RMIT UNIVERSITY

Academic Administration

School name: RMIT University Vietnam

Assignment Cover Sheet

for submission of individual and group work

Course/Unit code	Assignment number	Assignment due date	Group/Session name (if applicable)
INTE2512		19/9/2021	Team PunPoQue (NRA)
Course/Unit name			Program title
Object-Oriented Programming			
Lecturer/Teacher's name			Tutor / Marker's name (if applicable)
Mr. Quang Tran Ngoc			Mr. Bao Ho Nguyen Phu

This statement should be completed and signed by the student(s) participating in preparation of the assignment.

Declaration and statement of authorship:

1. I/we hold a copy of this assignment, which can be produced if the original is lost/damaged.
2. This assignment is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgment is made.
3. No part of this assignment has been written for me/us by any other person except where such collaboration has been authorized by the lecturer/teacher concerned and is clearly acknowledged in the assignment.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced and/or communicated for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the School for review by external examiners.
7. I/we understand that plagiarism is the presentation of the work, idea, or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to expulsion from the University. Plagiarized material can be drawn from, and presented in, written, graphic and visual form, including electronic data, and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited.
8. Enabling plagiarism is the act of assisting or allowing another person to plagiarize or to copy your work.

RMIT UNIVERSITY

Contributors

Family name	Given name	Student number	Student signature	Date
Ngo	Ngoc Thinh	S3879364		19/9/2021
Trieu	Hoang Khang	S3878466		19/9/2021
Nguyen	Van Quy	S3878636		19/9/2021
Nguyen	Tran Khang Duy	S3836280		19/9/2021

Further information relating to the penalties for plagiarism, which range from a notation on your student file to expulsion from the University, is contained in *Regulation 6.1.1 'Student Discipline'* ***www.rmit.edu.au/browse;ID=11jgnnjgg70y*** and *Academic Policy: 'Plagiarism'* ***www.rmit.edu.au/browse;ID=sg4yfqzod48g1***.

Assessor's comments	Grade	School date stamp
		(Office use only)

Final Project

NRA news aggregator App

System Documentation

Team PunPoQue (NRA)

19/9/2021

Ngo Ngoc Thinh – s3879364

Trieu Hoang Khang – s3878466

Nguyen Van Quy – s3879636

Nguyen Tran Khang Duy – s3836280

"I declare that in submitting all work for this assessment I have read, understood and agree to the content and expectations of the Assessment declaration."

RMIT UNIVERSITY

Table of Contents

Table of Figures

RMIT UNIVERSITY

I. Product Description

With the current technological revolution and modern communication era, reading newspapers through newspapers has become outdated and no longer convenient. That's why many news websites have appeared and provided users with hot news updated 24 hours a day. But this leads to an inconvenience that sometimes users have to visit multiple websites simultaneously to be able to read information from different sources. For this reason, our team would like to introduce a program that can integrate news from many various news sources, specifically 5 trusted Vietnamese news sources (VnExpress *[1]*, ZingNews *[2]*, TuoiTre *[3]*, ThanhNien *[4]*, and NhanDan *[5]*); hence bringing users seamless experiences.

Using Java programming language *[6]*, our news aggregator will scrape through 5 above Vietnamese news websites to display the data in separate categories sorted chronologically.

Using the JavaFX library *[7]*, our application's user interface was developed and inspired by the News app from Microsoft. As evident, this design language is entirely modern, trendy, and also very user-friendly.

The application's navigation mechanism is also a bright spot that allows users to conveniently pick different categories, even when the app is being minimized. As a result, the category menu will automatically highlight and move the current category to the top of the category menu to quickly notify the current category they are on.

The program will use a paging mechanism such as pagination to divide ten articles per page. The first page shows the ten most recent articles in that category as ten articles' headers, and users can click on one of these articles to start reading that article. Each article's header includes a photo, a title, a source, and the date it was first published, giving the user an intuitive look. In addition, each category will contain more than 50 articles (up to 200-300 articles in each category).

The app also develops extra features such as dark mode/light mode, real-time search, font size customization, and navigation between different articles to bring users valuable experiences. All these features will be present in detail in the Program's Features section.

II. Installation Instructions

Our team recommends using a 16:9 aspect ratio and full HD device to run this program for the best experience.

Because these are java source code files, we need an IDEA to compile and run the code. In this section, we will use IntelliJ IDEA:

1. Download and install **IntelliJ IDEA** *[8]* community version (choose the appropriate version with your operating system) from: https://www.jetbrains.com/idea/download/
2. Once you have installed IntelliJ IDEA, the next step is to install the appropriate JDK. We use the Liberica 16 Full JDK, which also includes the JavaFX library, as Mr. Quang has advised. Download and install **Liberica 16 Full JDK** *[9]* (make sure that you choose the Full JDK to download and select the appropriate version with your operating system), which contains the JavaFX library, from: https://bell-sw.com/pages/downloads/#/java-16-current

Figure 1. Choosing the correct Liberica 16 JDK.

3. Extract the java source code file, we will have the **"NRA"** folder, then start the IntelliJ IDEA. In IntelliJ choose **Projects** -> **Open** -> In the select path dialog, select the extracted **"NRA"** directory.
4. If IntelliJ IDEA asks to **trust the Maven** *[10]* **project**, choose the Trust **Project** option**.**

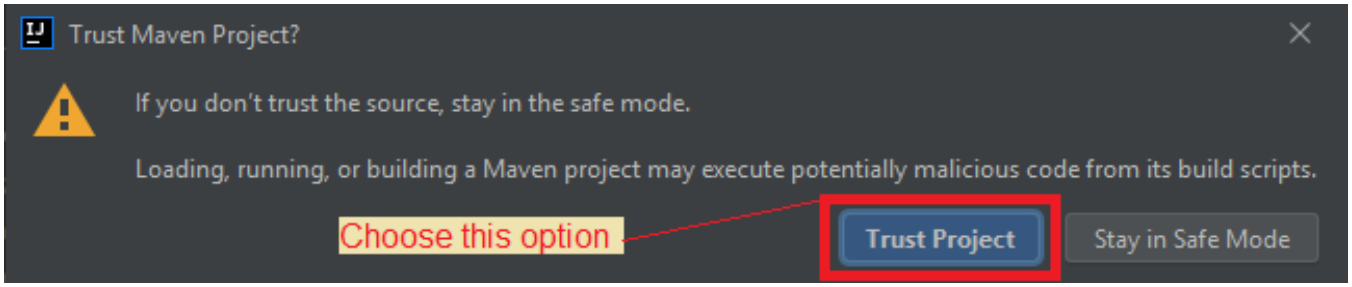

Figure 2. Trusting the project source code.

