



Fabiano Pecorelli

<https://fabiano-pecorelli.github.io>

fabiano.pecorelli@tuni.fi

+39 345 0594 401

PERSONAL INFORMATION

Name

Fabiano Pecorelli

Date of Birth

September 17, 1991

Place of Birth

Nocera Inferiore (SA), Italy

Address

Vaajakatu 5 H 174 - 33720, Tampere, Finland

JOB POSITIONS

Postdoctoral Researcher

CloudSea.AI research group with Prof. Davide Taibi

Nov 2021 - current

Tampere University

EDUCATION

Degree of European Doctor of Philosophy (Ph.D.) in Computer Science

Advisor: Prof. Andrea De Lucia

2022

University of Salerno

Master's Degree (MSc) in Computer Science

110/110 magna cum laude, Advisor: Filomena Ferrucci

2018

University of Salerno

Bachelor's Degree (BSc) in Computer Science

110/110 magna cum laude, Advisor: Andrea De Lucia

2016

University of Salerno

RESEARCH

My main research area is Software Maintenance and Evolution with a particular interest in Software Code Quality and Predictive Analytics. Following, a more specific focus on those topics:

Machine-Learning-based Code Smell Detection: Bad code smells, i.e., poor design and implementation choices, are usually introduced in software systems because developers poorly conceived the design of a code component. Recent empirical studies have shown that code smells hinder comprehensibility, and possibly increase change- and fault- proneness. Many heuristic detection strategies have been proposed over the last year but, unfortunately, all showing some key common limitations (e.g., thresholds-dependent performance). To overcome these limitations, during my Ph.D. I have analyzed the application of Machine-Learning to Code Smell Detection by several perspectives: (i) comparing its performance with the heuristics ones [1], (ii) studying the impacts caused by data balancing [2, 5], (iii) prioritizing their severity according to developers' perspective [4], and (iv) exploiting a novel set of predictors [13].

Test Code Quality: Testing is an activity performed to check whether the actual software product matches expected requirements and to ensure that the software product is defect-free. Test code quality has been often associated with the defect-proneness of production code in the past. Based on those findings, I've deeply studied the relation of test code quality to production code post-release defects [3, 15]. Besides looking at traditional software systems, I've also analyzed the quality of tests in Android mobile applications [8].

Quantum Software Engineering: Quantum computing is no longer only a scientific interest but is rapidly becoming an industrially available technology that can potentially overcome the limits of classical computation. This shift has led to the definition of a new discipline called quantum software engineering, which is demanded to define novel methods for engineering large-scale quantum applications. While the research community is successfully embracing this call, still there is a lack of systematic investigations into the state of the practice of quantum programming. Recently, I studied the current state of the practice on quantum programming by proposing a two-step investigation that includes a mining software repository analysis and a survey study to identify all possible software-engineering related challenges for quantum programming [23]. I have also started investigating about Quantum Programming for Software Engineering by studying the possible applications of quantum programming to provide better solutions to known Software Engineering problems. In this regard, I've co-organized the first workshop on Quantum Programming for Software Engineering (QP4SE).

Other Research Topics: Besides the two main topics analyzed above, I've also focused my research activity on other Software Engineering related aspects:

- **Defect Prediction:** I've started working on this topic since my Bachelor's Thesis, in which I developed a web application to build and compare Machine Learning models for Defect Prediction. More recently, I've conducted a large empirical investigation to assess the performance of several ML ensemble classifier for Software Defect Prediction [14];
- **Social Aspects:** In the context of Social Aspects, my research focused on (i) finding possible relations between Community Patterns and Community Smells [6], and (ii) defining a refactoring recommendation approach based on the optimization of socio-technical congruence [10];
- **Tools Development:** Some of the research activities conducted, resulted in the definition and the implementation of tools useful for developers and software engineers. In particular, my research lead to the definition of: (i) CASPER [12], an IntelliJ plug-in for the automatic detection and refactoring of code smells, (ii) DARTS [7], an IntelliJ plug-in for the automatic detection and refactoring of test smells, (iii) ADOCTOR [9], an Android Studio plug-in for the automatic detection and refactoring of android smells, and (iv) VITRUM [11], an IntelliJ plug-in for the computation and visualization of test-related factors.

