Tribhuvan University
Faculty of Humanities and Social Sciences

"AUTOMATED WEB SCRAPING AND ARCHIVING"

A PROJECT REPORT

Submitted to
Department of Computer Application
Asian College of Higher Studies

In partial fulfillment of the requirements for the Bachelors in Computer Application

Submitted by

Momik Shrestha [1811771365]

Shreeju Pradhan [1811771374]

April 26, 2021

Under the Supervision of

Mr. Deepesh Rahut

Tribhuvan University
Faculty of Humanities and Social Sciences
Asian College of Higher Studies

Supervisor’s Recommendation

I hereby recommend that this project is prepared under my supervision by Shreeju Pradhan and Momik Shrestha entitled “**AUTOMATED WEB SCRAPING AND ARCHIVING**” in partial fulfillment of the requirement for the degree of Bachelor Application is recommended for the final evaluation.

…………………………
Mr. Deepesh Rahut
Lecturer
Humanities and Social Science
Asian college of Higher Studies

Tribhuvan University
Faculty of Humanities and Social Sciences
Asian College of Higher Studies

LETTER OF APPROVAL

This is to certify that this project prepared by MOMIK SHRESTHA and SHREEJU PRADHAN entitled “WEB SCRAPING AND ARCHIVING” in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

Signature of Supervisor ……………………………. Mr. Deepesh Rahut Chettri Asian College of Higher Studies Dhobidhara, Putalisadak, Kathmandu	**Signature of Head of Department** ……………..………….. Mr. Brihat Singh Boswa Asian College of Higher Studies Dhobidhara, Putalisadak, Kathmandu
Signature of Internal Examiner ……………..…… Mr. Sachin Shrestha	**Signature of External Examiner** ……………..…… Er. Ranjan Raj Sharma

ABSTRACT

As the Internet and World Wide Web continue to grow in popularity, so do the various news providing sites. The number of users accessing the web that use various sites to read online news are growing exponentially, there is an urgent need to address issues related to this mode of access. Some of this issue includes different sites with different news headlines with unnecessary information. This issue can be addressed by making a useful website which will only display necessary information to a user. The proposed system for this project is a web scraper capable of accessing and extracting data from websites through a website that serves as an interface for user interaction. The extracted data is then saved in a database, and the website allows the user to search and query the saved results. After the system has been fully implemented, a review of the completed system is performed to determine whether a web scraper can be successfully implemented to solve the issues of users receiving unnecessary information.

Keywords: Web scraping, scraper, crawling, crawl, News portals.

ACKNOWLEDGEMENT

We are overwhelmed in all humbleness and gratefulness to acknowledge our depth to all those who have helped us to put these ideas, well above the level of simplicity and into something concrete. We would like to express our special thanks of gratitude to our supervisor Mr. Deepesh Rahut for guiding us throughout our project, by doing this project it helped us in doing a lot of research and we came to know about so many new things.

Any attempt at any level could not have been satisfactorily completed without the support and guidance of parents and friends.

We would also like to thank our parents and friends who helped us a lot in gathering different information, collecting data and guiding us from time to time in making this project, despite of their busy schedules, they gave us different ideas in making this project unique.

Thanking you.

Project Members

Momik Shrestha [1811771365]

Shreeju Pradhan [1811771374]

TABLE OF CONTENTS

LIST OF ABBREVATIONS

IT	Information Technology
TV	Television
HTML	Hyper Text Markup Language
KIM	Knowledge Information Management
AJAX	Asynchronous JavaScript and XML
URL	Universal Resource Location
OMAN	Online Media Association of Nepal
HLMC	High Level Media Commission
FNC	Federation of Nepalese Journalists
EJS	Embedded JavaScript
FR	Functional Requirement
NFR	Non-Functional Requirement
DFD	Data Flow Diagram
RAM	Random Access Memory
VM	Virtual Machine
JS	JavaScript
SQL	Structured Query Language
CMS	Content Management System
CRUD	Create Read Update Delete
CPU	Central Processing Unit
GHz	Gigahertz

TABLE OF FIGURES

TABLE OF TABLES

CHAPTER 1: Introduction

1.1 Introduction

Over 25 million people in Nepal have access to the internet, this converges to nearly 84% of Nepal’s total population. Out of these 25 million people it is safe to assume that the majority of the people do not solely rely on TV for news. There has been a rise in the trend of visiting news portals for news rather than relying on televisions. As demographics of news consumer has shifted from the elderly to the youth and middle-aged adults so has the medium that relays news information. A brief description of study of such topic can be shown below.

For the past few decades, television has been the undisputed leading source of news. In recent years though, a new medium for sourcing news has started to catch up and even surpass television, the Internet. Even in 2013, statistics from the UK’s Office for National Statistics already showed that 55% of adults, for the first time ever, access news on the web. The trend has been on the rise since then. Portal websites are the most commonly used online news sources, visited by over half of online news users on a typical day. While internet users who get news online tend to explore a wide variety of news topics, they are fairly modest in the number of internet sites they use to gather that information. [1]

1.2 Problem Statement

As there are different news sites with different news titles, it is time consuming to get the quick news. So, to solve this problem we can use web scraper which is a software that extracts data from webpages and stores the data in a database for further manipulation. Due to the current number of various news websites with unnecessary information which might mislead the reader it is very necessary to solve the problem.

1.3 Objectives

This project aims to conduct research in the area of web scraping and how it can be used as a tool for scraping only the necessary data. The main focus of this project is to provide user with useful information and save their valuable time. To do so, a web scrapper will be created to automate the process of extracting data and information from websites. Any potential data that are found will be stored in database.

Thus, the main aims of this project are:

- To understand web scraping strategies and web scrappers.

- To create a web scraper to scrape various news sites.
- To store scraped data into a database.
- To build a web application that utilizes the stored data.

1.4 Scope and limitations

1.4.1 Scope

This project when taken as a CMS web application, it is realized that the data scraped by the scraper is the content itself. The strength of this web application is highlighted when there is a vast amount of data accumulated by collecting them over a long period of time. With enough available data this web application can breathe its scopes. Currently the web application relies on dates of events, however with enough data the dependency from date can by eliminated. The web application can be enhanced in such ways that it allows for users to search for news events based on keywords.

For example, a user if desires can be able to enter ‘Kp Sharma Oli’ into the search field, which can then be used by the business logic to display news events relating to ‘Kp Sharma Oli’. Besides this, with enough data and using different statistical formula many estimations of news events can also be possible.

1.4.2 Limitations

As stated above the core strength of this web application is the number of data that it can scrape, store and read. However, this can be its limitation too, if there is not enough data the web application serves no purpose. The huge amounts of data that the web application requires need to be stored somewhere reliable, however this huge data could possibly have economic burdens.

According to the research done by the developing team of “The Wayback Machine”, the cost of storing this kind of data is exponential. For small time developers such kind of expense is not feasible. Legality can be another limitation for this web application, in Nepal where IT rules and regulations are not properly defined, the legality of web scraping becomes a grey area. Since there is no absolute ruling for web scraping, scraping even publicly available data can become troublesome legally.

1.5 Report Organization

The rest of the document is divided into three Chapters namely Background Study and Literature Review, System Analysis and Design, Implementation/Testing and finally Conclusion in the similar order. The Background study chapter describes the major concepts that this project revolves around. The Literature Review Chapter consists of brief analysis of study of existing or similar systems. The System Analysis and Design consists of research and analysis done by us regarding our project. This chapter also consists of diagrams such as use case, ER-diagram, Architectural Design, Database Schema, DFDs and Gantt Chart used as a reference by us while developing the project. It also gives a detailed explanation for each diagram that was used to help with design and implementation, and outlines the constraints regarding the website. The Implementation and Testing chapter contains the detailed explanation. And finally the conclusion chapter includes the Conclusion reached after creating the current version of the website to meet the system objectives.

CHAPTER 2: Background Study and Literature Review

2.1 Background Study

2.1.1 Innovation

When newslinenepal.com began service on the web, it had announced to launch an internet Television. However, it could not continue its operation for long and shut down within a year. Now, www.ana.com.np has launched its internet TV. ANA is the acronym for Associated News Agency Pvt. Ltd. It began its internet service following the success of People's movement in April 2006. NepalNews has also been availing video reports to major stories now.

Many of news and current affair-based programs of Nepal Television including prime news are available on Nepalnews. It also avails many news-based programs of Nepal television including prime news in Nepali and English and top talk shows. Nowadays, it has also been availing Sagarmatha Television live. Ekantipur.com avails live broadcast of Kantipur Television. Many portals are also leading from outside the valley. Lovelypokhara.com, sarasansar.com, danfe.com and cydberchitwan.com are the top online sites in the country operating outside the valley; first three from Pokhara and the next from Chitwan. Hamrodesh.com carries round the clock news updates from Chitwan. Besides, many other are also published, targeting the particular segment of the people. More than a dozen glamour sites are probably making more money than the established online news portals. Cybersansar, sarasansar, Lovelypokhara and danfe.com are the most popular sites in the country. They target youth focused, mainly producing glamorous contents and availing glamorous pictures of aspiring and established models. Cricket.com.np is renowned site operating since last year about news and updates in Nepalese cricket. But there are sites which have their grand initiation but short closure. However, the case is not similar to all others. A few news portals that arrived in the arena with great applaud have terminated their operation: www.parewa.com, www.nepaleyes.com, www.nepaljournal.com, www.newslinenepal.com. Major strength of online media is their accessibility. Unlike many traditional media, they are accessible to the large part of the country. Given topographical situation of Nepal, it takes days for a newspaper or magazine to reach some parts of the country. Online media are free from these constrains. They can immediately update their content and are accessible across the globe.

So far 59 district headquarters have been linked with internet i.e. they can access dialup internet through local telephones. The internet penetration has been increasing; it increased by 0.04 percent 5 in the first quarter of fiscal year 2007/2008 which is very encouraging (NTA, 2008). With this, the internet penetration in the country has reached 0.24%. With advancement in telephone, especially increase in the number of mobile telephone users, the number of internet users is increasing. Many people are accessing internet into their mobile. Mero Mobile provides internet service to all customers while Nepal Telecom is providing the internet service for its postpaid mobile users. This also signals better future of online media. [2]

2.1.2 Web Scraping

Extracting useful information from the web is the most significant issue of concern for the realization of semantic web. This may be achieved by several ways among which Web Usage Mining, Web Scrapping and Semantic Annotation plays an important role. Web mining enables to find out the relevant results from the web and is used to extract meaningful information from the discovery patterns kept back in the servers. Web usage mining is a type of web mining which mines the information of access routes/manners of users visiting the web sites. Web scraping, another technique, is a process of extracting useful information from HTML pages which may be implemented using a scripting language known as Prolog Server Pages(PSP) based on Prolog. Third, Semantic annotation is a technique which makes it possible to add semantics and a formal structure to unstructured textual documents, an important aspect in semantic information extraction which may be performed by a tool known as KIM(Knowledge Information Management). In this paper, we revisit, explore and discuss some information extraction techniques on web like web usage mining, web scrapping and semantic annotation for a better or efficient information extraction on the web illustrated with examples. [4]

2.1.3 Automated web Scraping

Extracting data from a website is fairly a simple and straightforward process. Images can be saved and text can be copied. However, this kind of data extraction is practically impossible when you need large amount of data from multiple websites for a business use case. This is when automated web scraping comes into the picture. To crawl and extract large amounts of data continuously, an automated web scraping setup can be employed. The benefit is minimal manual interference after the initial setup and fully automated web scraping thereafter. Let’s look at how an automated web scraping setup works. [4]

For a web scraping setup to work on full automation, the bot should be able to navigate through the different pages on a website and save the required data fields. Navigation is the key aspect when it comes to automating a web scraping task. This is because, different websites use different kinds of navigation systems and these vary greatly in terms of the complexity. While some websites use simple numbered navigation, there are some modern websites that use infinite scrolling and other AJAX based dynamic navigation techniques. In order to get past these hurdles, the developer writing the web crawler program must have sound technical knowledge. Once the machine is programmed to mimic a human user where required, automating the web scraping setup is a relatively simple process. A queuing system is used to stack up the URLs to be scraped and the crawler setup will visit these pages, one by one thereby extracting the data from them. [3]

2.1.4 Institutionalization of Online News Portals

Institutionalizing online news portals is foremost important for their development and sustainability. But in Nepalese context, they are in the initial stage of institutionalization. Consequently, they have neither clearly defined the functions of the staffs nor the salaries and benefits. As the government is yet to recognize such portals as media institution, many of them are free from taxes and other liabilities. But it is clear that initiations have begun from private endeavor. Online Media Association of Nepal (OMAN) is one major output in this respect. An NGO working in the media sector, Freedom Forum Nepal is providing them support. OMAN’s secretariat is established in the premises of Freedom Forum. [2]

2.2 Literature Review

2.2.1 Study of WayBack Machine

Introduction

The Internet Archive Wayback Machine is a service that allows people to visit archived versions of Web sites. Visitors to the Wayback Machine can type in a URL, select a date range, and then begin surfing on an archived version of the Web. Its founders, Brewster Kahle and Bruce Gilliant, developed the Wayback Machine with the intention of providing "universal access to all knowledge" by preserving archived copies of defunct webpages. Since its launch in 2001, over 531 billion pages have been added to the archive. [1]

Technical Details

Software has been developed to "crawl" the Web and download all publicly accessible information and data files on web pages, the Gopher hierarchy, the Netnews (Usenet) bulletin board system, and downloadable software. The information collected by these "crawlers"; does not include all the information available on the Internet, since much of the data is restricted by the publisher or stored in 8 databases that are not accessible. To overcome inconsistencies in partially cached websites, ArchiveIt.org was developed in 2005 by the Internet Archive as a means of allowing institutions and content creators to voluntarily harvest and preserve collections of digital content, and create digital archives. Crawls are contributed from various sources, some imported from third parties and others generated internally by the Archive. The Wayback Machine offers only limited search facilities. Its "Site Search" feature allows users to find a site based on words describing the site, rather than words found on the web pages themselves. [1]

2.2.2 Literature Review Conclusion

Conclusions can be drawn from the research conducted in the previous chapter. The use of news portals are now prevalent more than ever before. The pace of delivery and consumption of news information is very high. Other findings indicate that the use of web scrapers to scrape news portals and store them are doable.

For the creation of the web scraper and the web application, several tools must be utilized. To keep all code and software produced in a safe environment, this environment can be simulated by the use of a GUI based OS. For the creation of the web scraper the latest version of JS (EJS6) will be used with Node js (v14.16.0) which will be used to communicate with a server and Puppeteer (v8.0.0), which is a headless chromium browser will be used. Whereas for the creation of the web application HTML5, CSS3,PHP, Boostrap4, jQuery and some minute dependencies will be used. The latest stable releases of each language and software will be used to ensure code will be up to date and will not be deprecated in the foreseeable future.

CHAPTER 3: System Analysis and Design

3.1 System Analysis

3.1.1 Requirement Analysis

Based on the research conducted in the previous chapters, requirements can be produced and divided as mentioned next.

Functional Requirement

A functional requirement determines a system or a part of a system. It specifies the tasks that software must carry out. The functional requirements of the project are:

FR1. An unregistered user must be able to view news titles.

FR2. An un-registered user must be able to register.

FR3. A registered user must be able to view news titles.

FR4. A registered user must be able to view news articles.

FR5. A registered user must be able to logout.

FR6. Only the admin must be able to access admin dashboard.

FR7. There must be server-side validation on the registration and login form.

FR8. The scraper must crawl to two different news sites in a single run.

FR9. The scraped data must be stored into a database.

Non Functional Requirement

A non-functional requirement determines a software system's consistency attribute. They are a series of criteria used to assess the precise function of a system. A nonfunctional requirement is needed to ensure the overall usability and effectiveness of the software system.

NFR1. The system should be user friendly and easy to navigate.

NFR2. The web application should be responsive.

NFR3. The scraping script must be fast.

NFR4. There should not be data repetition.

3.1.2 Feasibility Analysis

Technical

The website shall require minimum hardware requirements. The hardware configurations required are a PC using the Core i3, 32 MB of free hard-drive space and 2GB of RAM and Internet Connectivity. For the software requirements, that are needed to run the system efficiently are Operating System: Windows (Vista/Windows 7 and above), Web Brower:

Internet Explorer (8.0 and above), Mozilla Firefox (3.0 and above), or Google Chrome, Drivers: Java Runtime Environment and Integrated Development Environment: Eclipse Juno or Apache Tomcat.

Operational

The users of this website are the ones who would search to test the website. They must have basic understandings about computers and the internet. The users should be able to perform the following functions using this system:

- Use a browser.
- Visit our URL
- Enter date into the search box
- Sign-on/login using a username and password.

Economic

A project must be completed within mentioned budget. In this project however, there is no development cost nor any operational cost. Thus, it can be said that the project is economically feasible. However in the future, if the application is hosted in the cloud, various fees will follow. As our current database is a SQL database, let us hypothesize using the Microsoft Azure database service. These fees can be structured as fo1llows:

Table 3.1 : Economic Feasibility Study

Expense Article	Rate
Gen 5 CPUs2 GB RAM 2.3GHz 1VCORE (Azure)	$0.034/hr =$24/per month
Storage (Azure)	$0.115/per month
Backup (Azure)	$.20/per month
Hosting (AWS)	$3.50/per month
Total	~$29 per month

We have taken into consideration the cheapest pricings provided by the Amazon on AWS and Microsoft on Microsoft Azure. The cheapest plan provided by Microsoft Azure on cloud computing is $0.03 per hour for a Gen 5 2.3 GHZ CUP with 2GB of RAM. This translates to the cost being $24 per month for renting server. Similarly Azure provides 1TB of online storage for $0.10 per month. 1 GB of cloud storage which can be used to store necessary data remotely, is more than enough for this web application to work smoothly, finally the backup plan of Azure tend to cost $0.20 per month. Similarly the hosting services provided by AWS is $3.50 per month. The total cost of keeping this web application in operation per month tends to become an average of $28. This when converted to domestic currency results in Rs.3343.76 per month as of current date (2021/04/20).

Schedule:

The expected timeline can be seen in the Gantt chart below:

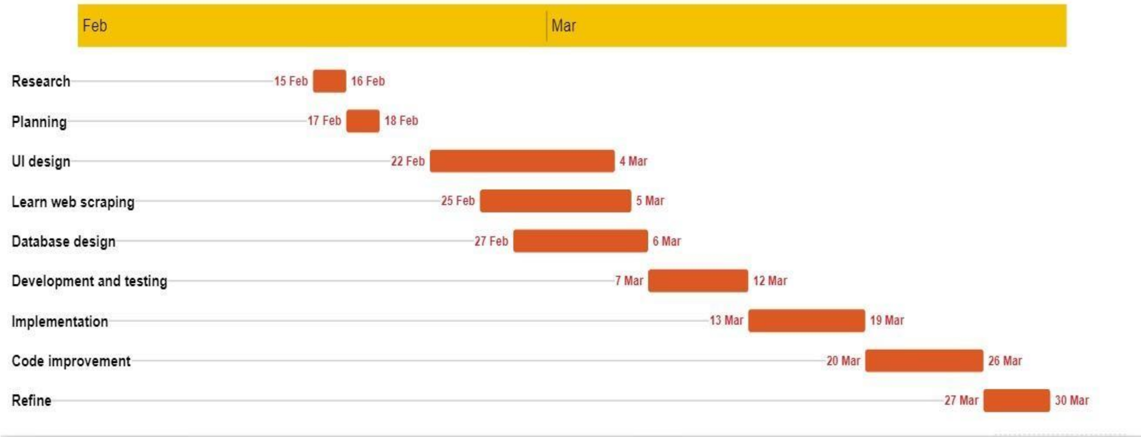


Figure 3.1: Expected Timeline Gantt Chart

Whereas the actual timeline can be seen below:

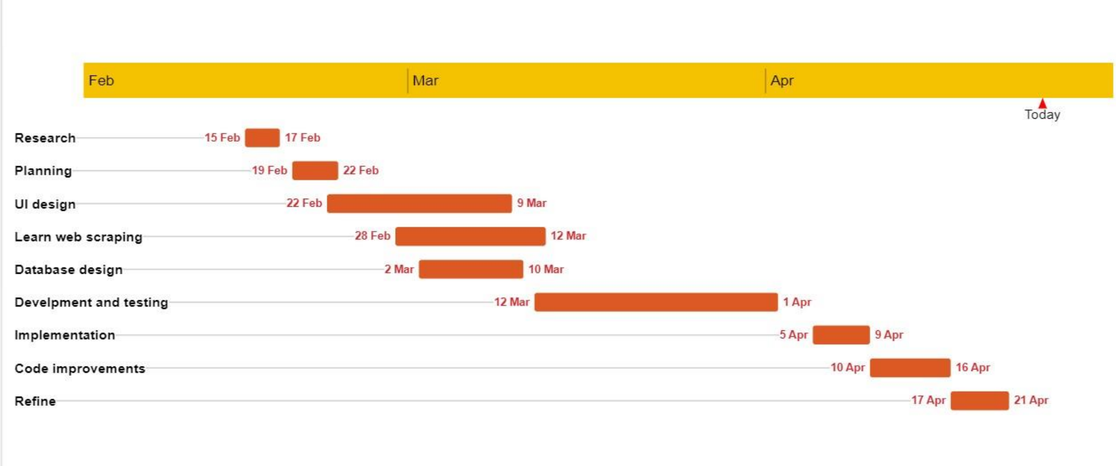


Figure 3.2: Real Timeline Gantt Chart

We had expected to complete all work by March 30th. However due to development and integration issues, our timeline got delayed. The successful development of the scraper ended up taking longer than that we had expected of. As we mentioned in earlier studies, our workflow followed the waterfall model, the delays in creation of the web scraper ended up creating further delays regarding implementation, testing and maintenance. Thus extending our timeline till 18th of April.

