

# UML Generator – An Automated System for Model Driven Development

Chamitha Ramal Narawita<sup>#1</sup>, Kaneeka Vidanage<sup>#2</sup>

<sup>#</sup>Department of Computer Science, Informatics Institute of Technology,  
Ramakrishna Road, Colombo - 06, Sri Lanka.

<sup>1</sup>chamithanara@gmail.com/chamitha.2012178@iit.ac.lk

<sup>2</sup>kaneeka.v@iit.ac.lk

**Abstract**— This research mainly focused on automation of Unified Modeling Language (UML) diagrams from the analyzed requirement text using Natural Language Processing (NLP). The proposed system is an efficient and accurate way to obtain elements of the use case and class diagrams from proposed methods. This research mainly focuses on the design phase of a software. Nowadays everybody needs a quick and reliable service. It was needed to have some sort of quick, accurate and intelligent software for generating UML based documentations to save time and budget of both the user and system analyst [1].

**Keywords**— UML Diagrams, Use Case Diagram, Class Diagram, Natural Language Processing, Artificial Intelligence, Machine Learning, Software Design Phase

## I. INTRODUCTION

The automation of UML Diagrams using natural language processing is a highly challenging task due to the following reasons [2].

- Natural language is arguable. Thus, detailed and error-free analysis is very difficult.
- There could be different ways of representing the same semantic.
- Concepts that were not explicitly expressed in a written source are often very difficult to model. Usually, expert domain knowledge is needed to identify the hidden classes.

Design phase is the most important among the other phases because blueprint of a system helps developers to avoid all the misunderstanding regarding the software by involving the users. Requirement engineers analyze requirements manually to come out with highly precise UML diagrams for their systems. By modeling a system, most important aspect is to capture the dynamic behaviors. Static behaviors are not sufficient to build models for a system rather using dynamic behaviors. Use case diagram shows dynamic aspects of a system with both internal and external interactions. They describe the events of a system and their flows. However, the use cases never describe how they are implemented. When it comes to class diagrams its other way around. The class diagrams can be define as static diagrams and they represent the static view of a system. Class diagram contains classes of the system, their attributes, operations and constraints imposed on the system. They can be

directly map with the object oriented languages as well. Therefore, to give a both dynamic and a static view of the scenario, UML Generator generates use case and class diagram by analyzing the input text. This research will be contributing towards filling the gap between gathered user requirements and the phase of the implementation by sorting out the problems that above mentioned [3].

According to the norms and conventions, before drawing the UML diagrams the system analyst has to do a lot of work by analyzing the business logics and figuring out the user requirements. In recent times, there is no software which provides services by manually drawing UML diagrams more efficiently except Rational Rose and Smart Draw and there is no doubt that these are reasonably good software but with many disadvantages. First, analysis need to investigate the requirements and then draw the models separately. Hence, there is wastage of so much time when using current available tools to create UML models for the required scenario. Nowadays everybody needs a quick and reliable service. Moreover, the time spent on analyzing systems and poor quality of human analysis shows the need of automated support.

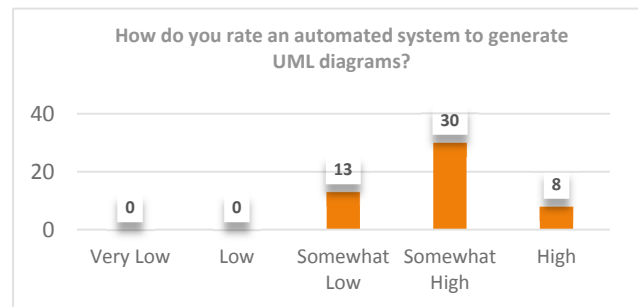


Fig. 1 Importance of an automated system

As shown in figure 1 according to the survey carried out to gather requirements, most of the users have agreed to the researcher's opinion, which 75% (38 out of 51) of overall responses have positive thoughts regarding this system.

## II. LITERATURE REVIEW

### A. Natural Language Processing (NLP)

After user enter the requirements, there should be a mechanism to extract the information and to understand the text from currently available techniques. To analyze large amount of text data, currently NLP is the only available technique for the developers. NLP is an area of