TEACHING

Academic Writing and Publishing - Tampere University <i>Teaching Assistant in the Ph.D. degree course of Prof. Davide Taibi</i>	2022
Programming 3 - Tampere University <i>Teaching Assistant in the Bachelor's degree course of Prof. Terhi Kilamo</i>	2022
Continuous Development and Deployment (DevOps) - Tampere University <i>Teaching Assistant in the Bachelor's degree course of Prof. Kari Systä</i>	2022
Software Dependability - University of Salerno <i>Teaching Assistant in the course of Prof. Fabio Palomba at the MSc in Computer Science</i>	2021
Software Engineering II: Maintenance and Evolution - University of Salerno <i>Teaching Assistant in the course of Prof. Andrea De Lucia at the MSc in Computer Science</i>	2021
Software Engineering - University of Salerno <i>Teaching Assistant in the course of Prof. Andrea De Lucia at the BSc in Computer Science</i>	2021
Software Dependability - University of Salerno <i>Teaching Assistant in the course of Prof. Fabio Palomba at the MSc in Computer Science</i>	2020
Software Engineering II: Maintenance and Evolution - University of Salerno <i>Teaching Assistant in the course of Prof. Andrea De Lucia at the MSc in Computer Science</i>	2020
Software Engineering - University of Salerno	2020

Teaching Assistant in the course of Prof. Andrea De Lucia at the BSc in Computer Science

Programming and Data Structures - University of Salerno	2020
<i>Teaching Assistant in the course of Prof. Maurizio Tucci at the BSc in Computer Science</i>	
Programming Languages - University of Salerno	2020
<i>Teaching Assistant in the course of Prof. Maurizio Tucci at the BSc in Computer Science</i>	
Software Dependability - University of Salerno	2019
<i>Teaching Assistant in the course of Prof. Fabio Palomba at the MSc in Computer Science</i>	
Software Engineering II: Maintenance and Evolution - University of Salerno	2019
<i>Teaching Assistant in the course of Prof. Andrea De Lucia at the MSc in Computer Science</i>	
Software Engineering - University of Salerno	2019
<i>Teaching Assistant in the course of Prof. Andrea De Lucia at the BSc in Computer Science</i>	
Object Oriented Programming - University of Salerno	2019
<i>Teaching Assistant in the course of Prof. Carmine Gravino at the BSc in Computer Science</i>	
Software Engineering II: Maintenance and Evolution - University of Salerno	2018
<i>Teaching Assistant in the course of Prof. Andrea De Lucia at the MSc in Computer Science</i>	
Software Engineering - University of Salerno	2018
<i>Teaching Assistant in the course of Prof. Andrea De Lucia at the BSc in Computer Science</i>	
Object Oriented Programming - University of Salerno	2017
<i>Teaching Assistant in the course of Prof. Vincenzo Deufemia at the BSc in Computer Science</i>	
Programming Languages - University of Salerno	2017
<i>Teaching Assistant in the course of Prof. Rosalba Zizza at the BSc in Computer Science</i>	
Programming Languages - University of Salerno	2017
<i>Teaching Assistant in the course of Prof. Michele Nappi at the BSc in Computer Science</i>	
Programming and Data Structures - University of Salerno	2016
<i>Teaching Assistant in the course of Prof. Andrea De Lucia at the BSc in Computer Science</i>	

THESES ADVISING / CO-ADVISING

Search-based vs deep learning automatic test case generation	2022
<i>M.Sc. Student: Andrea Cupito. Advisor: Prof. Fabio Palomba</i>	<i>University of Salerno</i>
Impacts of community smells on open source projects participation	2022
<i>M.Sc. Student: Antonio Russo. Advisor: Prof. Fabio Palomba</i>	<i>University of Salerno</i>
Automatic generation of code smells summaries in source code	2021
<i>M.Sc. Student: Michele Simone Gambardella. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
An Android Studio plug-in for automatic android smell detection and refactoring	2020
<i>B.Sc. Student: Antonio De Matteo. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
A plug-in for automatic android smell detection in continuous integration	2020
<i>B.Sc. Student: Francesco Gargiulo. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
An Android Studio linter for automatic energy smell detection and refactoring	2020
<i>B.Sc. Student: Girolamo Giordano, Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>