3.1.3 Data Modeling

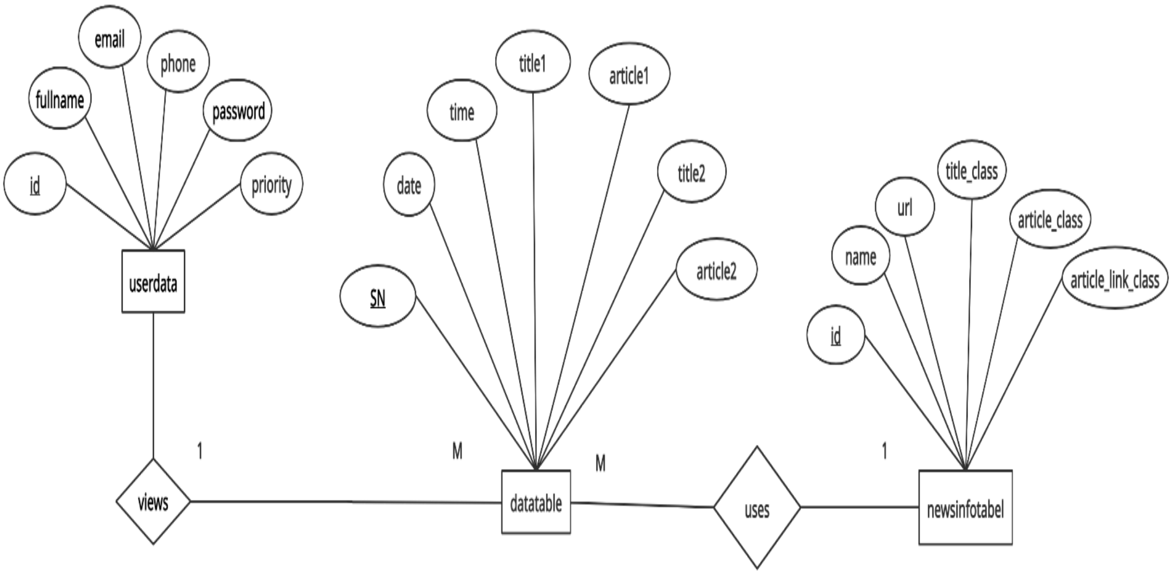


Figure 3.3: ER-Diagram

Entities:

- userdata:

  The table named 'userdata' consists of data of the users that have successfully registered to the system. 'userdata' has 6 fields, namely; id, fullname, email, phone, password and priority. Here the 'id' field is the Primary Key.

- datatable:

  The table named 'datatable' consists of data that has successfully been scraped by the scrape. 'datatable' has 7 fields, namely; SN, date, time, title1, article1, title2, article2. Here the 'SN' field is the Primary Key.

- newsinfotable:

  The table named 'newsinfotable' consists of data that acts as the crawl parameters for the web. 'newinfotable' has 6 fields, namely; id, name, url, title_class, article_class and article_link_class. Here the 'id' field is the Primary Key.

Relationships:

- userinfo and datatable:

  There exists a Many to One relationship between the entity ‘userinfo’ and and the ‘datatable’. This is because one single instance (row) of ‘userinfo’ has the ability to access multiple instances (row) of ‘datatable’ without any restrictions.

- newsinfotabel and datatable

  There exists a One to Many relationship between the entity ‘newsinfotabel’ and entity ‘datatable’. This is because a single instance (row) of ‘newsinfotable’ is used by the scraper as parameters to scrape multiple instances of data (row) of ‘datatable’.

3.1.4 Process Modeling

Flowcharts

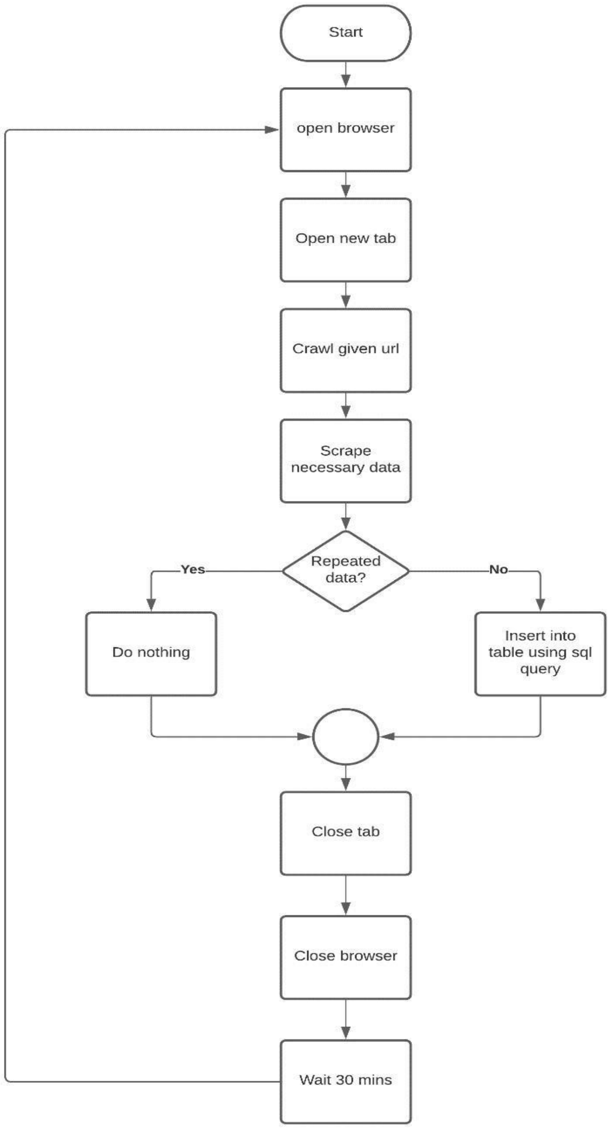


Figure 3.4: Web Scraper Flowchart

The life of the scraper starts when the super-user (admin) executes the command "node scrape.js" into the terminal. The "scrape.js" file consists of asyn function that manages the puppeteer Node JS library. The first step of the function is to open a headless chrome browser behind the scenes. The second step of the function is to open a new tab in the headless chrome. The third step is to go to the URL that is retrieved form the database. The fourth step is scrape necessary. Then scraper then checks the database if the data is repeated of not. If repetition is found, the scraper discards the scraped data. If there is no repetition of data the insert() function is called. The insert() function inserts scraped data into the database ‘datatable’. The previously mentioned async function then closes the previously opened tab. The final step is to close the headless chrome browser. The scraper then waits 30 minutes (say) and loops back to step one.

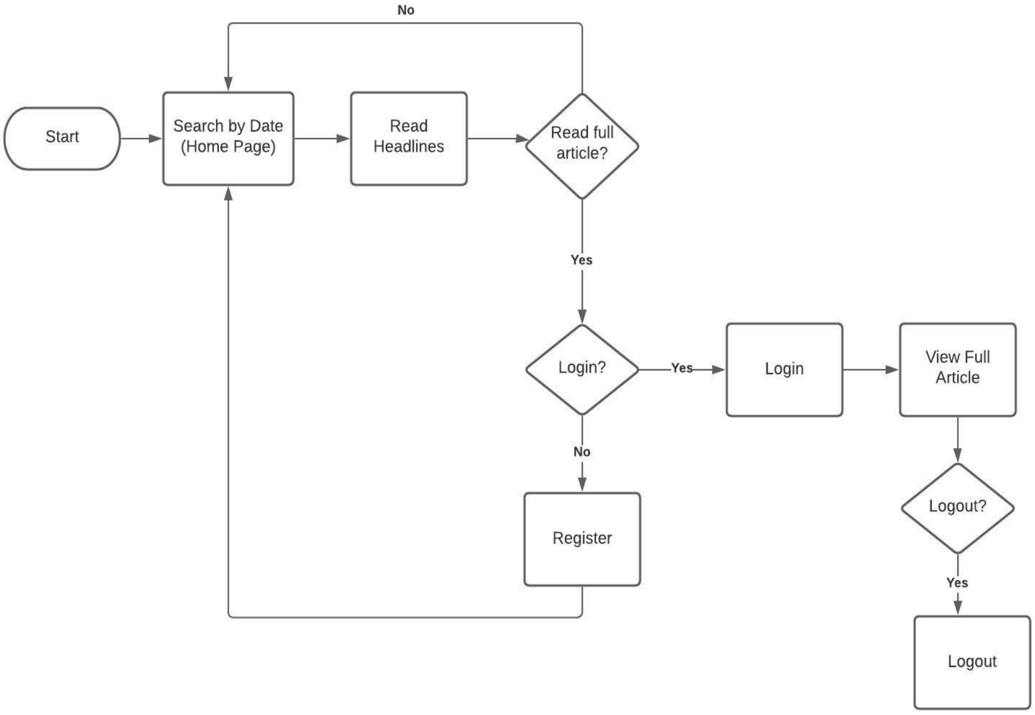


Figure 3.5: User Interaction Flowchart

The interaction of a user with the web application starts as soon as the user lands to the home page i.e index.php. The user then enters desired date into the search field. The web application loads necessary data based on the entered date into a new webpage which the user is redirected to. If the user desires to read complete article, the login status of the user is checked by the web application. If true, the user is redirected to another new page with related information, if false the user is redirected to the registration page. The user is able to logout if he/she decides to.

Data Flow Diagram

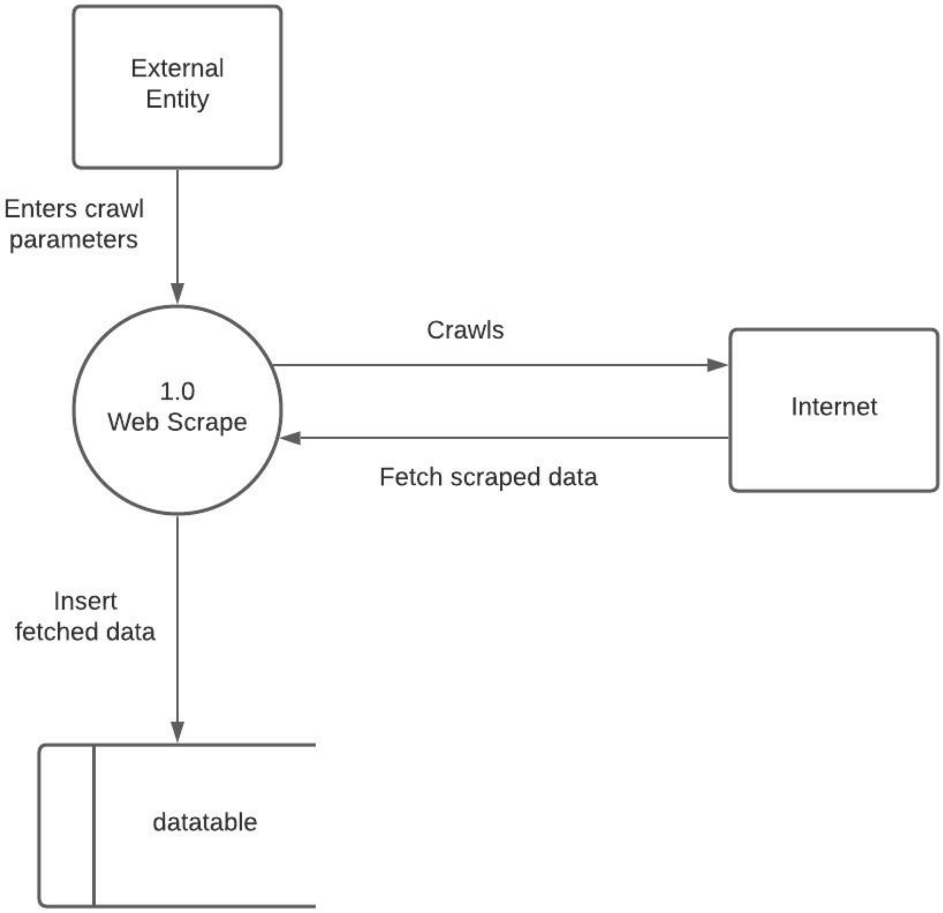


Figure 3.6: Web Scraper level 1 DFD

The figure above represents the Data Flow Diagram for the Web Scraper. The External Entity mentioned in the above figure is a combination of admin user and the database table ‘newsinfotable’. The web scraper uses the crawl parameter entered by the ‘External Entity’. The crawler uses those parameters to crawl through the internet and fetch/retrieve scraped data. The scraped data is then inserted into the database table ‘datatable’ after filtering out the repeated data.

The data in ‘datatable’ is the data used by the user as shown and mentioned in the next diagram.

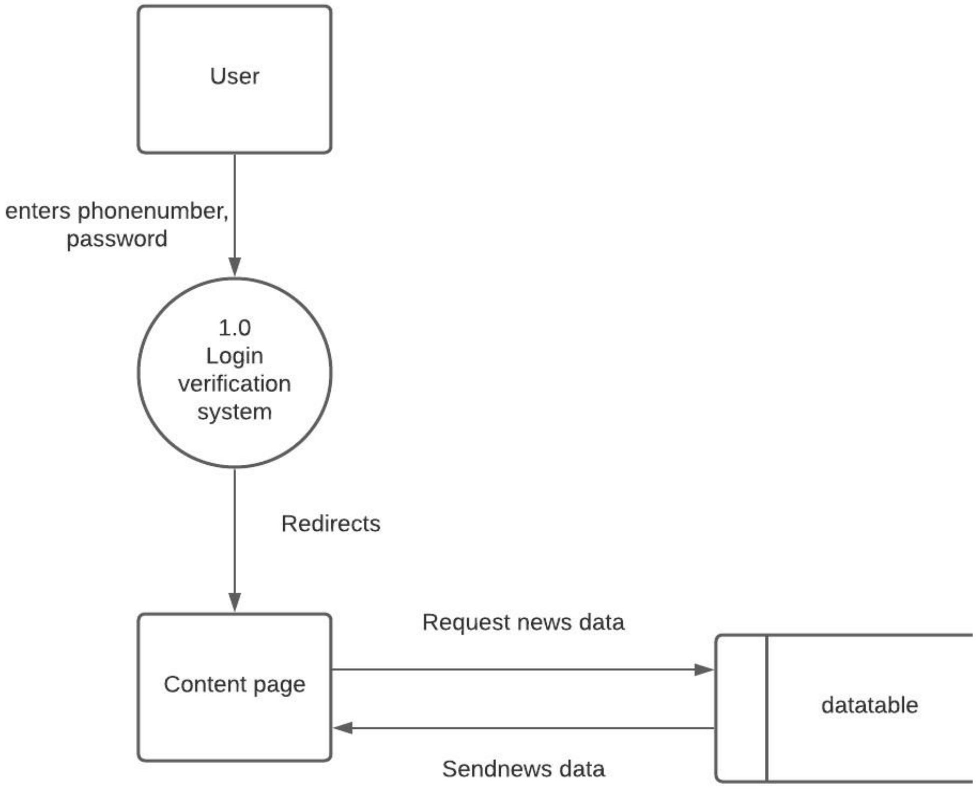


Figure 3.7: User-info level 1 DFD

The ‘User’ in the diagram refers to the end user of the web application. The user enters the phonenumber and password as data into the login page. Those data are verified by the login verification system. If the phonenumber and password is valid, the user is redirected to the content page. The content page is the page where the user is able to view necessary information. The information in the content page exists after the content page request for necessary data from the database table ‘datatable’ which consists of all news data. The ‘datatable’ returns necessary data based on the request made by the content page.

3.2 System Design

3.2.1 Architectural Design

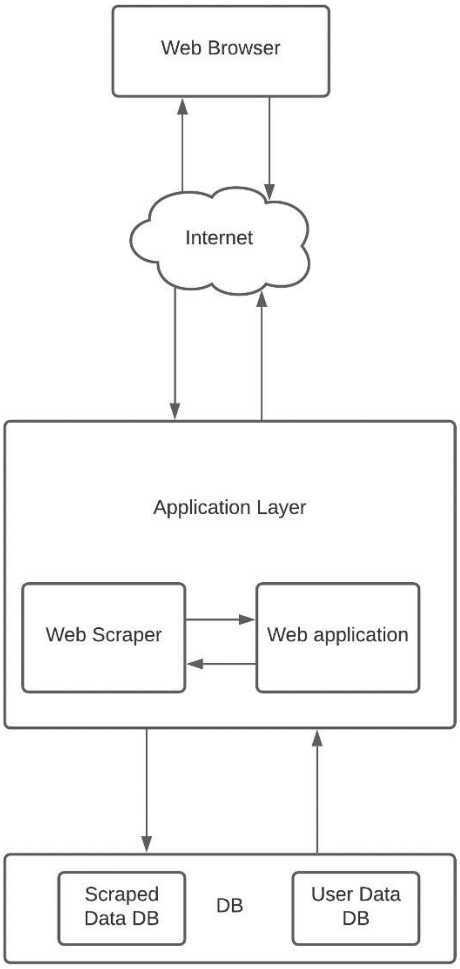


Figure 3.8: Architectural Design

At the most abstract level, Figure 3.2.1 shows the architectural design of the system. The web browser referred in the figure is the platform from where the user interacts with the system. The user hops onto the internet and has access to the application layer of the system, the application layer is made of the web scraper and the web application. The application has access to the database layer 'DB' as mentioned in the figure, the database can primarily be divided into two databases namely the database that consists the scraped data 'Scraped Data DB' and the database that holds user data 'User Data DB'.

