

<b>Module code and title:</b>	Level 5 Mobile (CO566, CO555 CO508)
<b>Assignment No. and type:</b>	Coursework 1 (CW1): Design and implement an Intelligent System with Report
<b>Student Name and ID:</b>	Nerizza Marie Flores (2201268)

## Contents

1.1 Introduction .....	3
1.2 Requirements Gathering .....	3
1.3 User Story .....	3
1.3.1 Browse News .....	3
1.3.2 Read Whole Article .....	3
1.3.3 Choose Category .....	3
1.3.4 Read About App .....	4
1.4 System Requirements .....	4
1.4.1 Homepage .....	4
1.4.2 Categories .....	4
1.4.3 Load News .....	4
1.4.4 Read More Link .....	4
1.4.5 About App .....	4
1.4.6 Usability .....	4
1.6 Target platform .....	4
1.7 Target audience .....	5
2.1 Design .....	5
2.2 Use case Diagram .....	5
2.3 Wireframe .....	5
2.3.1 Wireframe User Evaluation .....	6
2.3.2 Changes to the Wireframe .....	7
2.4 Usability of the design .....	7
3.1 Implementation .....	8
Result of Hamburger menu .....	8
Result of Form 1 .....	9
Result of Form 2 .....	10
Result of Form 3 .....	11
Result of Form 4 .....	12
Result of Form 5 .....	13
3.3 Future Improvements .....	13
3.4 App video demonstration .....	13
4.1 Testing .....	14
4.2 Test Cases .....	14
5.1 Publishing .....	19
References .....	20

## 1.1 Introduction

As a cybersecurity student, we were expected to always be updated to the latest news on cybersecurity world. It will be helpful for us, students, to understand more about the current threats and vulnerabilities around technology and the internet which can help us analyse more on how to prevent/mitigate the threats. This has bought an inspiration of developing a cybersecurity news app. The Cybersecurity News App was developed to bring the latest news from leading experts and news outlets about the world of cybersecurity, threats, exploits, hacking, malware and many more.

There has been similar app that was produced for the cybersecurity news, such as Cybersecurity News & alerts. However, to use the app, a user must be a paid subscriber. (Cyber Security News & Alerts, 2022). Thus, the Cybersecurity News App Project will be free to use.

## 1.2 Requirements Gathering

Firstly, the user story (section 1.3) was developed to analyse and define the requirements based on the users' action on the app. The user story helped formulate the system requirements (section 1.4). Additionally, the user story includes priority level and effort points which is used to prioritise the task as there are different levels of prioritisation on each task. (Cohn, 2022)

In the case of Cybersecurity News App, the most prioritised task is the Browse News (section 1.3.1) and Read Whole Article (section 1.4.2) as these tasks would take a great amount of time and effort to do. While the lowest priority would be the Read about app (section 1.3.4) which means that this functionality could be developed in a short period of time.

The User story was also used to create the Use Case Diagram (section 2.2). The diagrammatic notation makes it easier for the developer and users to see the relationship and constraints of each functionality.

### 1.3 User Story

1.3.1 Browse News	
As a user, I can browse the cybersecurity news from different various news outlet.	
High	13 points

1.3.2 Read Whole Article	
As a user, after browsing for the news, I should be able to read the whole article of the news that interest me.	
High	18 points

1.3.3 Choose Category	
As a user, I can choose between different categories of the news and browse the news based on the chosen category.	

<b>Mid</b>	<b>8 points</b>
------------	-----------------

1.3.4 Read About App	
As a user, I can access the about page of the app to let me know what to do when I do not know how to navigate the app.	
<b>Low</b>	<b>3 points</b>

## 1.4 System Requirements

### 1.4.1 Homepage

This page would show the hottest news on cybersecurity from various leading news outlets all around the world.

### 1.4.2 Categories

The news app would have a category section that will be placed on a menu for a quick access. The categories would be based on cybersecurity threats such as malware, ransomware, and hacking.

### 1.4.3 Load News

The news app would be able to load a list of news based on the category from various news sources. The news app would integrate a News API (<https://newsapi.org/>) for searching and retrieving live articles from all over the web.

### 1.4.4 Read More Link

The news app would incorporate a read more button on each news article which will be linked to the news source where users would be able to read the full article once the button is clicked.

### 1.4.5 About App

The news app would include an about page where it would include a short introduction about the app and instruction on how to use the app.

### 1.4.6 Usability

The news app interface would be user-friendly and easy to use.

