

# Developing a Taxonomy for Software Engineering Education Through an Empirical Approach

**Sebastián Pizard**

Universidad de la República, Facultad de Ingeniería,  
Montevideo, Uruguay, 11300,  
*spizard@fing.edu.uy*

and

**Diego Vallespir**

Universidad de la República, Facultad de Ingeniería,  
Montevideo, Uruguay, 11300,  
*dvallesp@fing.edu.uy*

## Abstract

**Background:** With just over 50 years since birth, software engineering gathers more and more topics. This diversity, which shows how broad and prolific the area is, also greatly fragments knowledge. Efforts to develop classifications and taxonomies can collaborate in ordering this knowledge. **Objective:** This work aims to contribute to organizing software engineering education knowledge, a sub-area in which formalization is still necessary. **Method:** We propose a process for the construction of controlled vocabularies. We instantiated this process twice; first, using automatic clustering techniques to analyze over 1,000 articles; and then, we focused on concepts related to teaching techniques and methods. **Results:** We present a taxonomy with 60 terms with covers concepts to be taught, methods to use, and where to do it. The ‘teaching approaches and methods’ category covers 26 terms with their definitions and most relevant references. **Implications:** The taxonomy can be used by teachers and researchers to understand the breadth of the field, to place their research initiatives in a broader context and to conduct more rigorous searches in the literature. We believe it is necessary to continue working on the taxonomy’s expansion and also to carry out validation activities, if possible, including experts’ validation.

**Keywords:** taxonomy for software engineering education, software engineering education, controlled vocabulary

## 1 Introduction

Software engineering is a discipline within engineering that develops all the aspects related to the production of software, from the conception of the software to its operation [1]. As such, software engineering education includes a great variety of topics such as communication skills, problem solving, design methodology, negotiation, human-computer interaction, leadership, and ethics among others [2, 3, 4].

Software engineering education research is published in different conferences and journals. Some of them are more specific and others are more general.

One important way of retrieving information is using keywords, which are assigned by the authors and editors before each article is published. Frequently, the quality of the information retrieved by researchers depends on the quality of the keywords used to label the articles. In areas that are not very mature or not thoroughly studied, such as software engineering education, there are no consensual keyword lists. This makes both labelling the articles and retrieving information difficult.

Furthermore, standardizing terminology in a thematic area provides benefits for several stakeholders [5]: it can serve as a platform allowing researchers and members of the community to place their research initiatives in a broader context; it can be a source of terms to understand the breadth of the field and conduct searches in the literature; it can be a mechanism for editors of scientific journals to organize related authors and research areas, identify the suitability of the submitted articles for the aim of the journal and categorize the database of reviewers for a more appropriate allocation; and it can provide a method for financing agencies to better classify the research areas in order to ensure a better distribution of the funds.

Another relevant benefit regards the improvement in the accumulation of evidence, for example, through systematic literature reviews and mapping studies. Kitchenham et al. summarize the current difficulty of not having a standard terminology to use in the systematic literature reviews for the entire area of software engineering:

“Software engineering lacks strong taxonomies. The terms that we use are often imprecise, and software engineers are rather prone to create new terms to describe ideas that may well be closely related to existing ones. This can complicate searching since we need to consider all possible forms of terminology that might have been used in the titles and abstracts of papers” [6]

The purpose of our work is to create a taxonomy of software engineering education. As a consequence, and due to the lack of a process with a suitable level of formality, we define a disciplined process to guide the creation or update of a controlled vocabulary. This process uses the material to classify as a basis for the gathering of the concepts to be considered.

In a previous paper [7], we proposed an initial software-engineering-education controlled vocabulary; here, we present the results of the next stage of our work: an extended taxonomy of software engineering education and an update of related work that provides a more general context of the work. Although it is still necessary to validate the taxonomy, its terms and definitions can be useful for teachers, researchers and editors.

This article is organized as follows: section 2, includes concepts about controlled vocabularies and taxonomies; section 3, reports related work; section 4, summarizes the work approach; section 5, briefly presents the first stage in which automatic clustering techniques were used to develop a first taxonomy of education in software engineering; then, section 6, shows the work done to expand the terms and definitions related to the teaching approaches and methods used in software engineering; and, finally, the conclusions and future work are presented in section 7.