5. Wait a little bit for IntelliJ IDEA to analyze the project and for **Maven** to download the required dependencies. In IntelliJ IDEA, choose **File** -> **Project Structure** -> **Project** -> **Project SDK** and make sure that you are choosing the **Liberia-16 SDK** in this section. Then **Apply** -> **OK**.

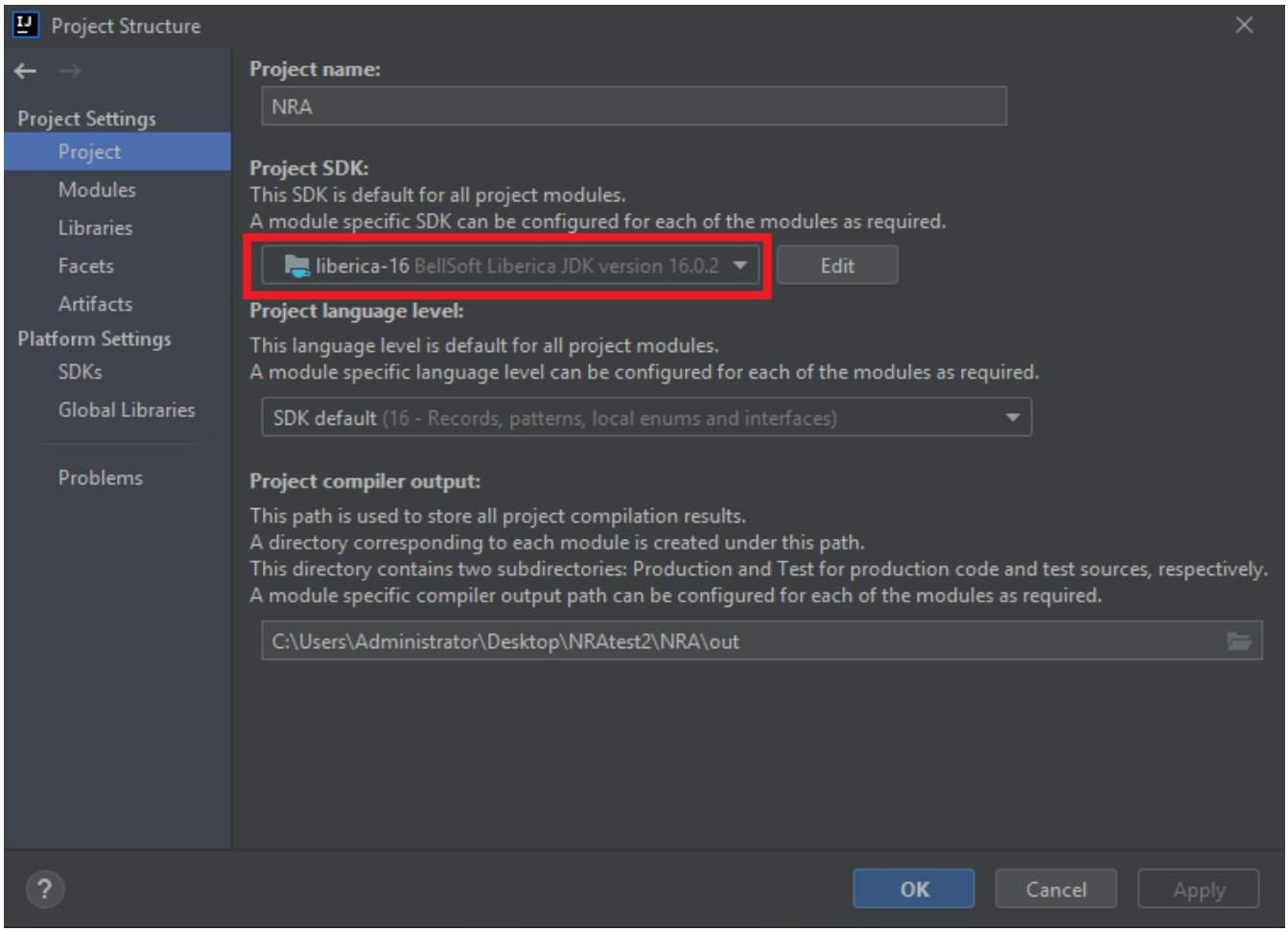

Figure 3. Choosing the correct SDK to run the project.

6. In IntelliJ IDEA, choose **Run** -> **Edit Configurations** -> **Build and run**: make sure that you also choose the **Liberica-16 SDK** here. Then **Apply** -> **OK**.