## 1.6 Target platform

Android and IOS for tablets and phones

## 1.7 Target audience

Target audience would be students and people that are interested in Cybersecurity news and students and professionals that are currently in the field of computing or/and cybersecurity.

## 2.1 Design

Diagrammatic notation was used to identify functions and non-functions relationship to the news app. Additionally, it would also help the users and the developer to understand any early stage of setbacks that may affect the functionality of the app. For instance, the wireframe (section 2.3) is used to determine whether the user interface design is appropriate with the target audience, as if the UI is too complex then the usability and reliability of the app will lessen.

## 2.2 Use case Diagram

The use case diagram (Figure 2.1) is used to illustrate the high-level function and the interaction between the functions within the News App based on the user story (section 1.3). It also describes how actors use the functionality. In the case of the News App, there is only two actor which is the user of the app and the News API actor which would act as a bridge to the other news outlet. The figure 2.1 shows which functionality is associated with the actor. The extended relation between two use cases just shows that other use cases can be triggered conditionally. In the case of the News App, the Browse News use case is extended to the Read Article as the User can browse News but may not want to read the whole article. In addition to that, the include relation between two cases just shows that the other use case is mandatory for the other use case to work. In the News App case, it would be that the Browse news include the Load News Use case as the user would not be able to browse news if the Load news is not used.

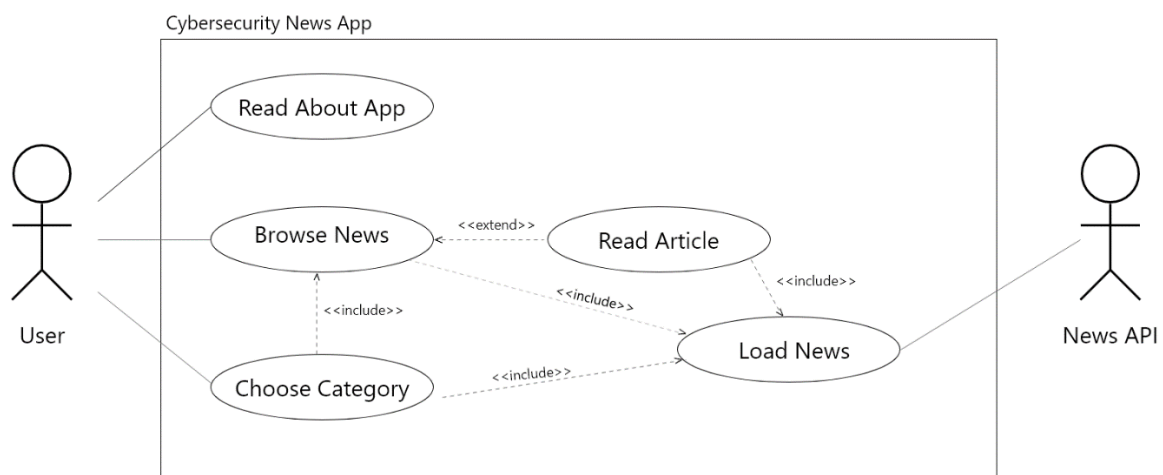


Figure 2.1 Use Case Diagram

## 2.3 Wireframe

The wireframe (Figure 2.2) illustrates the interface design of the News App. It illustrates the basic elements of each page of the News App. The News App wireframe was created with the use of Adobe XD.

