Jurnal Informatika Universitas Pamulang **ISSN: 2541-1004**
Penerbit: Program Studi Teknik Informatika Universitas Pamulang **e-ISSN: 2622-4615**
Vol. 6, No. 1, Maret 2021 (202-209) **10.32493/informatika.v6i1.10077**

Web Harvesting for Data Retrieval on Scientific Journal Sites

I Gede Surya Rahayuda[1], Ni Putu Linda Santiari[2]

[12]Faculty of Informatics and Computers, Institute of Technology and Business STIKOM Bali, Jalan Raya Puputan Renon No. 86, Denpasar, Bali, Indonesia, 80234
e-mail: [1]surya_rahayuda@stikom-bali.ac.id, [2]linda_santiari@stikom-bali.ac.id

Submitted Date: March 27th, 2021 Reviewed Date: June 02nd, 2021
Revised Date: June 05th, 2021 Accepted Date: June 15th, 2021

Abstract

Publishing scientific articles online in journals is a must for researchers or academics. In choosing the journal of purpose, the researcher must look at important information on the journal's web, such as indexing, scope, fee, quarter and other information. This information is generally not collected in one page, but spread over several pages in a web journal. This will be complicated when researchers have to look at information in several journals, moreover, the information in these journals may change at any time. In this research, web harvesting design is conducted to retrieve information on web journals. With web harvesting, information that is spread across several pages can be collected into one, and researchers do not need to worry if the information has changed, because the information collected is the last or updated information. Harvesting technique is done by taking the page URL of the page, starting the source code from where the information is retrieved and end source code until the information stops being retrieved. Harvesting technique was successfully developed based on the web bootstrap framework. The test data is taken from several scientific journal webs. The information collected includes name, description, accreditation, indexing, scope, publication rate, publication charge, template and quarter. Based on tests carried out using black box testing, it is known that all the features made are as expected.

Keywords: web harvesting; web mining; parsing; bootstrap; journal

1. Introduction

Information technology advancements, especially the internet, makes people choose to store data and everything that is digital on the internet. Online storage media is considered safe and has a large capacity which generally cannot be done when storing data offline on personal devices. Starting from unofficial publications such as file sharing, videos, social media to official publications such as open datasets, government data, conferences, journals and others. Currently, almost all researchers, lecturers, students and other academics choose to publish their scientific papers online in journals. On the internet all information can change quickly. In choosing the journal of purpose, the researcher must look at important information on the journal's web, such as indexing, scope, fee, quarter and other information. This information is generally not collected in one page, but spread over several pages in a web journal. This will be complicated when researchers have to look at information in several journals, moreover, the information in these journals may change at any time. In this study, the author will discuss the use of web harvesting methods for data collection in several scientific journals (Josi et al., 2014). By using this method, researchers can collect important information desired from several journals or journal pages. Researchers also don't need to worry if the information on the web journal changes. Web harvesting is a technique for extracting data and information from a website and then storing it in a specific format (Sahria, 2020)(Chifu & Leţia, 2015). Web harvesting is done by taking the page URL of the page, starting the source code from where the information is retrieved and end source code until the information stops being retrieved. Web harvesting will be made web-based with the addition of a bootstrap front-end framework. The test data used are several scientific journal webs. The information collected is, Name, Description, Accreditation, indexing, scope, publication rate, Publication Charge, Template and Quarter (Johnson & Sieber, 2012)(Josi et al., 2014). The website will be tested using black box method. The author hopes that this research will make it easier for researchers or academics to choose destination journals.

2. Literature Review

Literature study is done by searching for information in the form of theory and tutorials via the internet. Some of the literature used in this study are summarized in the next subchapter.