3.2.2 Interface Design

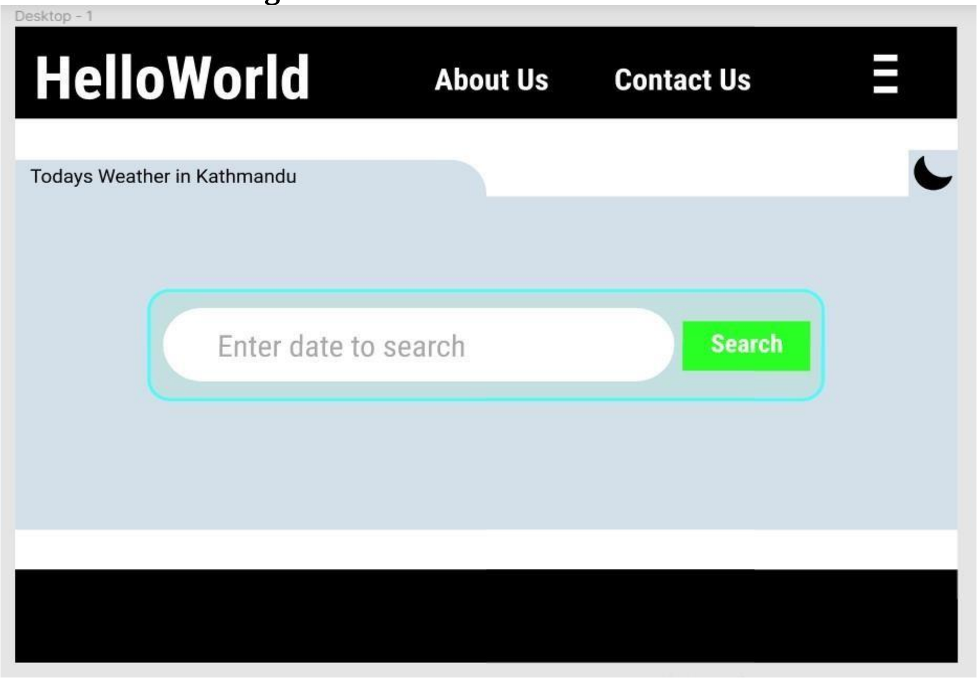


Figure 3.9: Index Page Design

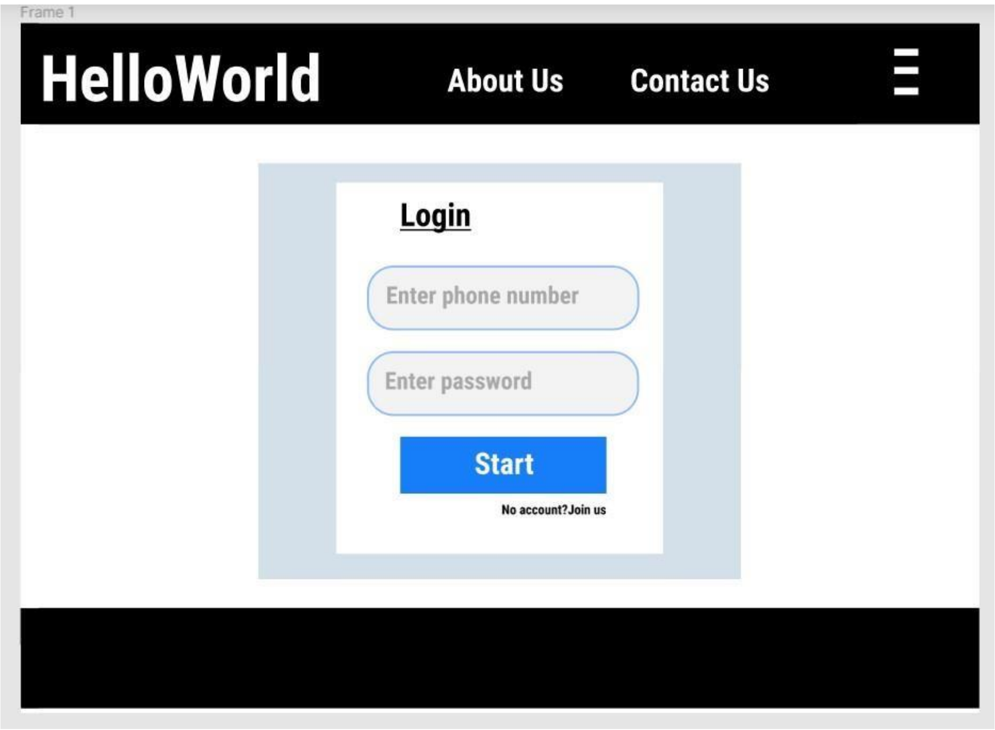


Figure 3.10: Login-page Design

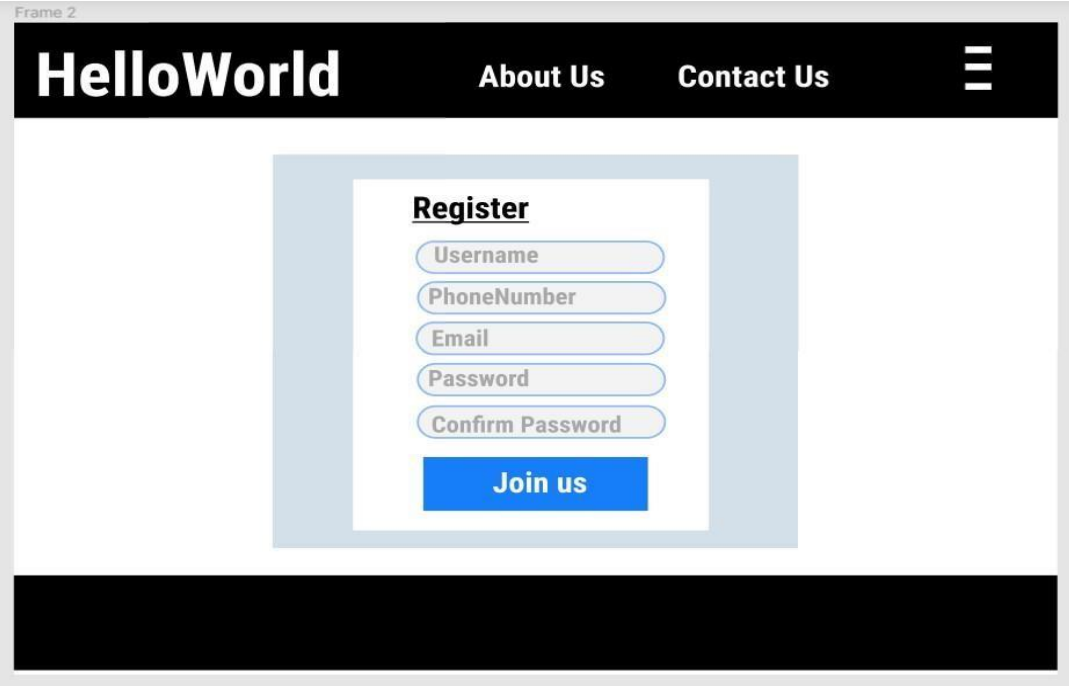


Figure 3.11: Registration Page Design

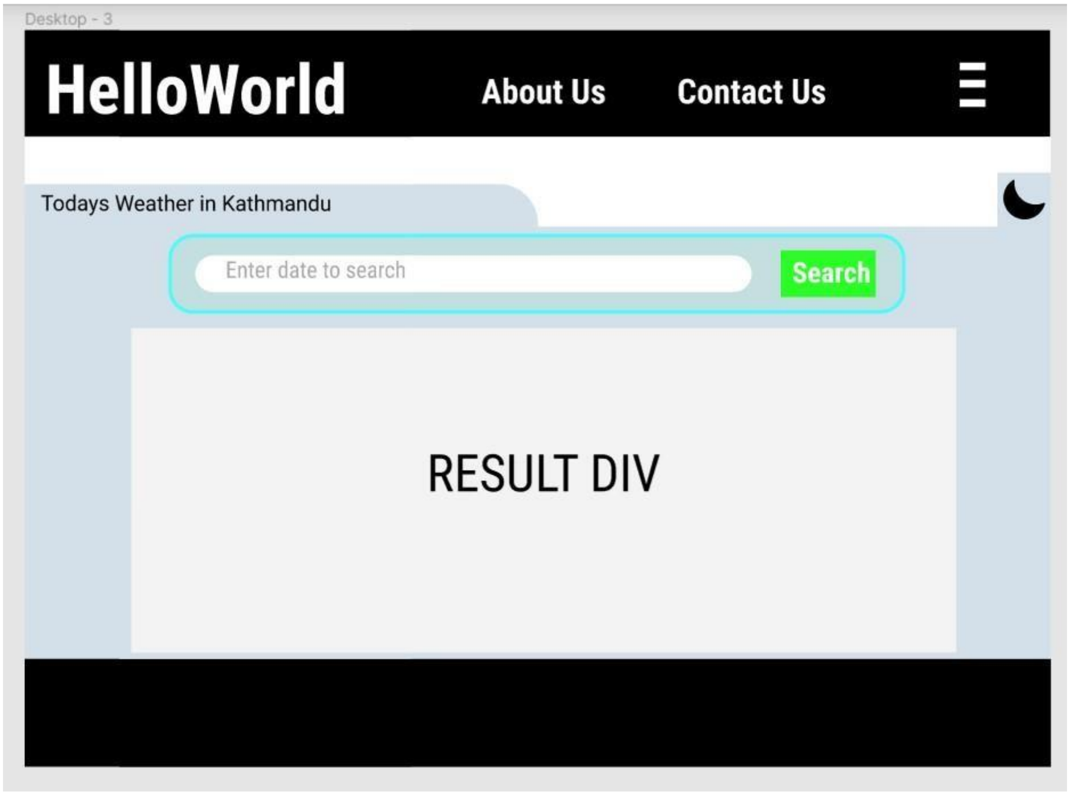
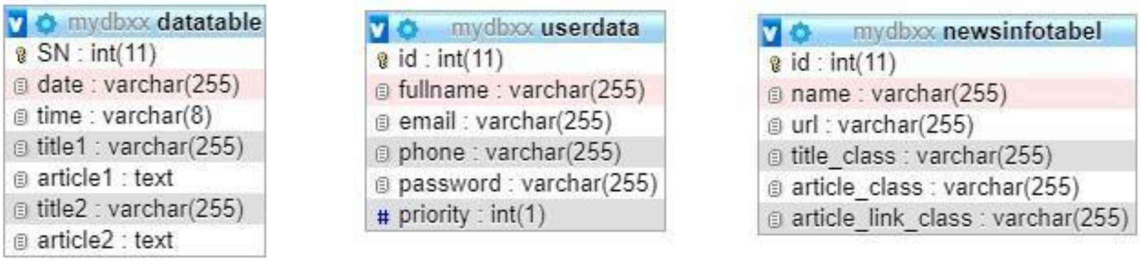


Figure 3.12: Output Page Design

3.2.3 Database Schema Design



mydbxx datatable
SN : int(11)
date : varchar(255)
time : varchar(8)
title1 : varchar(255)
article1 : text
title2 : varchar(255)
article2 : text

mydbxx userdata
id : int(11)
fullname : varchar(255)
email : varchar(255)
phone : varchar(255)
password : varchar(255)
priority : int(1)

mydbxx newsinfotabel
id : int(11)
name : varchar(255)
url : varchar(255)
title_class : varchar(255)
article_class : varchar(255)
article_link_class : varchar(255)

Figure 3.13: Database Schema Design

Figure 3.13 show Database Schema Design, as seen above there is no direct relationship that can be derived between the tables.

3.2.4 Sequence Diagram

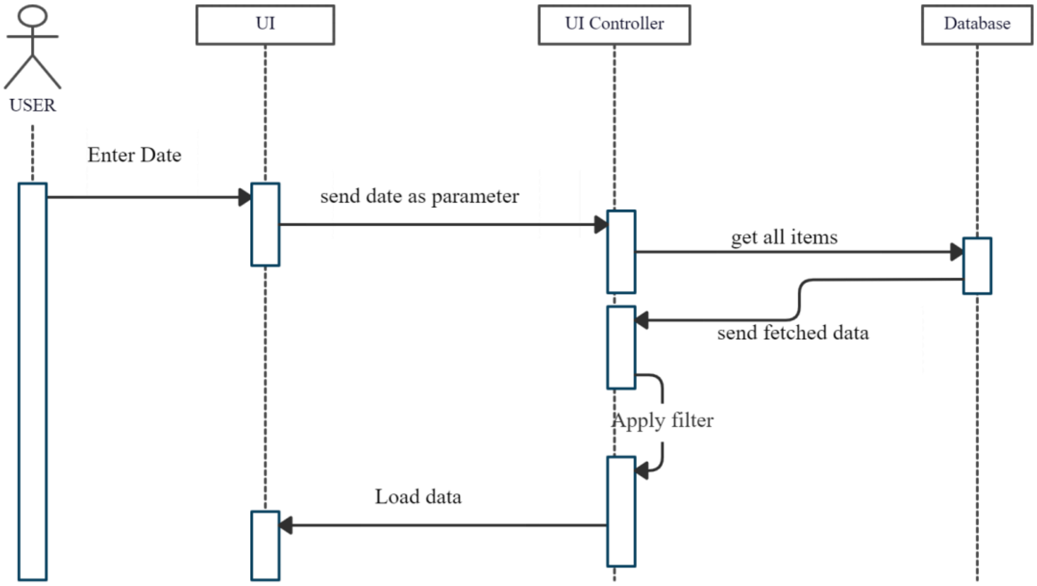


Figure 3.14: User Sequence Diagram

Assuming that the user has logged in with correct credentials. The things mentioned next is the sequence in which the user interacts with the web application. The first sequence as show by the sequence diagram above, begins when the user lands to the home page reffee ‘UI’ in the figure. The user then proceeds to enter the desired date of search into the search box. The entered data is processed by the ‘UI controller’, which consideres them to be parameters or arguments for being processed by various functions within them. The ‘UI controller’ processes a query that fetches data from the database. The fetched data is returned to the ‘UI controller’ that filters out data based on the user i.e(normal user or admin user). After the filteration process is complete, the ‘UI controller’ displays the data into the ‘UI’ such that the user is able to view it. The above mentioned set of sequence is repeated for each time the user searches by new date.

3.2.5 Use Case

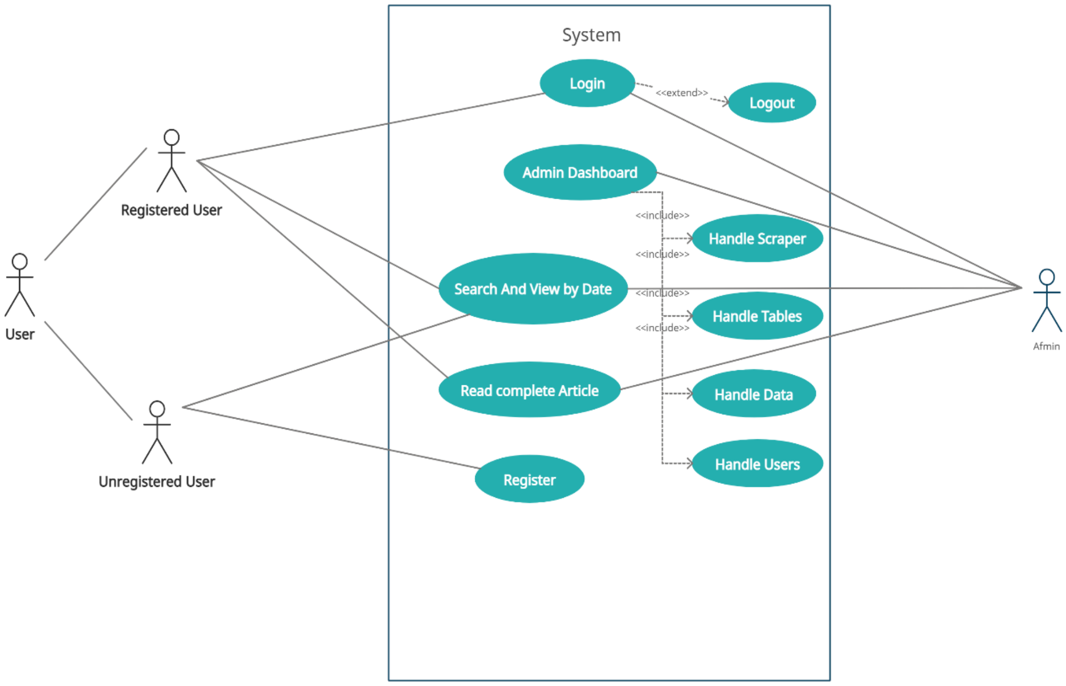


Figure 3.15: Use Case Diagram

In the above use case diagram, displays the two major factions of actors namely, User and the Admin. The ‘User’ actor can be divided into two types, Unregistered user and Registered user. Similarly the use case diagram showcases the 5 primary use cases: ‘Login’, ’Admin Dashboard’, ‘Search and View’, ‘Read Complete’ and ‘Register’ as 5 secondary use cases: ‘Handle Scraper’, ’Handle Table’, ‘Handle Data’, ‘Handle users’, and ‘Logut’.

Unregistered User - This actor is only allowed to do two things, i.e register or search and view by date.

Registered User - This actor is assumed to be registered in the systems database. He/she is allowed to Login into the system, search and view by date and also has the privilege to read complete article.

Admin -The admin can also be referred to as the super user as it is allowed to do all the things that was previously mentioned for the ‘User’ actor. Moreover the admin has access to the admin dashboard. The admin dashboard allows the manipulation of scraper, tables, and news and user data of the database

CHAPTER 4: Implementation and Testing

4.1 Implementation

4.1.1 Tools Used

Case Tools

- Lucidchart.com

  We used lucidchart.com to create all the diagrams needed, diagrams are listed but not limited to the following, ER-diagram, Use Case Diagram, Data Flow Diagram, Flow chart, Sequence diagram, etc. The ER-diagram show the logical relationships between

- Visual Studio Code

  Most of the code was written for this project were written in Visual Studio Code. Visual Studio Code is a very low weight and powerful code editor. The main reason for using Visual Studio Code is the inbuilt terminal that comes with it. Other editors like notepad, notepad++, Sublime Text or Atom do not have an inbuilt terminal, these editors require installation of external packages for terminals to be available. The terminal is very useful in installing the Node JS package. Since the web scraper is written in JavaScript, Node JS makes it easy to execute the JavaScript file trough the terminal. Similarly the 'live server' extension that can be added in Visual Studio Code makes coding and writing and testing HTML faster because of the live reload capability.

Programming Languages:

- PHP

  PHP plays an important part in this project. The web application is written in HTML within the PHP file. All the CRUD functionality available in the web application is written in PHP. PHP is used to connect and communicate with the local server that consists the database that is used by the web application.

- JavaScript

  JavaScript is used in this project to mostly develop the web scraper. The web scraper is written in plain vanilla JavaScript. The scraper does not render in the browser. The scraper is supposed to run in a server, thus Node JS is used to simulate a web server locally.Puppeteer which is a Node JS library that provides a high-level API to control headless Chrome or Chromium over the DevTools Protocol. The whole aspect of web scraping with is only possible if the 'puppeteer' package is installed in a system. Apart from using JavaScript to develop and execute the scraper, JS is also used within the web application. Within the web application, JS is used to make the web pages dynamic and responsive. For example, the index page of the web application features the use of Fetch API, the fetch() method is used to retrieve weather data from the Openweathermap weather api. Besides vanilla JS, jQuery is used to make use of Bootstrap elements responsive.

Database

- MySQL

MySQL is used to create and maintain the database required in this project.