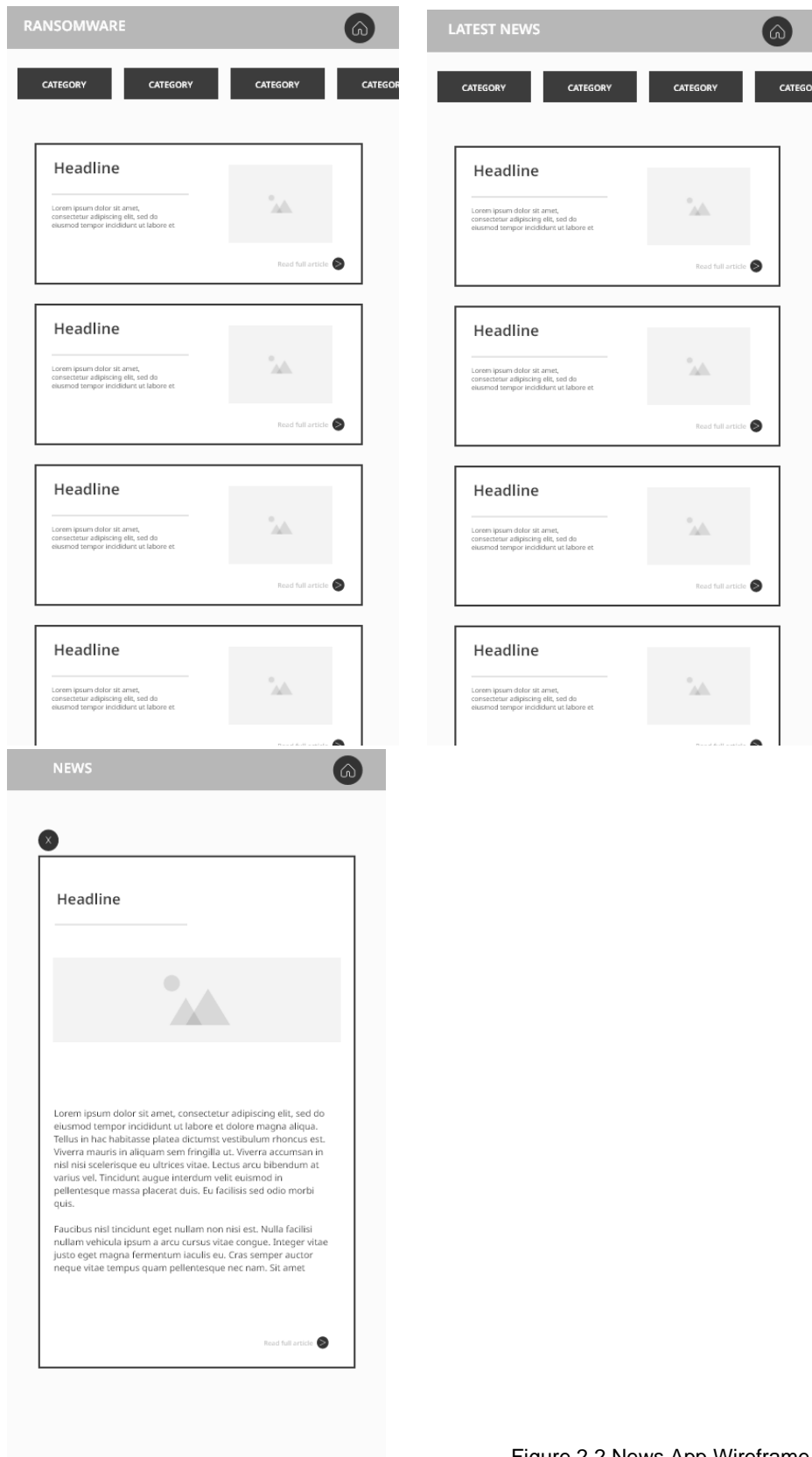


Figure 2.2 News App Wireframe 1

### 2.3.1 Wireframe User Evaluation

A survey was conducted to evaluate the News App Wireframe. The survey was conducted by the developer of the News App. Two types of users have answered the survey. User 1 is a university student

that is taking a graphic design course. User 2 is an adult, working in the medicine field. The two users have different technical knowledge.

#### User 1 Feedback:

*“Good: the overall design was simple and is pleasing to look at. It is minimalist and still looks sophisticated. Bad: the spacing is a little too big in the boxes which reduces the possible space that could be allotted for the content or summary of the article. Also, the small boxes stand out more than the content, which should be the other way around.”*

#### User 2 Feedback:

*“I like the consistency of the design. It looks easy to use and navigate. However, the categories placed after the header looks a bit cluttered and the space looks too tight.”*

### 2.3.2 Changes to the Wireframe

It is important to take all the suggestions of the users. As the user 1 and 2 feedback shows dissatisfaction around the category section and the spacing, The placement of the category was changed thus the category would now be placed on a hamburger menu thus the news section would be more focused.

