

PROJECT REPORT

NEWS TRACKER APPLICATION

1. Introduction

1.1 Project Overview

The main objective of this project is to provide people of all age groups right from a high-schooler, entrepreneur to a businessman, an easy-to-use web application via which those people can access all types of news ranging from entertainment to business and politics in a one single place. Through this web application, anyone, from anywhere in the world, at anytime can gain knowledge of the day-to-day happening of the world and their own surrounding in one single touch of the screen. Moreover, this application is like one click to gain knowledge in and around the world.

1.2 Purpose

The main purpose developing this web application is to eliminate any barrier in accessing or getting to know the day-to-day happenings. This application will provide all the day-to-days updates about various categories such as Entertainment, Sports, Business, Politics etc., in one single place. Hence, saves time and the effort of the user to search for different categories separately, also making this web application an efficient eventually. Also, main purpose of this application terminates any possible information redundancy that may cause while surfing the news.

2. Literature Survey

2.1 Existing Problem

Our research utilized most of the currently available trends and technology to design and develop such an application based on website for user with any type of device be it, Computer, Mobile phone, Tablet etc., Most of the existing News organization's web applications only provide users with their articles are missing out a big opportunity. There are multiple existing news web application used by a single user which are often spammed with a lot of unwanted push notifications and are susceptible to false news.

2.2 References

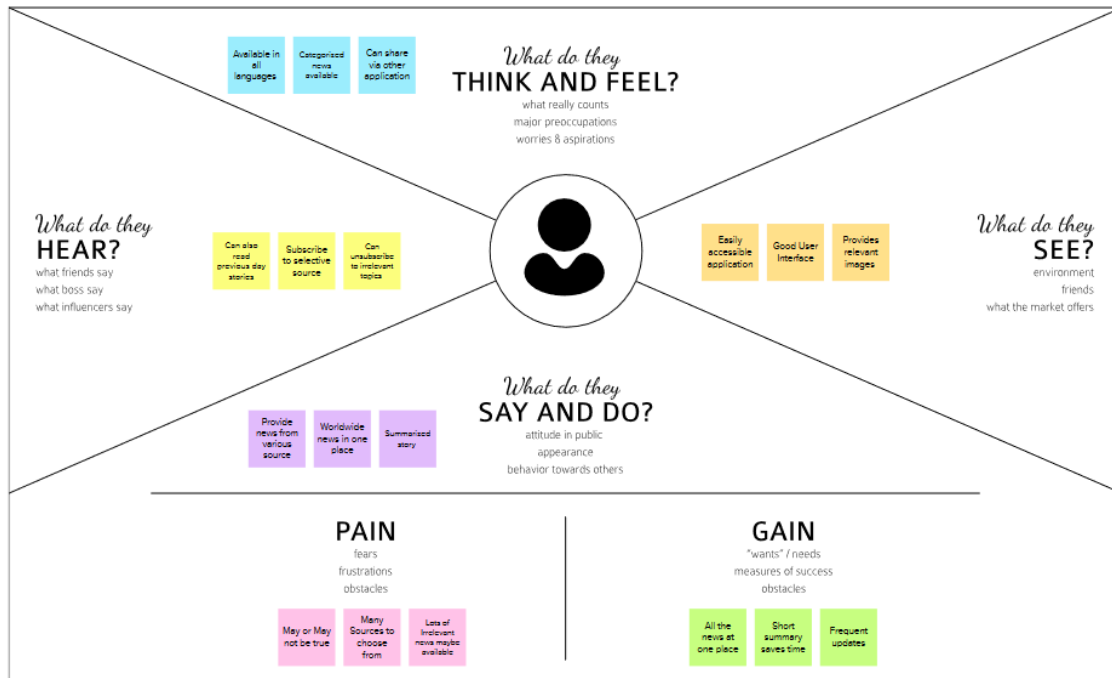
1. Exploring mobile news reading interactions for news app personalisation – Marios Constantinde & John Dowell.
2. New technology, old practices: Examining news websites from a professional perspective - Itai Himelboim & Steve McCreery
3. The future of personalization at news websites: Lessons from a longitudinal study - Neil Thurman & Steve Schifferes
4. Detection and Tracking in News Articles – Sagar Patel, Sanket suthar.
5. Following the fed with a news tracker – Michael william Mc Cracken