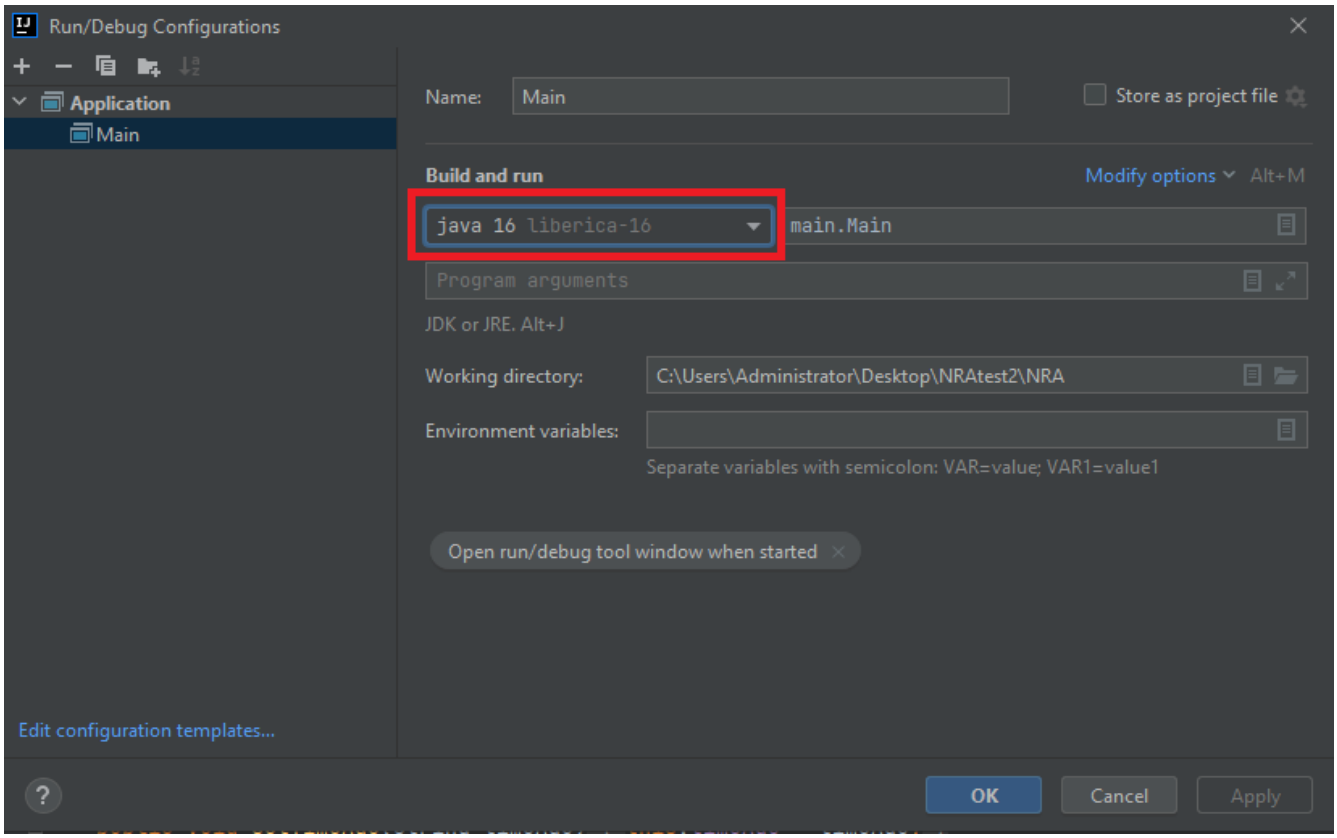

Figure 4. Choosing the correct compiler to run the project.

7. Now in IntelliJ IDEA check the **External Libraries** section to make sure that we have all required libraries. We will use two libraries for this project: **Liberia-16** and **Jsoup 1.14.2** *[11]*. The **Liberia-16 library** will be automatically added to the project when we choose the **Liberia-16 SDK** to run this project. For the **Jsoup library,** normally, IntelliJ IDEA will automatically download this library through **Maven** as soon as the project is opened. If you see that you have two required libraries, go directly to **step 10**. But in some cases, **Maven** can't automatically download the **Jsoup library**, move to the next step to fix this problem.

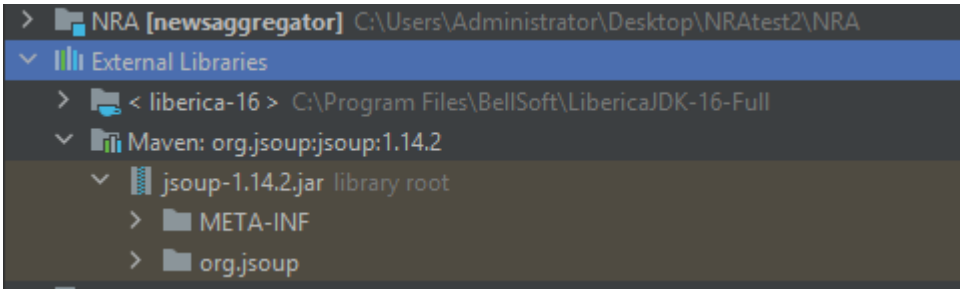

Figure 5. Checking the external libraries.

8. If you don't see the **Jsoup library** in the **External Libraries** section, try to make **Maven** reload. In IntelliJ IDEA, in the upper right, choose **Maven**, then a tab will popup: choose the **Reload All Maven Projects** option. Then wait for Maven to download the **Jsoup library**;

again, check the **External Libraries** section to ensure that you have two required libraries. If you still don't see the **Jsoup library**, move to the next step to manually add this library.

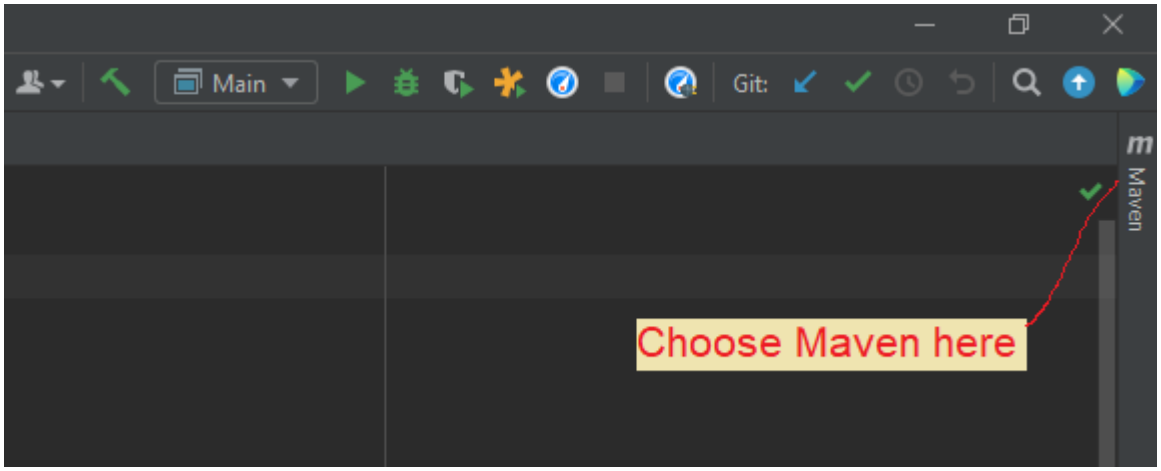

Figure 6. Maven button in Intellij IDEA.

