Generating Use Case Diagram from Requirements Using Natural Language Processing

Prepared by

Samah Ismail Elkhedir Elbashir

Thesis presented in fulfilment of the requirements for the degree of Master of Computer Science

Supervised By

Dr. Mohammed Ahmed Mohammed Alatta

Faculty of Post Graduate Studies

Red Sea University, Sudan

May, 2021

Abstract

Requirements analysis is the preliminary step in software development process, and going from requirements analysis to design phase is considered as one of the most complex and difficult activities in software development. Errors caused during this activity can be quite expensive to fix in later phases of software development.

One main reason for such potential problems is due to the specification of software requirements in Natural Language format. To overcome some of these defects we have proposed a technique, which aims to provide semi- automated way for developers to generate use case diagram from normalized natural language requirements using Natural Language Processing techniques, user writes the requirements in simple English in a few paragraphs and the designed system incorporates NLP methods to analyze the given script.

The objectives of the study are to achieve completeness and consistency of requirements for systems development for high quality of software products also to reduce time in drawing use cases.

We used (Food ordering system) as a case study, and implemented some rules to generate a use case diagram; by using java eclipse and regular expression, also using open NLP tool to process the requirements text and automatically generate the use case diagram. First the NL text is semantically analyzed to extract actors, use cases and relationships.

The results of proposed approach is drawing use case diagram automatically after clicking button just in less time and that reduced mistakes in early stages of software development.

المستخلص

تعتبر مرحلة تحليل المتطلبات الخطوة الأولى في عملية تطوير البرمجيات وأهمها، والانتقال منها الى مرحلة التصميم تعتبر من اصعب واكثر المراحل تعقيدا في تطوير البرمجيات.

الأخطاء التي تحدث أثناء هذه المرحلة تكون غالية الثمن ومن الصعب معالجتها في مراحل متاخرة في تطوير البرمجيات.

السبب الرئيسي للأخطاء التي تحدث أثناء وصف متطلبات البرمجيات هو انها مكتوبة باللغة الطبيعية.

لتلافي وحل هذه الأخطاء تم اقتراح تقنية جديدة لتوليد مخطط حالات الاستخدام من المتطلبات المكتوبة باللغة الانجليزية بإستخدام تقنيات معالجة اللغة الطبيعية. المستخدم يكتب المتطلبات بلغة إنجليزية بسيطة في شكل فقرات، والنظام المصمم يستخدم دوال معالجة اللغة الطبيعية لتحليلها.

من أهداف الدراسة تحقيق تكاملية و تماثلية المتطلبات لتطوير الأنظمة لتحقيق جودة عالية لمنتجات البرمجيات وايضا لتقليل زمن رسم مخطط حالات الإستخدام.

تم إستخدام (نظام طلب طعام)، وتطبيق بعض القواعد عليها لتوليد مخطط حالات الإستخدام، وذلك بإستخدام جافا اكليبس والتعبيرات المنطقية، وأيضا استخدمت أداة معالجة اللغة الطبيعية المفتوحة لمعالجة المتطلبات، وبعد ذلك يتولد مخطط حالات الإستخدام. النص اولا يكون مكتوب باللغة الطبيعية ويتم معالجته لإستنتاج اللاعبين وحالات الإستخدام والعلاقات بينها.

نتائج هذا الإسلوب المقترح هو رسم مخطط حالات الإستخدام تلقائيا بعد الضغط فقط على الزر وفي زمن أقل وبالتالي تقليل الأخطاء في مراحل أولية في تطوير البرمجيات.

Table of Contents

5.1 Conclusion

The approach will help software analyst for reducing their time in analysis process. Our proposed system understanding the input scenario via applying NLP techniques to extract knowledge from user requirements written in English, such as tokenization and POS tagging to parse the system specifications based on predefined set of syntactic heuristics rules.

As one of the most important issues in software engineering much work can be done, in process of UML generation more diagrams can be generated to give a full

design views like class diagram, active diagram and sequence diagram, this means that the new rules may be needed to assists in process of building diagrams. And that the limitation of our approach to generate use case diagram just, so we recommend to automated generation of other UML diagrams.

5.2 Recommendations

Our approach can generate only self-contained concrete use cases which constitutes a complete flow of events. It does not have the capability to reuse other existing use cases via include and generalize relationships. Therefore, we aim to provide this capability to reuse the existing use cases in order to reduce the efforts required to define the use cases in the system. On the other hand, we aim to extend our system to have more semantic analysis to infer more linguistic features and to improve the system performance.

One of the largest issues in requirements engineering is how to keep high consistency between requirements and design, this problem happen if we want to trace requirements to design phase from design phase to requirement in many cases the requirements engineering many want to change requirements after design was made, in this case it’s important to trace all made changes and apply these changes to design, and same steps happened to design, this process missed in current approach but was covered partially in many researches.