2.3 Problem Statement

The user need a way to get relevant and un-fake news based in his own choice of categories so that the user do not have to spend a lot of time searching for the specific category. Also, the news web application now a days are filled with a lots of advertisements and spams which might annoy and irritate a large group of users which in turn affect the user experience. Now-a-days, world is filled with wide range of news categories. The user might not be interested in all the categories, sometimes the user wants to read only a specific category of news based on their own personal choice of interest. The traditional way of tracking news is considered to be slow and obsolete, hence, the user looks for a new innovative kind of application to track news with personal choice of interest.

3. Ideation and Proposed Solution

3.1 Empathy map canvas



3.2 Ideation & Brainstorming



3.3 Proposed Solution

As a team, we worked out that we have got different and unique skill sets and perceived that we would all be able to cater to this problem statement in a very strategic and collaborative manner. Additionally, it is really fascinating that we all had different take on this news web application. A agreed together that a news application should help the user find relevant and important news everyday in an efficient and easy way. Also, we understood that the news published should not be fake and have an authentication that the news is not fake.

3.4 Problem Solution Fit

| | | | | |
|--|--|---|--|---------------------------|
| Define CS, fit into CC | 1. CUSTOMER SEGMENT(S) Who is your customer? i.e. working parents of 0-5 yo. Kids 1. Working class people of different sectors 2. Entrepreneurs 3. Adults of age above 18 4. School students 5. University graduates 6. PhD Scholars 7. Politicians 8. Scientists | 6. CUSTOMER CONSTRAINTS What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices. Budget friendly, Available devices, Time, Quality | 5. AVAILABLE SOLUTIONS Which solutions are available to the customers when they face the problem? Of need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital note-taking Solutions: News gets your message to a wider range of audiences and promotes brand awareness of your business. Being featured at an independent media like News gives credibility to the information. By tracking and maintaining positive news increases positive credibility. Pros: Allowing people access to an unlimited wealth of information on numerous topics ranging from cooking tips to sports trivia. Cons: It is responsible for a lot of other types of misinformation circulating the web. They may find themselves consuming inaccurate news reports or false encyclopedia entries. | Explore AS, differentiate |
| | 2. JOBS-TO-BE-DONE / PROBLEMS Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one, explore different sides. 1. The problem is that news tastes go beyond mere categories and keywords 2. Get to know a lots of facts to keep the citizens aware 3. Create awareness among the public about various issues 4. Provide aids for advertisements 5. Job Vacancies | 9. PROBLEM ROOT CAUSE What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations. 1. In need for knowing the day to day happening and in order to be updated 2. To connect news articles from all around the world and deliver it to user as fast as possible in a best visualized way 3. To increase reach for a product/ certain thing/ Job vacancy advertisement is needed | 7. BEHAVIOUR What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits, indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace) Permits the user to gather the information regarding her/his behaviour and interests and improve their interest to organize the content depending on their requirements For example: Several big publishing organizations use news apps just as a mobile friendly version of their sites Gives time since searching for the particular category is easy | |
| Focus on J&P, tip into BE, understand RC | 3. TRIGGERS What triggers customers to act? i.e. seeing their neighbor installing solar panels, reading about a more efficient solution in the news. 1. The urge to know the day to day happenings than others 2. Readers today want great content in easily digestible, bites. They don't want to have to trawl through a complicated website to get it. | 10. YOUR SOLUTION If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behavior. 1. A separate segment for day to day happenings as hot headlines 2. Advertisement column to display job vacancies, advertisements 3. Display category wise news as of user's wish to save time 4. Create awareness about the various issues prevailing all over the world | 8. CHANNELS of BEHAVIOR 8.1 ONLINE What kind of actions do customers take online? Select online channels from #7 1. Read news online via the user's PC or even using mobile phone 2. Share the read news to their friends and family 3. Subscribe to a particular news provider or a particular Author/ Journalist 8.2 OFFLINE | Identify strong TR & EM |
| | 4. EMOTIONS: BEFORE / AFTER How do customers feel when they face a problem or a job and afterwards? i.e. loss, insecure → confident, in control → use it in your communication strategy & design. 1. Confident that they know the day to day happenings 2. Information about various job vacancies 3. Source for various advertisements 4. Awareness for various issues | | | |