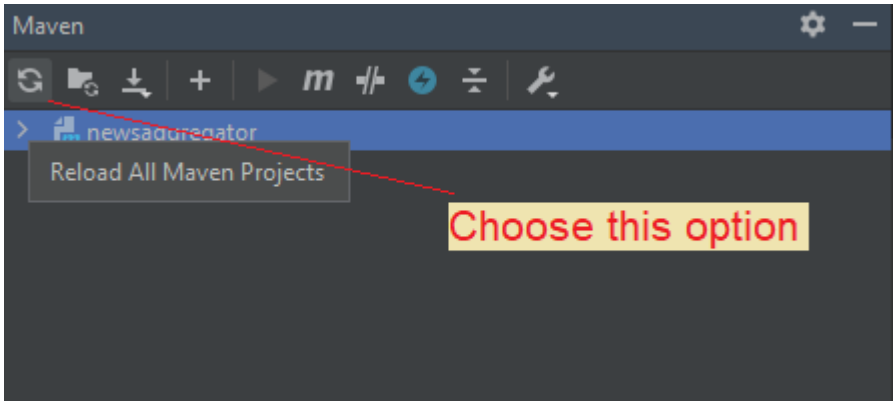

Figure 7. Reloading Maven to download the necessary external libraries.

9. Download the **Jsoup library** jar file from: https://jsoup.org/packages/jsoup-1.14.2.jar

   In IntelliJ IDEA choose **File** -> **Project Structure** -> **Libraries** -> **New Project Library (plus sign)** -> **Java** -> In the select path dialog select the downloaded file **"jsoup-1.14.12.jar"**. Then **Apply** -> **OK**.

10. Now we have all the required libraries then we can run the project. In IntelliJ IDEA, choose Run -> Run 'Main', then wait for IntelliJ to compile and run the application.

III. Program's Features

Display newest news from 5 trusted Vietnamese news source

The program can effectively scrape the latest news from 5 different websites:

1. VnExpress
2. ZingNews
3. TuoiTre
4. ThanhNien
5. NhanDan

Up to 50 latest news are displayed in ascending order of time and divided into 5 pages with 10 news per page.

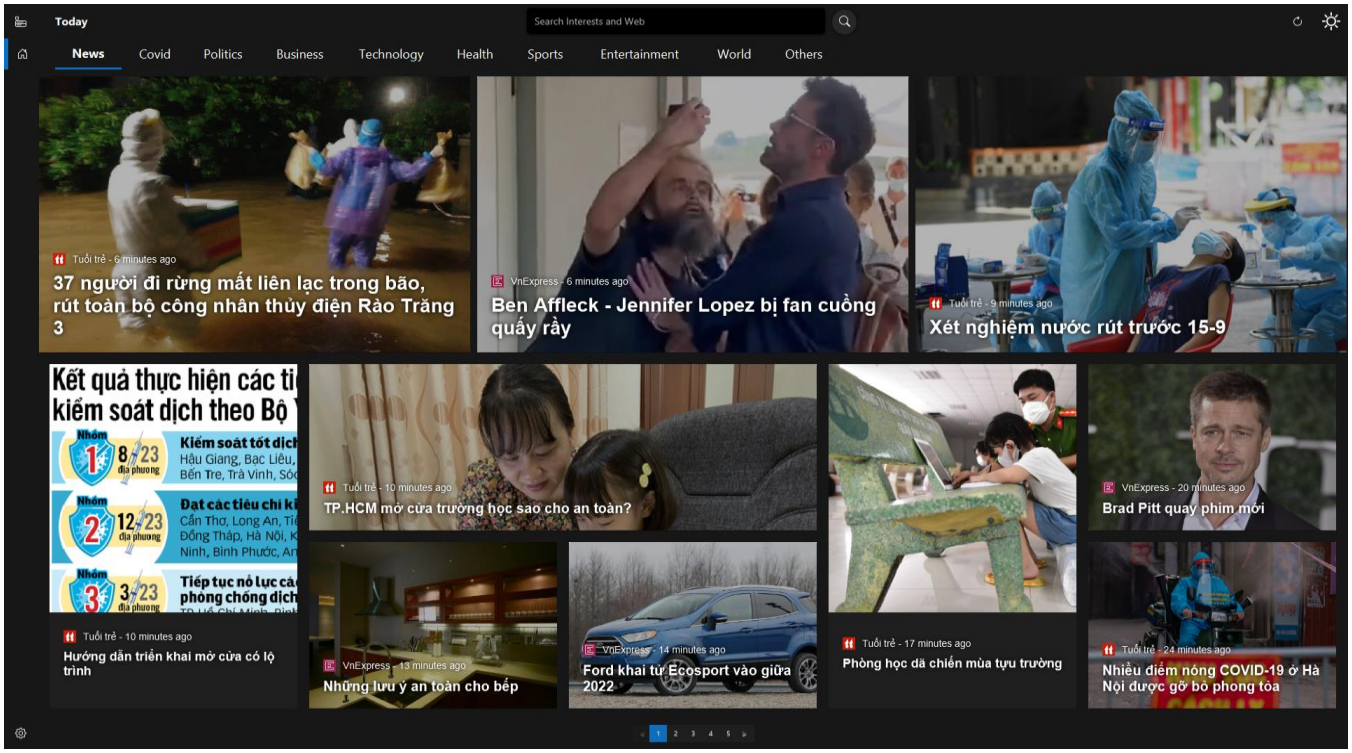

Figure 8. First page of the application with 10 latest news.