## 2 Controlled Vocabularies and Taxonomies

The purpose of controlled vocabularies is to provide a way of organizing information. As noted in the Z39.19-2005 guide provided by ANSI/NISO [8], “through the process of assigning terms selected from controlled vocabularies to describe documents and other types of content objects, the materials are organized according to the various elements that have been chosen to describe them”. A content object is any element that has to be described in order to be included in an information retrieval system, website, or another information source. Typical content objects are articles from scientific journals, technical reports and other types of documents.

In practice, a controlled vocabulary consists of a list of terms that have been listed explicitly to represent the concepts that have been chosen to describe content objects. It is controlled because only the terms in the list should be used to refer to the concepts of the thematic area covered by the vocabulary. It is also controlled since the modification of the list is subject to previously defined policies.

The aim of a controlled vocabulary is to ensure consistency in the application of the language [9]. An example of controlled vocabulary is the *ISO/IEC/IEEE International Standard - Systems and software engineering - Vocabulary* [10], which includes terms used in systems and software engineering.

A taxonomy is a controlled vocabulary that is made up of preferred terms connected in a hierarchy or a poli-hierarchy. It is very important to identify the hierarchy relationships between the terms and state them explicitly.

## 2.1 Construction

Three different approaches are suggested for the construction of a controlled vocabulary [8]:

- The Committee Approach - This approach implies that experts in the area draw a list with the key terms indicating their relationships. Lists of terms may be submitted by various experts or taken from various sources. This approach has the major advantage that the construction of the vocabulary is done by the experts in the area, who will generally be the future users.
- The Empirical Approach - This approach basically contemplates two methods:
  - Deductive Method - The terms are extracted from a reference set of content objects, but there is no attempt to control or determine possible relationships until a sufficient number of terms is collected. The terms are then revised, the possible hierarchies are identified and the remaining terms are assigned to each one of them. From that basis, the vocabulary is controlled.
  - Inductive Method - New terms are selected for their possible inclusion in the vocabulary as they are found in a reference set of content objects. The control of the vocabulary is applied from the beginning as a continuous operation. If the controlled vocabulary has some kind of hierarchy, each term is assigned to the corresponding classes as soon as its inclusion is decided.
- Combination of Methods - In practice, more than one approach can be employed in one phase or another in the construction of a controlled vocabulary.

## 2.2 Facet Analysis

Facet analysis is a type of classification system in which you can assign multiple terms to classify a content object; which are alternate aspects. “For example, a book would have a subject heading in a hierarchy but also an author, date, price and so on, allowing search across categories” [11]. The use of facet analysis is similar to the construction of a controlled vocabulary in various subsets, in which every subset is in itself a controlled vocabulary that represents a different aspect of information [9].

Facet analysis is particularly useful for [8, 12]:

- New areas of knowledge in which knowledge of the domain is incomplete or the relationships between concepts are unknown.
- Interdisciplinary areas in which there is more than one perspective from which to consider an object of content.
- Vocabularies for which multiple hierarchies are needed but may be inadequate because of the difficulty of clearly defined boundaries.

## 3 Related work

Cassel et al. carried out *The Computing Ontology* project [13, 14], updated in 2010 [15], whose purpose was the identification of overlapping areas in the computer domain to make recommendations on the curricula, the creation of new courses, and -according to the authors- also indexing digital libraries and support searches. The result of this project was a collection of various Computer Science disciplines, topics and subtopics. It covered a hierarchy of classes and objects modeled in Protégé<sup>1</sup> [16] and, although it is available online [17], it does not seem to be updated. In particular, the authors indicate that they used different curriculum guidelines as a basis [13], for example, those of ACM/IEEE for Computer Science, Computer Engineering and Software Engineering published before the project [18]. Currently, these guides have more updated and specific versions. On the ontology’s content, it is difficult to find the concepts about software engineering. Many of them are in categories of ‘Software Design’ and ‘Implementation’ mixed with other concepts (for

---

<sup>1</sup><http://protege.stanford.edu/>

example, types of algorithms), while concepts related to project management are in the ‘Organization and Management’ category. It does not contemplate software engineering education or computer science education concepts. These characteristics, the limited information on the taxonomy construction method, and the fact that it is outdated, motivate us to address an independent work on software engineering education.