4. Requirement Analysis

4.1 Functional Requirements

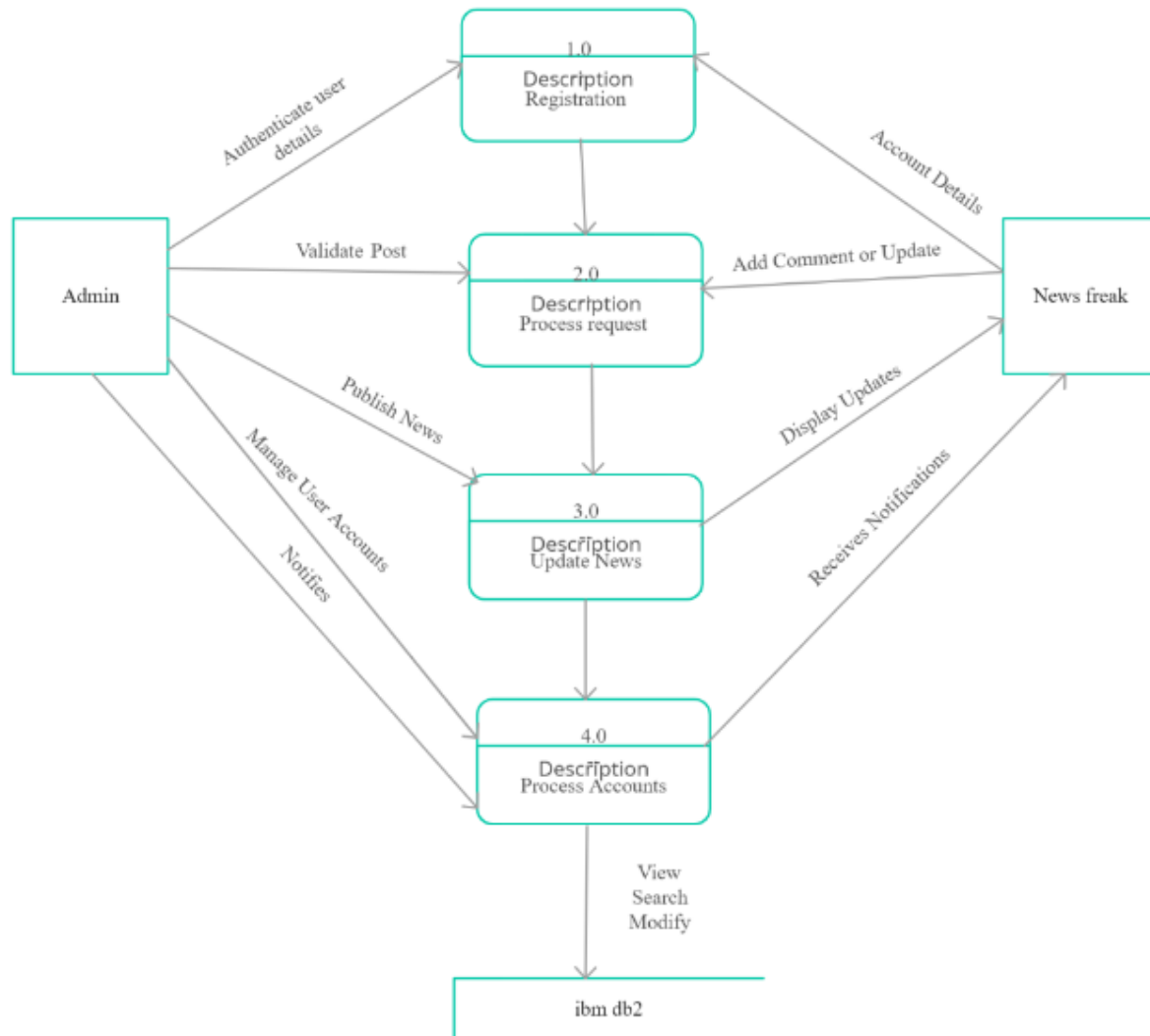
| FR No. | Functional Requirements (FR) | Sub Requirement (Story/ Sun-Task) |
|--------|------------------------------|--|
| FR-1 | User Registration | Registration through Gmail |
| FR-2 | User Confirmation | Confirmation via Google Auth |
| FR-3 | User Login | User Login with username and password Logi with Gmail |
| FR-4 | User Interface | User goes though the filtered news provided by the application |
| FR-5 | User Account Settings | User can update account credentials, update interests and choice of news category and report any issue |

