Philipps-Universität Marburg
Fachbereich Mathematik und Informatik
AG Softwaretechnik

Analysis of Conflicts and Dependencies between User Stories using Graph Transformation

Masterarbeit
zur Erlangung des akademischen Grades
Master of Science

vorgelegt von
Amir Rabieyan Nejad
Matrikel-Nr: 3350269

16. März 2024

1 Analysing Redundancy

In this Section, we present an approach for syntactically analysing redundancy using the CRF tool and Henshin.

In Section 1.1 we illustrate the requirements and functional needs that are used as input to the design phase to satisfy and bring value and benefit to the stakeholder. In the Section 1.2 we explain the design decisions of the workflow shown in **??** and we explain how the architecture is structured and what the components and their classes look like. In Section **??** comes the implementation to show what we have implemented and in Section **??** we show how we have tested it. Finally, we will apply this approach to 19 backlogs entered with the CRF tool[1] we enter all backlogs as input and the results are evaluated in section**??**.

1.1 Requirements

In order to accomplish the analysis of redundancy in USs we try to address following requirements:

- As a member of a project group, I want to syntactically analyse USs that belong to a specific backlog, so that redundancies between USs can be recognise and manage accordingly.
- As a member of a project group, I would like to have a collection of US-pairs as a report that syntactically have the same clause in parts of user stories, so that I can change them if necessary.
- As a member of a project group, I want to filter the report and only see the redundancy clauses that contain "Action"(as verb) and "Entity"(as noun) which is called "Targets" that are duplicated in US-pair, so that I can reduce the number of potential redundancy pairs in the report.
- As a member of a project group, I want to mark founded redundancy clauses with hash marks(#) and list those that contain "Persona"(as a noun) and "Action"(as a verb) which is called "Triggers", so that I can better see whether the persona in is also recognised as a redundancy.
- As a member of a project group, I want to mark justified redundancy clauses with hash marks (#) and list those that contain "Entity"(as a noun) and "Entity"(as a noun), which is called "Contains", so that I can better see whether the contained entity is also recognised as a redundancy.
- As a member of a project group, I would like to have a redundancy report that shows substantiated clauses in US-pairs and adds a hash symbol (#) at the beginning and end of the substantiated clauses as tag in each user story, so that I can see which words are duplicated in a part of the sentences.

[1] https://github.com/ace-design/nlp-stories

- As a member of a project group, I want to see how many clauses are in each US-pair, so that I can aggregate each redundant US-pair based on it.
- As a member of a project group, I want a table at the top of the redundancy report that lists the US-pairs and the number of clauses contained in each pair, so that I can quickly see all the US-pairs founded and the number of clauses.
- As a member of a project group, I want to know whether the redundancy clauses belong to the main or benefit part of the US, so that I can make a decision accordingly.

1.2 Design

In this section