PhD in Information Technology and Electrical Engineering
Università degli Studi di Napoli Federico II

PhD Student: Marco De Luca

Cycle: XXXVII – 37°

Training and Research Activities Report

Academic year: 2022-2023 - PhD Year: Second

student signature

Tutor: prof. Anna Rita Fasolino

tutor signature

Co-Tutor: Pasquale Cimmino (Micron Semiconductor Italia)

Date: October 31, 2023

Training and Research Activities Report

PhD in Information Technology and Electrical Engineering

Cycle: XXXVII **Author: Marco De Luca**

1. Information:

- **PhD student: Marco De Luca** **PhD Cycle: XXXVII**
- **DR number: DR995860**
- **Date of birth: 15/12/94**
- **Master Science degree: Computer Science Engineering University: Univerisità degli Studi Di Napoli Federico II**
- **Scholarship type: *Funded by Micron Semiconductor Italia S.R.L.***
- **Tutor: Prof. Anna Rita Fasolino**
- **Co-tutor: Pasquale Cimmino (*Micron Semiconductor Italia S.R.L.*)**

2. Study and training activities:

Activity	Type[1]	Hours	Credits	Dates	Organizer	Certificate[2]
Using Deep Learning Properly	Course	10	4	10-12-17-19 - 24/01/2023	Dr. Andrea Apicella	Y
Teaching activities regarding practical lectures/seminars during the courses of "Ingegneria del Software"	Tutorship	12	0.48	21/03/203 - 26/05/2023	Prof. Anna Rita Fasolino	N
Research activity in place at Micron about: Software Requirement Management and Modeling in compliance with ISO 26262	Research	-	4	01/11/2022 - 31/12/2022	-	-
Research activity and write for: Investigating the Robustness of End-to-End Test Cases in Template-based Web Applications using a GUI Change Classification Model	Research		3	01/11/2022 - 31/12/2022	-	-

Writing e submission of as co-author the paper "*Documenting Software Architecture Design in Compliance with ISO 26262*"	**Research**	**-**	**2**	**01/11/2022 - 31/12/2022**	**-**	**-**
Research activity in place at Micron about: Software Requirement Management in compliance with ISO 26262	**Research**	**-**	**3**	**01/01/2023 - 28/02/2023**	**-**	**-**
Research activity and write for: Investigating the Robustness of End-to-End Test Cases in Template-based Web Applications using a GUI Change Classification Model	**Research**	**-**	**3**	**01/01/2023 - 28/02/2023**	**-**	**-**
Research activity in place at Micron about: Software Requirement Management in compliance with ISO 26262	**Research**	**-**	**5**	**01/03/2023 – 31/04/2023**	**-**	**-**
Participation as "Presenting Author" to "International Conference on Software Architecture – ICSA 2023", L'Aquila, Università degli Studi dell'Aquila	**Research**	**-**	**4**	**13-17/03/2023**	**ICSA 2023 – Università degli studi dell'Aquila**	**Y**
Research activity in place at Micron about: Reverse Engineering of Legacy Code	**Research**	**-**	**7**	**01/05/2023 – 30/06/2023**	**-**	**-**
ISSSE 2023 - 16th International Summer School on Software	**Doctoral School**	**24**	**3**	**12/06/2023 - 15/06/2023**	**University of Salerno**	**Y**

Engineering, 12-15 June 2023, University of Salerno, Italy.						
Research activity in place at Micron about: Reverse Engineering of Legacy Code	**Research**	**-**	**5**	**01/07/2023 – 31/08/23**	**-**	**-**
Research activity about Java Code Architecture analysis with ArchiUnit and SonarQube	**Research**	**-**	**3.6**	**01/07/2023 – 31/08/34**	**-**	**-**
Writing of the paper: Analysis of Software Project Quality: A Comparative Study of Computer Science and Computer Engineering Degree Courses	**Research**	**-**	**2**	**01/09/2023 – 31/10/2023**	**-**	**-**
Writing of the paper: reverse engineering approach for the automatic extraction, generation, and evaluation of automotive software architectures	**Research**	**-**	**3**	**01/09/2023 – 31/10/2023**	**-**	**-**
Research Activity in place at Micron about compliance with ISO 26262	**Research**	**-**	**5**	**01/09/2023 – 31/10/2023**	**-**	**-**
Research activities in collaboration with "Gran Sasso Science Institute (GSSI)" about software metric to assess the compliance with safety standard	**Research**	**-**	**3**	**01/09/2023 - 31/10/2023**	**-**	**-**