4.1.2 Implementation Details of Modules

Database: The database that this web application uses locally is named, “mydbxx”. The Mysql database was created using the MySql administrator panel. The web application communicates to the said database and its tables through php. Whereas the webscraper accesses the database using the ‘mysql’ node js package. The tables inside the database can be showcased as shown below:

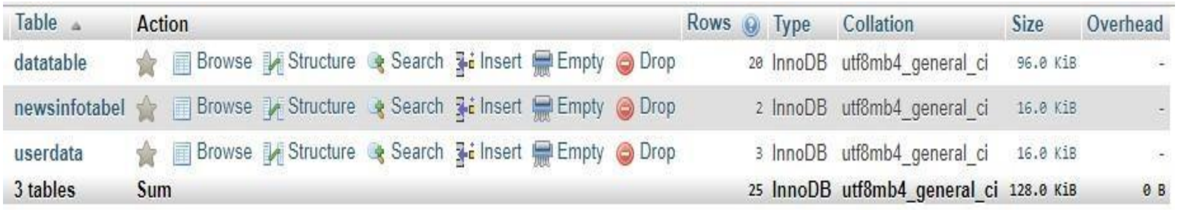
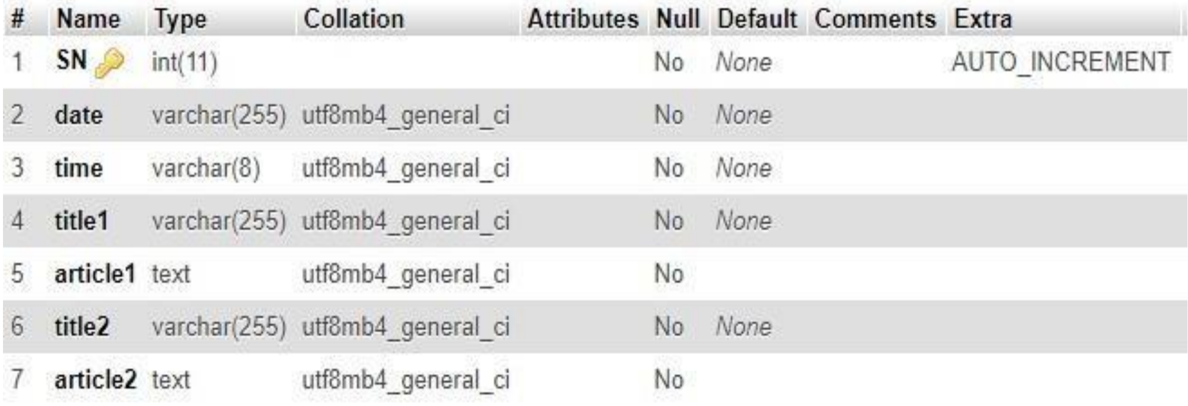


Table ▲	Action	Rows	Type	Collation	Size	Overhead
datatable	★ Browse Structure Search Insert Empty Drop	20	InnoDB	utf8mb4_general_ci	96.0 KiB	-
newsinfotabel	★ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
userdata	★ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	16.0 KiB	-
3 tables	Sum	25	InnoDB	utf8mb4_general_ci	128.0 KiB	0 B

Figure 4.1: Database 'mydbxx’

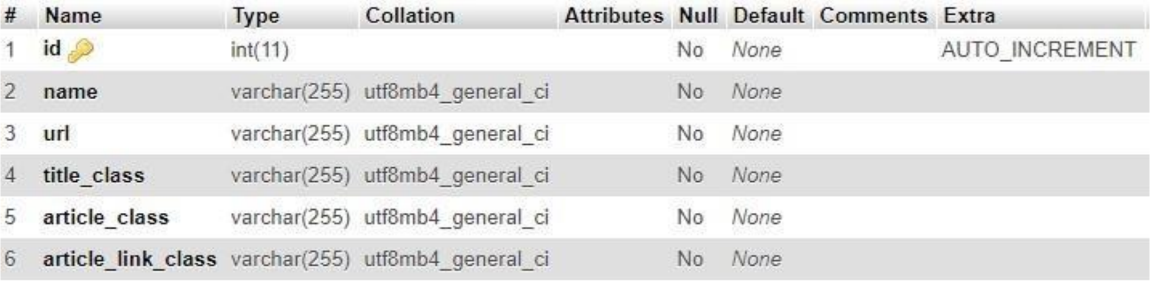
The table named ‘datatable’ is where the scraped data (titles and articles are stored).



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	SN	int(11)			No	None		AUTO_INCREMENT
2	date	varchar(255)	utf8mb4_general_ci		No	None		
3	time	varchar(8)	utf8mb4_general_ci		No	None		
4	title1	varchar(255)	utf8mb4_general_ci		No	None		
5	article1	text	utf8mb4_general_ci		No			
6	title2	varchar(255)	utf8mb4_general_ci		No	None		
7	article2	text	utf8mb4_general_ci		No			

Figure 4.2: Table 'datatable'

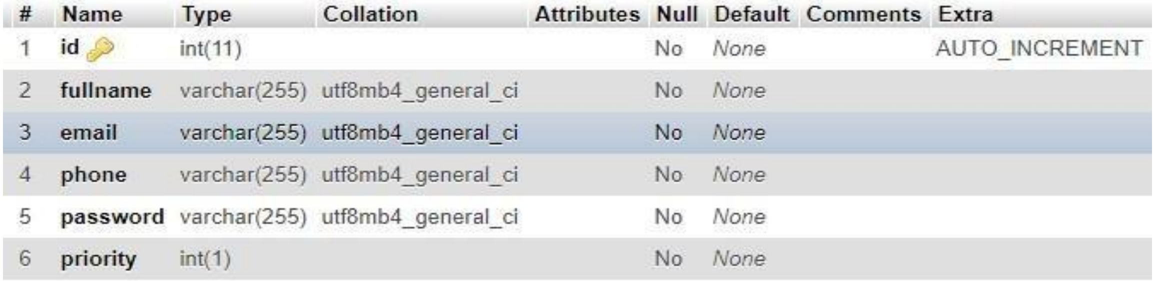
The table named ‘newsinfotable’ holds the parameters used by the web scrapers crawler.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_general_ci		No	None		
3	url	varchar(255)	utf8mb4_general_ci		No	None		
4	title_class	varchar(255)	utf8mb4_general_ci		No	None		
5	article_class	varchar(255)	utf8mb4_general_ci		No	None		
6	article_link_class	varchar(255)	utf8mb4_general_ci		No	None		

Figure 4.3: Table 'newsinfotabel'

Finally the table named ‘userdata’ stores datas related to users of the web application.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	fullname	varchar(255)	utf8mb4_general_ci		No	None		
3	email	varchar(255)	utf8mb4_general_ci		No	None		
4	phone	varchar(255)	utf8mb4_general_ci		No	None		
5	password	varchar(255)	utf8mb4_general_ci		No	None		
6	priority	int(1)			No	None		

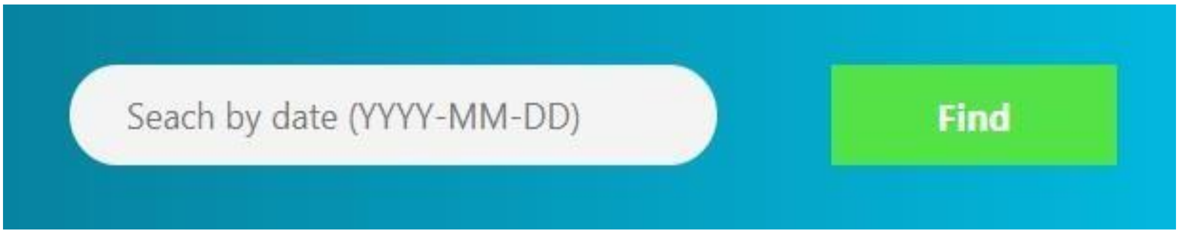
Figure 4.4: Table 'userdata'

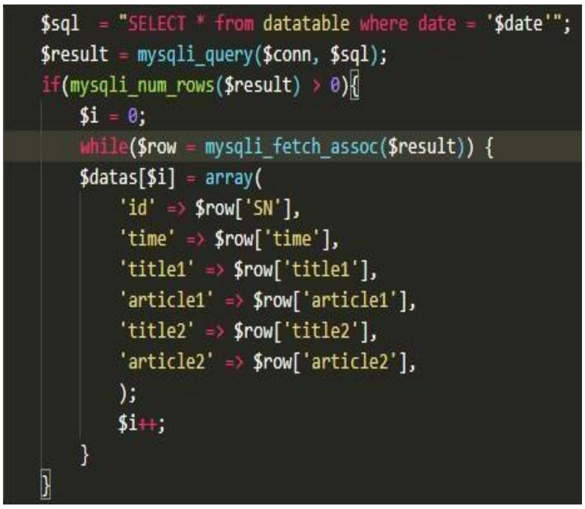
Web Application:

The web application was developed using plain html, css,bootstrap and js for the frontend and php for the backend. The homepage is rendered by the index.php for all types of user. The route is the path in the URL bar of the browser. For example:

Index.php “localhost/loginpage/index.php” is entered similarly to reach the login page “localhost/loginpage/login.php” is entered. The other URLs that can be rendered without a user being logged in are the ’localhost/loginpage/aboutus.php’, ‘localhost/loginpage/contactus.php’, ‘localhost/loginpage/register.php’ and ‘localhost/loginpage/view.php’. The rest of the webpages of the web application can only be rendered when a user sessions are set in place.

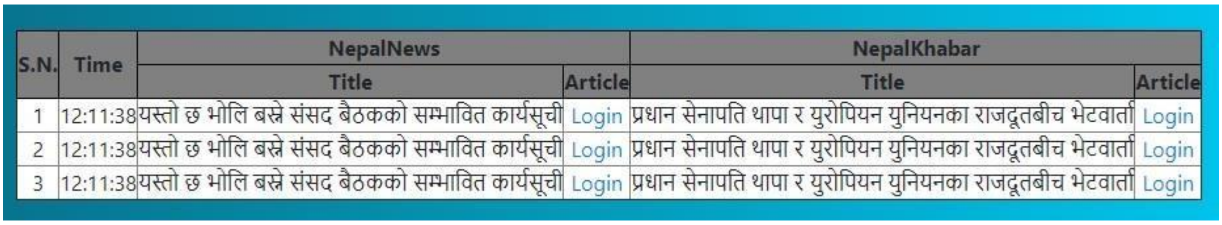
The figure below shows the search box of the home page that takes ‘date’ as an input which is then processed by two blocks of code as showcased in the next two images:





```
$sql  = "SELECT * from datatable where date = '$date'";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result) > 0){
    $i = 0;
    while($row = mysqli_fetch_assoc($result)) {
    $datas[$i] = array(
        'id' => $row['SN'],
        'time' => $row['time'],
        'title1' => $row['title1'],
        'article1' => $row['article1'],
        'title2' => $row['title2'],
        'article2' => $row['article2'],
        );
        $i++;
    }
}
```

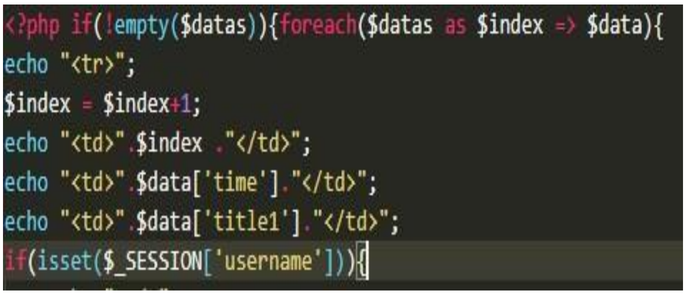
The blocks of code are used in view.php such as to display the fetched data from the database to the user as such:



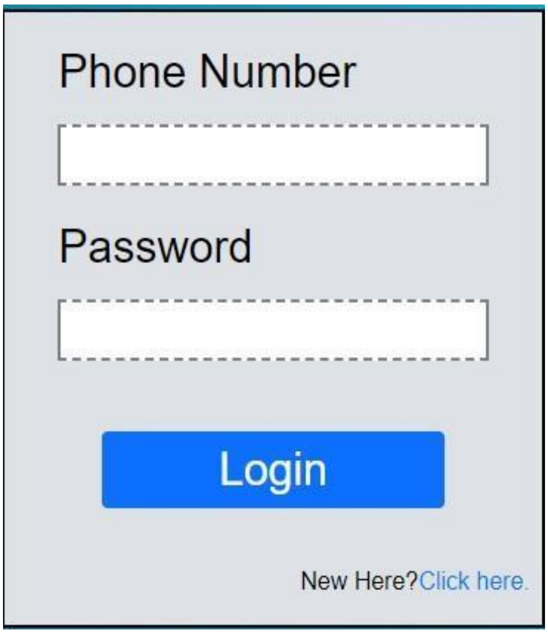
S.N.	Time	NepalNews		NepalKhabar	
		Title	Article	Title	Article
1	12:11:38	यस्तो छ भोलि बस्ने संसद बैठकको सम्भावित कार्यसूची	Login	प्रधान सेनापति थापा र युरोपियन युनियनका राजदूतबीच भेटवार्ता	Login
2	12:11:38	यस्तो छ भोलि बस्ने संसद बैठकको सम्भावित कार्यसूची	Login	प्रधान सेनापति थापा र युरोपियन युनियनका राजदूतबीच भेटवार्ता	Login
3	12:11:38	यस्तो छ भोलि बस्ने संसद बैठकको सम्भावित कार्यसूची	Login	प्रधान सेनापति थापा र युरोपियन युनियनका राजदूतबीच भेटवार्ता	Login

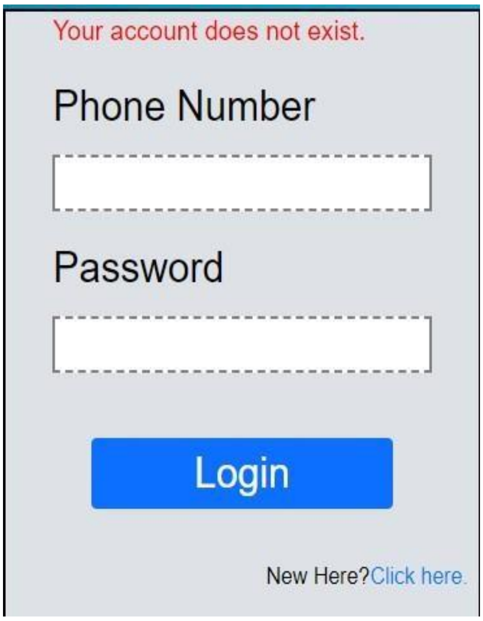
Login:

Shown below is the ‘login.php’ page, first with no error message, then with the error message displayed during a failed log-in attempt.

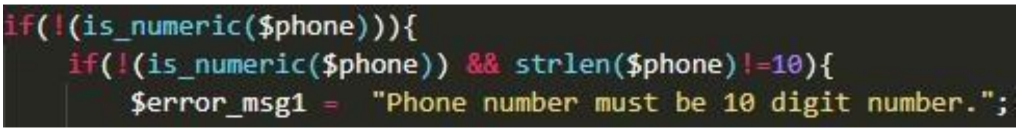


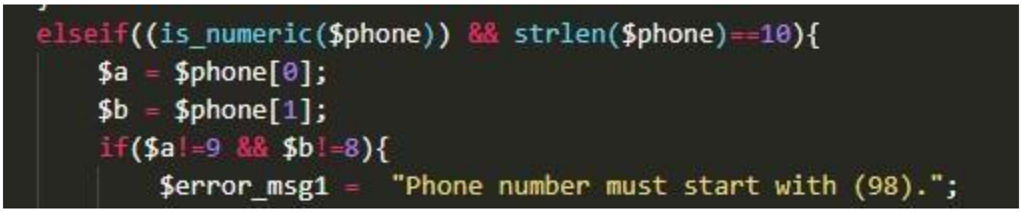
```
<?php if(!empty($datas)){foreach($datas as $index => $data){
echo "<tr>";
$index = $index+1;
echo "<td>".$index ."</td>";
echo "<td>".$data['time']."</td>";
echo "<td>".$data['title1']."</td>";
if(isset($_SESSION['username'])){
```




Your account does not exist.

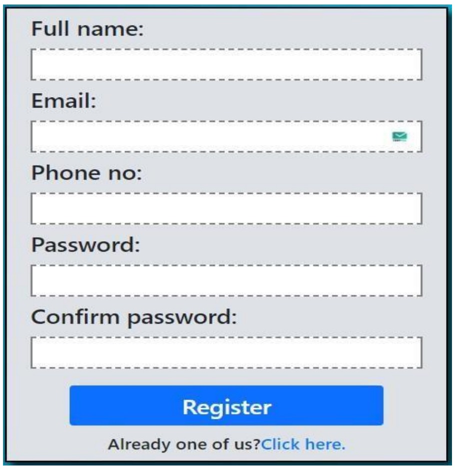
The entered phone number and password is checked using php, by first comparing the ‘phonenumber’ to ‘phone’stored in database inside the ‘userdata’. The phone number is validate in such a way that the only valid phone number is a phone number that have initial characters 9 and 8. The phone number must have length equals to 10. To check the password, the inbuilt password_verify(password, hashed_password) php method is called. This method is used to compare the hashed value generated by password_hash(password, PASSWORD_DEFAULT) which uses the default encryption algorithm. The password_hash(password, PASSWORD_DEFAULT) method is used in the registeration form of register.php which serves to add new users to the database.


```
if(!(is_numeric($phone))){
    if(!(is_numeric($phone)) && strlen($phone)!=10){
        $error_msg1 =  "Phone number must be 10 digit number.";
```


```
  elseif((is_numeric($phone)) && strlen($phone)==10){
      $a = $phone[0];
      $b = $phone[1];
      if($a!=9 && $b!=8){
          $error_msg1 =  "Phone number must start with (98).";
```

Register:

Shown below is the ‘register.php page’, first with no error message, then with the error message displayed during a failed register attempt.



Full name:

Email:

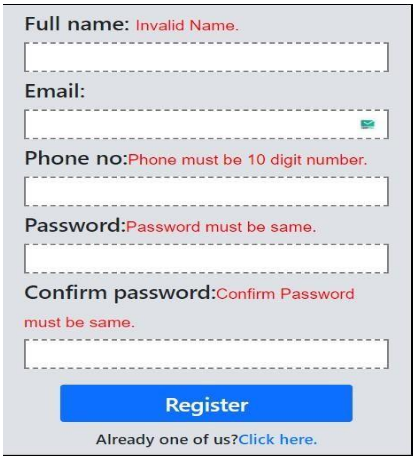
Phone no:

Password:

Confirm password:

Register

Already one of us?Click here.



Full name: Invalid Name.

Email:

Phone no:Phone must be 10 digit number.

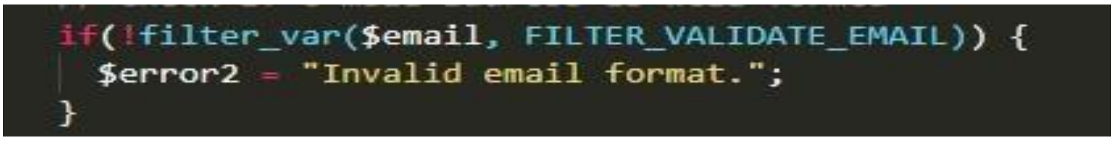
Password:Password must be same.

Confirm password:Confirm Password must be same.

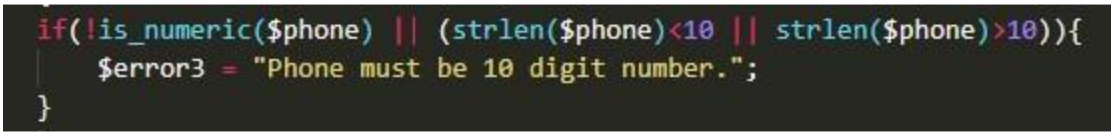
Register

Already one of us?Click here.

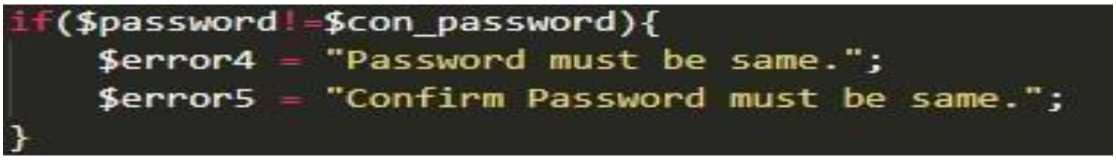
The serverside validation allows Full name to be a string with white space but no number or special symbols. The email must be the standard email pattern that consists of '@' and '.com'. The phone number is strictly validated such that only a 10 digit numeric string that starts with 9 and 8 is allowed. The password and confirm password field both must have same input.The password is hashed into a 60 character random string by using the password_hash(password, PASSWORD_DEFAULT) method of php. the hased and if all necessary validations are met, all the data is stored in the 'userdata' table of 'mydbxx' database.