Display news based on category

News is divided into ten categories. The user can choose one out of those categories, and the program can display 50 latest news of that category only, which is also in ascending order of time.

Real-time search

The program implemented a searching mechanism that utilized the existing search bar of each website. The program will use the user's keyword to put into each website search bar then scrape the search result of the keyword from those websites. After that, the news list will be arranged in ascending order of time, then displayed to the application.

For example, if the user searched "nông dân" in the search bar, the application would go to 5 websites and use that keyword to search for relevant news. After that, the application will scrape the result from the searches and display it onto the screen.

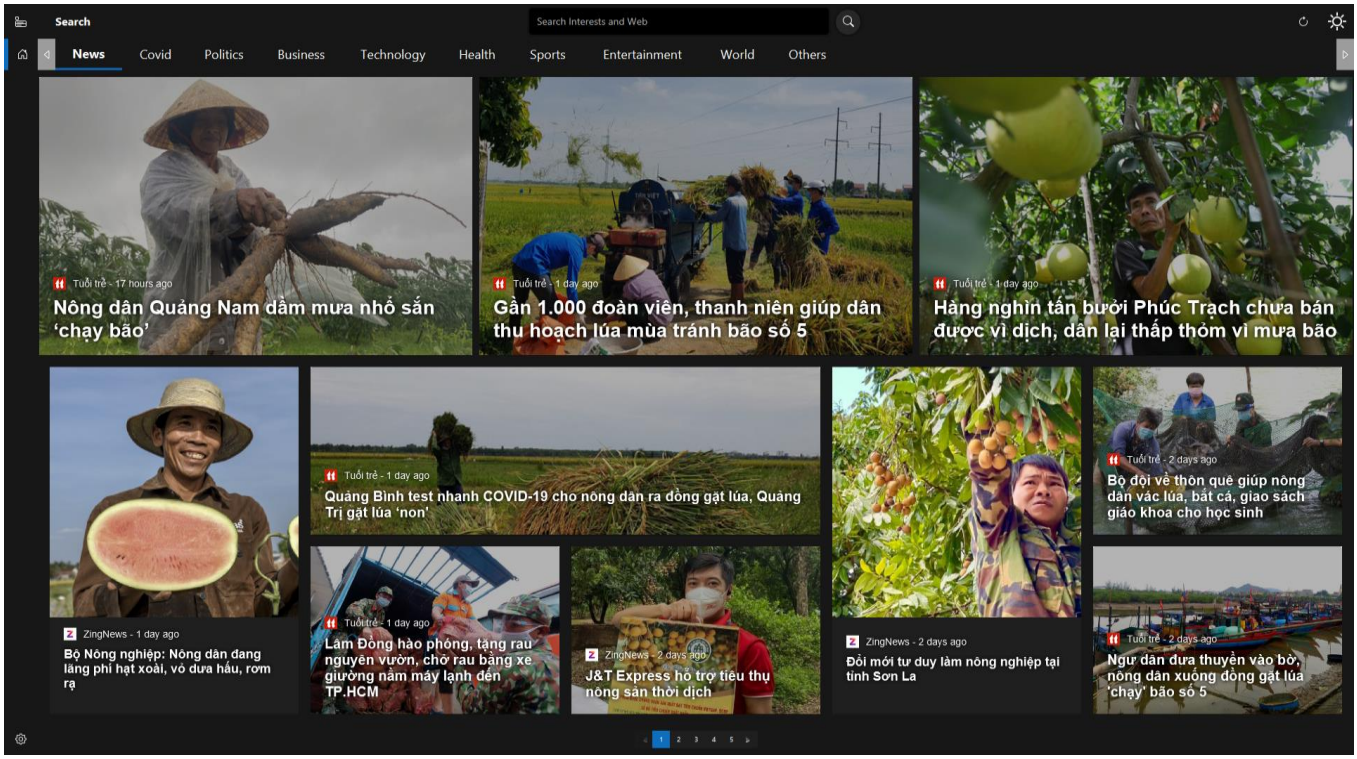

Figure 9. Searching the "nông dân" keyword example.

Dark mode / Light Mode

Users can switch the application color scheme between dark mode and light mode. Light mode uses a dark-colored text, icon with a light background, while dark mode uses light-colored text and a dark background.

Light mode is set as default. Users can switch between dark and light with a button on the top-right corner of the application. Alternatively, users can switch between dark and light modes in the setting.

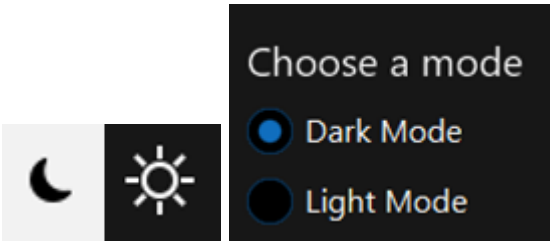

Figure 10. Two ways to choose between dark/light mode.