2.1 Web Harvesting

Web harvesting would be a method of extracting information on websites without having to physically copy it (Haralson, 2016)(Saurkar & Pathare, 2018). The purpose of a web scraper is to find certain information and then collect it in the new web. Web harvesting focuses on obtaining data by means of retrieval and extraction (Zhao, 2017). The benefit of web harvesting is that the extracted information is more focused, making it easier to search for something (Rabby, 2017)s. Harvesting web applications only focus on how to obtain data through data collection and extraction with various data sizes. Web harvesting has a number of steps including:

- Create Template: The programmer obtains the HTML document from the website, from which the content for the HTML element that encloses the data to be obtained will be extracted.
- Investigate Web Navigation: The programmer studies website navigation strategies, which will be used to model the harvesting webpage that would be made.
- Automate Navigation and Extraction: On the basis of the data gathered in stages before, the web harvesting is designed in order to optimize the gathering of data from the webpage that has been identified.
- Optimize Routing and Filtration: The web harvesting application is constructed information collected in wave 1 and 2 here to organize data processing from the selected website.

2.2 Harvesting Technique

Web harvesting is a technique for mechanically gathering or obtaining data from the internet (Indra et al., 2019)(T, 2019). It's a burgeoning area with a shared objective with the logical search perception, an target undertaking that nevertheless necessitates advancements in text mining, analogize, intelligent systems, and human-computer interactions.

- Copying and pasting by humans
  The most basic type of web harvesting is manually scraping and duplicating information to the website into a text document or worksheet. Even the greatest online harvesting technology can't always substitute a person's human investigation and edit and grab, and in certain cases, this is the only viable option when harvesting websites put up hurdles to prevent computer automation.
- Pairing textual patterns
  Regular adjective computer language capabilities or the UNIX grep function (e.g., Perl or Python) can be used as an easy but efficient method to extract knowledge from web pages.
- HTTP scripting
  Socket programming allows you to get dynamic and static websites by making Html code toward a remote server.
- HTML parsing
  Many websites contain enormous collections of content built continuously with an underpinning structured source, similar to a database. Data from a certain class is frequently encoded into comparable pages using a common code or concept. A wrapper is a software that recognizes particular themes in a given source of data, extracts its information, and turns it into a basic unit in data mining. The creation of wrappers techniques suppose that the wrapper's input pages expansion device follow a common template and are easily identifiable in accordance with the chosen Web address system. In addition, HTML pages can be parsed and page content extracted and converted using certain moderately query languages, like XQuery and HTQL.
- Document object model parsing
  Embedding a complete browser like chrome or brave browser management, programs may obtain dynamically content produced by scripts on the client side. These browser features may even parse pages together into document object model tree, depending on what are the programs may obtain sections of the sites. In order to parse the resulting document object model tree, languages such as Xpath may be used.
- Vertical aggregation

There are many businesses which have built unique vertical harvesting platforms. For verticals where there is no "man in the loop" (no human intervention engagement) and when there is no work tied toward a identified target area, these networks build and track a multitude of "bots". The bots are developed automatically by the platform when the conceptual framework for the whole vertical has been created. The accuracy of the work it gathers (typically the amount of fields) and the platform's scalability are used to determine its resilience (how rapidly it could expand up to hundreds or thousands of places). This ability to scale is frequently used to find sites in the furthest reaches of the internet are difficult or time-consuming to acquire material from using traditional aggregators.

- Semantic annotation recognizing
  Annotations and metadata, often known as semantic markups and metadata, could be used included in the pages being scraped, it might be utilized find unique data fragments. If indeed the captions are incorporated in the phrase, like small format does, this method might be considered as a specific case of document object model parsing. In another scenario, the observations are saved and kept independently from the web pages, arranged into a semantic layer, so that scrapers may extract the before this layer's structure and directives harvesting the pages.
- Analysis of web using image recognition
  Image recognition and artificial intelligence are being used to detect and abridged knowledge graphically extracting information from online pages scanning them like a human would.