4.2 Non-Funtional Requirements

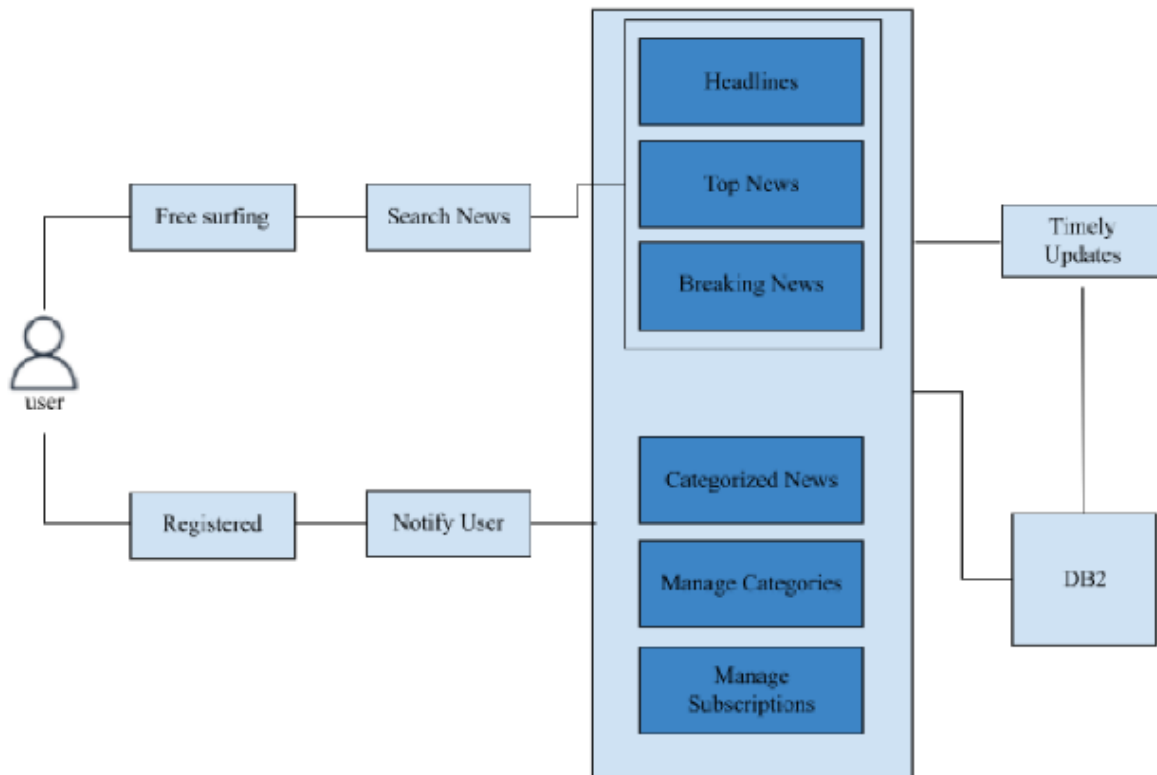
| NFR No. | Non-Functional Requirement | Description |
|----------------|-----------------------------------|--|
| NFR-1 | Usability | Ease of use and user-friendly interface, that allows the user to seamlessly interact with the news updates |
| NFR-2 | Security | Access control, private data processing and external attack risk reduction |
| NFR-3 | Reliability | App behaviour incase of alarm status, Notification Like Automatic restart and operation recovery |
| NFR-4 | Performance | Handles simultaneous users and reduces response time |
| NFR-5 | Availability | Requirements for app continuous running, for example 24/7, minimum idle time, etc., |
| NFR-6 | Scalability | Ways to expand the system and avoid adversely affected performance |