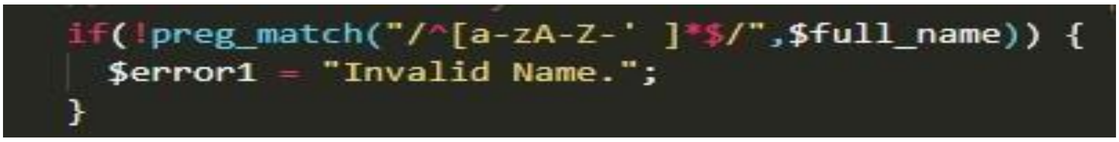
```
if(!filter_var($email, FILTER_VALIDATE_EMAIL)) {
  $error2 = "Invalid email format.";
}
```



```
if(!is_numeric($phone) || (strlen($phone)<10 || strlen($phone)>10)){
    $error3 = "Phone must be 10 digit number.";
}
```



```
if($password!=$con_password){
    $error4 = "Password must be same.";
    $error5 = "Confirm Password must be same.";
}
```

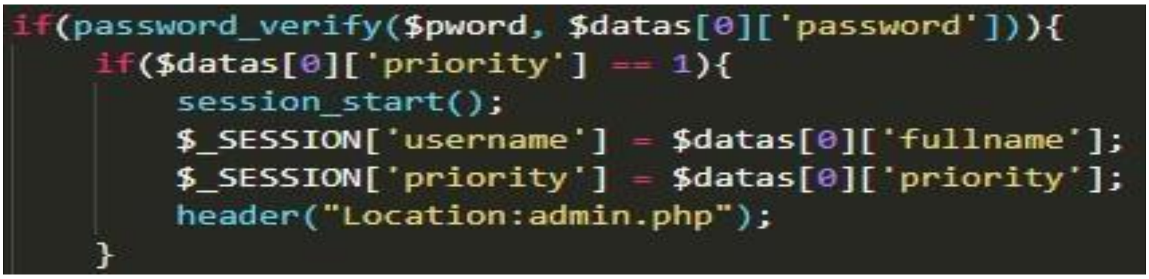


```
if(!preg_match("/^[a-zA-Z-' ]*$/",$full_name)) {
  $error1 = "Invalid Name.";
}
```

Sessions:

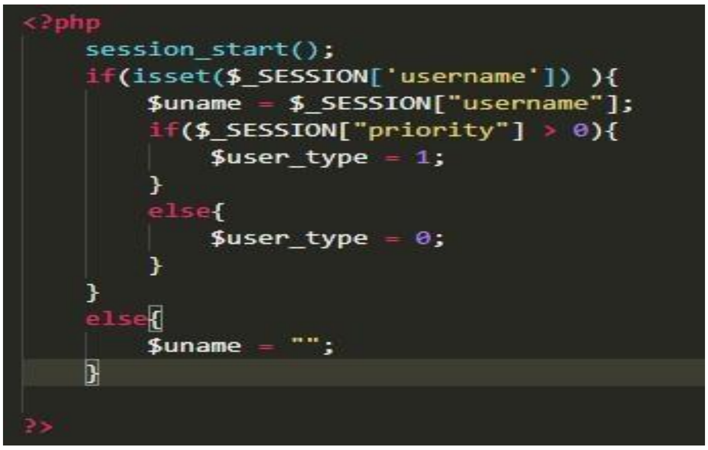
The session is set when a user enters valid phone number and password in the login form. The fullname and the priority of the user that is set in the database are taken as session variable. The session variables are used to redirect users and give the necessary privellages.

The session_start() is a inbuilt php function that starts the session. The session variables are assigned using $_SESSION['session_variable_naem'], as shown below:



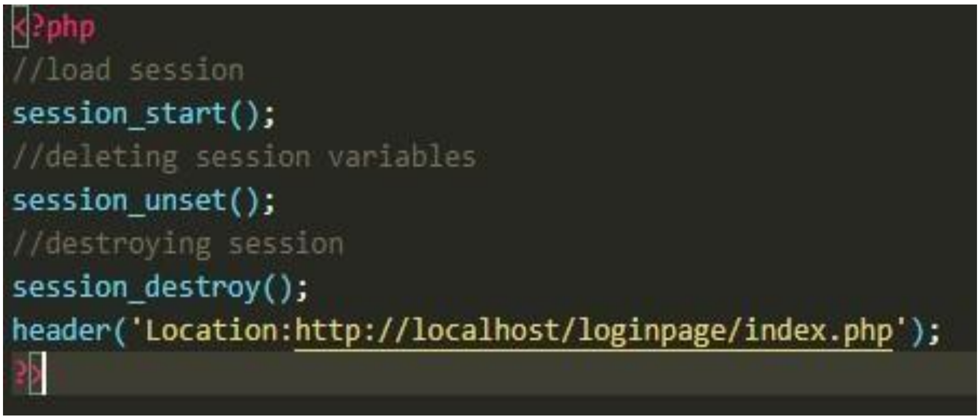
```
if(password_verify($pword, $datas[0]['password'])){
    if($datas[0]['priority'] == 1){
        session_start();
        $_SESSION['username'] = $datas[0]['fullname'];
        $_SESSION['priority'] = $datas[0]['priority'];
        header("Location:admin.php");
    }
```

The session is verified by 'session_verification.php' which consists the following code block.



```
<?php
    session_start();
    if(isset($_SESSION['username']) ){
        $uname = $_SESSION["username"];
        if($_SESSION["priority"] > 0){
            $user_type = 1;
        }
        else{
            $user_type = 0;
        }
    }
    else{
        $uname = "";
    }

?>
```

Session can be unset and destroyed when a user clicks on the logout button on the navbar. Upon clicking the logout button 'logout.php' is executed.

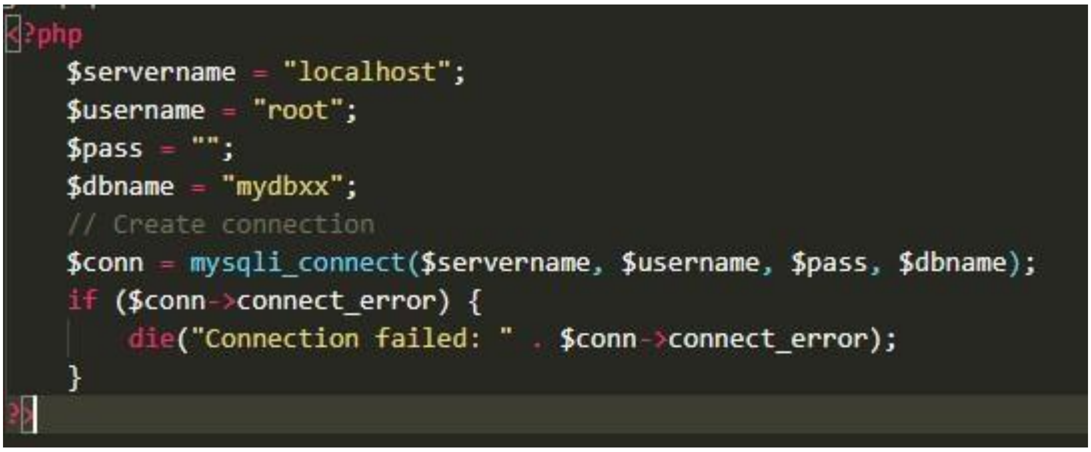


```
<?php
//load session
session_start();
//deleting session variables
session_unset();
//destroying session
session_destroy();
header('Location:http://localhost/loginpage/index.php');
?>
```

4.2 Testing

For the testing of this web application, unit testing was don’t manually as no frameworks were used such that automated testing was not possible.

To begin testing the web application, the necessary connections to the database were made. The configurations were done with the file ‘configure.php’ which contains code block shown below:



```php
<?php
    $servername = "localhost";
    $username = "root";
    $pass = "";
    $dbname = "mydbxx";
    // Create connection
    $conn = mysqli_connect($servername, $username, $pass, $dbname);
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }
?>
```

4.2.1 Test Cases for Unit testing

A. Register Form => Full Name field unit testing

For this testing, the test data taken are “Momik Shrestha”, “   , “12345” and “mom12k” for 4 test scenarios.

Table 4.1: Fullname Field Unit Testing

SN	Action	Input	Expected Outcome	Actual Outcome	Result	Comment
1	Enter Valid Full Name, press register.	Momik Shrestha	No error message	No warning message	Pass	Momik Shrestha is a valid name
2	Do not fill full name, press register.	__________	“Cannot be empty.” error message must be displayed	“Cannot be empty.” error message is displayed	Pass	Full name cannot be empty

3	Enter numeric value as name, press register.	12345	“Invalid Name” error message must be displayed	“Invalid Name” error message is displayed	Pass	Full name cannot be numeric
4	Enter alphanumeric and symbolic value as name, press register.	Mom12K	“Invalid Name” error message must be displayed	“Invalid Name” error message is displayed	Pass	Full name cannot be alphanumeric

B. Register Form => Email field unit testing

For this unit testing, the test data taken are “Momik12339@gmail,com”, “ “ and “Momik1233” for 3 test scenarios.

Table 4.2: Email Field Unit Testing

SN	Action	Input	Expected Output	Actual Outcome	Result	Comment
1	Enter Valid email address, press register.	Momik12339@gmail.com	No error message	No warning message	Pass	Momik12339@gmail.com is a valid email address
2	Do not fill email address, press register.	__________	“Email is required.” error message must displayed	“Email is required.” error message is displayed	Pass	Email cannot be empty
3	Enter invalid email, press register.	Momik123	“Enter proper email” error message must be displayed	“Enter proper email” error message is displayed	Pass	Email has to consist ‘@’ and ‘.com’

C. **Register Form => Phone Number unit testing**

For this testing, the test data taken are “9860222338”, “ “ , “9860222xyz” , “9860222” , “98602223388” and “1111111111” for 6 test scenarios.

Table 4.3: Phone Number Unit Testing

SN	Action	Input	Expected Output	Actual Outcome	Result	Comment
1	Enter valid phone number, press register.	9860222338	No error message.	No error message.	Pass	9860222338 is a valid phone number.
2	Do not fill phone number, press register.	__________	“Phone is required.” error message must be displayed.	“Phone is required.” error message is displayed.	Pass	Phone number field cannot be empty.
3	Enter phone number with string, press register.	9860222xyz	“Enter proper phone number” error message must be displayed.	“Enter proper phone number” error message is displayed.	Pass	Phone number must be numeric.
4	Enter phone number with length less than 10, press register.	9860222	“Phone number must be 10 digit number” error message must be displayed.	“Phone number must be 10 digit number” error message is displayed.	Pass	Phone number must have length = 10.

5	Enter phone with invalid start	1111111111	“Phone number must start with (98)” error message must be displayed.	“Phone number must start with (98)” error message is displayed.	Pass	Phone number must start with a 98.
6	Enterr Phone number with special symbol like @, $, %, ! and press login	986@50205	“Phone number must be 10 digit number” error must be displayed	“Phone number must be 10 digit number” error is displayed.	Pass	Phone number must be numeric.
7	Enter phone number with length more than 10, press register.	98602223388	“Phone number must be 10 digit number” error message must be displayed	“Phone number must be 10 digit number” error message is displayed.	Pass	Phone number must have length = 10.

D. Register Form => Password and Confirm Password testing

For this testing, the sets of test data taken are “Password1, Password1”, “ , Password1”, “ , Password1,” and “Password1, Password2” for 4 test scenarios**.**

Table 4.4: Password Unit Testing

SN	Action	Inputs	Expected Outcome	Actual Outcome	Result	Comment
1	Enter same password, same confirm password, press register.	Password1, Password1	No error message	No error message	Pass	Both password fields need to be same
2	Enter empty password, and non- empty password, press register.	________, Password1	“Password required” and “confirm password must be same” error message must be displayed.	“Password required” and “confirm password must be same” error message is displayed.	Pass	Both password fields need to be same
3	Enter non-empty password, and empty password, press register.	Password1, _________	“Password must be same”, and “confirm password required” error message must be displayed.	“Password must be same”, and “confirm password required” error message is displayed.	Pass	Both password fields need to be same
4	Enter different values in password	Password1, Password2	“Password must be same”, and	“Password must be same”, and	Pass	Both password

	field and confirm password field, press register		“confirm password must be same” error message must be displayed.	“confirm password must be same” error message is displayed.		fields need to be same

4.2.2 Test Cases for System testing

A. Login System Testing

a. Admin Login System Testing

Pre-requisites: Admin credentials must exist in database.

Table 4.5: User Login System Testing

SN	Action	Inputs	Expected Outcome	Actual Outcome	Test Result	Test Comment
1	Enter valid phone number, valid password, press login	9860222338 admin	Redirected to admin dashboard	Redirected to admin dashboard (admin.php)	Pass	Admin dashboard can only be accessed by 9860222338, admin.
2	Enter valid phone number, invalid password, press login	9860222338 Apple	Error message “Account doesn’t exist.” pops up	login.php loads with error message	Pass	Admin dashboard cannot be accessed.
3	Enter invalid phone number,	9860222558 Admin	Error message “Account doesn’t exist.” pops up	login.php loads with error message	Pass	Admin dashboard cannot be accessed.

	valid password, press login					
4	Enter invalid phone number, invalid password, press login	9860222558 Apple	Error message “Account doesn’t exist.” pops up	login.php loads with error message	Pass	Admin dashboard cannot be accessed.

b. User Login System Testing

Pre-requisites: Admin credentials must exist in database.

Table 4.6: Admin Login System testing

SN	Action	Inputs	Expected Outcome	Actual Outcome	Test Result	Test Comment
1	Enter valid phone number, valid password, press login	9860222337 user	Redirected to Home page.	Redirected to admin dashboard (index.php)	Pass	User login is possible
2	Enter valid phone number, invalid password, press login	9860222337 Apple	Error message “Account doesn’t exist.” pops up	login.php loads with error message	Pass	User login is not possible with false credentials.

3	Enter invalid phone number, valid password, press login	9860222558 user	Error message “Account doesn’t exist.” pops up	login.php loads with error message	Pass	User login is not possible with false credentials.
4	Enter invalid phone number, invalid password, press login	9860222558 Apple	Error message “Account doesn’t exist.” pops up	login.php loads with error message	Pass	User login is not possible with false credentials.

B. Scraper System Testing

Pre-requisites: node JS, internet connection.

Table 4.7: Scraper System Testing

SN	Action	Internet Speed	Scrape Success	Scrape execution time	Database Insertion	Number of data inserted	Expected Outcome	Actual Outcome
1	Run scrape.js	<=10Mbps	NO	>1 minute	NO	-N/A-	Connection Timeout	Connection Timeout
2	Run scrape.js	>10 Mbps && <15Mbps	NO	>1 minute	YES	Empty	Connection Timeout	Connection Timeout
3	Run scrape.js	>15Mbps && <25Mbps	YES	~27 seconds	YES	1	2 records added	1 record added
3	Run scrape.js	>25Mbps	YES	~18 seconds	YES	2	2 records added	2 records added

4	Run scrape.js	>35 Mbps	YES	~13 seconds	YES	2	2 records added	2 records added
5	Run scrape.js	>45Mbps	YES	~13 seconds	YES	2	2 records added	2 records added

Comment: Speed over 35Mbps has no adverse affect to the scraping function.

C. Search System Testing

Table 4.8: Search System Testing

SN	Action	Inputs	Expected Outcome	Actual Outcome	Test Result	Test Comment
1	Enter valid date format (YYYYMMDD) and valid date (<feb 20 and >current date).	2021-04-20	Redirect to result (view.php)	Redirected to result (view.php)	Pass	Result can only be viewed of date is in correct format
2	Enter invalid date format (YYYYMMDD) and valid date (<feb 20 and >current date).	2021/02/21	“enter proper date format” error message must pop up	“enter proper date format” error message does not pop up	Fail	Haven’t written code for this logic. but possible
3	Enter valid date format (YYYY-MM-DD) and invalid date (<feb 20 and >current date).	2021-01-01	“Enter valid date” error message must pop up.	“Enter valid date” error message pops up.	Pass	Simple if condition was implemented.

4	Enter invalid date format (YYYY-MM-DD) and invalid date (<feb 20 and >current date).	2020/01/01	“enter proper date format” error message must pop up	“enter proper date format” error message does not pop up	Fail	Haven’t written code for this logic. but possible

4.3 Evaluation

Evaluation of Functional Requirements.

Table 4.9: Functional Requirement Evaluation

Functional Requirements	Requirement Met?	Comment
FR1	YES	An unregistered user is able to search by data and view title.
FR2	YES	An unregistered user is able to register if desires
FR3	YES	A registered user is able to search by date and view title.
FR4	YES	A registered user is able to view related new articles
FR5	YES	A registered user is able to logout.
FR6	YES	Only the registered admin is able to access admin dashboard, no one else can.
FR7	YES	Partial client side form validation is implemented through php.
FR8	YES	The Scraper is able to crawl 2 new portals in a single run.
FR9	YES	The scraped data is successfully stored in a database.

Evaluation of Non-functional requirements.

Table 4.10: Non-Functional Requirement Evaluation

Non-Functional Requirements	**Requirements Met?**	**Comment**
NFR1	YES	The web application is considerably user friendly and easy to navigate.
NFR2	YES	The web application supports mobile view.
NFR3	YES	The scraping script takes maximum 30 seconds to be complete.
NFR4	NO	Logic for data redundancy has not been implemented.

CHAPTER 5: Conclusion and Future Recommendations

5.1 Lesson learnt/Outcome

The primary object of this project was to understand web scraping and web scraping tools and technologies and the secondary objective of this project was to have a better understanding of HTML, CSS, JavaScript, PHP and other web application building technologies. As the completion of this project has come to terms, the outcome is visible as well. We were able to learn about web scraping which helped us build a scraping script of our own. We then proceeded to build a working web application that relied on the scraped data. We learned many things during the course building this project, from the development technologies to the scheduling of the project phases. The communication and teamwork of the team members developed overtime as the project development time went on as well. We have come to learn about the things it take to complete a project.

5.2 Conclusion

The project proposal suggested that the project consists a web scraper and a web application that is able to make use of the huge amount of data prepared by the scraper. We used JavaScript to create a simple yet effective web scraper that can successfully scrape data based on crawl parameters. Similarly we successfully built a web application that can perform CRUD functionalities based on the user's desire. The web application was successfully built using HTML, CSS and PHP. We've managed to fulfill our listed objectives. Various types of testing were successfully conducted as well. The end result of our project can be seen in the 'APPENDICES' chapter next.

