

# CSCI 370 Project Proposal

Name: Ethan Posner

Date: March 17th, 2023

## Introduction

The goal of this project will be to create an application and a supporting database to display news articles. It will include the ability for users to login and view articles. Users will interact with the application through a command-line interface.

They will be able to view recent articles, view articles by category, and view them based on tags. Each article should show what category and tags it has.

Users should also have the ability to comment on posts, and other users should be able to see those comments.

Users of this application include anyone interested in keeping up with new knowledge, or in learning new things in general.

## Data Description

### Semantic Schema

To support all application functionality, the database will be require storing the following data:

- Users: Login information. Users also have roles. For now, they will only be either “user” or “admin”.
- Articles: Title, content, tags, categories, number of views, and who viewed it,

- Comments: Who commented, what article the comment was made on, and what the comment contains.
- Article tags: For example, articles talking about technological breakthroughs in nuclear fusion energy production might all be tagged “Nuclear Fusion Energy”.
- Article Categories: For example, articles about political events might all be put under the category of “Politics”.

## Schema

```
UserRoles(roleName, description)
Users(userID, username, password, registerDate, roleName)
Articles(articleID, title, author, publishDate, content)
Comments(commentID, articleID, userID, commentDate, content)
Categories(catName, description)
Tags(tagID, tagName, catName)
ArticleTags(articleID, tagID)
ArticleViews(articleID, userID, viewedAt)
```

where

- In Users, roleName references UserRoles
- In Comments, articleID references Articles and userID references Users
- In Tags, catName references Categories
- In ArticleTags, articleID references Articles and tagID references Tags
- In ArticleViews, articleID references Articles and userID references Users

## Application Requirements

Below are the major functionalities that will be supported by this application.

### 1. User Login

Upon entering the application, users will be prompted to enter a username and password. They will be unable to access the application without logging in.

### 2. View articles by date

Upon logging into the application, users should be shown a list of articles. These articles should be displayed in order based on their publication dates, with newer ones being shown first. Articles in this list should display their title, author, publication date, category, tags, and the first few sentences of the actual article.

Users should also be able to select a specific article to view, and once selected, should be shown the entire article's contents. If the article is large, they should be able to scroll down to see the rest of it.

### **3. View articles in specific categories**

After logging in and having been shown the initial display of articles, users should be able to select a specific category of articles to view. Upon selecting a category, they should be shown a list of articles in that category. These should be ordered by their publication date with newer articles being shown first.

### **4. View articles with specific tags**

Similarly to categories, users should be able to view all articles that have a specific tag.

### **5. Comment on articles and see other user comments**

After selecting an article, users should be able to make comments on them. They should also be able to view other user comments at the bottom of the article.

### **6. Admins can view reports for article viewing**

A logged in admin user should be able to view reports for most viewed articles, most popular tags, most popular categories, most active users, etc for a given year. For example, they could generate a report called "most popular categories in the year 2022". If there is time, they should also be able to view metrics for specific articles, categories, and tags. For example, a report of this type might read "most popular articles in the 'Politics' category in 2022".

## **Stretch Goals**

### **1. User Registration**

With the current development plan, users will require the database administrator to setup their accounts. If there is time, it would be more convenient for users to be able to create accounts themselves.

This might also include other complicated components, such as allowing users to reset their passwords with a recovery email. An initial implementation of this feature would be more simple, however.

## **2. Searching for Articles**

An interesting and useful feature would be to allow users to search for specific articles by typing the name, or keywords, into a prompt. This could also include the ability to filter the search results further. For example, to search specifically for articles under the category of “Politics” with the tag “Canada” and make a search for “election results”.