5. Project Design

5.1 Data Flow Diagram



5.2 Solution & Technical Architecture



5.3 User Stories

| User Type | Functional Requirement (Epic) | User Story No. | User Story/ Task | Acceptance criteria | Priority | Release |
|----------------------------------|-------------------------------|----------------|---|------------------------------------|----------|----------|
| Customer (Mobile user/ Web User) | Registration | USN-1 | As a user, I can register for the application via email | I can access my account/ dashboard | High | Sprint-1 |
| | | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email | High | Sprint-1 |

| | | | | | | |
|------------------------|-------------------------|--------|---|--|--------|----------|
| | | USN-3 | As a user, I can register fir the application via Gmail | | Medium | Sprint-1 |
| | Login | USN-4 | As a user, I can log into the application by entering the email and password | | High | Sprint-1 |
| | Dashboard | USN-5 | As a user, I can enter the category and choice of news I want in my dashboard | | High | Sprint-2 |
| | | USN-6 | As a user, I can go throught the feed filtered by categories | | High | Sprint-3 |
| | | USN-7 | As a user, I can logout my account according to my wish | I can click confirm to log out and end the session | Medium | Sprint-3 |
| | Settings | USN-8 | As a user, I can update my category list | | Medium | Sprint-4 |
| Cusomer care executive | Chat Bot/ Query Section | USN-9 | Solves issues brought by the customer | | Medium | Sprint-4 |
| Administrat or | | USN-10 | Roll out updates and bug fixes | | High | Sprint-4 |

6. Project Planning & Scheduling

6.1 Sprint Planning & Estimation

| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Subtotal |
|------------|------------|------------|------------|------------|----------|
| By Design | 5 | 1 | 2 | 1 | 9 |
| Duplicate | 1 | 0 | 0 | 0 | 1 |
| External | 1 | 0 | 0 | 1 | 2 |

| | | | | | |
|-----------------------|----|---|---|---|----|
| Fixed | 7 | 1 | 2 | 2 | 12 |
| Not Reproduced | 0 | 0 | 0 | 0 | 0 |
| Skipped | 0 | 0 | 0 | 0 | 0 |
| Won't Fix | 0 | 0 | 0 | 0 | 0 |
| Totals | 14 | 2 | 4 | 4 | 24 |

6.2 Sprint Delivery Schedule

7. Coding & Solutioning

7.1 Feature 1 - Search Filter

```
@app.route('/search', methods=['POST', 'GET'])
✓ def searchNews():
    searchQuery = request.form['search']
    searchResult = newsapi.get_everything(q=searchQuery)
    ✓ return render_template('search.html', session=session.get('user'),
        searchResult=searchResult, categories=allCategories,
        searchQuery=searchQuery)
```

7.2 Feature 2- Chat-bot

```
window.watsonAssistantChatOptions = {
    IntegrationID: "1713fc15-945b-4ab2-63f66447b180",
    region: "jp-tok",
    serviceInstanceID: "fc27de01-58f6-4706-9655-ec879954c69",
    onLoad: fucntion(instance) { instance. render(); }
};
setTimeput(function(){
    const t=document.createElement('script');
    t.src="http://web-chat.global.assistant.watson.appdomain.cloud/version/"
    + (window.watsonAssistantChatOptions.clientVersion || 'latest' +
    '/WatsonAssistantChatEntry.js');
    document.head.appendChild(t);
});
```

7.3 Category Filter

```
@app.route('/category/business')
def business():
    sources = businessArticles()

    return render_template('business.html', sources = sources)

@app.route('/category/tech')
def tech():
    sources = techArticles()

    return render_template('tech.html', sources = sources)

@app.route('/category/entertainment')
def entertainment():
    sources = entArticles()

    return render_template('entertainment.html', sources = sources)

@app.route('/category/science')
def science():
    sources = scienceArticles()

    return render_template('science.html', sources = sources)

@app.route('/category/sports')
def sports():
    sources = sportArticles()

    return render_template('sport.html', sources = sources)
```