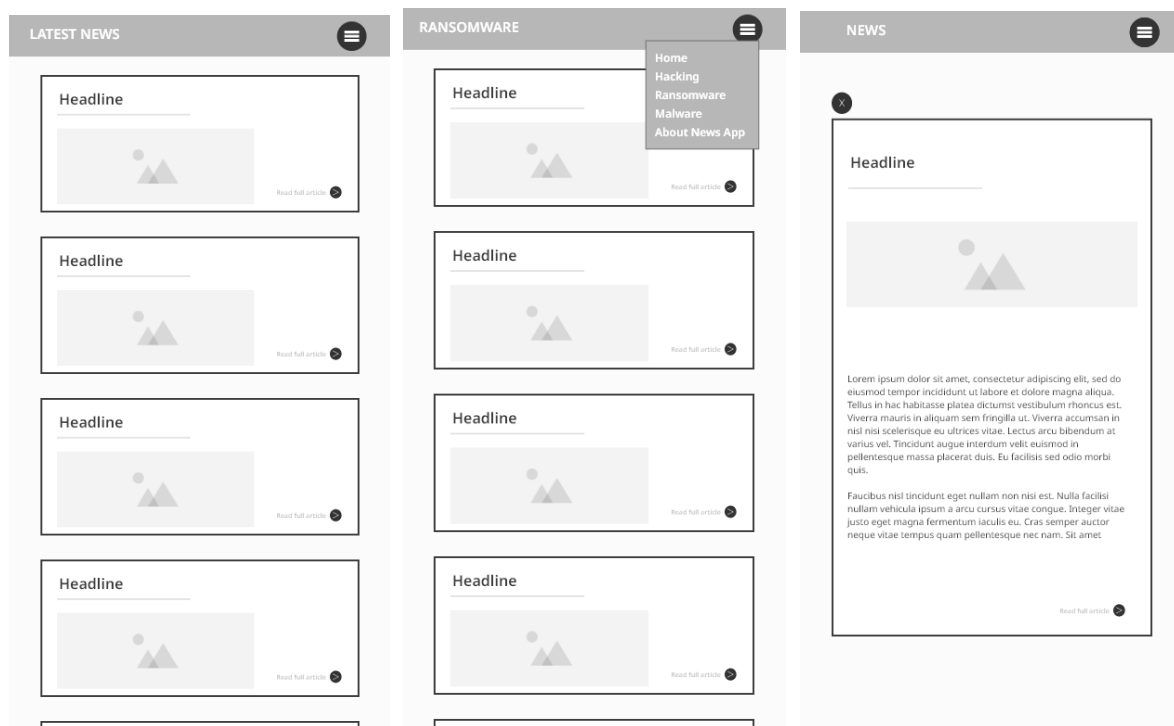


Figure 2.3 News App Wireframe 2

## 2.4 Usability of the design

The Heuristics Theory is applied on the News App user interface design as a guideline for the usability of the user Interface Design. According to the Heuristics Theory (1994), *“#8 Aesthetic and minimalist*

design, the interfaces should not contain information which is irrelevant or rarely needed". Applying this theory to the News App, the app is solely focused on cybersecurity news. Thus, rather than showing all general news, it would only output relevant news about cybersecurity. Additionally, the news app included categories such as ransomware and malware in the hamburger menu thus it would be easier for the user to distinguish what news they want. Furthermore, for the user to focus on the news article, the colours to be used in the design would be a simple different shade of blue.

### 3.1 Implementation

The Cybersecurity News App was developed in NSB App Studio. The NSB App Studio uses a mix of a drag and drop design that uses the latest framework of bootstrap and CSS and programming in JavaScript. (Wiki.appstudio.dev, 2022) The News App uses elements of the header and hamburger menu which was dragged and dropped in Form 1. The label within the header was edited to show the name of the app. In the right corner of the header is the hamburger menu which is a ready-made element in app studio. The selection within the hamburger menu is set within the code. The selection states that once the item from the menu is selected, it will be changed based on the form that was stated. The header was also declared as global in Form 1. This would retain the elements and functionality of the header and the hamburger menu in different page/forms.



```
1 -Hamburger1.onclick=function(s){
2   if (typeof(s) == "object") {
3     return;
4   }
5   if (s === "Home") {
6     ChangeForm(Form1);
7   }
8   if (s === "Hacking") {
9     ChangeForm(Form2);
10  }
11  if (s === "Malware") {
12    ChangeForm(Form3);
13  }
14  if (s === "Ransomware") {
15    ChangeForm(Form4);
16  }
17  if (s === "About CSN") {
18    ChangeForm(Form5);
19  }
20 };
21
22 -Form1.onshow=function(){
23   NSBPage.appendChild(HeaderGlobal)
24 };
25
```

