System Requirement Conflict Identification and Resolution using LLMs

Trung Kien Nguyen, Phong Tran, Quoc Co Luc
Swinburne University of Technology, Hawthorn, Victoria, Australia
Email: {104053642, 104334842, 103830572}@student.swin.edu.au

***Abstract*—System requirements often emerge from different stakeholders with diverse goals, priorities, challenges, motivations—leading to inconsistencies and conflicts that slow down system design. This project presents an automated framework for identifying and resolving conflicts using Large Language Models (LLMs). By utilizing the natural language understanding and reasoning capabilities of LLMs, the system extracts detailed requirements and usage contexts from multiple perspectives, detects contradictions across them, and proposes resolutions through guided strategies. The pipeline supports both functional and non-functional requirements and incorporates structured reasoning, persona-based simulation, and conflict identification analysis. As no existing solution offers fully automated conflict identification and resolution in requirements engineering, this research demonstrates a significant advancement toward intelligent, scalable, and inclusive software specification.**

***Index Terms*—Requirements Engineering, Requirements Clustering, Conflict Identification and Resolution,**

I. Introduction

In requirements engineering, one of the most persistent and complex challenges lies in reconciling conflicting needs and expectations from diverse stakeholders. As modern systems increasingly cater to heterogeneous user bases, it becomes critical to ensure that system requirements represent each user's perspective accurately, while avoiding internal inconsistencies and contradictions. This report investigates an automated framework for detecting and resolving such requirement conflicts—particularly at the user story level, using the capabilities of the State-of-the-art (SOTA) Large Language Models (LLMs).

To contextualize our study, we explore the ALFRED system—a virtual assistant platform designed to support older adults in living independently while enabling caregivers, medical staff, and developers to engage meaningfully with its ecosystem. ALFRED's documentation includes detailed user personas and use cases, which serve as an ideal foundation for investigating requirement-level conflicts across different user groups. These personas often reflect contrasting values: for example, one older adult may prioritize privacy and minimal interaction, while another may demand constant motivational feedback from the system. Such divergences can lead to requirement conflicts that, if left unresolved, hinder system integrity and usability [1].

Early iterations of our pipeline revealed a major limitation in using LLMs like ChatGPT for conflict identification. The model frequently misclassified the relationship between user stories—often labeling two unrelated or mutually supportive user stories as "contradictory," simply due to superficial differences in wording or inferred intent. This issue, raised during Weeks 6 and 7 of client consultations, underscored the opacity of LLM reasoning and the risk of over-interpreting vague language patterns as conflicts. It highlighted the need for a more structured conflict detection approach that could distinguish between genuine contradictions and contextually compatible expectations.

This report aims to answer the following research questions:

- **RQ1:** How can we generate user stories that are specific, unambiguous, highlighting the persona's singularities?
- **RQ2:** How can we reliably identify subtle or indirect requirement conflicts between user stories—especially when there is no obvious contradiction?
- **RQ3:** What resolution strategies are most suitable for addressing different types of conflicts identified between user stories?

Ultimately, this study contributes a methodology for automated system requirement conflict identification and resolution using LLMs, while grounding the approach in a realistic system (ALFRED). The framework is extensible and applicable to other domains where persona-driven requirement modeling is essential.

Structure. Section II provides the background of our study.

II. Background

In this section, we introduce the background related to ...

III. Methodology

Fig. 1 provides an overview of our approach.