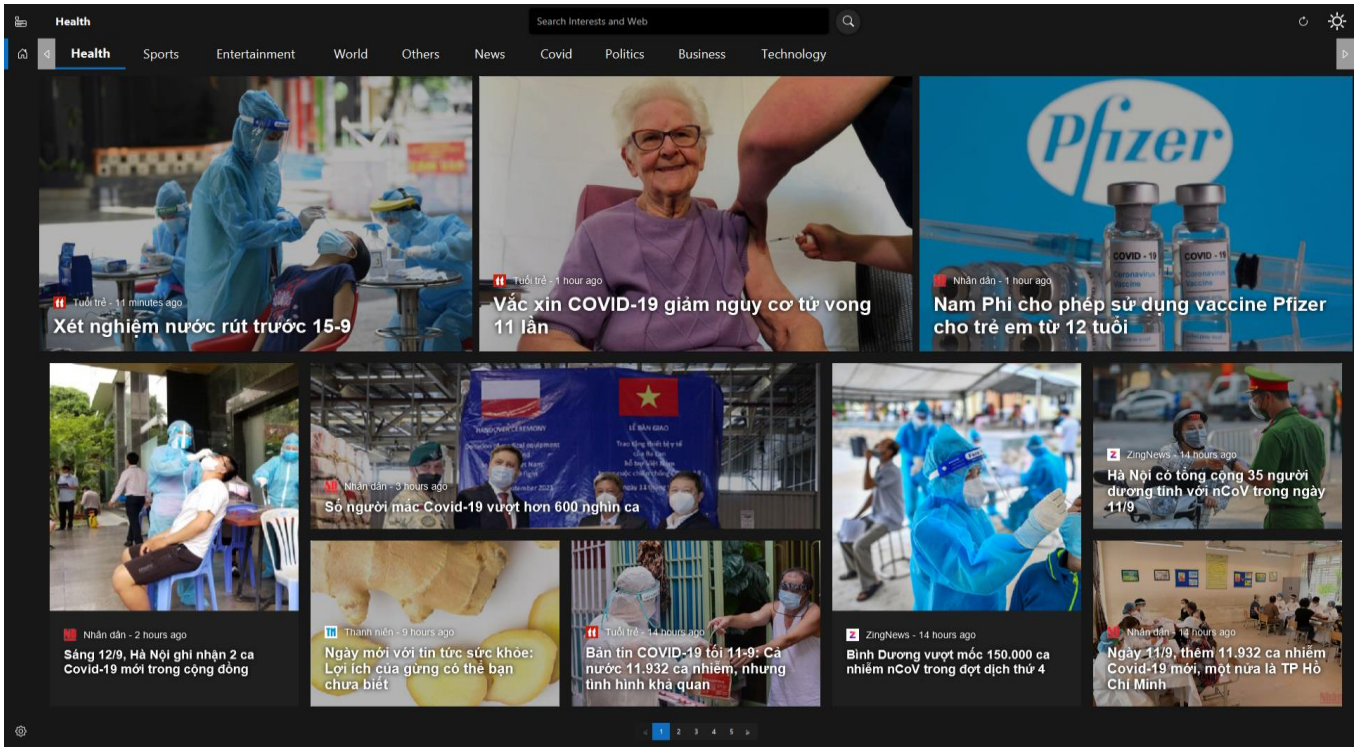

Figure 11. Dark mode of application.

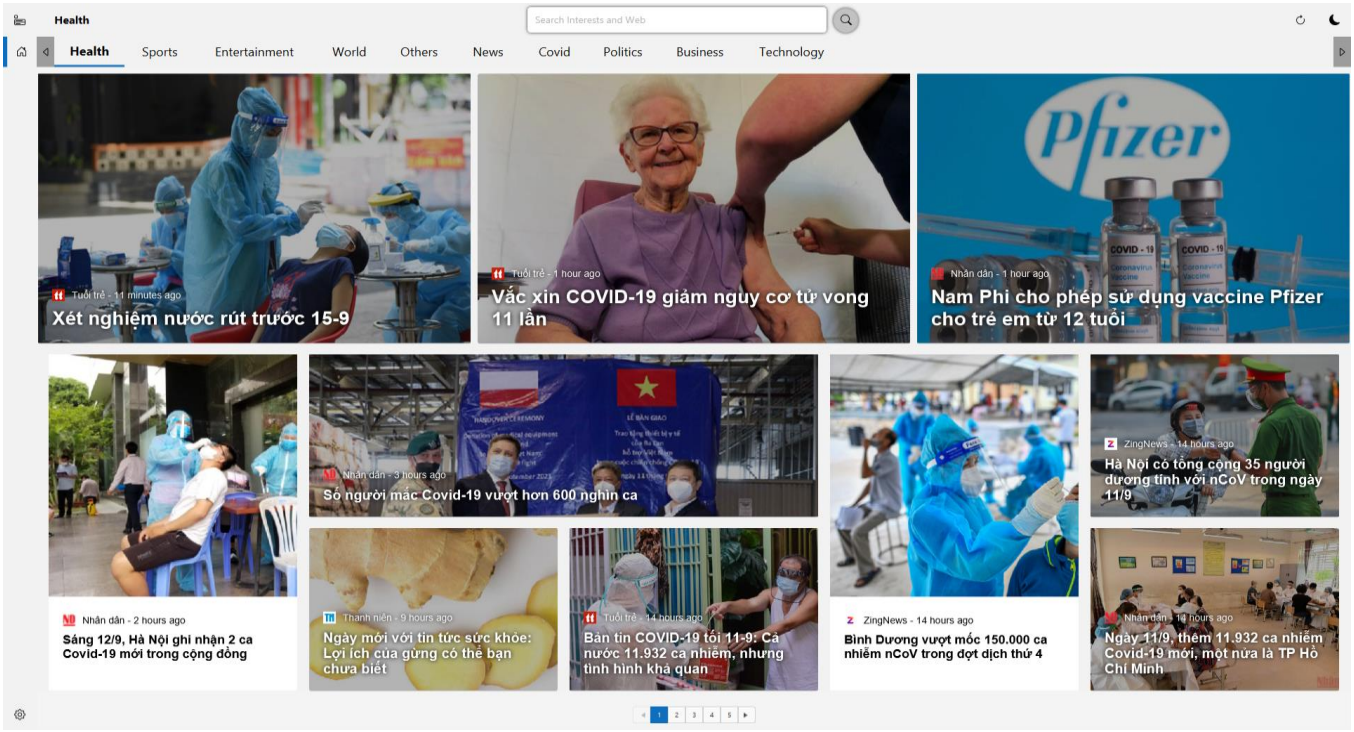

Figure 12. Light mode of application.

Font size customization

The program comes with three font size options to choose from, ranging from “Normal”, “Large” to “Very Large”. Users can change font size in the setting.