Result of Hamburger menu





The News API was integrated in form 1 to fetch the news from various news outlet and generate news based on the category given. The form 1 is fetching a list of news articles of cybersecurity category which was received from the get request that was made to the news API using the fetch method. The title, author and imageURL of the article is inserted inside the HTML Page using innerHTML() method. The container that was placed in the form 1 as a placeholder of the news article is selected using document.getElementById ().

```
let news_data = null
let getNewsData = async () => {

  await fetch('https://newsapi.org/v2/everything?language=en&q=cybersecurity&sortBy=publishedAt&apiKey=8b7984a')
  .then(res=> res.json())
  .then(res=>{
    console.log(res)
    console.log(res.articles)
    news_data = res.articles
  })
  .catch(err=>{
    console.log(err)
  })
  return news_data
}

let fetchNews = async()=>{
  let data = await getNewsData()
  for (let i = 0; i < data.length; i++) {
    const n1 = document.getElementById('Container1')
    const newsSegment = `<div class = "jumbotron">
      <h4> ${data[i].title}</h4>
      <p>by: ${data[i].author}</p>
      <img src=${data[i].urlToImage} /><br>
      <input type=button class="btn btn-primary" onClick="parent.open('${data[i].url}')" value=
      </div><br>`;
    n1.innerHTML += newsSegment
  }
}
fetchNews()
```

Result of Form 1



The form 2 has a similar functionality as the script on Form 1. The only difference is that the Form 2 is fetching articles based on different genres which is hacking news.

```
Form1 Form2 x Form3 Form4 Form5 News_App_FINAL +
X {}
1 let newsdetails = null
2 -const fetchNewsData = async () => {
3
4     await fetch('https://newsapi.org/v2/everything?q=hacking&sortBy=popularity&apiKey=8b7984aeb82f431a82b1ae58abc1
5     .then(res=> res.json())
6     .then(res=>{
7         console.log(res)
8         console.log(res.articles)
9         newsdetails = res.articles
10    })
11    .catch(err=>{
12        console.log(err)
13    })
14    return newsdetails
15  }
16
17 -const displayNews = async()=>{
18     let newsDataArr = await fetchNewsData()
19     for (let i = 0; i < 38; i++) {
20     const n2 = document.getElementById('Container2')
21     const newsQuery = `<div class = "jumbotron">
22         <h4> ${newsDataArr[i].title}</h4>
23         <p>by: ${newsDataArr[i].author}</p>
24         <img src=${newsDataArr[i].urlToImage} /><br>
25         <input type=button class="btn btn-primary" onClick="parent.open('${newsDataArr[i].url}')"
26         </div><br>`;
27     n2.innerHTML += newsQuery
28   }
29 }
30
31 displayNews()
```

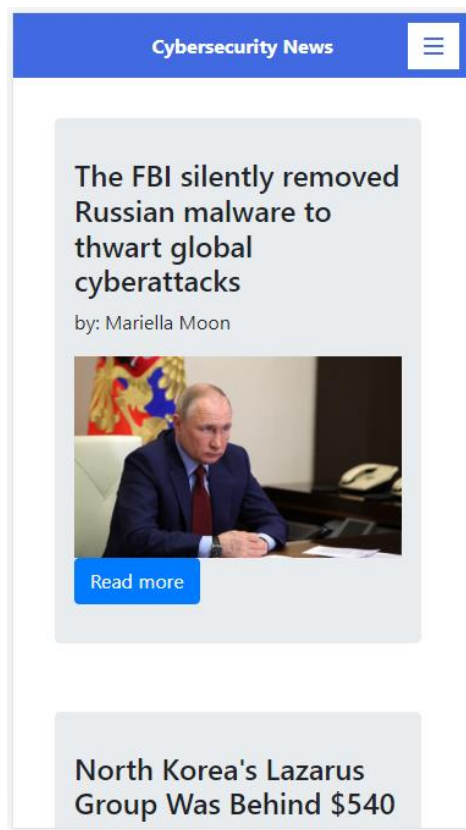
Result of Form 2



The form 3 below has a similar functionality as the script on Form 1. The only difference is that the Form 3 is fetching articles based on different genres which is malware news.

```
Form1 Form2 Form3 Form4 Form5 News_App_FINAL +
{ }
1 let newsdets = null
2 -const fetchNewsD = async () => {
3
4     await fetch('https://newsapi.org/v2/everything?q=malware&sortBy=popularity&apiKey=8b7984aeb82f431a82b1ae58')
5     .then(res=> res.json())
6     .then(res=>{
7         console.log(res)
8         console.log(res.articles)
9         newsdets = res.articles
10    })
11 }
12 .catch(err=>{
13     console.log(err)
14 })
15 return newsdets
16 }
17
18 -const displayNewsData = async()=>{
19     let newsData = await fetchNewsD()
20     for (let i = 0; i < 30; i++) {
21         const n3 = document.getElementById('Container3')
22         const newsSign = `<div class = "jumbotron">
23             <h4> ${newsData[i].title}</h4>
24             <p>by: ${newsData[i].author}</p>
25             <img src=${newsData[i].urlToImage} /><br>
26             <input type=button class="btn btn-primary" onClick="parent.open('${newsData[i].url}')"
27             </div><br>`;
28         n3.innerHTML += newsSign
29     }
30 }
31 displayNewsData()
```