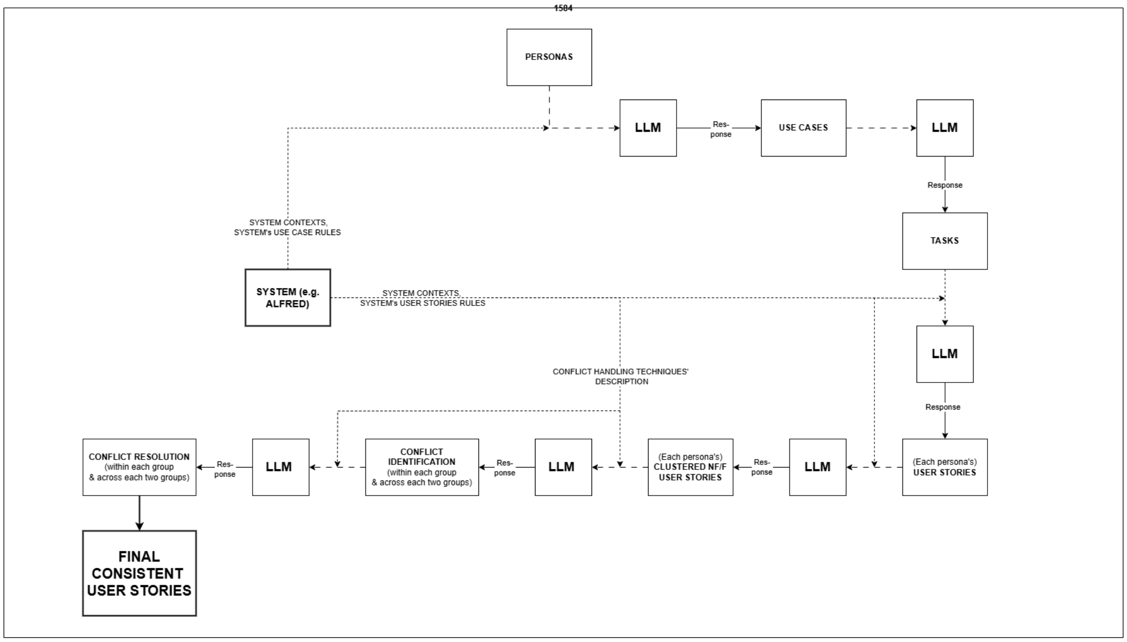

Fig. 1. Pipeline Overview - Summary version

A. Overall Pipeline

The methodology employed in this project follows a structured and iterative pipeline designed to ensure comprehensive and consistent generation of system requirements driven by user personas and use cases. The overall pipeline comprises the following key steps:

1) **Persona and System-specific Contexts Collection:** This initial step involves collecting detailed user personas along with system-specific contexts that capture the environment, constraints, and user persona (stakeholder) perspectives. These contexts provide essential background to ground subsequent use case and requirement generation, validation, and analysis.
2) **Requirement (User Story) Generation:** Building upon the collected personas and contexts, use case generation and task extraction are performed as intermediate steps. Use cases describe goal-directed interactions between users and the system, while task extraction breaks these scenarios down into specific user tasks. These form the basis for generating detailed requirements, also referred to as user stories, which reflect concrete system functionalities and constraints.
3) **Requirement Clustering (Categorization)** The pipeline clusters similar requirements into thematic groups based on their functional or non-functional nature. This categorization facilitates easier management and understanding of requirement domains.
4) **Conflict Identification:** An essential phase involves detecting conflicts and inconsistencies among requirements, especially those arising from different user groups or conflicting stakeholder needs. The system leverages automated analysis to highlight potential requirement clashes.
5) **Conflict Resolution and Negotiation:** Following conflict identification, the pipeline supports the resolution process by proposing reconciled or unified requirements that balance competing needs, ensuring system coherence. The final set of system requirements, in which no conflicts or inconsistencies arise, forms a solid basis for system design and implementation.

This pipeline is iterative and modular, allowing continuous refinement as new personas, use cases, or system constraints emerge. The integration of automated LLM-based generation and conflict resolution tools significantly accelerates the requirements engineering process while maintaining high quality and relevance.

B. Data in This Project

Fig. **??** illustrates the initial data inputs, including system guidelines and persona groupings. This project relies on a rich and structured set of input data that forms the foundation for automated requirement generation, conflict detection, and resolution. The data can be broadly categorized into system-specific guidelines and user personas, which reflect diverse stakeholder perspectives.

a) System-Related Guidelines: These consist of high-level descriptions and formalized rules defining the system context, fundamental pillars, and the structure and format of use cases and user stories. For example, the ALFRED system acts as a reference case, with its documented system summary and core pillars guiding the generation and validation of requirements. Additional use case-related rules specify definitions, expected structures, scenario examples, and task extraction methodologies. Likewise, user story-related rules provide guidance on the user story format, examples, and crucially, clustering techniques that group requirements by thematic or functional similarity.

b) User Personas and Stakeholder Groups: Personas represent archetypal users or stakeholders, each characterized by detailed attributes, goals, challenges, and unique singularities. These personas are organized into distinct user groups reflecting their roles within the system ecosystem—for example, developers and app creators, medical staff and caregivers, and older adult users. Each group is governed by tailored guidelines to capture their specific needs and constraints. The personas provide the human-centric perspective essential for personalized and context-aware requirement engineering.

By integrating these data—the system guidelines and personas—our approach ensures that automated user story generation and conflict analysis are both comprehensive and contextually relevant, faithfully reflecting the diverse viewpoints of all stakeholders involved.

IV. Conclusion

In this paper, we ...

Acknowledgement. XXX

References

[1] A. Consortium, “D2.3 – User Stories and Requirement Analysis,” ALFRED Project, Tech. Rep. Version 1.4, March 2022, 001-D23UserStoriesAndRequirementAnalysisv14ForApproval.pdf.