2.3 Framework Bootstrap

Bootstrap is a CSS framework toolkit designed primarily for building a website's front end. Bootstrap is a CSS, HTML, and JavaScript framework that is extremely popular among web designers and developers. Bootstrap is a framework for creating responsive webpages (Krause, 2020). Of course, the website page may change to the size of the display device with this bootstrap. If you want to use your phone, tablet, or computer, this is the place to go. The bootstrap was originally known as Twitter Blueprint. Created and developed by Jacob Thornton and Mark Otto, who were looking for a workable tool to promote consistency in their internal tools when they saw it on Twitter (Gaikwad & Adkar, 2019). Of course, by utilizing bootstrap, a developer may make creating a front end for a website easier and faster. Users need just call each class they use, such as navigation, tables, grinds, buttons, and so on. A website can make use of a variety of bootstrap functions. The following functionalities are available:

- Can speed up the processing time for the front end of a website
- Displays a more current aspect of the website that is likewise representative of today's children
- The bootstrap's look is incredibly responsive, so it works well with many sorts of resolutions, including tablets, smartphones, and PCs and laptops
- Bootstrapped websites are often lighter since they are more structured

3 Research Method

The research was conducted using the SDLC Waterfall method (Eason, 2016). The Waterfall method would be a technology process that runs in a consecutive order. the steps of planning, modeling, implementation, and testing flow down (like a waterfall) in a continuous stream (Rani, 2017).

- Collecting and analysing requirements
  After collecting all of the requirements, the team assessed and established the requirements that the software must meet. This step must be completed in its entirety in order to generate a finished design.
- Conceptualization
  The developer will create an overall system and decide the software flow to a detailed algorithm during this stage.
- Development
  It is at this point that the complete concept is transformed into program code. The program code that results is still in the form of modules that will be combined into a full system.
- Evaluating and integrating
  At this point, the modules that have been created are integrated, and a test is run to see if the program that has been created is in line with the design and function of the program, or if there are any flaws.
- Authentication

At this point, the user verifies that the system is operating in line with the authorized specifications.

4 Result and Discussion

The research was conducted in accordance with the research methods described. Starting from collecting all the requirements needed in the research, then continuing with designing the system to be built, after the design is complete, it is continued with the implementation of web harvesting based on the bootstrap web framework. Then the results of the implementation of the program were tested using black box testing and finally verified.

4.1 Requirement

Some of the components needed in this study are data in the form of the desired information from web journals and several destination journals (Josi et al., 2014). Several destination journals sites that will be used as test data on web harvesting are:

- Journal of ICT Research and Applications
- International Journal on Advanced Science, Engineering and Information Technology
- Bulletin of Electrical Engineering and Informatics

Some of the data or information that you want to obtain from the journal are:

- Name
- Description
- Accreditation
- Indexing
- Focus and scope
- Publication Rate
- Publication Charge
- Template
- Quarter

4.2 Data Flow Diagram

Before building a web harvesting system, a system design is made, the system design is made using a data flow diagram (Chong & Diamantopoulos, 2020). Data flow diagrams are made in three levels. The first level is the context diagram, the context diagram consists of one executor, namely the user and one process, namely the system (Irhamn & Siahaan, 2019). The user enters the required data parsing into the system then the system will store and process the input and display the results to the user (Aleryani, 2016). As shown in Figure 1.

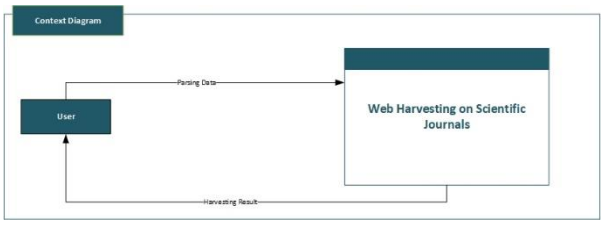

Figure 1. Context Diagram

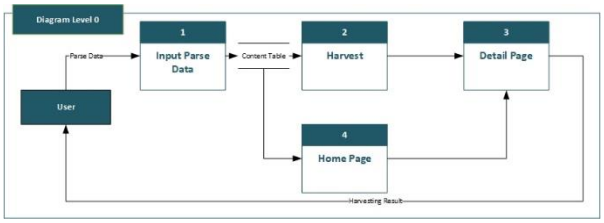

Figure 2. Level 1 Diagram

The first level diagram shown in Figure 2 is the development of the web harvesting process in the context diagram. The system is translated into 4 processes, namely:

- Input parse data: In this process, the user enters the data obtained from the html source code of the journal web page. These data are name, description, accreditation, indexing, scope, rate, charge, template and quarter.
- Harvest: In this process the input data that has been stored is then retrieved and then harvested which will produce an output in the form of some important information in the destination journal.
- Detail page: This process functions to receive data harvest results and display it on the detail page.
- Home page: This process serves to display a list of data that has been stored in the database in the form of names and descriptions.