5.3 Future Recommendations

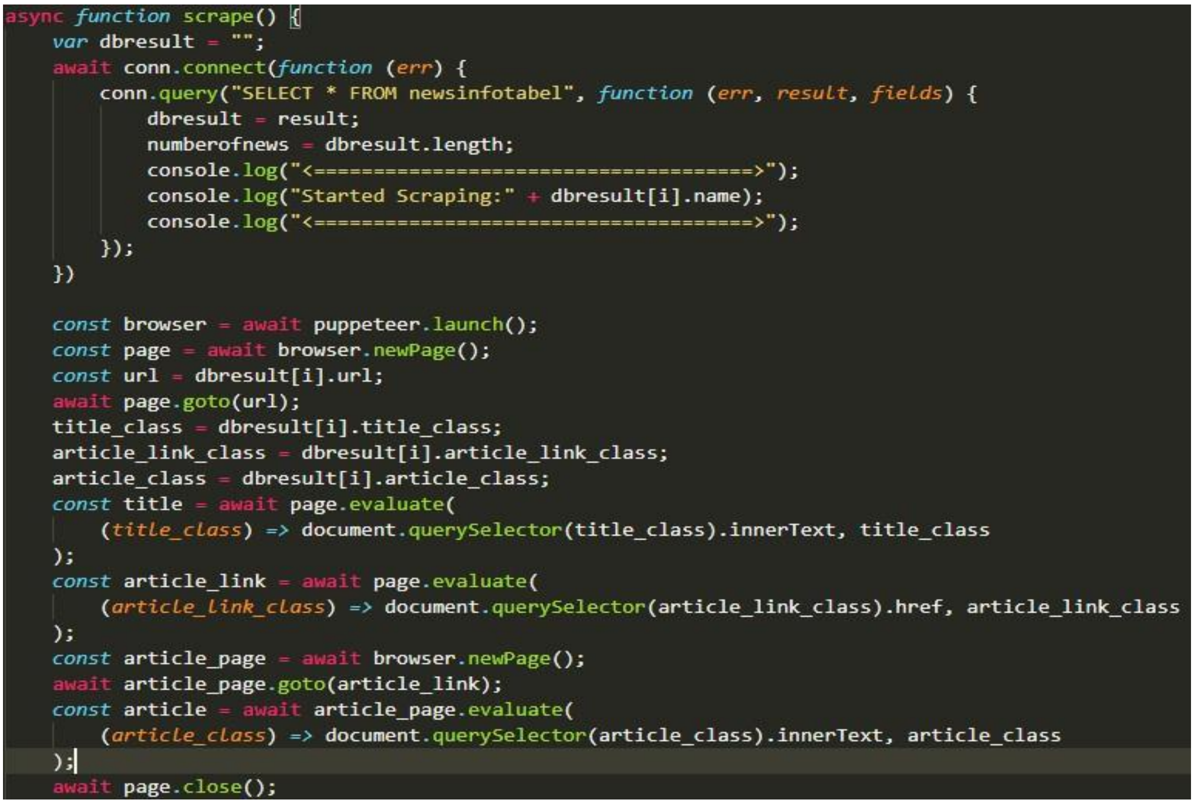
The created system works as per intended and works efficiently but new features can be added in the near future for increasing the user experience for both the end user as well as the administrator. The future ventures for this project can be listed below:

- Enhance database: The completion of this project has highlighted the flaws in the current database schemas. In the near future, the database can be made better.
- Enhance UI: The current UI uses no dedicated framework, nor is it designed by experienced designers. It is safe to say that better UI designs are an integral future venture.
- Optimized scraper: The current web scraper despite doing its job correctly has one major flaw, that being its dependency on high speed internet. An optimized version of the scraper

can be developed such that the scrape time is lowered and dependency on internet speed is decreased in the future.

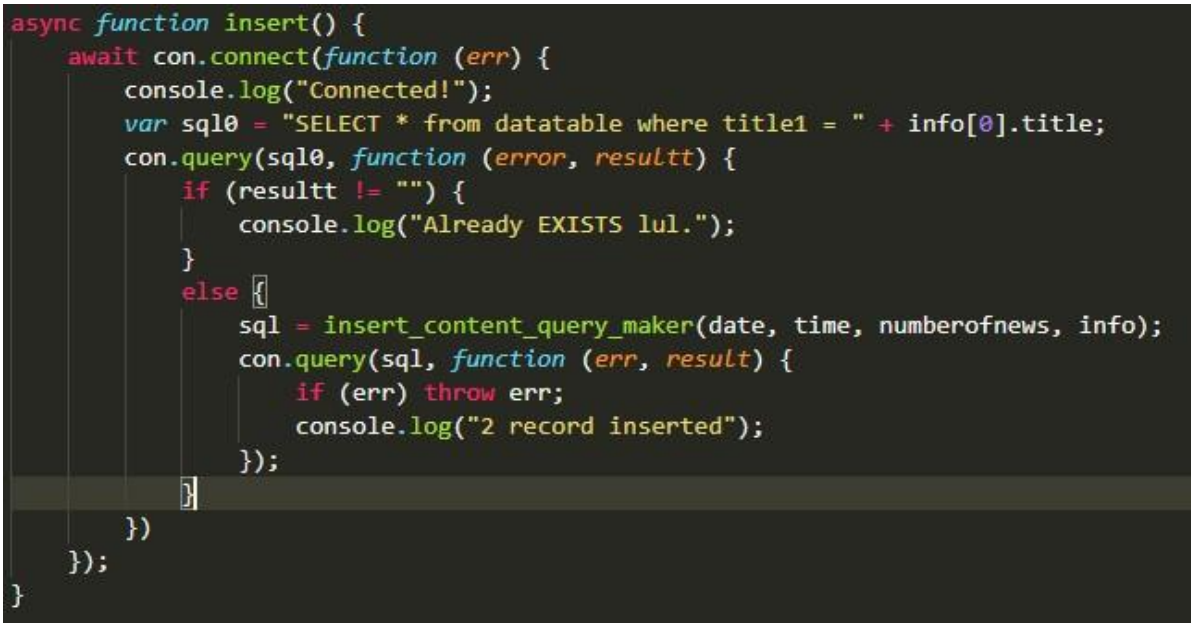
- Enhance scraper: Right now the scraper servers minimum usage, however the scraper can be enhanced to scrape complex sites, scrape more data and be built such that the scraper scrapes data in shorter intervals as well as scraper intelligently.

APPENDICES



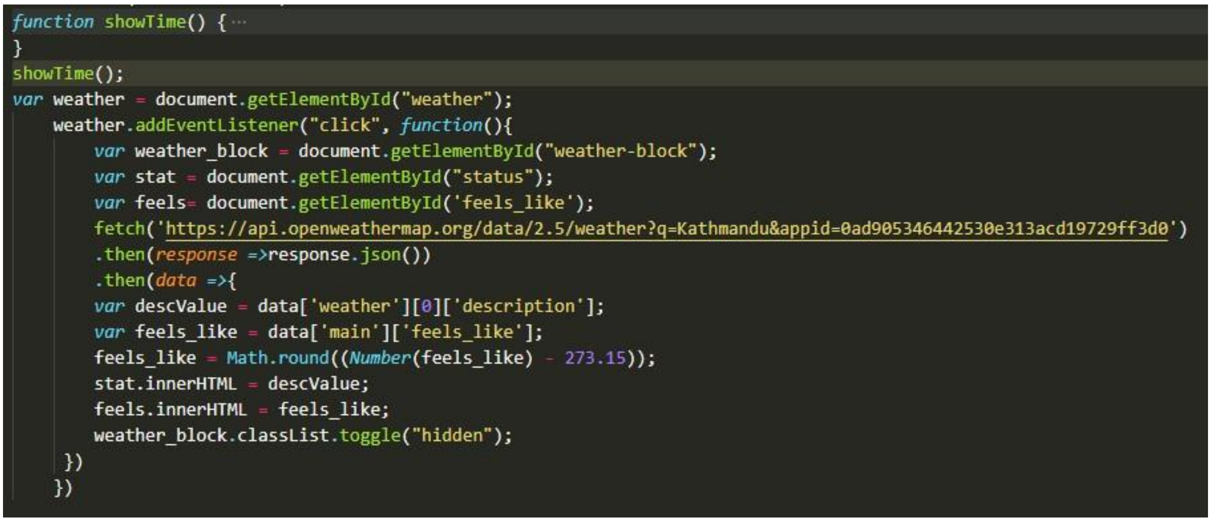
```
async function scrape() {
    var dbresult = "";
    await conn.connect(function (err) {
        conn.query("SELECT * FROM newsinfotabel", function (err, result, fields) {
            dbresult = result;
            numberofnews = dbresult.length;
            console.log("<====================================>");
            console.log("Started Scraping:" + dbresult[i].name);
            console.log("<====================================>");
        });
    })

    const browser = await puppeteer.launch();
    const page = await browser.newPage();
    const url = dbresult[i].url;
    await page.goto(url);
    title_class = dbresult[i].title_class;
    article_link_class = dbresult[i].article_link_class;
    article_class = dbresult[i].article_class;
    const title = await page.evaluate(
        (title_class) => document.querySelector(title_class).innerText, title_class
    );
    const article_link = await page.evaluate(
        (article_link_class) => document.querySelector(article_link_class).href, article_link_class
    );
    const article_page = await browser.newPage();
    await article_page.goto(article_link);
    const article = await article_page.evaluate(
        (article_class) => document.querySelector(article_class).innerText, article_class
    );
    await page.close();
```

Figure 6.1: scrape() JS function



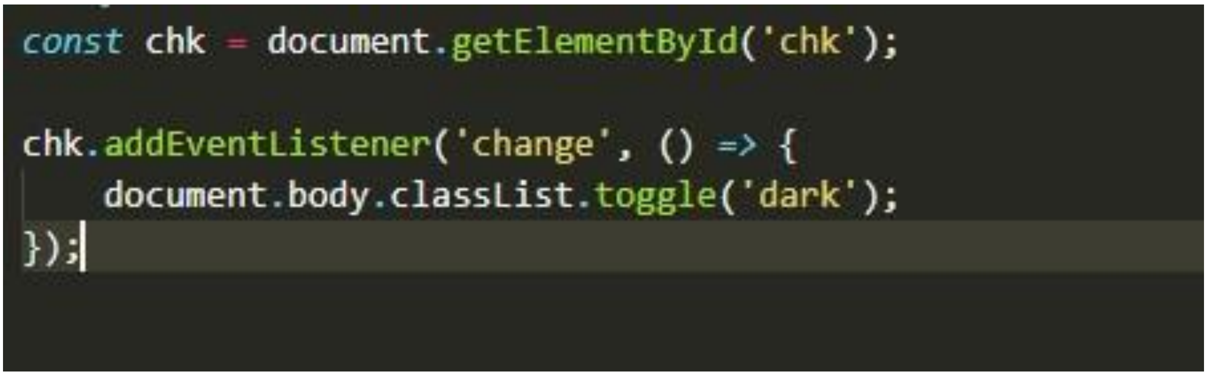
```
async function insert() {
    await con.connect(function (err) {
        console.log("Connected!");
        var sql0 = "SELECT * from datatable where title1 = " + info[0].title;
        con.query(sql0, function (error, resultt) {
            if (resultt != "") {
                console.log("Already EXISTS lul.");
            }
            else {
                sql = insert_content_query_maker(date, time, numberofnews, info);
                con.query(sql, function (err, result) {
                    if (err) throw err;
                    console.log("2 record inserted");
                });
            }
        })
    });
}
```

Figure 6.2: insert() JS function



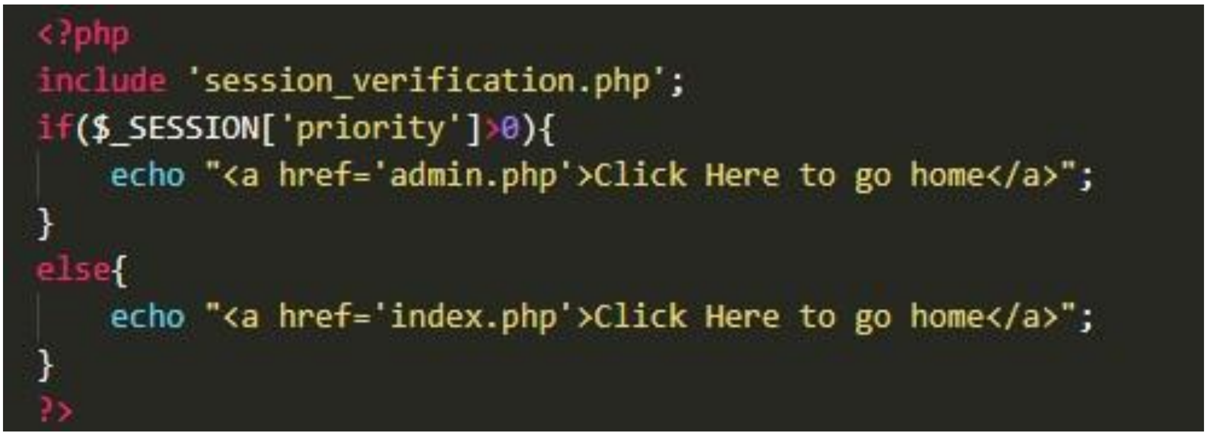
```
function showTime() {···
}
showTime();
var weather = document.getElementById("weather");
    weather.addEventListener("click", function(){
        var weather_block = document.getElementById("weather-block");
        var stat = document.getElementById("status");
        var feels= document.getElementById('feels_like');
        fetch('https://api.openweathermap.org/data/2.5/weather?q=Kathmandu&appid=0ad905346442530e313acd19729ff3d0')
        .then(response =>response.json())
        .then(data =>{
        var descValue = data['weather'][0]['description'];
        var feels_like = data['main']['feels_like'];
        feels_like = Math.round((Number(feels_like) - 273.15));
        stat.innerHTML = descValue;
        feels.innerHTML = feels_like;
        weather_block.classList.toggle("hidden");
     })
    })
```

Figure 6.3: Weather module



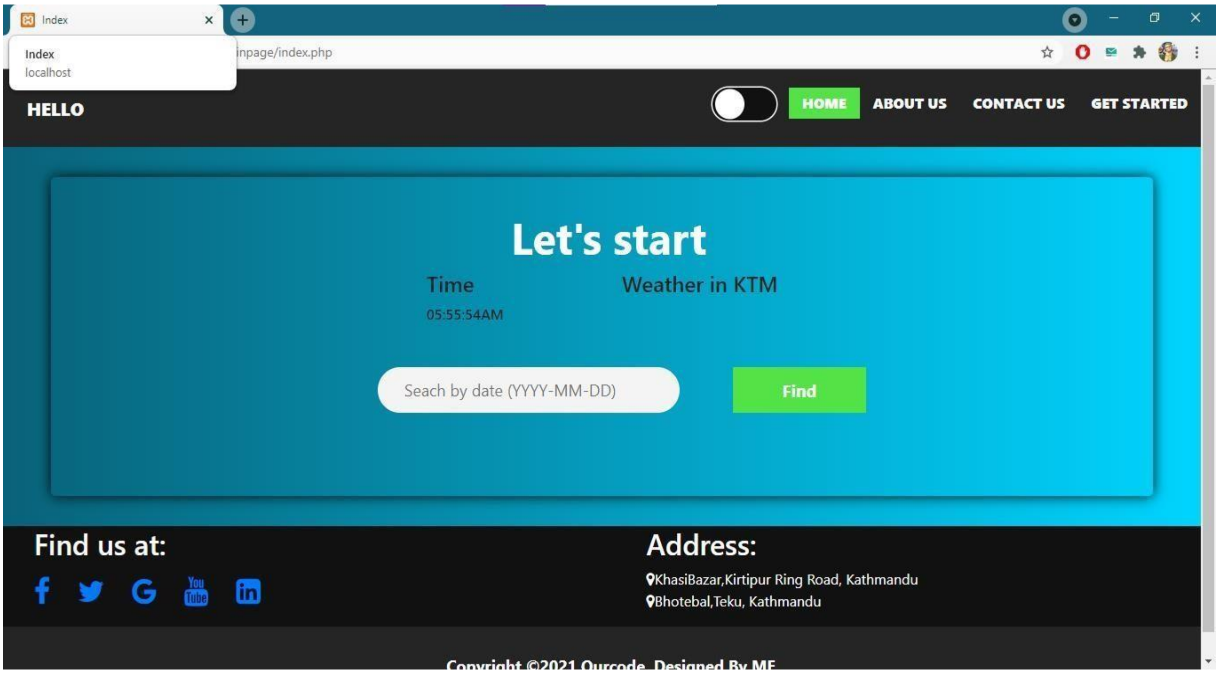
```
const chk = document.getElementById('chk');

chk.addEventListener('change', () => {
    document.body.classList.toggle('dark');
});
```

Figure 6.4: Darkmode module



```
<?php
include 'session_verification.php';
if($_SESSION['priority']>0){
    echo "<a href='admin.php'>Click Here to go home</a>";
}
else{
    echo "<a href='index.php'>Click Here to go home</a>";
}
?>
```

Figure 6.5: Error module



Index

Index
localhost

inpage/index.php

HELLO

HOME ABOUT US CONTACT US GET STARTED

Let's start

Time

05:55:54AM

Weather in KTM

Seach by date (YYYY-MM-DD)

Find

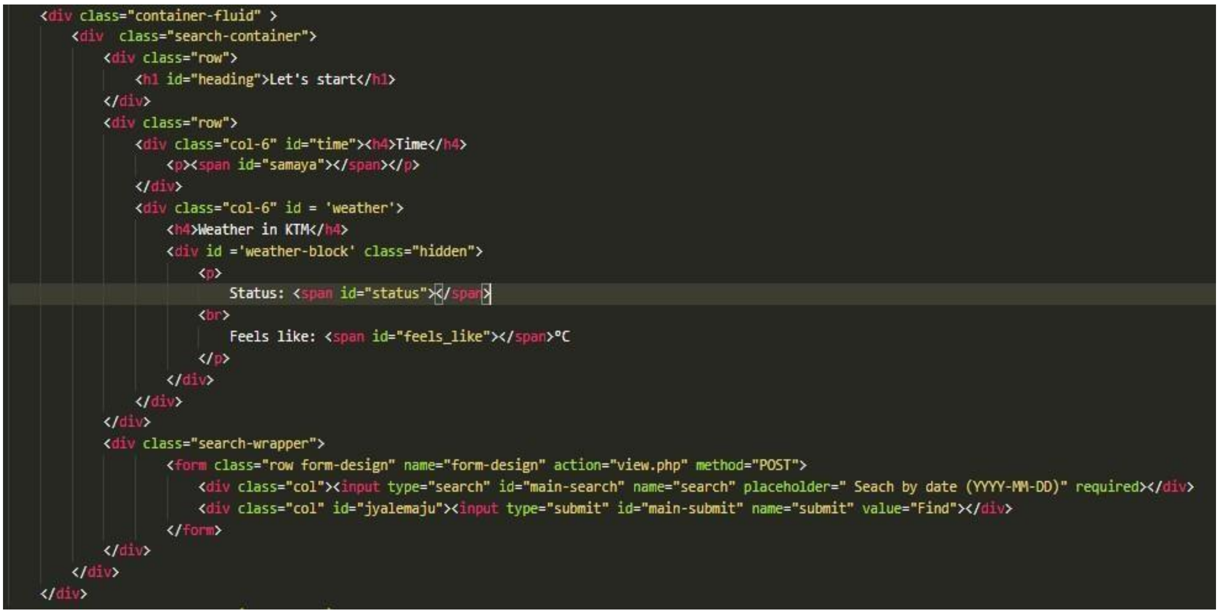
Find us at:

Address:

KhasiBazar,Kirtipur Ring Road, Kathmandu
Bhotebal,Teku, Kathmandu

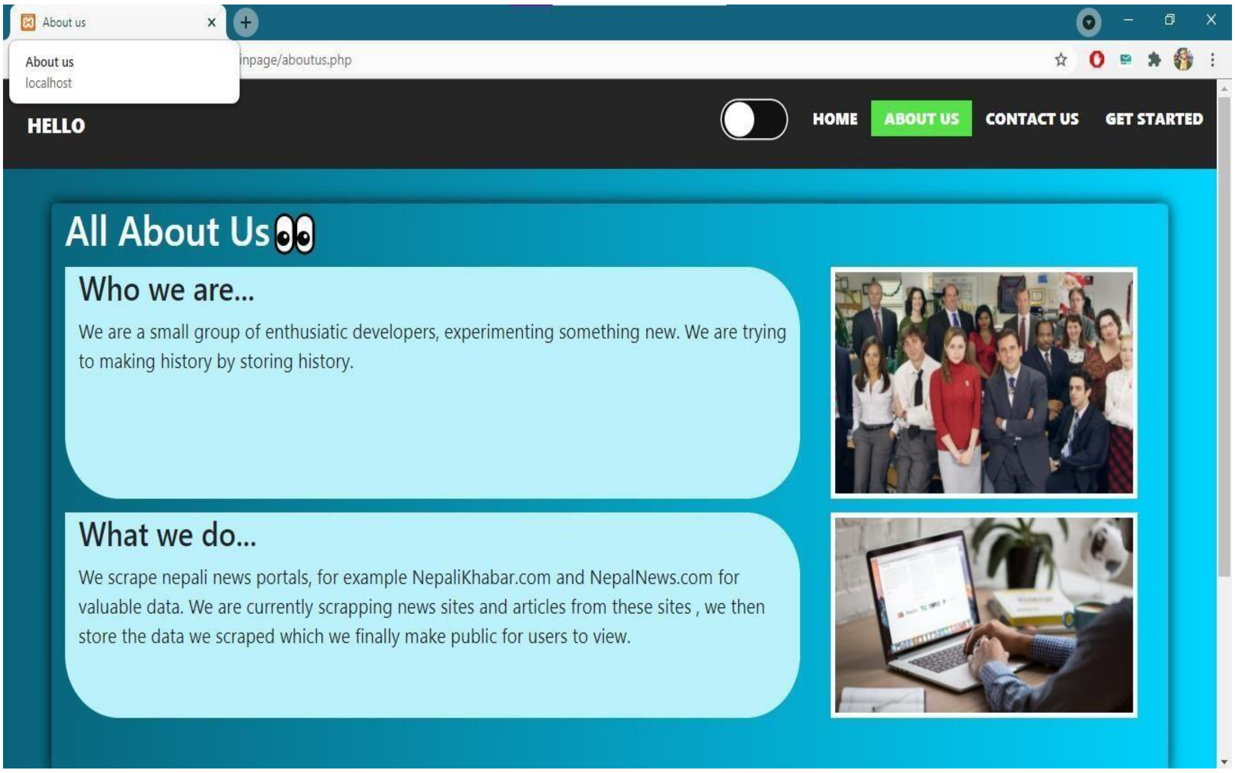
Copyright ©2021 Ourcode. Designed By ME

Figure 6.6: Home page



```
<div class="container-fluid" >
    <div  class="search-container">
        <div class="row">
            <h1 id="heading">Let's start</h1>
        </div>
        <div class="row">
            <div class="col-6" id="time"><h4>Time</h4>
                <p><span id="samaya"></span></p>
            </div>
            <div class="col-6" id = 'weather'>
                <h4>Weather in KTM</h4>
                <div id ='weather-block' class="hidden">
                    <p>
                        Status: <span id="status"></span>
                    <br>
                        Feels like: <span id="feels_like"></span>°C
                    </p>
                </div>
            </div>
        </div>
        <div class="search-wrapper">
                <form class="row form-design" name="form-design" action="view.php" method="POST">
                    <div class="col"><input type="search" id="main-search" name="search" placeholder=" Seach by date (YYYY-MM-DD)" required></div>
                    <div class="col" id="jyalemaju"><input type="submit" id="main-submit" name="submit" value="Find"></div>
                </form>
        </div>
    </div>
</div>
```

Figure 6.7: Homepage code



All About Us👀

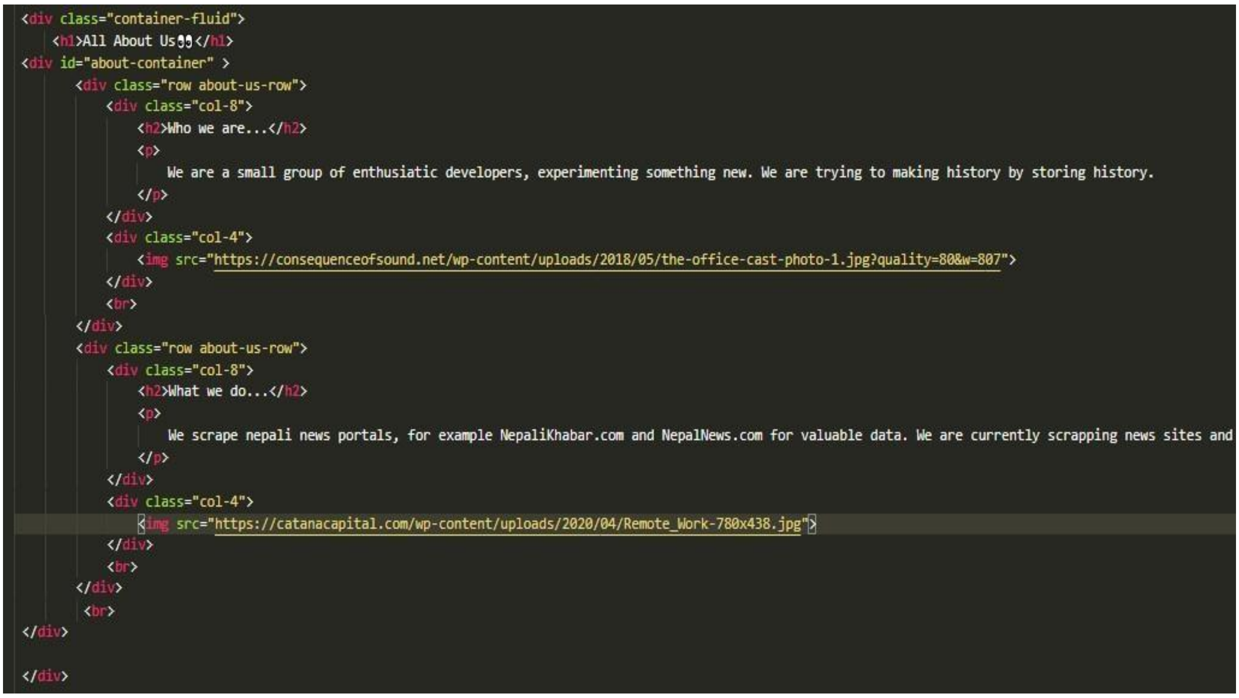
Who we are...

We are a small group of enthusiatic developers, experimenting something new. We are trying to making history by storing history.

What we do...

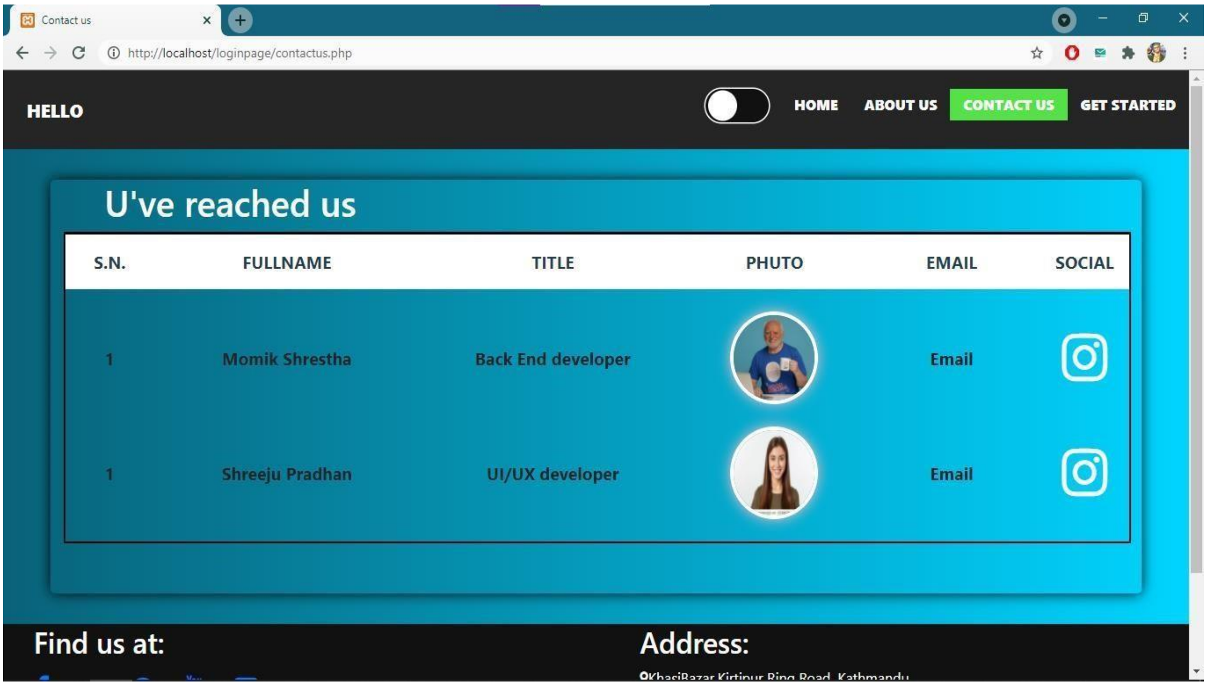
We scrape nepali news portals, for example NepaliKhabar.com and NepalNews.com for valuable data. We are currently scrapping news sites and articles from these sites , we then store the data we scraped which we finally make public for users to view.

Figure 6.8 : About Us page



```
<div class="container-fluid">
    <h1>All About Us👀</h1>
<div id="about-container" >
       <div class="row about-us-row">
           <div class="col-8">
               <h2>Who we are...</h2>
               <p>
                   We are a small group of enthusiatic developers, experimenting something new. We are trying to making history by storing history.
               </p>
           </div>
           <div class="col-4">
               <img src="https://consequenceofsound.net/wp-content/uploads/2018/05/the-office-cast-photo-1.jpg?quality=80&w=807">
           </div>
           <br>
       </div>
       <div class="row about-us-row">
           <div class="col-8">
               <h2>What we do...</h2>
               <p>
                   We scrape nepali news portals, for example NepaliKhabar.com and NepalNews.com for valuable data. We are currently scrapping news sites and
               </p>
           </div>
           <div class="col-4">
               <img src="https://catanacapital.com/wp-content/uploads/2020/04/Remote_Work-780x438.jpg">
           </div>
           <br>
       </div>
        <br>
</div>

</div>
```

Figure 6.9: About Us code



U've reached us

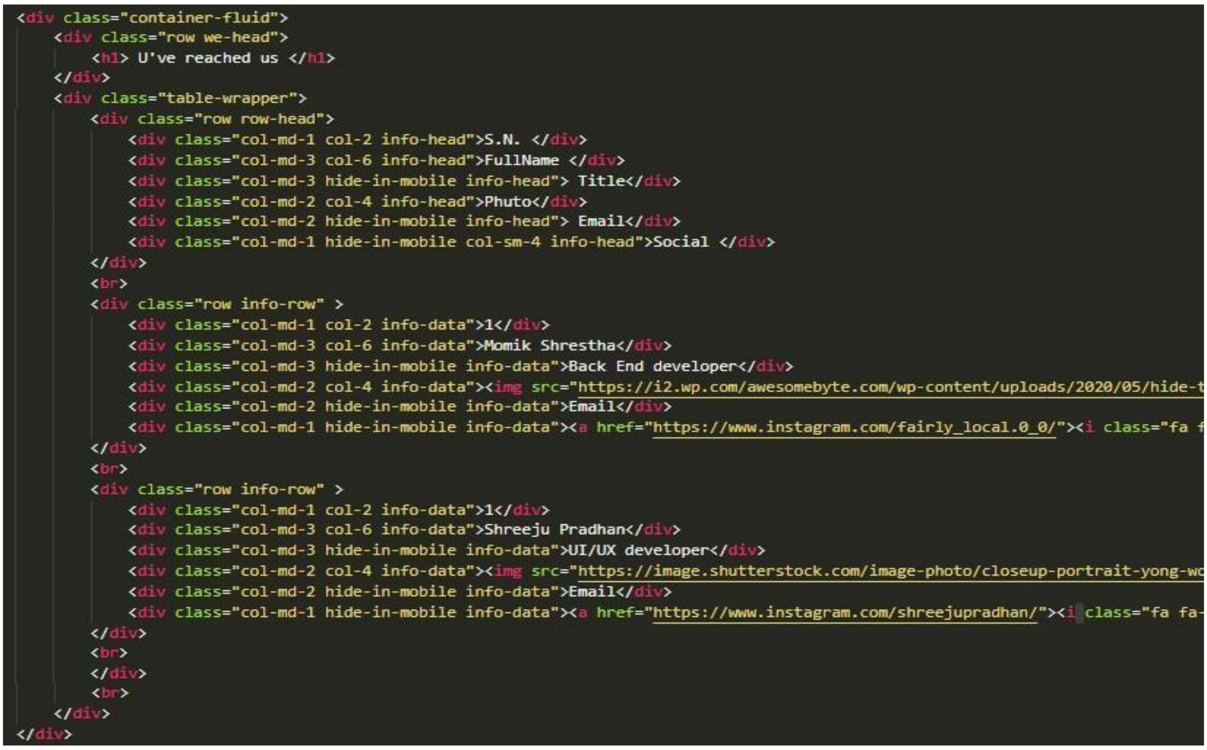
S.N.	FULLNAME	TITLE	PHUTO	EMAIL	SOCIAL
1	Momik Shrestha	Back End developer		Email	
1	Shreeju Pradhan	UI/UX developer		Email	

Find us at:

Address:

KhasiBazar Kirtipur Ring Road, Kathmandu

Figure 6.10: Contact Us Page



```
<div class="container-fluid">
    <div class="row we-head">
        <h1> U've reached us </h1>
    </div>
    <div class="table-wrapper">
        <div class="row row-head">
            <div class="col-md-1 col-2 info-head">S.N. </div>
            <div class="col-md-3 col-6 info-head">FullName </div>
            <div class="col-md-3 hide-in-mobile info-head"> Title</div>
            <div class="col-md-2 col-4 info-head">Phuto</div>
            <div class="col-md-2 hide-in-mobile info-head"> Email</div>
            <div class="col-md-1 hide-in-mobile col-sm-4 info-head">Social </div>
        </div>
        <br>
        <div class="row info-row" >
            <div class="col-md-1 col-2 info-data">1</div>
            <div class="col-md-3 col-6 info-data">Momik Shrestha</div>
            <div class="col-md-3 hide-in-mobile info-data">Back End developer</div>
            <div class="col-md-2 col-4 info-data"><img src="https://i2.wp.com/awesomebyte.com/wp-content/uploads/2020/05/hide-t
            <div class="col-md-2 hide-in-mobile info-data">Email</div>
            <div class="col-md-1 hide-in-mobile info-data"><a href="https://www.instagram.com/fairly_local.0_0/"><i class="fa f
        </div>
        <br>
        <div class="row info-row" >
            <div class="col-md-1 col-2 info-data">1</div>
            <div class="col-md-3 col-6 info-data">Shreeju Pradhan</div>
            <div class="col-md-3 hide-in-mobile info-data">UI/UX developer</div>
            <div class="col-md-2 col-4 info-data"><img src="https://image.shutterstock.com/image-photo/closeup-portrait-yong-wo
            <div class="col-md-2 hide-in-mobile info-data">Email</div>
            <div class="col-md-1 hide-in-mobile info-data"><a href="https://www.instagram.com/shreejupradhan/"><i class="fa fa-
        </div>
        <br>
        </div>
        <br>
    </div>
</div>
```

Figure 6.11: Contact Us code

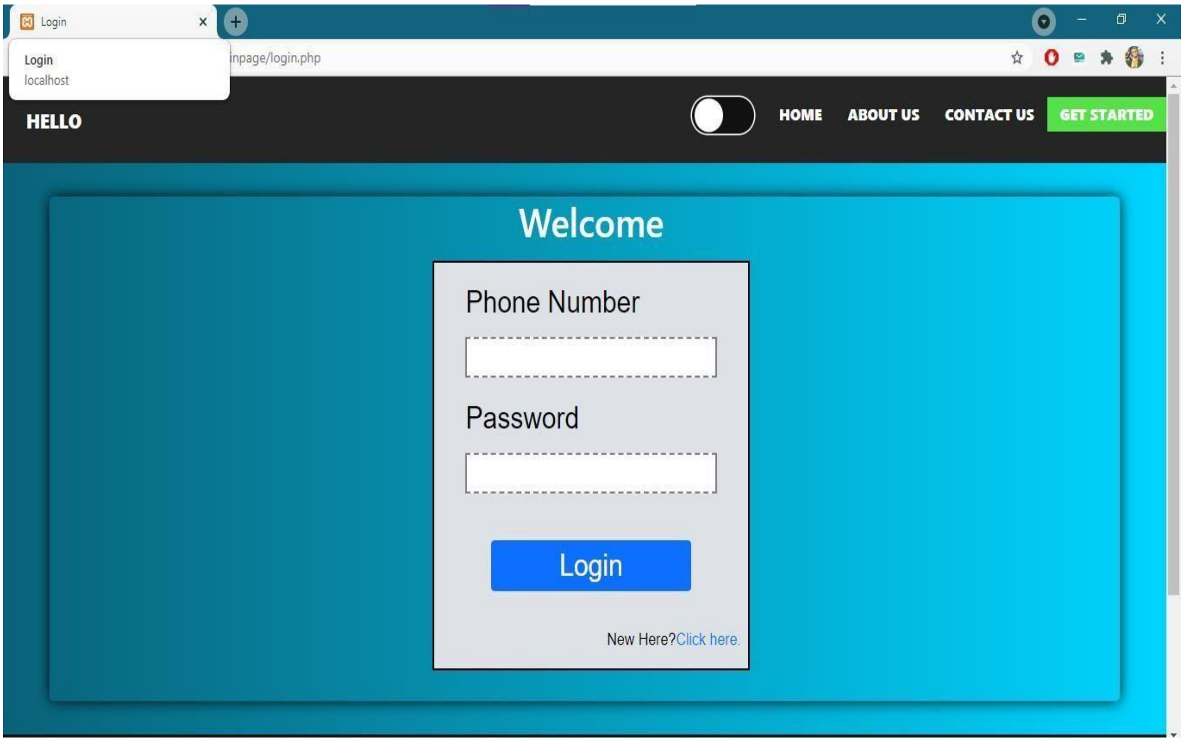
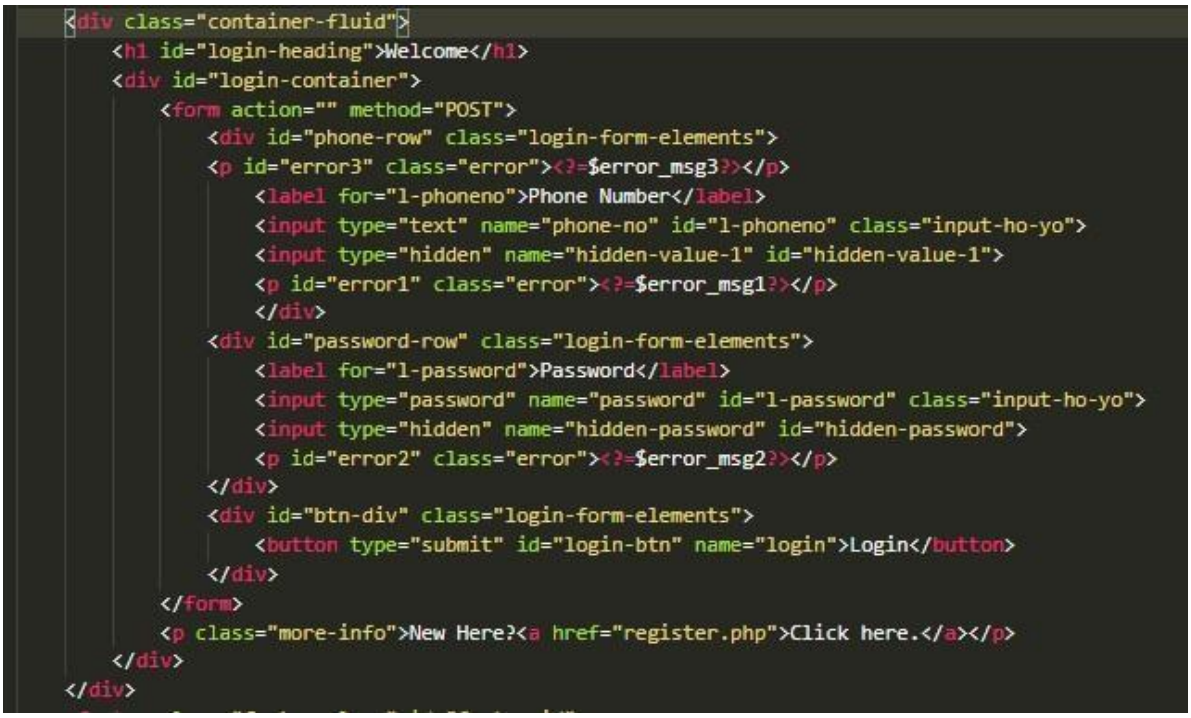
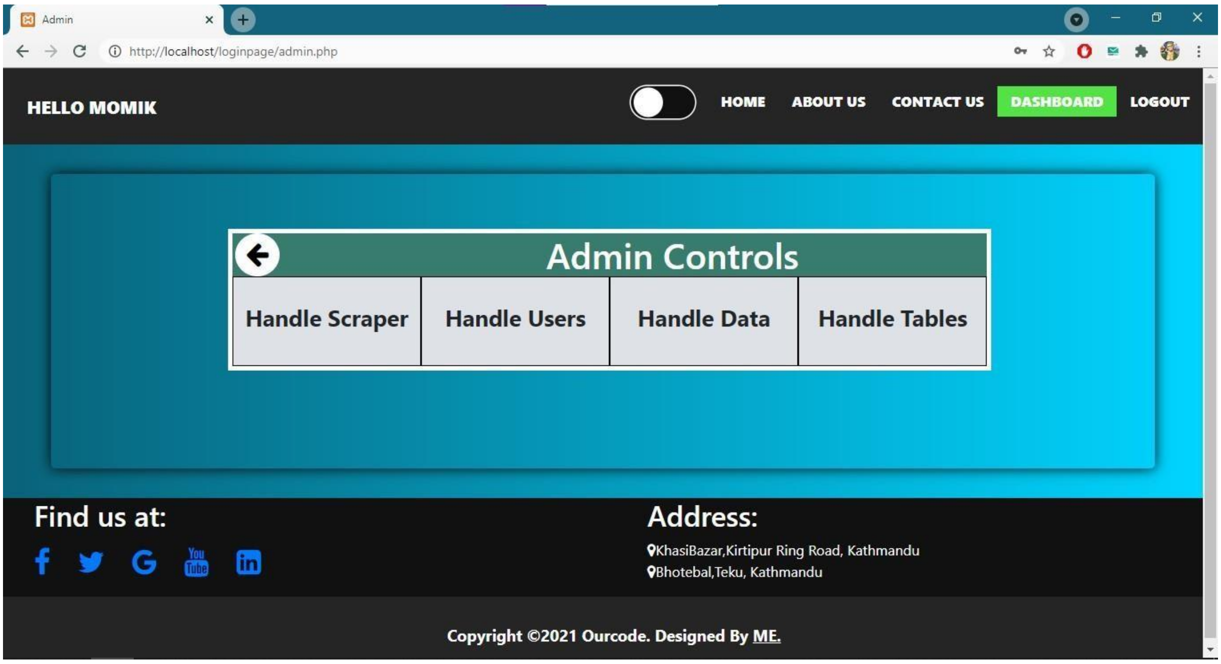