8. Testing

8.1 Test Cases

| Test Case ID | Feature Type | Component | Test Scenario | Pre-Requisite | Steps to Execute | Test Data | Expected Result | Actual Result | Status | Comments | TC for Automation (Y/N) | Bug ID | Executed By |
|---------------------|--------------|---------------|--|---------------|--|---|--|---------------------|--------|----------|-------------------------|--------|--------------------------|
| LoginPage_TC_001 | UI | Login Page | Verify user is able to see the Login/ Signup popup when goes to app's URL | | 1. Enter App's URL and click go 2. Verify Login/ Signup popup displayed or not | | Login/Signup block should display | Working as expected | pass | | | | Swathi Kalyan K, Saranya |
| LoginPage_TC_002 | Functional | Login Page | Verify the UI elements in Login/Signup Block | | 1. Enter App's URL and Click go 2. Verify login/signup popup with below ui elements a) Enter test box b) password test box c) Login button d) New User? Create Account link | | Application should show below elements a) Enter test box b) password test box c) Login button d) New User? Create Account link | Working as expected | pass | | | | Thillai M, Vasundhara S |
| RegisterPage_TC_003 | Functional | Register Page | Verify user is able to register into application with required credentials | | 1. Enter app url 2. Click register button 3. Enter valid username and email 4. Enter valid password 5. Click register button | Username: newsapitesting@gmail.com Password: newsapi@123 | the user registration details should be saved in the database and the user should be navigates to the home section | Working as expected | pass | | | | Swathi Kalyan K, Saranya |
| LoginPage_TC_004 | Functional | Login Page | Verify users able to log into application with valid credentials | | 1. Enter app url and click login page 2. Enter valid username and email 3. Enter valid password 4. Click login button | Username: newsapitesting@gmail.com Password: newsapi@123 | The user should be navigated to the home page where the categories and news is displayed | Working as expected | pass | | | | Thillai M, Vasundhara S |
| LoginPage_TC_005 | Functional | Login Page | Verify user is able to log into application with invalid credentials | | 1. Enter app url and click login page 2. Enter valid username and email 3. Enter valid password 4. Click login button | Username: karan123@gmail.com Password: karan@123 | application should show incorrect email or password validation message to the user | Working as expected | pass | | | | Swathi Kalyan K, Saranya |
| Homepage_TC_006 | UI | Home Page | Verify user is able to see the home page after logging in | | 1. Enter app url to login 2. login in to the page with credentials | Username: newsapitesting@gmail.com Password: newsapi@123 | application should show the home page and its elements | Working as expected | pass | | | | Thillai M, Vasundhara S |
| HomePage_TC_007 | Functional | Home Page | Verify user is able to see the news and categories | | 1. enter homepage 2. traverse through the categories | Username: newsapitesting@gmail.com Password: newsapi@123 | Application should show the news and categories to the user | Working as expected | pass | | | | Swathi Kalyan K, Saranya |
| HomePage_TC_008 | Functional | Home Page | Verify user is able to see all categories of news | | 1. Enter homepage 2. Traverse through the categories | Categories: Business, Technology, Entertainment, Science, Art, Sports, Health | Application should show different categories of news to the user | Working as expected | pass | | | | Thillai M, Vasundhara S |
| HomePage_TC_009 | Functional | Home Page | Verify user is able to search news with keywords | | 1. Enter the home page 2. Click search bar and enter search query | Keyword - Quarter World Cup, Elon musk, Kanye West | Application should display the news related to the keywords | Working as expected | pass | | | | Swathi Kalyan K, Saranya |
| HomePage_TC_010 | Functional | Home Page | Verify user is able to logout from the application | | 1. Enter the homepage 2. Click logout | Logout | Application should be redirected to the news page but the account should be logged out | Working as expected | pass | | | | Thillai M, Vasundhara S |

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the News Tracker Application project at the times of the release to User Acceptance Testing (UAT)