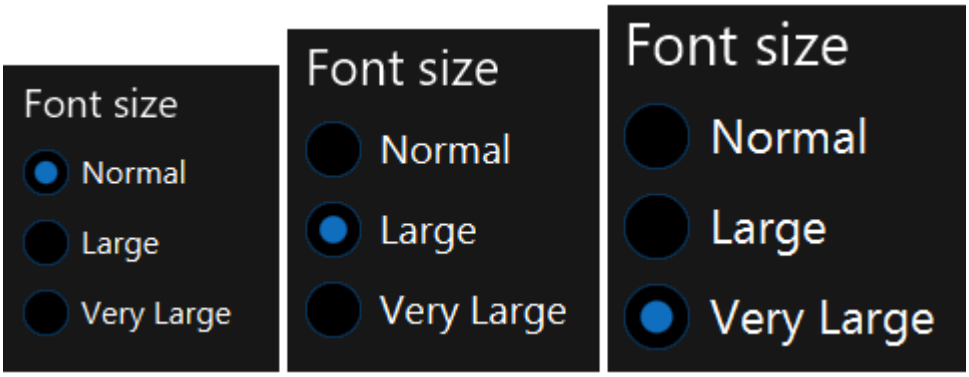

Figure 13. Choosing three font size modes.

Figure 14. Three font size modes (from left to right is normal, large, very large).

Refresh

The refresh button is used to prompt the application to scrape the latest news from the website. The user can use this to read the news posted minutes or seconds ago without reopening the application.

Figure 15. Refresh button.

Interaction with original post

Users can choose to open the original post in the browser by clicking the hyperlink found at the bottom of the news.

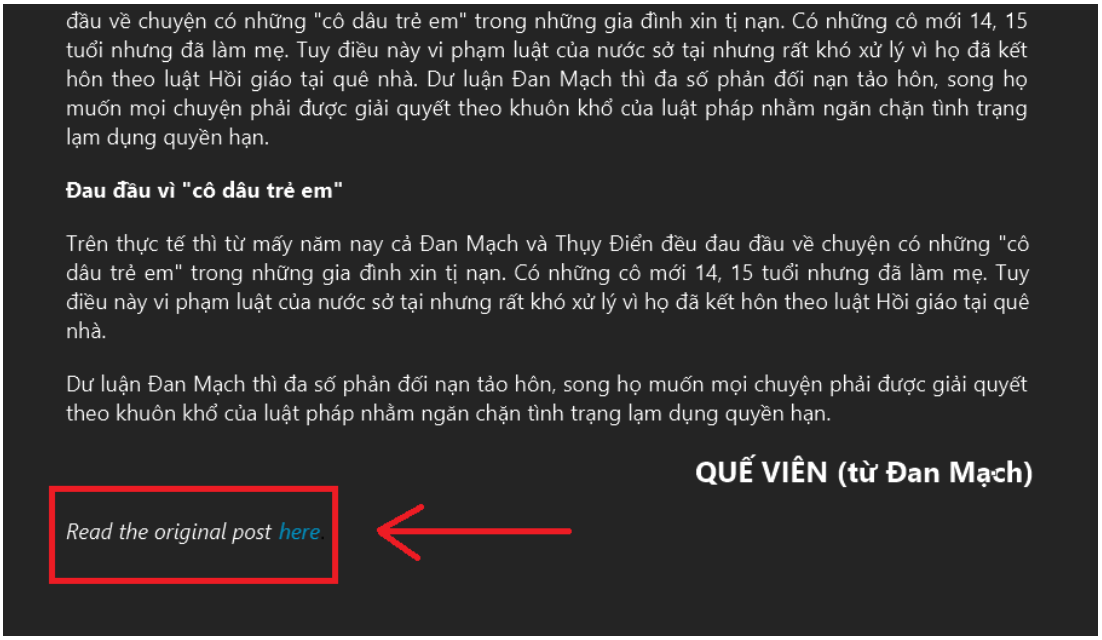

Figure 16. "Read the original post here" text.

Alternatively, users can open the original news in the browser by navigating to the top-right corner of the application and press the button with the globe icon.

Users can also copy the link of the original news by pressing the button with the chain icon.

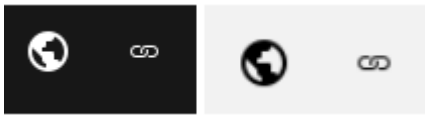

Figure 17. Open in browser and link copy button.

Navigation assistance

Users can navigate the news forwards and backward without needing to exit the news. Press the left arrow button to navigate backward and the right arrow button to navigate forwards.

Figure 18. Navigation between articles.

Like the news navigation, two arrow buttons allow users to move between categories (these two arrow buttons will appear when the user moves the mouse in the categories bar). Press the left arrow button to navigate backward and the right arrow button to navigate forwards.

Figure 19. Navigation between categories.

IV. Bugs and Work-around

Heap memory problem: At the application's initial state when the app just loaded, the app only uses about 40MB of heap memory. But when the application runs for too long and loads too many articles, the heap memory will take up from 600MB to 1GB, which is not reasonable for a news aggregator application.

We have been working on this problem and discovered a problem in the responsive design coding part of the app. Because the size of the loaded images will be bound to a listener, which listens to the app's height and width change, then the image size will change following this height and width. As a result, when we open a new article without deleting these listeners, these images will still be located in the heap memory. The Java Garbage Collector (GC) can't collect these images leading to the rapid memory increase. For this problem, we try to save all the listeners to a list and then delete all the list elements each time a user picks a news article. Below is the result in the JProfiler [*] that shows heap memory will decrease very fast when the user exits from an article back to the menu:

Figure 20. The ability of decreasing heap memory.

Different layout problem: Our data scraping strategy and our scraping algorithm might not be practical and general. Sometimes when the application counters an article with a strange layout, our app can't get complete information, including texts and pictures of these articles.

The first workaround for this bug is that we will rebuild the entire scraping algorithm and strategy. But this change required us to restructure almost the whole project, which is too expensive; hence our team might do this first solution in the future. So, our team chose the second solution, that we will fix the strange layout article one by one, fix as many as possible until there are almost no articles with unconventional layouts that the app can't scrape. With this second solution, we can continue to use what we have built, but there will still be a few articles with weird layouts that we haven't found in the future, and the app can't scrape all the data from it.