Our work began in 2015, since then three recent initiatives that update the context of this work have been published. These are: a survey of academics and literature to determine the nomenclature issues involved with computing education [19], the development of a taxonomy of keywords for engineering education research [5], and a systematic mapping study on taxonomies made in the field of software engineering together with a revised taxonomy development method [20].

Firstly, Simon et al. present a study of international issues involving computing education terminology worldwide [19]. In this work, carried out by a working group of the 2015’s edition of the Annual Conference on Innovation and Technology in Computer Science Education, there is a wide discussion related to the accreditation’s names in different countries and the current difficulties in having a common terminology. They believe that it is unlikely that any project could unify such diverse terminology on computing education, but that there could be guidelines on the different uses of terminology to know and tolerate them. On the topics discussed, they reached a list of 18 topics included in the education programs, software engineering being one of those topics. Although there are no topics that can be used for our work, we believe it is important to recommend different terms for the same concept according to regions or countries.

Secondly, Finelli et al. present a project for the development of a taxonomy of keywords for engineering education research (called EER Taxonomy) which is funded by the National Science Foundation [5]. The focus of the taxonomy is on the research conducted in the United States, although the authors achieved an inclusive work involving 266 individuals from 30 countries. The work comprises three major items: the creation of the taxonomy, its validation, and the creation of user guides. As context, the authors include a list of initiatives that seek to organize knowledge in the area of engineering education research [21, 22, 23, 24, 25, 26]. For the creation of the taxonomy, the authors included various collaborators, among which experts from Access Innovations stand out, an information management company that regularly creates and maintains taxonomies. With their help, they developed an initial taxonomy from three initial schemes obtained by analyzing different sets of published articles. This initial taxonomy was refined in a series of five subsequent workshops with the help of researchers from the area. Regarding the results obtained, the EER taxonomy has 455 terms organized in 14 branches of six levels. These branches include, for example, assessment, educational level, professional practice and research approaches. We believe, as we will discuss later in section 6.6, that the taxonomy that we present here complements the EER and can be considered as a specialization in the area of software engineering education.

Thirdly, the objective of the work by Usman et al. is to survey the state of the art in the research of taxonomies in software engineering (SE) through a literature mapping study [20]. As a result, they obtained 271 taxonomies that they classified according to 3 different facets:

- Research type - Classification by type of study (i.e., evaluation research, validation research, solution proposal, philosophical paper).
- SE knowledge area - Classification by software engineering knowledge area of each taxonomy (i.e. based on the SWEBOK [2], software requirements, software design, software construction, software testing, software maintenance, software configuration management, software engineering management, software engineering process, software engineering models and methods, software quality, software engineering professional practice, and software engineering economics).
- Presentation approach - Classification by the presentation approach of each taxonomy (i.e., textual and graphical).

They found that although SE taxonomies were being published since 1987, their number had greatly increased since 2000. The results indicate the following proportion in thematic areas: requirements - 15.50%, design - 19.55%, construction - 19.55%, testing - 9.96% and maintenance - 11.81%. Some recent areas such as economics or professional practice have a 1.11% each. Although it is not a SWEBOK topic, the authors do not report taxonomies in software engineering education either. On another note, only 16.24% of the taxonomies have a description of their classification process. The authors then study the work by Bayona-Oré et al. in detail [27], which describes a systematic approach to develop a taxonomy, to propose a revised method to create taxonomies. The paper presents a systematic approach to develop a taxonomy with activities organized in four phases, as presented in Table 1.

The process used in our work and presented in the following section was proposed prior to the work of Usman et al. and Bayona-Oré et al. Although the works of these authors are not based on the Z39.19-2005 guide, they share many similarities with our process, which will be presented in section 6.6.