A plug-in for code smell detection and refactoring based on historical information	2020
<i>B.Sc. Student: Luigi Bozzoli. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
A test-smell-aware genetic algorithm for automatic test case generation	2019
<i>B.Sc. Student: Gaetano Mauro. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
A genetic algorithm for automatic test case generation based on structural metrics	2019
<i>B.Sc. Student: Simone Avolicino. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
An IntelliJ plugin for the extraction and display of test-related factors	2019
<i>B.Sc. Student: Gianluca Di Lillo. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
An IntelliJ plug-in for automatic test smell refactoring	2019
<i>B.Sc. Student: Andrea Cupito. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
A plug-in for code smell detection and refactoring based on textual information	2019
<i>B.Sc. Student: Michele Simone Gambardella. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
An IntelliJ plug-in for automatic test smell detection	2019
<i>B.Sc. Student: Stefano Lambiase. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>
A Travis plug-in for continuous test case (Re)prioritization	2018
<i>B.Sc. Student: Michele Santarsiere. Advisor: Prof. Andrea De Lucia</i>	<i>University of Salerno</i>

RESEARCH INTERNSHIPS

Visiting PhD Student at University of Zurich (UZH)	2019
<i>5 Months as Visiting PhD Student. The research conducted in this period, resulted in the publication of [15]</i>	

Visiting Master Student at University College of London (UCL)	2018
<i>5 Months as Erasmus Student working on my Master's Thesis under the supervision of Prof. Federica Sarro</i>	

PROFESSIONAL ACTIVITIES

1. Organizing Committee Participation

Poster Track Chair	2022
<i>23rd International Conference on Product-Focused Software Process Improvement Profs 2022, Jyväskylä, Finland</i>	
Program Co-Chair	2022
<i>1st International Workshop on Quantum Programming for Software Engineering QP4SE 2022, Virtual</i>	
Virtualization Chair	2022
<i>5th International Conference on Technical Debt TechDebt 2022, Virtual</i>	
Program Committee Member of RENE/NIER track	2022
<i>14th Symposium on Search-Based Software Engineering SSBSE 2022, Singapore</i>	
Program Committee Member of NIER track	2022
<i>9th IEEE/ACM International Conference on Mobile Software Engineering and Systems</i>	

MobileSoft 2022, Pittsburgh, Pennsylvania (United States)

Web Chair 2021
*29th IEEE/ACM International Conference on Program Comprehension
ICPC 2021, Virtual*

Program Committee Member of Tool Demo track 2021
*29th IEEE/ACM International Conference on Program Comprehension
ICPC 2021, Virtual*

Web Chair 2021
*28th edition of the IEEE International Conference on Software Analysis, Evolution and Reengineering
SANER 2021, Virtual*

Program Committee Member 2020
*12th International Conference on Advances in Databases, Knowledge, and Data Applications
DBKDA 2020, Venice, Italy*

Program Committee Member 2020
*6th International Conference on Advances and Trends in Software Engineering
SOFTENG 2020, Lisbon, Portugal*

Program Committee Member 2019
*14th International Conference on Software Engineering Advances
ICSEA 2019, Valencia, Spain*

Program Committee Member 2019
*9th International Conference on Mobile Services, Resources, and Users
MOBILITY 2019, Nice, France*

2. Journal Services

Science of Computer Programming 2022
Guest Editor - Special issue on Software Quality Assurance for Artificial Intelligence

Transactions on Software Engineering (TSE) 2022 - current
Reviewer

Transactions on Software Engineering and Methodology (TOSEM) 2021 - current
Reviewer