Connection problem: Normally, the application needs under 1 second to load a category, but sometimes due to the slow connection between the computer and the server of news websites, the application can load very slowly. Slow connection leads to unnecessary discomfort.

To tackle this problem, we will retry to connect to the server if the connection is not complete in under 3 seconds and the maximum retry number is 5, which is reasonable for the user to wait. During the waiting time, we also display the beautiful loading animation, loading bar, and a meaningful message to notify the user what websites are being retried to connect to.

Text selection/highlight problem: When reading newspapers, some users like to highlight the text they are reading and select text to copy it. But this feature is not present in our project. This problem comes from choosing the library unwisely from the start of the project: we use Text and TextFlow of the JavaFX library to display scraped text, which doesn't support text selection.

We looked to a few other libraries, such as **RichTextFX** *[12], as a perfect alternative to Text and TextFlow. But the replacement takes a long time to completely replace what we have built, so this recommendation may be implemented in the future.*

V. Project Demonstration

The project demonstration video link can be found here: https://youtu.be/KEmnBOmm5oc

Our team members take so much effort to recap this video, but our pronunciation still must be improved. So for a better understanding, we have generated a caption for the video. Please turn on the caption while you are watching the video.

1. Nguyen Tran Khang Duy: From 0’22’’ to 2’26’’
2. Nguyen Van Quy: From: From 2’26’’ to 4’33’’
3. Trieu Hoang Khang: From 4’34’’ to 6’16’’
4. Ngo Ngoc Thinh: From 6’17’’ to 8’26’’

RMIT UNIVERSITY

Acknowledgment

We want to send our deepest gratitude to a vast constellation of mentors, teammates, and friends who helped finalize this project. Our first debt goes to our Course Coordinator - Mr. Quang Tran, and our Teaching Assistant – Mr. Bao Ho, for providing us invaluable guidance and consultations through the different iterations of this project. The Covid-19 outbreak has made this course challenging for us; however, we managed to complete this project thanks to their excellent advice and insightful explanation. We are incredibly grateful to our friends-among them our teammates Thinh Ngo, Quy Nguyen, Khang Trieu, Duy Nguyen for their encouragement and generous engagement with this project. We would not have been able to complete this project if it were not for their commitments.

Again, the deepest thanks to these lecture and laboratory slides from our teachers:

- 1. INTE2512 Week 1 lecture + lab slides: W1 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 2. INTE2512 Week 2 lecture + lab slides: W2 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 3. INTE2512 Week 3 lecture + lab slides: W3 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 4. INTE2512 Week 4 lecture + lab slides: W4 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 5. INTE2512 Week 5 lecture + lab slides: W5 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 6. INTE2512 Week 6 lecture + lab slides: W6 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 7. INTE2512 Week 7 lecture + lab slides: W7 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 8. INTE2512 Week 8 lecture + lab slides: W8 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 9. INTE2512 Week 9 lecture + lab slides: W9 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 10. INTE2512 Week 10 lecture + lab slides: W10 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 11. INTE2512 Week 11 lecture + lab slides: W11 - What's happening this week?: Object-Oriented Programming (instructure.com)
- 12. INTE2512 Week 12 lecture + lab slides: W12 - What's happening this week?: Object-Oriented Programming (instructure.com)

RMIT UNIVERSITY

References

[1] "VnExpress - Báo tiếng Việt nhiều người xem nhất", *vnexpress.net*, 2021. [Online]. Available: https://vnexpress.net/. [Accessed: 11- Sep- 2021].

[2] *Zingnews.vn*, 2021. [Online]. Available: https://zingnews.vn/. [Accessed: 11- Sep- 2021].

[3] T. ONLINE, "Tin tức, tin nóng, đọc báo điện tử - Tuổi Trẻ Online", *TUOI TRE ONLINE*, 2021. [Online]. Available: https://tuoitre.vn/. [Accessed: 11- Sep- 2021].

[4] "Tin tức 24h mới nhất, tin nhanh, tin nóng hàng ngày", *Thanh Niên*, 2021. [Online]. Available: https://thanhnien.vn/. [Accessed: 11- Sep- 2021].

[5] "Báo Nhân Dân điện tử", *Báo Nhân Dân*, 2021. [Online]. Available: https://nhandan.vn/. [Accessed: 11- Sep- 2021].

[6] "Java Tutorial", *W3schools.com*, 2021. [Online]. Available: https://www.w3schools.com/java/default.asp. [Accessed: 18- Sep- 2021].

[7] "JavaFX", *Openjfx.io*, 2021. [Online]. Available: https://openjfx.io/. [Accessed: 18- Sep- 2021].

[8]*Tutorials.jenkov.com*,2021.[Online].Available:http://tutorials.jenkov.com/javafx/index.html?fbclid=IwAR00PfMH6tVx6fBOUIzHP3LbNIwAeW-o-DlaxO9zMyD8YijM_Z3i2XqtYIY. [Accessed: 12- Sep- 2021].

[9] "Download OpenJDK builds of Liberica JDK, Java 8, 11, Java 17 Linux, Windows, macOS", *BellSoft LTD*, 2021. [Online]. Available: https://bell-sw.com/pages/downloads/#mn. [Accessed: 18- Sep- 2021].

[10] J. Redmond, "Maven – Maven Documentation", *Maven.apache.org*, 2021. [Online]. Available: https://maven.apache.org/guides/?fbclid=IwAR2J6wZokvEVKj-RqRTw8qH3xJN1kr5TQY_WqnfzBsv4JoCXN1bOqEX6Bzw. [Accessed: 12- Sep- 2021].

[11] J. Hedley, "jsoup: Java HTML parser, built for HTML editing, cleaning, scraping, and XSS safety", *Jsoup.org*, 2021. [Online]. Available: https://jsoup.org/. [Accessed: 11- Sep- 2021].

[12] "GitHub - FXMisc/RichTextFX: Rich-text area for JavaFX", *GitHub*, 2021. [Online]. Available: https://github.com/FXMisc/RichTextFX. [Accessed: 17- Sep- 2021].