# Web-based Newsgroup (NewW-B)

Constant Marks and Michael Nutt

October 30, 2019

# Project Description

We will create a web based newsgroup platform that will allow users to browse newsgroup articles. The platform will include a user management and authentication system, a topic based subscription service, a ranking and comment system, a browsing interface that prioritizes highly rated articles, and both simple and advanced search tools.

The system will integrate several key features:

### Article Download System

Scripts will pull down an RSS feed from Ars Tehnica, a news and opinion website that produces articles in technology, science, politics, and society. The scripts will parse the RSS feed of recent articles, extract the relevant metadata as attributes, and upload the articles to our database.

### Manage User Account

The user management system will allow users to create and manage user profiles including an initial setup that validates a unique usernames and secure passwords. Users will be able to view and modify some of their information as well as delete their accounts. In addition users can subscribe to and unsubscribe from topic feeds, view their favorite and bookmarked articles, view the comments they have made on articles and view their current rank on the website.

### Browsing Service

The main website application and homepage of the website will be the article browsing interface. This page will display either a default view for users that have not logged in or a customized view for logged in users. The display will show the most recent highly ranked articles by topic. The default view will show all topics and users can navigate to subscribed topics with a tab system.

An article view page will be displayed when an article is selected. Below the article a comment section section will be displayed. Logged in users will be be able to leave comments, upvote or downvote the article, and upvote and downvote other user's comments.

### Filter and Search Service

The system will include a filtering system that will allow a user to filter the view by multiple parameters (publish date, rank, topic, etc.) and search capabilities to perform simple text search and advanced search.

# Implementation

We will implement the backend of or system in NoSQL and JavaScript using MongoDB for our DBMS and Node.js to serve our web application. We selected NoSQL as our DB language so that we could easily scale

the system and because it will easily allow us to inserting new articles and users without the need to populate all attributes and add new attributes as the need arises when we expand the functionality of the newsgroup system.

The front end of our platform will be a web based application created in JavaScript and HTML.

A draft of our NoSQL collection schema is shown below:

```javascript
const UserSchema = new Schema({
    u_id : {type: String, required: true},
    f_name : String,
    l_name : String,
    email : String,
    pw : {type: String, required: true},
    create_date : String,
    rank : Number,
    follows : [Tag.schema],
    favorites : [mongoose.Schema.Types.ObjectId],
    voted_on : [
        {article: mongoose.Schema.Types.ObjectId,
         vote : Number}],
    comments : [mongoose.Schema.Types.ObjectId]
});

const ArticleSchema = new Schema({
    title : String,
    author : String,
    date : String,
    URL : String,
    tags : [Tag.schema],
    rank : Number,
    comments : [Comment.schema]
});

const CommentSchema = new Schema({
    u_id : {
        type: mongoose.Schema.Types.ObjectId,
        required: true
    },
    date : String,
    txt : String,
    rank : Number
});

const TagSchema = new Schema({
    tag:String
});
```

# Testing

The backend testing of our system will be performed with the Mocha test framework. We will create tests for all server side functions including the fundamental CRUD operations as well as the more advanced functions (user creation, article commenting, filter, search, etc.).

The frontend testing of the system will be performed with user alpha testing of all web application features with a detailed check list of all features to test (user creation, article commenting, filter, search, etc.).

Load testing to simulate hundreds or thousands of users will be performed to evaluate the performance of the systems. The specifics will be determined later.