Table 1: List of activities, grouped in four phases, of the systematic process to develop a taxonomy proposed by Usman et al. [20]

Planning		
1	Define SE knowledge area	
2	Describe the objectives of the taxonomy	
3	Describe the subject matter to be classified	
4	Select classification structure type	
5	Select classification procedure type	
Identification and extraction		
6	Identify the sources of information	
7	Extract all terms	
Design and construction		
8	Perform terminology control	
9	Identify and describe taxonomy dimensions	
10	Identify and describe categories of each dimension	
11	Identify and describe relationships	
12	Define the guidelines for using and updating the taxonomy	
Testing and validation		
13	Validate the taxonomy	

We have not found initiatives proposing controlled vocabularies related to software engineering education. However, we believe that the community will be interested in its use. In this regard, many researchers have used Bloom's taxonomy, which covers skill levels in cognitive, affective and psycho-motor domains, to classify the learning outcomes of their courses. In fact, between 2000 and 2014, 26 studies were conducted on the application of Bloom's taxonomy in areas of software engineering education [28].

## 4 Work Method

Figure 1 shows the process used in this work for the creation of the taxonomy; we defined our process in accordance with guide Z39.19-2005 [8] and the recommendations by Hedden [9]. The process has four well-defined phases.

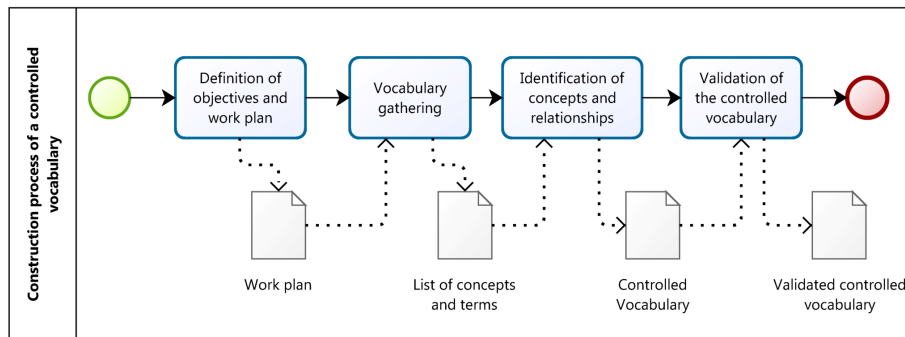


Figure 1: Construction process of a controlled vocabulary through an empirical approach

In the first phase, the objectives of the controlled vocabulary to be constructed must be defined as well as a plan for its development. The plan must include characteristics of the vocabulary (intended use, profile of the users, and scope or content to consider) and limitations or restrictions of the construction process. It must also define two very important aspects: the structure the controlled vocabulary would have, and the approach for its construction.

In the vocabulary gathering phase, we collect all the vocabulary and possible knowledge on a thematic area to be covered by the controlled vocabulary. In order to do this, we carry out one or more research activities, like interviews or analysis of the material to be classified. As a result of this phase, we obtain a list of possible concepts to be included in the controlled vocabulary. Possible synonyms of these potential concepts can also be registered, which can be associated, for example, with regions or countries, in order to follow the recommendations of Simon et al. [19].

In the third phase, we review the list of possible concepts obtained in the previous phase, and we incorporate the ones suitable for the controlled vocabulary.

Finally, in the validation phase of the controlled vocabulary, we try to determine its suitability for use. Usman et al. [20] point out that validation reinforces the usefulness and reliability of taxonomies and indicates three ways to validate them extracted from other previous work [29, 30]. These are:

- Demonstration of orthogonality of the dimensions and categories of the taxonomy.
- Benchmarking: comparison with similar classification schemes.
- Demonstration of utility. This can be done or exemplified by classifying the existing literature or using expert opinion, or with more rigorous validation approaches, such as a case study or an experiment.

This process was developed and instantiated to build a first taxonomy of software engineering education. In a first stage (explained in detail in section 5), over 1,000 articles published during the 1988-2014 period were analyzed using automatic clustering techniques. From this analysis, we drafted a first taxonomy that we later expanded in a second stage (explained in detail in section 6) to include terms and definitions about techniques and methods to teach software engineering.