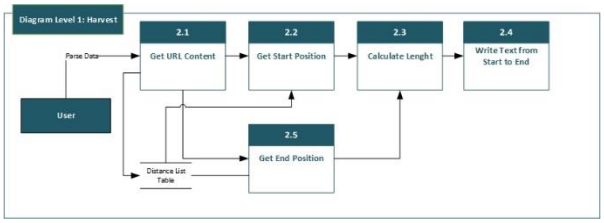

Figure 3. Diagram of Level 2 Harvesting Process

The level 1 diagram shown in Figure 3 is the development of the harvest process in the level 0

diagram (Haralson, 2016). In the harvest process there are 5 processes, namely:

- Get URL content: This process serves to receive the URL code data in the destination journal.
- Get start position: This process functions to receive data start position code or start parse from the desired information text in the URL content.
- Get start position: This process functions to receive end position code data or end parse from the information text in the URL content, which is the closing parse.
- Calculate length: This process functions to calculate the character length of the harvest text, which is the subtraction from the start parse and end parse.
- Write result: This process functions to print the results of harvest text obtained from the parsing results.

4.3 Implementation

The implementation of web harvesting is based on the web bootstrap framework. Web harvesting is created on several pages:

1. Parsing Page

This is the very first page of the system. Towards this section, the user will be asked to input the required data for web harvesting. The data is in the form of URL page from journal page, start source code and end source code. The page URL, start point and end point will be filled in for any important information you want to find. The data entered is part of the destination journal source code. Users can view the data through the menu inspect element which is generally available on any web browser. Some of the parsing code entered is shown in figure 4.

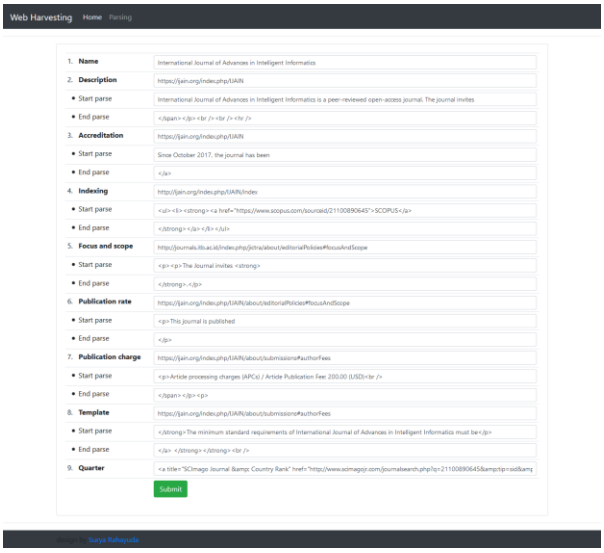

Figure 4. Parsing Page

2. Home Page

This page is the second page, after the user enters all the completeness of the data on the data input page, then the journal data will be stored in the database and the journal list page will be displayed. All journals that have been inputted will be displayed in the form of a list in the form of a number, name and action. Users can see name and description and can click details button for more detail. The list of names and descriptions is displayed on a stack of cards with a maximum of three cards per page. As shown in figure 5.

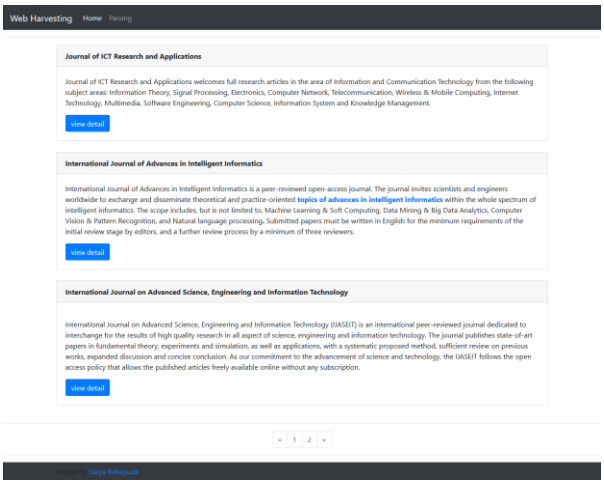

Figure 5. Home Page

3. Detail Page

The detail page is the main page of the system, on this page the user can see all the important information they want from the destination journal. On this page all important information will be displayed. The information is in the form of:

- Name: is the name of the journal
- Description: contains information in the form of an explanation of the journal description
- Accreditation: contains information in the form of sinta accreditation of journal
- Indexing: contains information in the form of an index list from the journal
- Focus and Scope: contains information in the form of focus and scope of journals, such as fields of science or research fields
- Pub. Rate: contains information in the form of journal publication scale in a year or in what month the journal was published
- Pub. Charge: contains information in the form of article publication fees and also other information related to payment.
- Template: contains information in the form of a link to download template file from the journal
- Quarter: contains information in the form of quarter levels from the journal

The detail page is shown in figure 6.

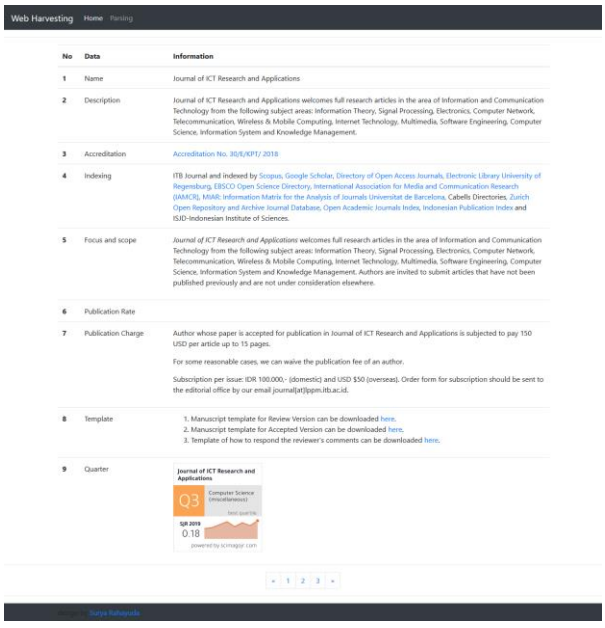

Figure 6. Detail Page

On this page, all parse data contained in the database is retrieved and then processed using the php script (Haralson, 2016). This process is called web harvesting, here is a snippet view of the harvesting process program:

```
$start_parse = $data['index_start'];
$end_parse = $data['index_end'];
$content = file_get_contents($data['index_content']);
$start = stripos($content, $start_parse);
$end = stripos($content, $end_parse, $offset = $start_parse);
$length = $end - start
$result = substr($content, $start, $length
```

Figure 7. Harvesting Code

4.4 Black Box Testing

In this study, the black box testing method was used (Roman, 2018). Testing is done by testing the interface part of the information system, each part of the interface is tested in order to determine whether the system is running according to the expected function (Xu & Chen, 2016). The purpose of this test is to find out errors in the system being made (Henard & Papadakis, 2016). The following are the test results using black box technique of evaluation which are displayed in tabular form:

Table 1. Black box testing results

No	Test Class	Testing Scenarios	Test result
1	Home Page	Users can access the homepage, the card list is successfully displayed with a maximum of 3 stacks in 1 page, the more detail button functions properly	compatible
2	Parsing Page	Users can input all data and when the submit button is pressed all data is stored properly	compatible
3	Harvest	The process of retrieving parsing data from the database went well. The calculation of data length from the start and end points went well.	compatible
4	Detail Page	The text data generated from the harvesting process is successfully displayed on the detail page.	compatible

4.5 Verification

At this stage, verification is carried out, by running or operating the finished system, whether the system is in accordance with the test results in the previous stage. This is done to ensure whether the system is operating properly, finding errors or bugs, improving system performance, ensuring that the application can run in a new scope. Besides that, maintenance is also carried out, such as repairing errors, improving the installation of the system unit and the enhancement of system services in response to new requirements. Verification also was carried out by accessing the system on several different devices such as cell phones, tablets, laptops and also testing by moving program code and databases on web hosting. Based on the trials conducted, it is known that the system can run well on all devices, by using the bootstrap framework it can keep the system display stable on different devices. When the system is moved to web hosting, the system can run properly and can be accessed online.