Result of Form 3



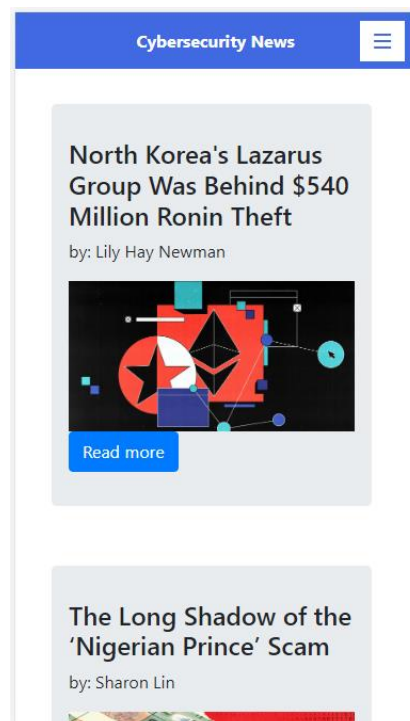
The Form 4 has a similar functionality as the script on Form 1. The only difference is that the Form 4 is fetching articles based on different genres which is ransomware news.

```

1 let newSd = null
2 -const fetchN = async () => {
3
4     await fetch('https://newsapi.org/v2/everything?q=ransomware&sortBy=popularity&apiKey=Bb7984aeb82f431a82b1ae58a
5     .then(res=> res.json())
6     .then(res=>{
7         console.log(res)
8         console.log(res.articles)
9         newSd = res.articles
10    })
11    .catch(err=>{
12        console.log(err)
13    })
14    return newSd
15 }
16
17 -const displayN = async()=>{
18     let newsData = await fetchN()
19     for (let i = 0; i < 10; i++) {
20         const n4 = document.getElementById('Container4')
21         const newsArt1 = `<div class = "juehotron">
22             <h4> ${newsData[i].title}</h4>
23             <p>by: ${newsData[i].author}</p>
24             <img src=${newsData[i].urlToImage} /><br>
25             <input type=button class="btn btn-primary" onClick="parent.open('${newsData[i].url}')" va
26             </div><br>";
27         n4.innerHTML += newsArt1
28     }
29     displayN()
30 }

```

Result of Form 4



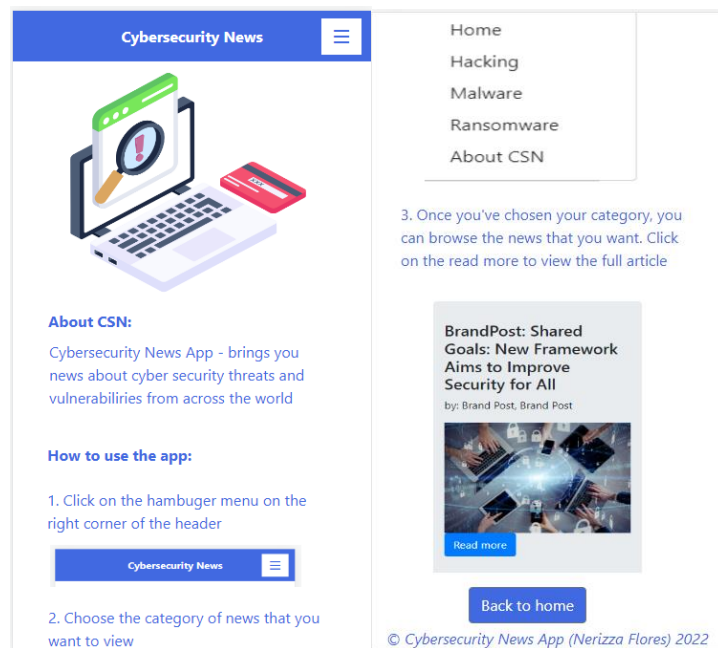
The Form 5 displays the about page of the app where it has a short introduction of the app and instruction on how to use the app. The script below adds a functionality in a button within the form 5.

```

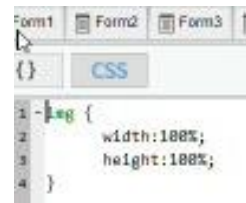
1
2 -Button1.onClick=function(){
3     ChangeForm(Form1);
4 }
5