1) Courses, Seminar, Doctoral School, Research, Tutorship
2) Choose: Y or N

2.1. Study and training activities - credits earned

	Courses	Seminars	Research	Tutorship	Total
Bimonth 1	-	-	9	0	9
Bimonth 2	4	-	6	0	10
Bimonth 3	-	-	9	0.24	9.24
Bimonth 4	-	3	7	0.24	10.24
Bimonth 5	-	-	8.6	0	8.6
Bimonth 6	-	-	13	0	13
Total	4	3	52.6	0.48	60.08
Expected	30 - 70	10 - 30	80 - 140	0 – 4.8	

3. Research activity:

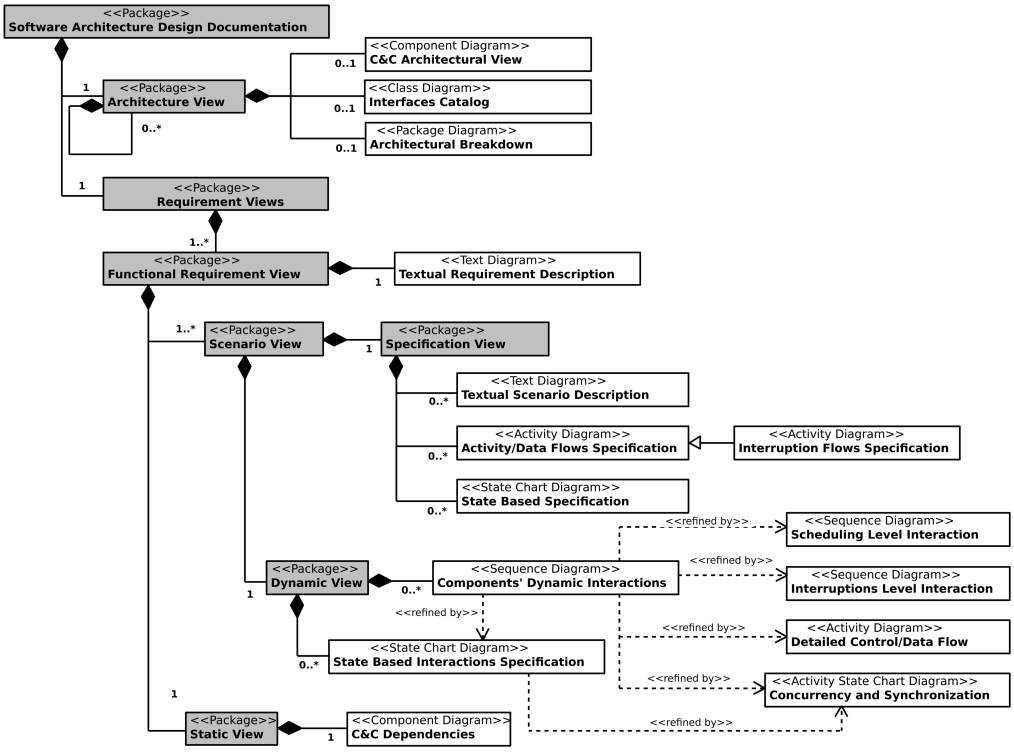
During my second year of my PhD program, I continued my research in the field of software engineering. My research activities can be divided into two main categories: software testing and software architecture development and documentation in the context of safety-critical systems where compliance with the guidelines and requirements of ISO 26262 is required.

In the context of software testing, my research focused on studying techniques to enhance the robustness of locators used for testing template-based web applications.

The second area of my research activities is about the software documentation process in an industrial environment, which necessitates adherence to the principles and guidelines outlined in ISO 26262. My contribution to this research area led to the development of a documentation template that assists automotive practitioners in documenting software in compliance with ISO 26262. To facilitate the use of this proposed documentation template, my ongoing research activities are focusing on reverse engineering techniques to automatically extract and generate software documentation from legacy code. Additionally, the proposed documentation template enables the extraction of metrics that help verify ISO requirements for Software Architecture Design (SAD), such as restricted size of software interface, minimizing software module complexity, and achieving low coupling and high cohesion.
In the following the detailed activities of my research:

- **Software Documentation in the automotive domain:** The complexity of automotive systems has significantly increased in recent years. ISO 26262 is a standard that addresses the functional safety of the E/E (Electric and Electronic) component of road vehicles. This standard defines a functional safety development V-process model that automotive manufacturing must follow and document to achieve compliance with the standard, otherwise the produced device want be suitable to run in commercial vehicles.
  This Standard covers the development lifecycle of all the different subsystems (both hardware and software ones) of which an Electrical/Electronic Unit is made. My research activities focused on Chapter 6 Clause 7 of the ISO 26262 (ISO 26262-6§7) that discusses *Software Architecture Design (SAD)*. This section of the Standard specifies a set of *requirements, recommendations* and *principles* that need to be followed during software architecture development.
  Several works in literature have described the challenges encountered for implementing different ISO Standard requirements and have presented possible solutions. Even if all these

works supply valuable indication and suggestion from implementing ISO 26262 compliant SAD, it is not easy to translate these guidelines into practical solutions.
To better understand the significant challenges faced by practitioners in producing software documentation compliant with the standard, I conducted a survey involving industry experts from various companies in the automotive sector. Based on the survey results and a literature review, we identified that most of the challenges are primarily related to the inadequate documentation process for SAD and the improper use of design tools to support traceability and consistency management among various architectural views and models.
To overcome these problems, we proposed a *documentation template* for describing the SAD in compliance with ISO 26262§6.7. The template is depicted in the following Figure, and offers solutions for: i) managing and ensuring the consistency among the different views and models' elements of the SAD ii) supporting the SAD refinement process from the highest levels of abstraction towards the lowest levels of detail closer to code implementation iii) aiding the verification process of the design principles defined by the ISO Standard iv) introducing new models for describing both static and dynamic safety related characteristics of the software architecture.



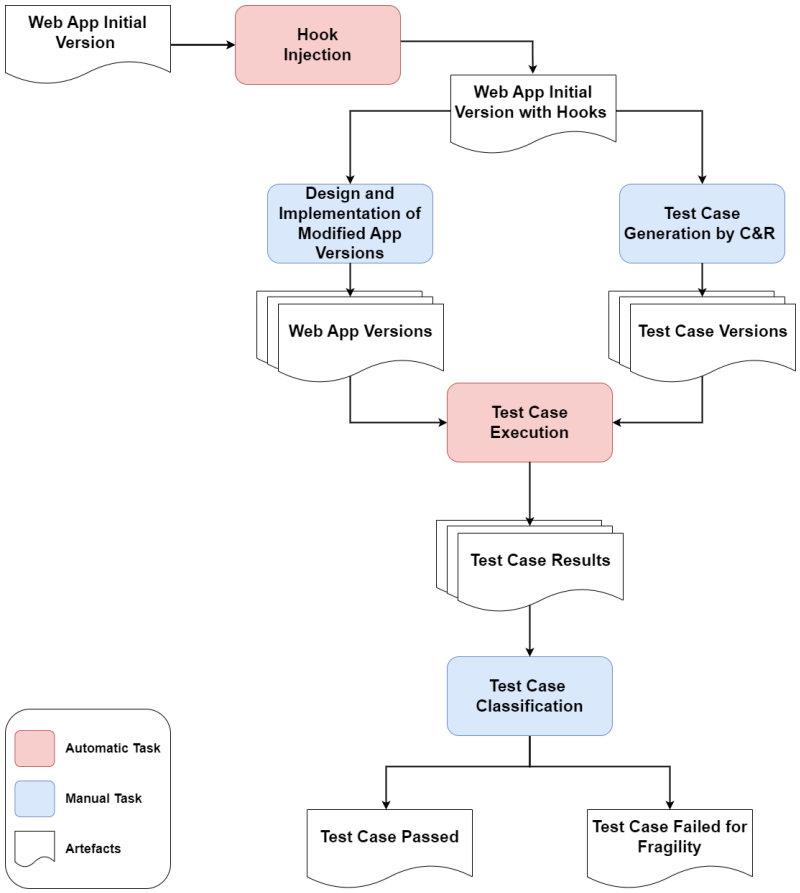
Metrics can be derived from the documentation produced by following the guidelines of the proposed template. These metrics aid in assessing the principles outlined in the standard, such as the number of functions per interface, cohesion, and module coupling, among others. My research in the field of metrics is advancing in partnership with the Gran Sasso Science Institute (GSSI). Together with them, we are conducting research on software and architectural metrics designed to facilitate compliance with ISO 26262. The primary objective of this research is to establish a "Continuous Compliance" software

development process. This process, driven by metrics, offers continuous support for compliance and can be adaptable to iterative development approaches.

Finally, the research I am currently conducting in this field is focused on defining an environment for the recovery of the architectural documentation of software system, referred to as an "ART" (Architectural Recovery Tool). Through the use of reverse engineering techniques, I am exploring the possibility of reconstructing documentation for legacy software while simultaneously applying transformations to make it compliant with the proposed documentation template.

