Development of automatic spam detection application based on modular programming

S Subhan\textsuperscript{1*}, R Arifudin\textsuperscript{1}, D A Efrilianda\textsuperscript{1}, B Prasetiyo\textsuperscript{1}

\textsuperscript{1}Department of Computer Science, Universitas Negeri Semarang

*Corresponding author: subhan@mail.unnes.ac.id

Abstract. This article discusses developing an application for automatic spam detection based on modular programming. In general website development, certain development features cannot be used on other similar websites. We use modular programming to solve this problem. The problem is how to develop an automatic spam detection application based on modular programming. We used the software prototyping method for application development. The results of this study are obtained an automatic spam detection website application based on modular programming.

A. Introduction
The public is getting easier to access the internet. One form of service on the internet is the website. A website is a facility that contains a distributed database. This internet service is the most widely known and the fastest growing technology. This service uses hypertext links called hyperlinks to refer to and retrieve web pages from the server. Web pages can contain sound, images, animation, text, and software programs that compose them into dynamic documents. Users can access the website with a browser, a program that can display HyperText Markup Language (HTML) or web page scripts [1]. Many internet users in Indonesia (64.8%) use a browser to browse while accessing the internet [2].

A large number of internet users is accompanied by many institutions that use websites to publish their institutions. The institutional website that is built can be used to convey specific information. One part of the website is providing a comment facility so that the information conveyed becomes interactive.

According to WordPress, a blog-publishing service provider states that its users make an average of 70 million new posts and 77 million new comments every month [3]. Unfortunately, with a large amount of traffic, there is a low management gap so that the website can become a target for spammers. Spammers usually give spam comments. These comments usually do not want by the website owner. Spam comments are a problem in itself, namely the decline in the website’s quality of comments. For that, we need special handling. In handling it, applications need to be developed without interrupting existing processes. Modular programming is used to handle this process. Davis states that modular programming can also reduce the complexity of programming [4]. In this article, the problem raised relates to developing an automatic spam detection website application based on modular programming.

Spam uses electronic devices to send messages without being desired by the recipient or in the form of junk messages [5]. People who do spam are called spammers. This act of spam is known as spamming. Commonly known spam forms include email spam, instant message spam, Usenet newsgroup spam, web search-engine spam, website spam, wiki spam, online classifieds spam, social network spam. Spam
on websites is done by posting (usually automatically) random comments, copying material from other places that are not authentic, or promoting commercial services to blogs, wikis, guestbooks, or other publicly accessible online discussion forums. Any web application that receives and displays hyperlinks sent by visitors may be a target. This characteristic of spam is essential in spam detection [6].

Adding links are pointing to spammers' websites artificial increases the site's ranking in search engines where URL popularity contributes to its fundamental value. An example is the PageRank algorithm, as used by Google search. It will increase the spammer's commercial sites listed in front of other sites for specific searches, increasing potential visitors and paying customers.

Besides providing comments by providing links, spam comments can also provide inappropriate comments for children to read. Besides, spam comments will waste resources owned by the website, such as bandwidth, storage, and website productivity [7]. For this reason, this spam selection process is vital [8].

Text preprocessing is the process of processing the original data prepared for processing at a later stage. This text preprocessing is part of the text processing used in the spam detection process [9]. Enter the initial data, namely in the form of documents. The text preprocessing stages include Case Folding, Filtering, Tokenization, Stopword Removal, and Stemming. Case folding is a stage of text processing where all text is converted into the same case. In this study, all letters in the document text are uniformed into lowercase letters.

Filtering refers to the process of deciding which terms should be used to represent the document so that it can be used to describe the contents of the document and differentiate the document from other documents in the collection. This filtering will reduce the original text from the existing data [10]. In other words, filtering is a process where text other than the characters "a" to "z" and spaces are removed.

The tokenization process is cutting the input string based on each word that composes it, breaking the document, and separating it into each word. The document splitting into single words is done by scanning the document, and each word is identified or separated from another word by a space separator. This feature consists of the capitalization type (case type), digits, punctuation marks, special characters, etc. The output from the tokenization process will be used as input in the text transformation stage.

The stopword removal process is intended to determine whether a word is entered into the stopword or not. The stoplist contains a set of 'irrelevant' words but often appears in a document. The stoplist to be used is from https://www.ranks.nl/stopwords. In other words, a stoplist contains a set of stopwords. Stopwords removal is the process of removing "irrelevant" words from a text document by comparing them to an existing stoplist. This stopword is an essential process in text processing [11].

Stopwords usually contain words that often appear in the form of conjunctions, prepositions, pronouns, conjunctions. However, the meaning is not descriptive and has no relation to a particular topic. Stemming is a process to find the root of a word by removing all affixes [12]. Based on the assumption that terms with the same root word will always have the same meaning, stemming is used for information retrieval to improve retrieval accuracy (information retrieval).