```

## Result of Form 5



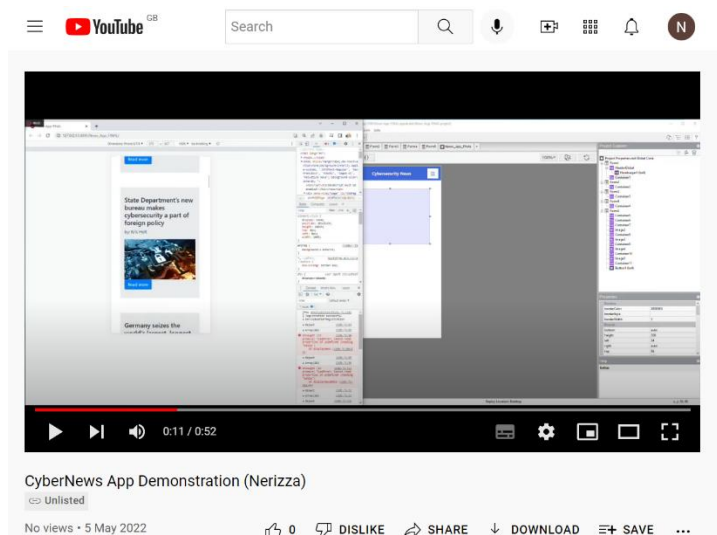
Added a simple script in the CSS to change the style of the image that was shown in the news article thus it would be resize appropriate to the mobile size.



### 3.3 Future Improvements

In the future development, the News app could include bookmark option where user can save the news that they want to reread. It can be implemented with the use of the JavaScript local storage where it can store user input in their local device which in this case would be the bookmark that they want/store. Although there are only limited amount of storage using the JavaScript local storage. (LePage, 2022) Additionally, the app could include a lot more categories.

### 3.4 App video demonstration



Github Link for the code:

<https://github.com/BNU-Comp-Nerizza/MSS-News>

Youtube Link for the demonstration:

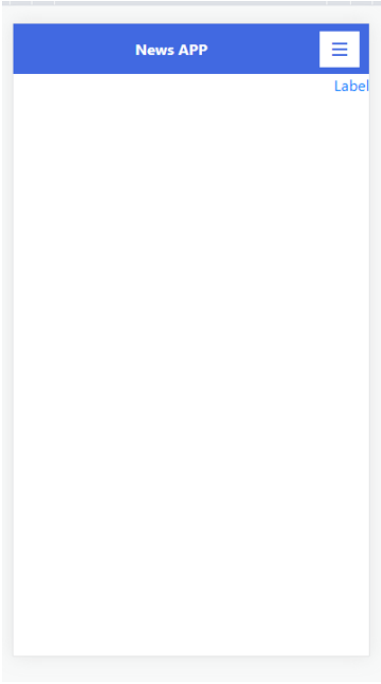
<https://youtu.be/7r-O89fyihk>


## 4.1 Testing

While developing the News App, there were some errors that were encountered. Test case (section 4.2) was made to monitor the error and how the solution was made which allows the developer to ensure that the app would satisfy the system requirements. (Brush, 2022)

### 4.2 Test Cases


Test No.	Test Description	Expected Result	Actual Result	Pass/Error	Solution Taken	Evidence (image)
1	Switching Form 1 to Form 2	The Form 1 would be switched to the Form 2	The page stayed on the Form 1	Error		
2	Switching Form 1 to Form 2	The Form 1 would be switched to the Form 2	The page was switched to Form 2 but the Hamburger menu was not retained	Error		
3	Switching to different page while the Hamburger Menu would be retained	The Form 1 would be switched to Form 2 while all elements and functionality of the hamburger menu would be retained	The page was switched, and the elements and functionality of the hamburger menu was retained	Pass	Did a hard refresh on the application.	