Figure 6.12: Login Page



```
<div class="container-fluid">
    <h1 id="login-heading">Welcome</h1>
    <div id="login-container">
        <form action="" method="POST">
            <div id="phone-row" class="login-form-elements">
            <p id="error3" class="error"><?=$error_msg3?></p>
                <label for="l-phoneno">Phone Number</label>
                <input type="text" name="phone-no" id="l-phoneno" class="input-ho-yo">
                <input type="hidden" name="hidden-value-1" id="hidden-value-1">
                <p id="error1" class="error"><?=$error_msg1?></p>
                </div>
            <div id="password-row" class="login-form-elements">
                <label for="l-password">Password</label>
                <input type="password" name="password" id="l-password" class="input-ho-yo">
                <input type="hidden" name="hidden-password" id="hidden-password">
                <p id="error2" class="error"><?=$error_msg2?></p>
            </div>
            <div id="btn-div" class="login-form-elements">
                <button type="submit" id="login-btn" name="login">Login</button>
            </div>
        </form>
        <p class="more-info">New Here?<a href="register.php">Click here.</a></p>
    </div>
</div>
```

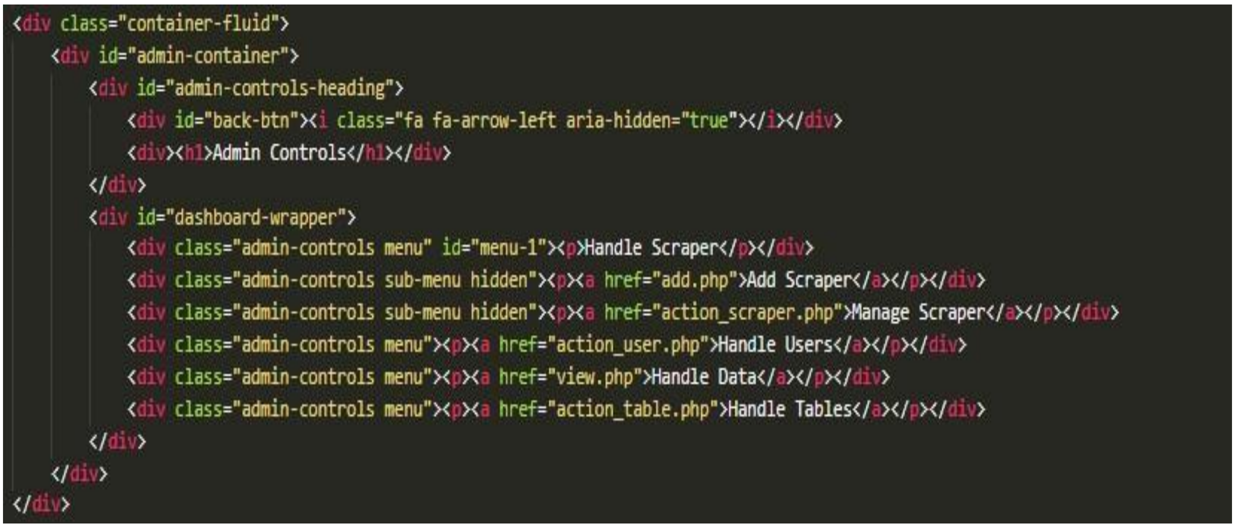
Figure 6.13: Login page code



Address:
KhasiBazar,Kirtipur Ring Road, Kathmandu
Bhotebal,Teku, Kathmandu

Copyright ©2021 Ourcode. Designed By ME.

Figure 6.14: Admin Dashboard



```
<div class="container-fluid">
    <div id="admin-container">
        <div id="admin-controls-heading">
            <div id="back-btn"><i class="fa fa-arrow-left aria-hidden="true"></i></div>
            <div><h1>Admin Controls</h1></div>
        </div>
        <div id="dashboard-wrapper">
            <div class="admin-controls menu" id="menu-1"><p>Handle Scraper</p></div>
            <div class="admin-controls sub-menu hidden"><p><a href="add.php">Add Scraper</a></p></div>
            <div class="admin-controls sub-menu hidden"><p><a href="action_scraper.php">Manage Scraper</a></p></div>
            <div class="admin-controls menu"><p><a href="action_user.php">Handle Users</a></p></div>
            <div class="admin-controls menu"><p><a href="view.php">Handle Data</a></p></div>
            <div class="admin-controls menu"><p><a href="action_table.php">Handle Tables</a></p></div>
        </div>
    </div>
</div>
```

Figure 6.15: Admin Dashboard code

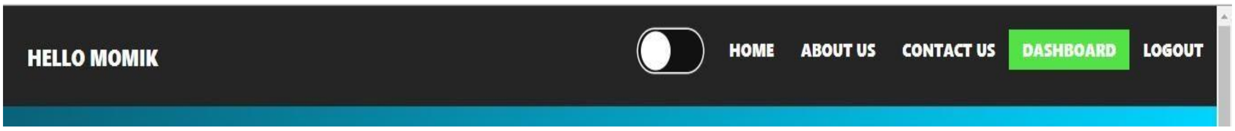
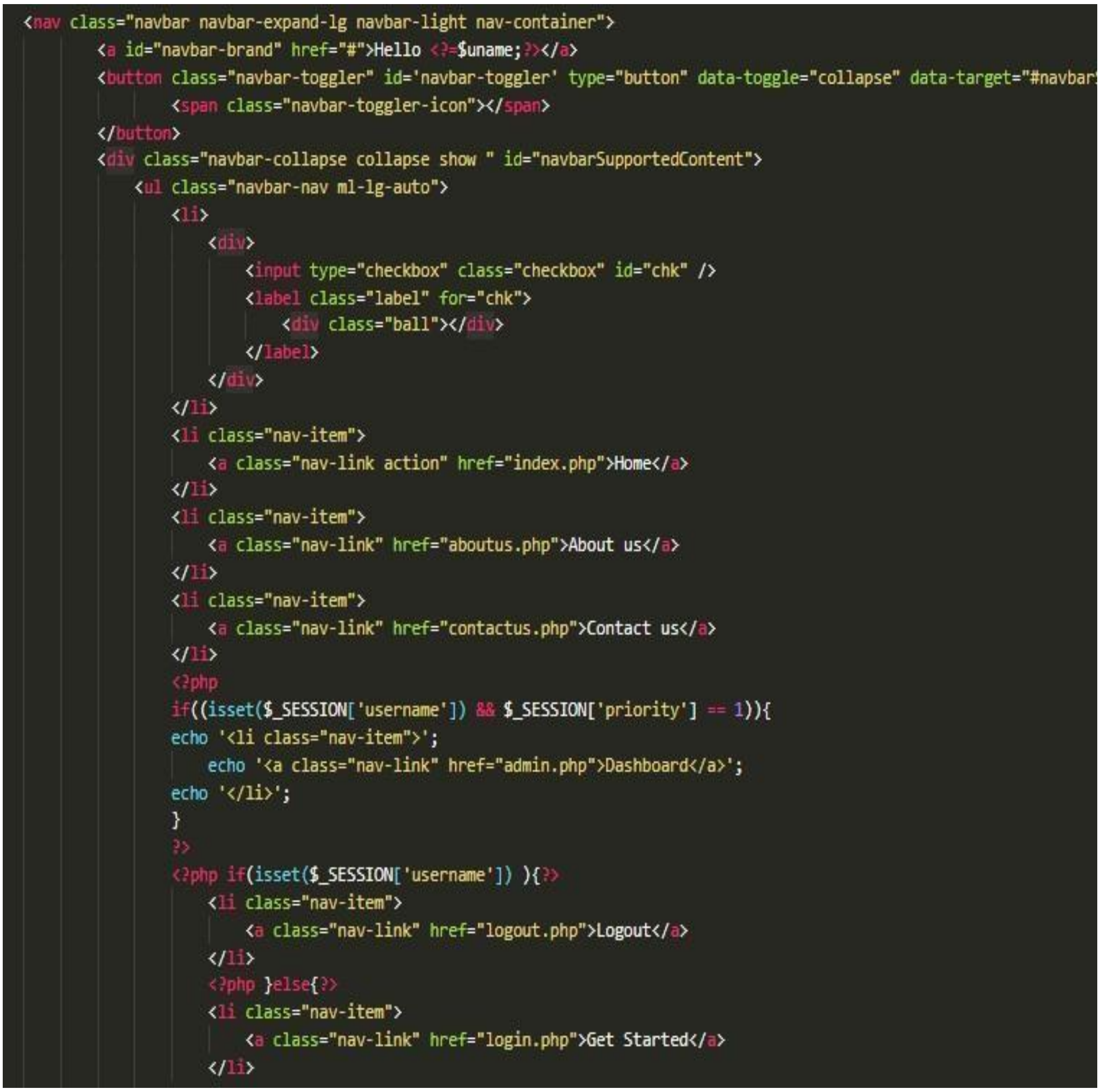
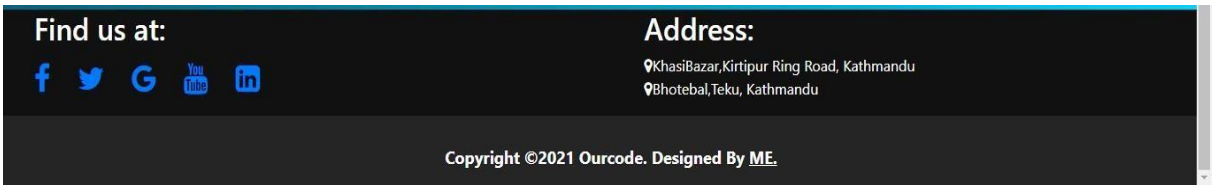


Figure 6.16: Header



```
<nav class="navbar navbar-expand-lg navbar-light nav-container">
        <a id="navbar-brand" href="#">Hello <?=$uname;?></a>
        <button class="navbar-toggler" id='navbar-toggler' type="button" data-toggle="collapse" data-target="#navbar
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="navbar-collapse collapse show " id="navbarSupportedContent">
          <ul class="navbar-nav ml-lg-auto">
            <li>
              <div>
                <input type="checkbox" class="checkbox" id="chk" />
                <label class="label" for="chk">
                  <div class="ball"></div>
                </label>
              </div>
            </li>
            <li class="nav-item">
              <a class="nav-link action" href="index.php">Home</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="aboutus.php">About us</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="contactus.php">Contact us</a>
            </li>
            <?php
            if((isset($_SESSION['username']) && $_SESSION['priority'] == 1)){
            echo '<li class="nav-item">';
              echo '<a class="nav-link" href="admin.php">Dashboard</a>';
            echo '</li>';
            }
            ?>
            <?php if(isset($_SESSION['username']) ){?>
              <li class="nav-item">
                <a class="nav-link" href="logout.php">Logout</a>
              </li>
              <?php }else{?>
              <li class="nav-item">
                <a class="nav-link" href="login.php">Get Started</a>
              </li>
```

Figure 6.17: Header partials



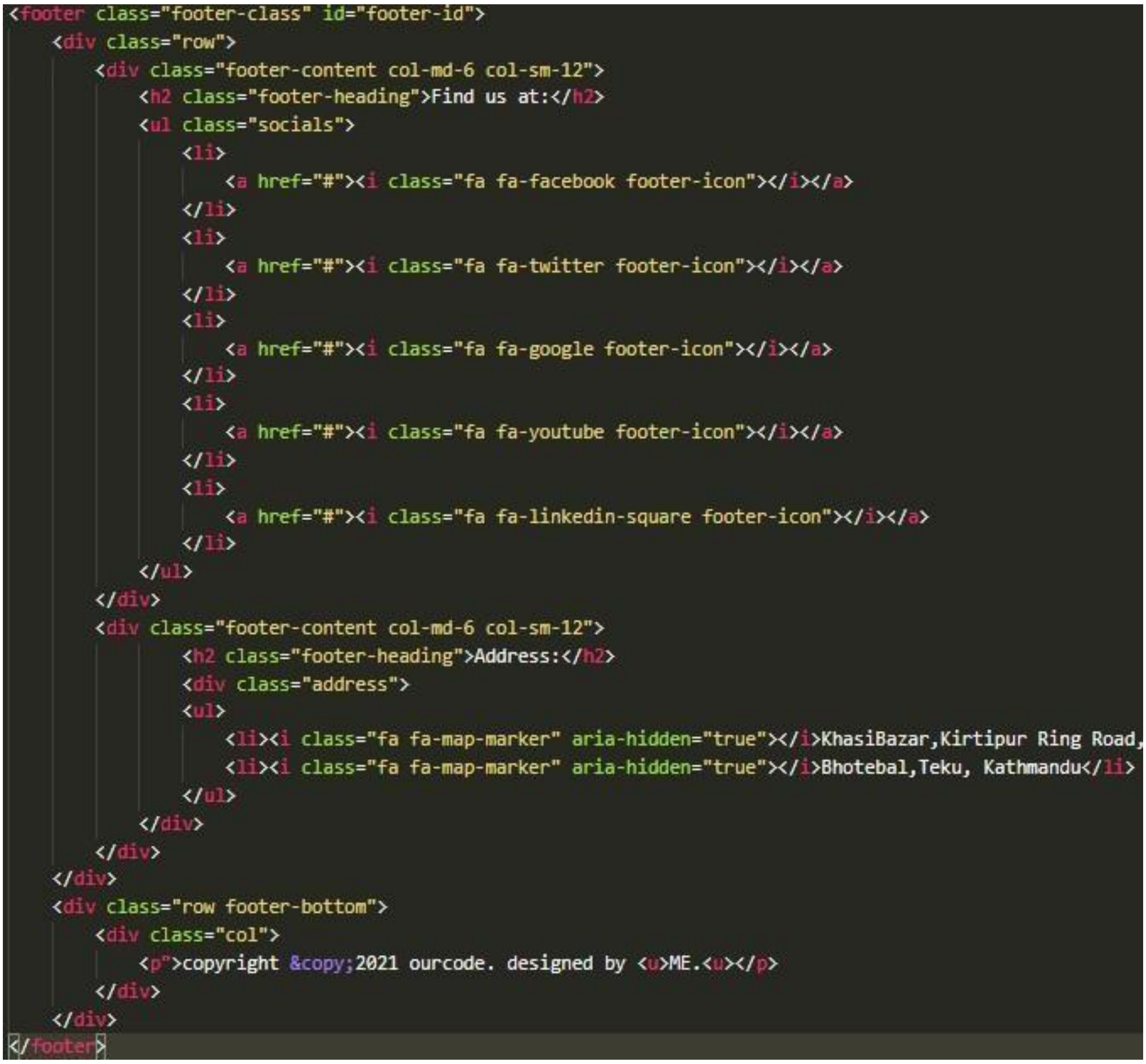
Find us at:

Address:

KhasiBazar,Kirtipur Ring Road, Kathmandu
Bhotebal,Teku, Kathmandu

Copyright ©2021 Ourcode. Designed By ME.

Figure 6.18: Footer



```
<footer class="footer-class" id="footer-id">
    <div class="row">
        <div class="footer-content col-md-6 col-sm-12">
            <h2 class="footer-heading">Find us at:</h2>
            <ul class="socials">
                <li>
                    <a href="#"><i class="fa fa-facebook footer-icon"></i></a>
                </li>
                <li>
                    <a href="#"><i class="fa fa-twitter footer-icon"></i></a>
                </li>
                <li>
                    <a href="#"><i class="fa fa-google footer-icon"></i></a>
                </li>
                <li>
                    <a href="#"><i class="fa fa-youtube footer-icon"></i></a>
                </li>
                <li>
                    <a href="#"><i class="fa fa-linkedin-square footer-icon"></i></a>
                </li>
            </ul>
        </div>
        <div class="footer-content col-md-6 col-sm-12">
                <h2 class="footer-heading">Address:</h2>
                <div class="address">
                <ul>
                    <li><i class="fa fa-map-marker" aria-hidden="true"></i>KhasiBazar,Kirtipur Ring Road,
                    <li><i class="fa fa-map-marker" aria-hidden="true"></i>Bhotebal,Teku, Kathmandu</li>
                </ul>
            </div>
        </div>
    </div>
    <div class="row footer-bottom">
        <div class="col">
            <p">copyright &copy;2021 ourcode. designed by <u>ME.<u></p>
        </div>
    </div>
</footer>
```

Figure 6.19: Footer partials

REFERENCES

[1] "The Wayback machine," Internet Archive, 1996. [Online]. Available: https://archive.org/about/. [Accessed 20 02 2021].

[2] E. Gallagher, "Scraping Websites for Law Enforcement," School of Computing, Engineering & Intelligent System, 2018.

[3] "PaperLit," Datrix S.A, 2020. [Online]. Available: https://www.paperlit.com/blog/where-dopeople-get-their-news-nowadays/. [Accessed 01 03 2020].

[4] PromptCloud, "Automated-Web-scraping," [Online]. Available: https://www.promptcloud.com/automated-web-scraping/.

[5] Shenesh Perera, "web-scraping-javascript," 02 03 2021. [Online]. Available: https://www.scrapingbee.com/blog/web-scraping-javascript/.

[6] S. K. Malik, "Research Gate," 10 2011. [Online]. Available: https://www.researchgate.net/publication/241626213_Information_Extraction_Usinh _Web_ Usage_Mining_Web_Scrapping_and_Semantic_Annotation. [Accessed 01 03 2020].

[7] I. D. Kshetri, "AN INTERDISCIPLINARY JOURNAL," *BODHI:,* p. 9, 2008.