Porter stemmer is a stemming algorithm that has long been used. Porter Stemmer was first published in 1980. Also, Porter stemmer is a stemmer that does not use a dictionary because based on the fact that resources such as an extensive digital dictionary are costly.

Modular programming is programming by making code into many different code modules that are developed separately. Modular programming is the leading solution for managing massive software complexities [4]. This modular programming allows different developers to separate parts of the system and design and implement them without understanding anything else. With modular programming, applications are easy to use again on new systems [13].

With this modular programming, the modules that are created must be isolated from other programs. Local reasoning is used in developing code modules in modular programming: reasoning about modules and the contracts that need to be fulfilled for the rest of the program. After the modules have been made separate, they can then be linked together to form a program.
B. Method
This automatic spam detection application based on modular programming was developed based on the software prototyping method. Garcia-valls (2016) states that this method makes the application creation process more flexible. Making software consists of four stages: (1) determining the purpose of making the application, (2) selecting the function, (3) making the application, (4) evaluating the application as shown in Figure 1.

![Figure 1. software prototyping method](image)

Both the developer and the customer must know what the application is trying to achieve. Setting application creation goals is one of the first activities a developer has to do. In the function selection step, the application must have directed the functions that the application will perform.

Prototyping involves the actual development process required to produce a prototype. Prototyping is influenced by the nature of the prototype to be produced. Prototype evaluation is perhaps the most crucial step in the prototyping process and for which there is little knowledge. Evaluation prototypes must have proper practice and the ability for evaluation sessions. In this study, this method is used in modular applications. Modular applications are made, as shown in Figure 2. Installation of this spam application does not interfere with the website application.

![Figure 2. Design Modular Application](image)

C. Result
This automatic spam detection application based on modular programming was developed based on the software prototyping method. Making a prototype consists of four stages: (1) determining the purpose of making the application, (2) selecting the function, (3) making the application, (4) evaluating the application.

This application was created at the goal-setting stage to build an automatic spam detection application with modular programming. In the function selection section, the function is selected (1) the application can be enabled to detect spam automatically, (2) the application can be used by various websites easily.

At the application development stage, the researcher has created an application to detect spam, as shown in Figure 3. This application is then applied to websites where there is no spam detection application. This website has a lot of spam, as in figure 4.
At the evaluation stage, this application is evaluated for its spam detection capabilities. With the existence of this anti-spam application, spam is automatically disposed of in the trash. Pay attention to one of the following comments in Figure 5. It can be seen that this spam has characteristics as conveyed by Aisyar [15], namely the content of the comments is not appropriate. This comment on the previous website was not detected as spam. With this application, comments are immediately thrown into the trash section. The application meets modular programming standards, which is easy to use on new websites, especially WordPress-based websites [16]

D. Conclusion
This paper describes the development of an automatic spam detection application based on modular programming. The application can detect spam automatically and easy to implement on new websites.

References
[1] Laudon, Kenneth C, Jane P, Laudon 2012 Management System: Managing the Digital Firm Twelfth Edition (New Jersey: Prentice Hall)
[2] Asosiasi Penyedia Jasa Internet Indonesia. 2019 Penetrasi & Profil perilaku Pengguna Internet Indonesia 2018 (Asosiasi Penyelenggara Jasa Internet Indonesia).

[3] WordPress 2020 “Stats – WordPress.com”, [Online]. https://wordpress.com/activity/. [Accessed 04 May 2020]

[4] Davis D, Burry J, & Burry M 2011 Int. J. Archit. Comput. 9(4) 361

[5] Faris H, Al-Zoubi A M, Heidari AA, Aljarah I, Mafarja M, Hassonah MA, & Fujita H 2019 Inf. Fusion 48 67

[6] Crawford M, Khoshgoftaar T M, Prusa JD, Richter AN, & Al Najada H 2015 J. Big Data 2(1) 23

[7] Mohammad AH, Zitar RA 2011 Appl. Soft Comput. 11 3827

[8] Günel S, Ergin S, Gülmezoğlu MB, Gerek ÖN 2006 International Workshop on Multimedia Content Representation, Classification and Security, Springer 635

[9] Almeida TA, Silva TP, Santos I, & Gómez Hidalgo JM 2016 Knowledge-Based Syst 108 25

[10] Li J, Fong S, Zhuang Y, & Khoury R 2015 Soft Comput. 20(9) 3411

[11] Raulji, J K and Saini JR 2016 Int. J. Comput. Appl. 150 15

[12] Singh J & Gupta V 2016 ACM Computing Surveys 49(3) 1

[13] Park J, Esmaeilzadeh H, Zhang X, Naik M & Harris W 2015 Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2015

[14] Aiyar S & Shetty NP 2018 Procedia Comput. Sci. 132 174

[15] Coronado E, Mastrogiovanni F & Venture G 2018 2018 IEEE Workshop on Advanced Robotics and Its Social Impacts (ARSO). doi:10.1109/arso.2018.8625839