IEEE Software 2021 - current
Reviewer

Empirical Software Engineering Journal (EMSE) 2020 - current
Reviewer

Journal of Systems and Software (JSS) 2020 - current
Reviewer

Software Quality Journal (SQJ) 2020 - current
Reviewer

Science of Computer Programming 2020 - current
Reviewer

CONFERENCES AND SCHOOL PARTICIPATIONS

3rd International Workshop on Quantum Software Engineering - Q-SE <i>Talk on [23]</i>	2022 <i>Virtual</i>
19th International Conference on Mining Software Repositories - MSR <i>Presentation of [22]</i>	2022 <i>Virtual</i>
9th Int. Conf. on Mobile Software Engineering and Systems - MobileSoft <i>Presentation of [18]</i>	2022 <i>Virtual</i>
29th IEEE Int. Conf. on Software Analysis, Evolution and Reengineering - SANER <i>Presentation of [17]</i>	2022 <i>Virtual</i>
42st ACM/IEEE International Conference on Software Engineering - ICSE <i>Attendee</i>	2021 <i>Virtual</i>
29th IEEE/ACM International Conference on Program Comprehension - ICPC <i>Presentation of [14]</i>	2021 <i>Virtual</i>
4th International Workshop on Software Health - SoHeal <i>Presentation of [15]</i>	2021 <i>Virtual</i>
4th Int. Workshop on Machine Learning Techniques for Software Quality Evolution <i>Presentation of [13]</i>	2020 <i>Virtual</i>
International Conference on Advanced Visual Interfaces 2020 - AVI <i>Presentation of [11], [12]</i>	2020 <i>Ischia Island, Italy</i>
42st ACM/IEEE International Conference on Software Engineering - ICSE <i>Attendee</i>	2020 <i>Virtual</i>
28th IEEE/ACM International Conference on Program Comprehension - ICPC <i>Presentation of [7], [8]</i>	2020 <i>Virtual</i>
3rd International Workshop on Software Health - SoHeal <i>Presentation of [6]</i>	2020 <i>Virtual</i>
17th International Conference on Mining Software Repositories - MSR <i>Presentation of [4]</i>	2020 <i>Virtual</i>
2nd International Summer School on Software Engineering - SIESTA <i>Attendee</i>	2019 <i>Termoli, Italy</i>
41st ACM/IEEE International Conference on Software Engineering - ICSE <i>Attendee</i>	2019 <i>Montreal, Quebec, Canada</i>
16th International Conference on Mining Software Repositories - MSR <i>Attendee</i>	2019 <i>Montreal, Quebec, Canada</i>
27th IEEE/ACM International Conference on Program Comprehension - ICPC <i>Presentation of [1]</i>	2019 <i>Montreal, Quebec, Canada</i>
14th International Summer School on Software Engineering - ISSSE <i>Scientific Secretariat</i>	2019 <i>Salerno, Italy</i>
13th International Summer School on Software Engineering - ISSSE <i>Attendee</i>	2017 <i>Salerno, Italy</i>