- **Software Testing:** Test cases generated by GUI-based Capture and Replay tools are often vulnerable to the well-known fragility problem. Even small layout changes in a web application can lead to test case failures without impacting the application's functionality. These test case failures are commonly referred to as 'test breakages' to distinguish them from failures that reveal application faults during normal testing. For instance, a simple layout change, such as moving an HTML tag to a different location on the page, can break a test case that previously located the tag using an XPath expression based on the tag's prior position in the HTML tree.
  Capture and Replay (C&R) techniques have been widely used in the industry to perform GUI testing without requiring advanced testing or programming skills. Testers can use C&R tools to automatically generate GUI-based test scripts based on real user interactions with the application's GUI, including mouse clicks, keyboard entries, and navigation commands. These recorded sequences are then translated into executable test scripts by the tool.
  To enhance the robustness of test cases, we've proposed a technique that relies on 'Hook-based' locators. These locators use an HTML tag attribute, known as a hook, to uniquely identify each tag on a web page. What sets our approach apart is that it introduces these hook attributes into the web page templates from which each rendered web page originates. Consequently, when a new page is generated, its tags already include the hook attributes, which make them more resilient to layout changes. This technique demonstrated its effectiveness in a preliminary validation study, where it significantly reduced the number of test case locator failures during regression testing of web applications created by students.
  In the research we compared the robustness of hook-based test cases against state-of-the-art and state-of-the-practice techniques for locating GUI objects. We proposed a three-dimensional model for classifying different types of layout changes and used it to define a benchmark of realistic changes. Thanks to the model, we systematically compared the robustness of test cases generated by different techniques with respect to specific types of changes and studied the relationship between fragility issues and types of changes in different test case generation techniques.
  Our experimental procedure consisted of five steps, outlined in the figure below. The automated steps were implemented using GitHub Actions in YAML scripts, while the artefacts were stored in a GitHub repository. In the first step we automatically injected hooks in the source code of the considered applications. We manually recorded a test case consisting of a sequence of actions (such as click, fill-in a text field, etc..) to be performed on different Web page items and assertions to check the test results. As to the implementation of the changes, we conveniently chose some GUI items associated to the locators included in the recorded test case and applied to them the types of change compatible to that specific type of object.

The modified versions of the Web pages/templates were manually implemented and stored in GitHub repositories. As to the step of Test Case Execution, we exploited the GitHub Actions through the definition of a YAML script that allowed us to setup the same test case execution environment for each test case. Finally, in the Test Case Results Classification step, we manually analyzed the test case results in order to obtain passed test cases and test cases failed for fragility issues.



4. Research products:

- **M. De Luca**, A.R. Fasolino, A. Ferraro, V. Moscato, G. Sperlì, P. Tramontana; *"A community detection approach based on Network Representation Learning for repository mining"*, Expert Systems with Applications, published, 2022. https://doi.org/10.1016/j.eswa.2023.120597
- D. Amalfitano, **M. De Luca**, A.R. Fasolino; *"Documenting Software Architecture Design in Compliance with the ISO 26262: an Industrial Case Study in the Automotive Domain"*, International Conference On Software Architecture (ICSA), published, 2023. https://doi.org/10.1109/ICSA-C57050.2023.00022

- **M. De Luca**, A.R. Fasolino, P. Tramontana; *"Investigating the Robustness of Locators in Template-based Web Application Testing using a GUI Change Classification Model",* Journal of System and Software (JSS), accepted, 2023

5. Conferences and seminars attended

- *20th IEEE International Conference On Software Architecture, during 13-17 March 2023, L'Aquila, Italy.* I attended this conference as presenting author for the paper "Documenting Software Architecture Design in Compliance with the ISO 26262: an Industrial Case Study in the Automotive Domain".
- *16th International Summer School on Software Engineering University of Salerno, Italy, during June 12-15, 2023.*

6. Periods abroad and/or in international research institutions

-

7. Tutorship

12 hours of teaching activities regarding practical lectures/seminars during the course "Ingegneria del Software", Bachelor Degree in Computer Engineering

8. Plan for year three

In the next year, I plat to:

- Extend my research activities in the fields of software testing.
- To continue my research activities in collaboration with the "Gran Sasso Science Institute (GSSI) about software metric to asses compliance with safety standards and about the concept of continuous compliance with the safety standards.
- To investigate reverse engineering approach to keep up to date the Software Documentation within the safety standards requirement and recommendation.
- To continue my research activities in Micron about Software Documentation in compliance with ISO 26262.
- Write my thesis about Software Documentation and Software Process in compliance with ISO 26262.