StArt - State of the Art through Systematic Review

SYSTEMATIC REVIEW - USING OPEN SOURCE PROJECTS IN SOFTWARE ENGINEERING EDUCATION SYSTEMATIC

Date: 05/08/2018 - 04:39:21

Title:	Using Open Source Projects in Software Engineering Education Systematic Mapping Study
Researchers:	Moara Sousa Brito; Fernanda Gomes Silva; Christina von Flach Garcia Chavez; Debora Maria Coelho Nascimento; Roberto Bittencourt;
Description:	One possibility to bring practice to Software Engineering Course is to make students participate in Open Source Projects with supervision of teachers. Many researches have been approaching this idea. The aim of this systematic mapping study is to summarize all existing information about the use of Open Source Projects in Software Engineering or Computer Science Courses, and how student learning has been evaluated in order to identify possible gaps that justifies further research activities.
Objectives:	The objective is to identify how Open Source Projects have been used in Software Engineering Education, like a possiblility to students participate in a real software project.
Main Question:	How have Open Source Projects been used in Software Engineering Education?
Intervention:	the use of Open Source Projects to improve student learning about Software Engineering Concepts.
Control:	(not applicable in this context)
Population:	Software Engineering Undergraduate Courses
Results:	How Open Source Projects have been used in Software Engineering Courses, methodologies, proposals, opportunities, strategies, results obtained, which specific Software Engineering areas, they have been used.
Application:	Universities
Keywords:	Apprentice; Coaching; Competencies; Computer Science; Computing; Course; Curriculum; Design Patterns; Education; Educational; FLOSS; FOSS; Free Software; Learning; Libre Software; Mentoring; OSP; OSS; Open Source; Skills; Software Analysis; Software Architecture; Software Configuration Management; Software Design; Software Development; Software Engineering; Software Evolution; Software Maintenance; Software Management; Software Metrics; Software Modeling; Software Process; Software Quality; Software Requirement; Software Testing; Software Validation; Software Verification; Teaching; Training; Tutoring;
Source Selection Criteria:	Important sources for Software Engineering and Computer Science studies;
Studies Languages:	English;
Source Search Methods:	Search through Web search engines and manual search;
Source Engine:	IEEE; Science Direct; Scopus; ACM; Engineering Village; Springer;

Studies inclusion and exclusion criterias:	(I) The study explore the use of Open Source Projects to learn/teach Software Engineering content; (I) The study is presented like full paper; (I) The study is presented like short paper; (I) The study is presented like work in progress; (E) Document not written in English; (E) Document which the full text is not available; (E) The study's content isn't associated with learning/teaching Software Engineering; (E) The study's focus is only to use Open Source Software as a tool in the infrastructure of the laboratories; (E) The study's focus is only to use Open Source Software as a tool to develop applications; (E) The document dealing with topics irrelevant to the purpose of this map; (E) The study only available as abstract; (E) The study only available as Powerpoint presentation; (E) It is a description of a panel in a conference; (E) The document is a proceedings; (E) The Open Source Software is a tool used to help learning; (E) The document is a Table of Contents; (E) The study's focus is to teach about or how to use a specific Open Source Software; (E) Open Source Projects are not used; (E) There isn't focus in using an Open Source Project in learning process, the Open Source Software is only a tool; (E) Open Source Projects are used as data source for search; (E) The document is a letter from the editor; (E) It is a description of a keynote or workshop in a conference; (E) The document describes a tutorial;
Studies types definition:	;
Initial studies selection:	;
Studies quality evaluation:	;
Information Extraction Fields:	Areas in Software Engineering addressed={Software Engineering (general),Software Requirements,Software Modeling/Analysis,Software Design and Architecture,Software Quality: Verification and Validation ,Software Testing,Software Evolution and Maintenance,Software Process,Software Management,Software Configuration Management,Software Development/Construction}; goal of the approach proposed in the paper={To learn about SE concepts using OSS projects,To learn about OSS Projects in a Software Engineering Course}; Active Learning approach applied={Not explicitly specified,Problem-Based Learning,Project-based learning,Project method,Case-based learning,Studio-base learning,Peer learning,Game-based learning,Other}; perspective of learning assessment={Not explicitly specified,student's perspective,teacher's perspective,product assessment: evaluation criteria applied to student's production}; type of assessment applied to verify students' learning={ none,Exams,report,portfolio,software artifacts,passing tests,seminar}; How Open source Projects are used in curriculum?={No explicitly specified,extra activity,optional discipline,compulsory discipline}; control level about the project={full control,inside control ,no control }; What role user does a student assumes?={passive user,active user,co-developers,core developers}; Research Type={Experience report,Case study,Experiment,Survey/questionário,Etnografia,Philosophical,opinion paper,Proposal of solution};
Results Summarization:	;