Specmate: Automated Creation of Test Cases from Acceptance Criteria

Jannik Fischbach[1] Andreas Vogelsang[2] Dominik Spies[3] Andreas Wehrle[4] Maximilian Junker[5] Dietmar Freudenstein[6]

Abstract: We summarize the paper *Specmate: Automated Creation of Test Cases from Acceptance Criteria* [Fi20b], which was presented at the 2020 edition of the IEEE International Conference on Software Testing, Verification and Validation (ICST).

Keywords: test case creation; natural language processing; model-based testing; user stories; agile software development

1 Introduction

User stories are an established instrument for the notation of system requirements in agile software projects. A user story is fulfilled if all specified acceptance criteria (AC) are satisfied. This requires testing the defined AC by creating, executing, and maintaining both positive and negative test cases (*Acceptance Testing*). Test case design is a very laborious activity that easily accounts for 40-70 % of the total effort in the testing process: First, the right input-output combinations need to be determined to comprehensively test the requirement, which is not trivial, especially for complex system requirements [Fi20a]. Secondly, the number of test cases should be kept to a minimum to avoid unnecessary testing efforts. Furthermore, the creation of test cases has to be mostly done manually since there is a lack of tool support. Existing approaches support the test case generation from formal and semi-formal requirements, but are not suitable for informal requirement descriptions based on unrestricted natural language. Unrestricted natural language, however, is the dominant way of formulating AC, as we found in the analysis of 961 user stories from two projects together with our industry partner Allianz Deutschland. We address this research gap and present Specmate as an approach to reduce the manual effort of deriving test cases from AC by applying *Natural Language Processing* (NLP).

[1] Qualicen GmbH, Munich, Germany, jannik.fischbach@qualicen.de
[2] University of Cologne, Germany, vogelsang@cs.uni-koeln.de
[3] Qualicen GmbH, Munich, Germany, dominik.spies@qualicen.de
[4] Qualicen GmbH, Munich, Germany, andreas.wehrle@qualicen.de
[5] Qualicen GmbH, Munich, Germany, maximilian.junker@qualicen.de
[6] Allianz Deutschland AG, Munich, Germany, dietmar.freudenstein@allianz.de

doi:10.18420/SE2021_07

2 Our Approach

We argue that a valuable automated solution for generating test cases from user stories and their AC can only be created by understanding both their content and form. For this purpose, we analyze 961 user stories provided by our industrial partner to determine requirements for the automated approach. Based on these requirements, we design an approach based on NLP to generate corresponding test cases automatically. We follow the idea of *Model-Based-Testing* and introduce an intermediate layer between the user stories and the final test cases. We extract the AC from the user stories and transfer each into a test model. Since we found in our case study that the expected system behavior is usually described in the form of cause and effect relationships (e.g. *In the case of <cause>, the system shall <effect>*), we use *Cause-Effect-Graphs* (CEG) as test models. In order to transfer the AC into a CEG, the relevant causes and effects must be identified within the AC. For this purpose we apply *Dependency Parsing* and first convert each AC into a dependency tree. Subsequently, we traverse the dependency tree and generate the CEG. Finally, we derive the minimum number of test cases from the CEG by applying the *Basic Path Sensitization Technique*.

3 Our Results

Our case study demonstrates that not every user story provides functional information to generate test cases. Depending on the project, user stories are increasingly used as a means of communication. In contrast, about 31.1 % to 50.1 % of the observed user stories describe the functionality by AC. We hypothesize that there is a high automation potential in test case derivation from these *functional user stories*. In this context, a major challenge arises in processing the informal nature of the AC, which is the dominant type of notation. Despite the use of unstructured language, the majority of AC are characterized by recurring patterns, of which cause-effect-relationships have the broadest application. We evaluated Specmate based on 604 test cases that have been manually derived from 72 user stories by test designers from our industry partner. Our experiments underline the practical benefits of Specmate. 56 % of the manually created test cases could be generated automatically and missing negative test cases are added. The missing 44 % stems from required domain knowledge and poor data quality within the AC. We hypothesize that a full automation of the test creation from AC can hardly be achieved. Our approach should therefore be seen as a supplement to the existing manual process.

Bibliography

[Fi20a] Fischbach, J.; Femmer, H.; Mendez, D.; Fucci, D.; Vogelsang, A.: What Makes Agile Test Artifacts Useful? An Activity-Based Quality Model from a Practitioners' Perspective. In: ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). 2020.

[Fi20b] Fischbach, J.; Vogelsang, A.; Spies, D.; Wehrle, A.; Junker, M.; Freudenstein, D.: SPECMATE: Automated Creation of Test Cases from Acceptance Criteria. In: IEEE 13th International Conference on Software Testing, Validation and Verification (ICST). 2020.