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Subtotal |
|-----------------------|-------------------|-------------------|-------------------|-------------------|-----------------|
| By Design | 5 | 1 | 2 | 1 | 9 |
| Duplicate | 1 | 0 | 0 | 0 | 1 |
| External | 1 | 0 | 0 | 1 | 2 |
| Fixed | 7 | 1 | 2 | 2 | 12 |
| Not Reproduced | 0 | 0 | 0 | 0 | 0 |
| Skipped | 0 | 0 | 0 | 0 | 0 |
| Won't Fix | 0 | 0 | 0 | 0 | 0 |
| Totals | 14 | 2 | 4 | 4 | 24 |

3. Test Case Analysis

This report shows the number of test cases that have passed, failed and untested

| Section | Total Cases | Not Tested | Fall | Pass |
|--------------------------|--------------------|-------------------|-------------|-------------|
| Login Page | 4 | 0 | 0 | 4 |
| Registration Page | 1 | 0 | 0 | 1 |
| Home Page | 5 | 0 | 0 | 5 |

9. Advantage

1. Viewers can get their news straight off their smartphone or tablet computer.
2. News is at their fingertips in an instant. An online newspaper can be read more elaborate than a printed newspaper.
3. You can read the old issues too very easily at the click of the mouse.
4. This will help the users to share news on various platforms such as Twitter and Facebook. This will not only give an amazing user experience

Disadvantage

1. It can be limited by time.
2. It may rely too heavily on personalities, emotions, opinions... not facts.
3. It can shortchange complex stories or avoid them altogether.

10. Conclusion

The Motivation and scope behind this project are to connect people through this application and provide a medium to share their views on the topic/news/information. Then, People with the same interest can interact with each other. However, they can even share more information on the topic. This app while cross-checks the redundancy in the information along with the false and misleading information, which later results in panic in the people.

11. Future Scope

Location feature with automation can be implemented which means as user move from one city to other local news will change as per it. Offline Reading can be improve will more efficient way on full articles. Data quality check needed. If API can't reach to certain article source it gives null value which can cause problem in JSON Parsing.

12. Appendix

Source Code

Configurations

class Config:

```
    NEWS_BASE_URL_SOURCES =  
'https://newsapi.org/v2/top-headlines/sources?apiKey={}'  
    NEWS_BASE_EVERYTHING_URL =  
'https://newsapi.org/v2/everything?domains={}&apiKey={}'  
    NEWS_BASE_HEADLINES_URL =  
'https://newsapi.org/v2/top-headlines?country=us&apiKey={}'  
    NEWS_BASE_SOURCE =  
'https://newsapi.org/v2/top-headlines?sources={}&apiKey={}'  
    API_KEY = "f1a683b7df544ace8de3d9ce54790eb1"
```

```
class ProdConfig(Config):  
    pass
```

```
class DevConfig(Config):  
    DEBUG = True
```

```
config_options= {  
    'development': DevConfig,  
    'production': ProdConfig  
}
```

Routes

```
from app import app  
from flask import render_template  
from .request import businessArticles, entArticles, get_news_source, healthArticles,  
publishedArticles, randomArticles, scienceArticles, sportArticles, techArticles, topHeadlines
```

```
@app.route('/')  
def home():  
    articles = publishedArticles()  
  
    return render_template('home.html', articles = articles)
```

```
@app.route('/headlines')  
def headlines():  
    headlines = topHeadlines()  
  
    return render_template('headlines.html', headlines = headlines)
```

```
@app.route('/articles')  
def articles():  
    random = randomArticles()  
  
    return render_template('articles.html', random = random)
```

```
@app.route('/sources')  
def sources():  
    newsSource = get_news_source()
```

```
        return render_template('sources.html', newsSource = newsSource)

@app.route('/category/business')
def business():
    sources = businessArticles()

    return render_template('business.html', sources = sources)

@app.route('/category/tech')
def tech():
    sources = techArticles()

    return render_template('tech.html', sources = sources)

@app.route('/category/entertainment')
def entertainment():
    sources = entArticles()

    return render_template('entertainment.html', sources = sources)

@app.route('/category/science')
def science():
    sources = scienceArticles()

    return render_template('science.html', sources = sources)

@app.route('/category/sports')
def sports():
    sources = sportArticles()

    return render_template('sport.html', sources = sources)

@app.route('/category/health')
def health():
    sources = healthArticles()

    return render_template('health.html', sources = sources)
```