4	Connecting News API to	The Form 1 would load a list of news article including the news headline and image with the use of the News API	The page did not have any changes, no news was shown.	Error		
5	Connecting News API to	The Form 1 would load a list of news article including the news headline and image with the use of the News API	The page only shown one news article with the headline and the image	Error		
6	Connecting News API to	The Form 1 would load a list of news article including the news headline and image with the use of the News API	The page shown a list of news articles with the headline and image. But the image was too big.	Error	Added a loop function to iterate over the news and return the news article	

7	Resizing the news image	The image would be resize appropriately for the mobile resolution	The image was able to fit based on the mobile resolution	Pass	Edit the style of the image in CSS to make the width and height 100%	
8	Editing Form 2 for the new News category for the news (Ransomware News)	The Form 2 page would show a list of news article based on the category (Ransomware News)	The Form 2 page did not show any changes, no news article was shown.	Error		
9	Editing Form 2 for the new News category for the news (Ransomware News)	The Form 2 page would show a list of news article based on the category (Ransomware News)	The Form 2 page only shown the same news article from the Form 1	Error		
10	Editing Form 2	The Form 2 page	The Form 2 could	Error		



	for the new News category for the news (Ransomware News)	would show a list of news article based on the category (Ransomware News)	not connect to News API			
11	Editing Form 2 for the new News category for the news (Ransomware News)	The Form 2 page would show a list of news article based on the category (Ransomware News)	The Form 2 loaded a list of news article based on the category (Ransomware News)	Pass	The news API URL was changed into ransomware category	
12	Editing Form 3 for the new News category for the news (Malware News)	The Form 3 page would show a list of news article based on the category (Malware News)	The Form 3 loaded a list of news based on the category (Malware News)	Pass		
13	Adding Read more button	A read more button would be added to each news article	A read more button was added to each news article	Pass		
14	Check if the read more button links to the news source	Once the read more button is clicked, it would load the actual news source	No changes happened	Error		
15	Check if the read more button links to the news source	Once the read more button is clicked it would load the actual news source	Error came up	Error		
16	Check if the read more links to the news source	Once the read more button is clicked it would load the actual news source	The read more button was linked to the actual news source	Pass	changes to the button onclick added the parent open instead of just using button onclick	

17	Changing the style of the news article	Each news article would be contained in the Jumbotron(bootstrap)	Each news article was contained in the Jumbotron(bootstrap)	Pass		
----	----------------------------------------	------------------------------------------------------------------	-------------------------------------------------------------	------	--	-------------------------------------------------------------------------------------

## 5.1 Publishing

There are different options on how to publish the app through NSB App studio. However, as the news app was only developed through the free version of the NSB there are limited options on how to publish it. Voltserver is the default web server that can be used for testing and publishing the app however, it is exclusive for the subscribed users which in my case, I only use the free version thus I could not use it. (Wiki.appstudio.dev, 2022).

Furthermore, although the news API are free to use, the request outside the localhost is not allowed. To be able to use the news API outside the localhost, the developer, myself, must upgrade the plan. (Documentation - News API, 2022)

## References

2022. *Cyber Security News & Alerts*. Loyal Foundry.

Brush, K., 2022. *What is a Test Case?*. [online] SearchSoftwareQuality. Available at: <<https://www.techtarget.com/searchsoftwarequality/definition/test-case#:~:text=The%20purpose%20of%20a%20test,or%20defects%20within%20the%20system.>>> [Accessed 21 April 2022].

Cohn, M., 2022. *Agile Estimation: Why The Fibonacci Sequence Works*. [online] Mountain Goat Software. Available at: <<https://www.mountaingoatsoftware.com/blog/why-the-fibonacci-sequence-works-well-for-estimating>> [Accessed 4 May 2022].

LePage, P., 2022. Storage for the web. [online] web.dev. Available at: <<https://web.dev/storage-for-the-web/#:~:text=LocalStorage%20should%20be%20avoided%20because,not%20be%20used%20for%20storage.>>> [Accessed 4 May 2022].

Newsapi.org. 2022. *Documentation - News API*. [online] Available at: <<https://newsapi.org/docs>> [Accessed 21 April 2022].

Nielsen, J., 1994. *10 Usability Heuristics for User Interface Design*. [online] Nielsen Norman Group. Available at: <<https://www.nngroup.com/articles/ten-usability-heuristics/>> [Accessed 21 April 2022].

Wiki.appstudio.dev. 2022. Distributing your App - NSB App Studio. [online] Available at: <[https://wiki.appstudio.dev/Distributing\\_your\\_App](https://wiki.appstudio.dev/Distributing_your_App)> [Accessed 21 April 2022].

Wiki.appstudio.dev. 2022. *NSB App Studio*. [online] Available at: <<https://wiki.appstudio.dev/Introduction>> [Accessed 21 April 2022].