Publications

- [1] Pecorelli, F., Palomba, F., Di Nucci, D., and De Lucia, A. (2019, May). Comparing heuristic and machine learning approaches for metric-based code smell detection. In 2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC) (pp. 93-104). IEEE.
- [2] Pecorelli, F., Di Nucci, D., De Roover, C., and De Lucia, A. (2019, August). On the role of data balancing for machine learning-based code smell detection. In Proceedings of the 3rd ACM SIGSOFT international workshop on machine learning techniques for software quality evaluation (pp. 19-24). ISO 690
- [3] Pecorelli, F. (2019, August). Test-related factors and post-release defects: an empirical study. In Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (pp. 1235-1237).
- [4] Pecorelli, F., Palomba, F., Khomh, F., and De Lucia, A. (2020, May). Developer-Driven Code Smell Prioritization. In International Conference on Mining Software Repositories.
- [5] Pecorelli, F., Di Nucci, D., De Roover, C., and De Lucia, A. (2020, November). A large empirical assessment of the role of data balancing in machine-learning-based code smell detection. Journal of Systems and Software, 110693.
- [6] De Stefano, M., Pecorelli, F., Tamburri, D. A., Palomba, F., and De Lucia, A. (2020, June). Splicing Community Patterns and Smells: A Preliminary Study. In Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops (pp. 703-710).
- [7] Lambiase, S., Cupito, A., Pecorelli, F., De Lucia, A., and Palomba, F. (2020, July). Just-In-Time Test Smell Detection and Refactoring: The DARTS Project. In Proceedings of the 28th International Conference on Program Comprehension (pp. 441-445).
- [8] Pecorelli, F., Catolino, G., Ferrucci, F., De Lucia, A., and Palomba, F. (2020, July). Testing of Mobile Applications in the Wild: A Large-Scale Empirical Study on Android Apps. In Proceedings of the 28th International Conference on Program Comprehension (pp. 296-307).
- [9] Iannone, E., Pecorelli, F., Di Nucci, D., Palomba, F., and De Lucia, A. (2020, July). Refactoring Android-specific Energy Smells: A Plugin for Android Studio. In Proceedings of the 28th International Conference on Program Comprehension (pp. 451-455).
- [10] De Stefano, M., Pecorelli, F., Tamburri, D. A., Palomba, F., and De Lucia, A. (2020, September). Refactoring Recommendations Based on the Optimization of Socio-Technical Congruence. In 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 794-796). IEEE.
- [11] Pecorelli, F., Di Lillo, G., Palomba, F., and De Lucia, A. (2020, September). VITRuM: A Plug-In for the Visualization of Test-Related Metrics. In Proceedings of the International Conference on Advanced Visual Interfaces (pp. 1-3).
- [12] De Stefano, M., Gambardella, M. S., Pecorelli, F., Palomba, F., and De Lucia, A. (2020, September). cASPER: A Plug-in for Automated Code Smell Detection and Refactoring. In Proceedings of the International Conference on Advanced Visual Interfaces (pp. 1-3).
- [13] Lujan, S., Pecorelli, F., Palomba, F., De Lucia, A., and Lenarduzzi, V. (2020, November). A preliminary study on the adequacy of static analysis warnings with respect to code smell prediction. In Proceedings of the 4th ACM SIGSOFT International Workshop on Machine-Learning Techniques for Software-Quality Evaluation (pp. 1-6).
- [14] Pecorelli, F., Di Nucci, D. (2021, January). Adaptive selection of classifiers for bug prediction: A large-scale empirical analysis of its performances and a benchmark study. Science of Computer Programming, 102611.
- [15] Pecorelli, F., Palomba, F., De Lucia, A. (2021, February). The Relation of Test-Related Factors to Software Quality: A Case Study on Apache Systems. Empirical Software Engineering.

- [16] De Stefano, M., Pecorelli, F., Palomba, F., De Lucia, A. (2021, August). Comparing within-and cross-project machine learning algorithms for code smell detection. In Proceedings of the 5th International Workshop on Machine Learning Techniques for Software Quality Evolution (pp. 1-6).
- [17] Nguyen, H., Lomio, F., Pecorelli, F., Lenarduzzi, V. (2022). PANDORA: Continuous mining software repository and dataset generation. In Proceedings of the 29th IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER2022).
- [18] Iannone, E., De Stefano, M., Pecorelli, F., De Lucia, A. (2022). Predicting The Energy Consumption Level of Java Classes in Android Apps: An Exploratory Analysis. In Proceedings of the 9th IEEE/ACM International Conference on Mobile Software Engineering and Systems (MobileSoft 2022).
- [19] De Stefano, M., Iannone, E., Pecorelli, F., Tamburri, D. A. (2022). Impacts of software community patterns on process and product: An empirical study. *Science of Computer Programming*, 214, 102731.
- [20] Pecorelli, F., Catolino, G., Ferrucci, F., De Lucia, A., Palomba, F. (2022). Software testing and Android applications: a large-scale empirical study. *Empirical Software Engineering*, 27(2), 1-41.
- [21] Li, X., Moreschini, S., Pecorelli, F., Taibi, D. (2022). OSSARA: Abandonment Risk Assessment for Embedded Open Source Components. *IEEE Software*.
- [22] Pecorelli, F., Grano, G., Palomba, F., Gall, H. C., De Lucia, A. (2022). Toward Granular Automatic Unit Test Case Generation. *arXiv preprint arXiv:2204.05561*.
- [23] De Stefano, M., Pecorelli, F., Di Nucci, D., Palomba, F., De Lucia, A. (2022). Software engineering for quantum programming: How far are we?. *Journal of Systems and Software*, 111326.
- [24] Pecorelli, F., Lujan, S., Lenarduzzi, V., Palomba, F., De Lucia, A. (2022). On the adequacy of static analysis warnings with respect to code smell prediction. *Empirical Software Engineering*, 27(3), 1-44.