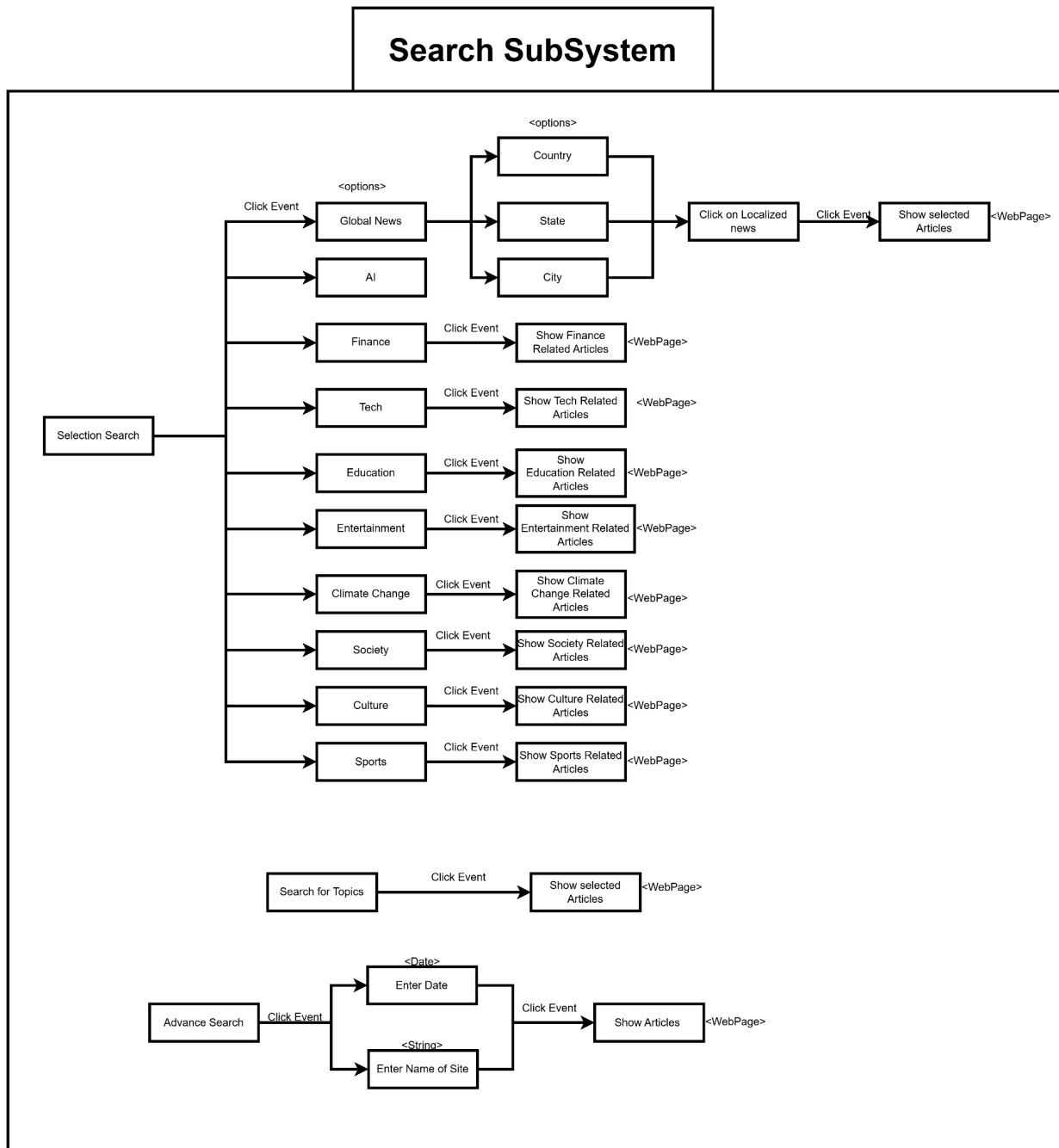
## Sprint 5



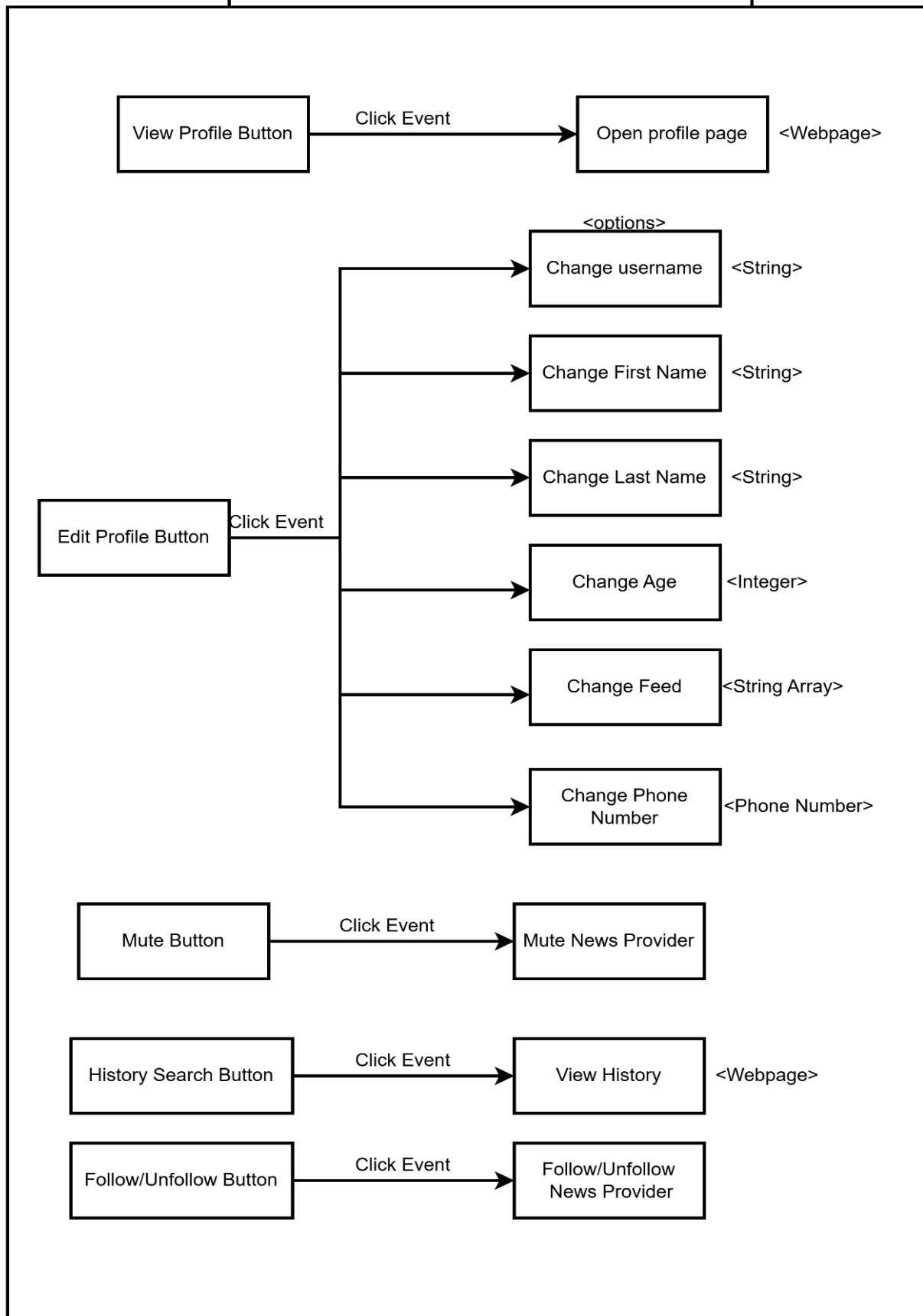
## 5.4 System Design



## 5.5 Object Design



## Reader Profile SubSystem





## 6. Technology Used

The Platform is designed to make user interactions and social connections easy and smooth. It is built using the MERN stack: MongoDB, React.js, and Node.js for a full-stack application.

The frontend, created with React.js, provides a dynamic and interactive user experience.

The backend, powered by Node.js and MongoDB, ensures strong performance and smooth data handling.

Visual Studio Code (VSCode) is used as the main development tool. Its features, like debugging and version control, help speed up the development process. GitHub is used for version control, making team collaboration easier.

## 7. References

- 1.GeeksforGeeks:<https://www.geeksforgeeks.org/web-development/>
- 2.ChatGPT:<https://chatgpt.com/>