|  |  |
| --- | --- |
| PROJECT ANALYSIS  Week 3 Assignment | ABSTRACT  This document was created for UMUC Course, CMSC 495, and analyzes aspects of the (TNC)  Group 3 Members  Name: Christiano, Andrew  Name: Fernandez, Yrume  Name: Orwick, Brian  Name: Sell, Julia  Class: CMSC 495 - Current Trends and Projects in Computer Science Professor: Dr. Hung Dao  Due: 10 September 2018 |

**Version Control**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Revision # | Date | Name | Descriptions | Contact Info |
| TNC\_0001 | 9/4/2018 | Brian Orwick | Created | Orwick12@outlook.com |
| TNC\_0002 | 9/5/2018 | Yrume Fernandez | Revisions | Yrume.fernandez@gmail.com |
| TNC\_0003 | 9/6/2018 | Andrew Christiano | Revisions | ajchristiano91@gmail.com |

Table of Contents

[1. Outside Systems 4](#_Toc523947450)

[2. Input Data 4](#_Toc523947451)

[3. Output Data 4](#_Toc523947452)

[4. Data Processing 5](#_Toc523947453)

[5. Subsystem Requirements 5](#_Toc523947454)

[6. Data Interface 5](#_Toc523947455)

[7. Potential Risk and Mitigation 5](#_Toc523947456)

[8. Future Enhancements 6](#_Toc523947457)

# Outside Systems

The Trusted News Code (TNC) software connects to various news websites using various python libraries. This software requires internet connection. The following is a list of trusted sources that are hardcoded for TNC to check, and a level 0 Data Flow Diagram (DFD):

* <https://www.foxnews.com>
* <https://www.usatoday.com>
* <https://www.cnn.com>
* <https://www.bbc.com>
* <http://www.apnews.com>



# Input Data

TNC does not require any input data from a user. A user will navigate to the TNC Server Web Page and receive statistical information on articles posted within the list of trusted news sources. TNC takes input data from web pages that are part of a hardcoded list of trusted news websites. This data is then parsed and structured for storage and retrieval using the SQLite Data Base that is part of the system. The retrieval of the data from the news websites will occur asynchronously in separate threads.

# Output Data

The data that has been aggregated by the TNC will presented from the SQLite Data Base (DB). This database will house all the required information to statistically analyze the trustworthiness of posted articles. The following provides information on the proposed structure of the database table used during execution of the TNC:

|  |  |  |  |
| --- | --- | --- | --- |
| NEWS ARTICLE (TABLE NAME) | | | |
| ID (TEXT) | NUMBER (INT) | NAME (TEXT) | COUNT (INT) |
| Web-Page URL | Article Number | Article Name | Article Count |

# Data Processing



# Subsystem Requirements

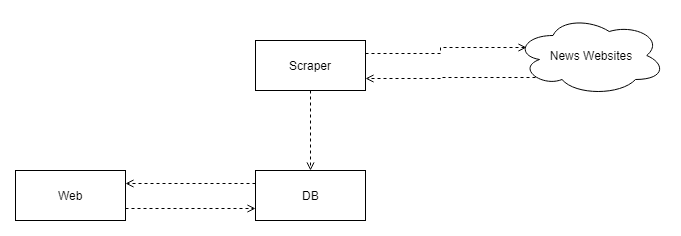
The TNC Software requires connections to local database (SQLite DB) which provides a structure to maintain information for analysis in a persistent manner. This DB also provides the capability of running analysis on articles while the system is offline. It also requires access to local systems networking services to retrieve the required information from external websites while online.

# Data Interfaces

The proposed Data Interfaces are as follows:

1. Web
   1. routes()
   2. queryDB()
2. Scraper
   1. get\_articles()
   2. parse()
   3. download()
   4. generate\_news()
3. DB
   1. create\_table()
   2. db\_insert()
   3. db\_update()
   4. db\_query()
   5. mass\_update()

Below is a flow diagram that shows how the different classes will interact:



# Potential Risk and Mitigation

Potential risk when operating the system are:

1. The system may fail to retrieve information from trusted websites (due to https certificate error, no internet connection, or other unknown circumstances)
   1. Mitigation: System will check for error code 200 to ensure website is available prior to submitting get-request for html.
2. System may fail to provide adequate information if fake-news goes viral and a number of articles are found on the internet based on false information.
   1. Mitigation: System will have a fake-news option that will lower the trustworthiness of the article.

# Future Enhancements

1. Sentiment analysis:
   1. Attempt to show if there is any correlation between rate of occurrence of certain words and their impact on the trustworthiness of an article
2. Search functionality:
   1. Provide a way for users to search through all content to discover the trustworthiness of a specific article, or specifically tagged articles.
3. User interaction:
   1. Provide a way for users to have a dialogue about articles
   2. Provide a way for users to propose additional articles for analysis
   3. Provide a way for users to propose additional websites for automated analysis