1. Introduction

Software is built based on the preferences and requirements of the client. In software development, interviews or questionnaires are required so the developers know the software specifications that match the client's requirements. The results of this interview or questionnaire are commonly referred to as Requirement Elicitation [1], [2]. In the flow, a Software Requirement Specification document is created as a developer's guide in software development after making a Requirement Elicitation [3]–[6].

When the software has been completed, the developer will send the progress reports to the client. In this process, developers are prone to receiving revisions and changes ranging from small-scale changes to major overhauls, significantly when the results of the developer's understanding differ from the client's preferences and requirements. This is certainly not effective in terms of the time and energy that has been spent on the development of software. In addition, the results will also not satisfy the client's needs [7], [8]. The discrepancy between the developer's understanding of the client's preferences in the Software Requirement Specification document is one of the backgrounds in this research. So this research was conducted to measure the similarity between Requirement Elicitation and Deployment Diagram to find out how much the developer's understanding of the specifications requested by the client.

This research applies the Software Requirement Specification from the Baker's Corner application. Baker's Corner is an Android-based cake and pastry product ordering platform. This research will analyze the similarity value between Requirement Elicitation and Deployment Diagram through the Non-Functional Requirement reference on Software Requirement Specification in the Baker's Corner application.

Before calculating the similarity value between artifacts, extracting text data on Requirement Elicitation, Non-Functional Requirement, and Deployment Diagrams is necessary. After that, text pre-processing will be implemented. Text pre- processing involves several steps, including case folding, defining the dataset to be utilized, tokenization, stemming, and stopwords. The results obtained from text pre-processing will then calculate the similarity value with the cosine similarity formula and then be tested for validity and reliability with Gwet's AC1 formula [9]. After obtaining the validity test results and the cosine similarity value, the similarity value between Requirement Elicitation, Non-Functional Requirement, and Deployment Diagrams will be obtained.

The main purpose of this research is to process text data on Non-Functional Requirement (NFR) where the extraction results can be used as a reference for comparing the suitability between Requirement Elicitation and Deployment Diagram with text pre-processing method. There are contributions to this research, namely:

1. Extracting text data on Requirement Elicitation, Non- Functional Requirement, Deployment Diagram through text pre-processing.
2. Calculating similarity between Requirement Elicitation and Deployment Diagram against Non-Functional Requirement as a reference for its suitability.
3. Conduct validity and reliability testing using Gwet's AC1 formula.
4. Produce a similarity value between Requirement Elicitation with Non-Functional Requirement and Non-Functional Requirement with Deployment Diagram.
5. Provides recommendations for improvement to documents that have low kappa value.