5 Conclusion

According to the research conducted, It is possible to deduce that web harvesting has been successfully implemented for data retrieval in bootstrap framework-based scientific journals. The system design is successfully illustrated using data flow diagrams up to level 1 diagrams. Web harvesting can help academicians when determining their choice of journal destinations. With web harvesting, important information in several journals can be viewed easily. Based on testing and verification, it can be concluded that the system that has been made is running well, can be accessed stably on all devices and can be accessed online.

References

Aleryani, A. Y. (2016). Comparative Study between Data Flow Diagram and Use Case Diagram. *International Journal of Scientific and Research Publications*, *6*(3).

Chifu, E. Ş., & Leţia, T. Ş. (2015). Web Harvesting and Sentiment Analysis of Consumer Feedback. *Acta Technica Napocensis Electronics and Telecommunications*, *56*(3).

Chong, H.-Y., & Diamantopoulos, A. (2020). Integrating Advanced Technologies to Uphold Security of Payment: Data Flow Diagram. *Automation in Construction*, *114*.

Eason, O. K. (2016). *Information Systems Development Methodologies Transitions: An Analysis of Waterfall to Agile Methodology*. University of New Hampshire.

Gaikwad, S. S., & Adkar, P. (2019). A Review Paper on Bootstrap Framework. *IRE Journals*, *2*(10).

Haralson, D. (2016). *Automating Website Crawling using Web Scraping Techniques Provided by PHP*. Helsinki Metropolia University of Applied Sciences.

Henard, C., & Papadakis, M. (2016). Comparing White-Box and Black-Box Test Prioritization. *International Conference on Software Engineering (ICSE)*.

Indra, E., Steffanily, & Dinesh, T. (2019). Designing Android Gaming News & Information Application Using Java-Based Web Scraping Technique. *Journal of Physics: Conference Series*.

Irhamn, F., & Siahaan, D. (2019). Object-Oriented Data Flow Diagram Similarity Measurement Using Greedy Algorithm. *International Conference on Cybernetics and Intelligent System (ICORIS)*.

Johnson, P. A., & Sieber, R. E. (2012). Automated Web Harvesting to Collect and Analyse User Generated Content for Tourism. *Current Issues in Tourism*, *15*(3).

Josi, A., Abdillah, L. A., & Suryayusra. (2014). Penerapan Teknik Web Scraping pada Mesin Pencari Artikel Ilmiah. *Jurnal Sistem Informasi*, *5*, 6.

Krause, J. (2020). *Introduction to Bootstrap*. Apress, Berkeley, CA.

Rabby, S. I. (2017). *The Web Application Based On Web Scraping*. Stamford University Bangladesh.

Rani, S. B. A. S. U. (2017). A detailed study of Software Development Life Cycle (SDLC) Models. *International Journals of Engineering and Computer Science*, *6*(7).

Roman, A. (2018). *Black-Box Testing Techniques*. Springer, Cham.

Sahria, Y. (2020). Implementasi Teknik Web Scraping pada Jurnal SINTA untuk Analisis Topik Penelitian Kesehatan Indonesia. *Proceeding of The 11th University Research Colloquium 2020: Bidang Sains Dan Teknologi*.

Saurkar, A. V., & Pathare, K. G. (2018). An Overview On Web Scraping Techniques And Tools. *International Journal on Future Revolution in Computer Science and Communication Engineering*, *4*(4).

T, K. (2019). Personalized Content Extraction and Text Classification Using Effective Web Scraping Techniques. *International Journal of Web Portals*, 12.

Xu, S., & Chen, L. (2016). A Comparative Study on Black-Box Testing with Open Source Applications. *International Conference on Software Engineering*.

Zhao, B. (2017). Web Scraping. *Springer International Publishing AG*.