## 5 First Stage: Automatic clustering techniques and published papers

We reported the first stage of the construction of our software engineering education taxonomy in detail in an article published in the *European Journal for Engineering Education*. [7]. Here we include a brief summary of it. Each sub-section corresponds to a phase of the process presented in section 4 and describes the details of its execution in this case.

### 5.1 Definition of objectives and work plan

The aim of the controlled vocabulary is the classification of the research related to software engineering education. We chose a taxonomy structure, since the bibliography indicates it is the most used structure with most values for contexts similar to the one proposed [5, 31].

We expect that the taxonomy will be of interest for the software engineering education community. This includes researchers, students, educators, and editors of journals that classify research. This community is scattered geographically and the language it uses the most for the diffusion of research is English.

Since this is an unexplored area in terms of taxonomies and controlled vocabularies, we decided to prioritize the literary warrant, so that the vocabulary will be appropriate to use. The literary warrant implies that the included concepts appear in published research, and that the terms used to represent these concepts are the most used in the literature. According to Hulme, who proposed this approach:

“Literary warrant meaning that the basis for classification is to be found in the actual published literature rather than abstract philosophical ideas or concepts in the universe of knowledge or the order of nature and system of the sciences” [32].

Therefore, the exploration and analysis of research material already published is proposed as a research activity.

### 5.2 Vocabulary gathering

Vocabulary gathering was conducted through the content auditing technique, i.e., the search of significant concepts within the content of the material to be classified.

Because the terms of the taxonomy should be good at classifying content that has already been published, we decided to take into account the material published in the Conference on Software Engineering Education and Training (CSEET) between 1988-2014 and in the Software Engineering Education and Training Track (SEET) of the International Conference on Software Engineering (ICSE) between 2000-2014. We selected these conferences because they are, to the best of our knowledge, the most recognized in the area, and also because there are no specialized scientific journals on software engineering education. We included all the available issues published by CSEET and SEET up to 2014 in our research.

We used the technique of automatic cluster analysis (or clustering) for the extraction of terms taking the article abstracts as a basis. Although the use of abstracts can be questionable, it prioritizes the literary warrant and allows us to obtain results without having to handle the complete papers and the difficulties that this would bring (that is, excessive number of pages and terms, and difficulty establishing the focus of each paper automatically). Shaw made an analysis of the papers sent to ICSE 2002 and came to the following conclusion [33]: “Whether you like it or not, people judge papers by their abstracts and read the abstract in order to decide whether to read the whole paper”. Although this does not indicate that the terms used in the abstracts correctly classify papers, it supports the claim that people use the abstract (and the terms they contain) to classify them.

Cluster analysis methods attempt to separate documents into groups, where each group represents a topic that is different from the topics represented by the other groups [34, 35]. Automatic clustering is a non-supervised learning method (in this case, it does not require human intervention) in which a set of clusters is obtained from a set of documents or texts (called “corpus”). In general, automatic clustering techniques are based on syntactic patterns recognition using the frequency of similar terms within the texts to discover similarities. Although these techniques allow enormous savings in effort, they have certain limitations that must be considered. On the one hand, the clustering tools only considers syntactic properties of the language used in the articles. In this way, we obtain a set of groups of documents that may or may not have semantic consistency. An analysis of the material made by experts would take into account semantic properties, surely obtaining more relevant concepts and prioritizing the thematic focus of the taxonomy. On the other hand, the results of the automatic clustering analysis are biased by the algorithm used by the tool. Despite these limitations, we used automatic clustering because this technique is very interesting to explore areas under investigation from the knowledge organization and controlled vocabularies point of view, and allowed us to work with a broader set of papers in a reasonable time. We selected Carrot2 as a clustering tool.

Carrot2 is a search clustering engine that allows automatically organization of small sets of documents into thematic categories [36]. It is an open-source tool, with a data-driven process, and that includes three clustering algorithms (Lingo, STC, k-means) [37]. The first of them, Lingo, was ranked third in general, and is the first open source, by Carpineto et al. in a label-driven subtopic comparison of clustering algorithms [38]. Search clustering algorithms seem like a very suitable work strategy since we have a set of short documents: abstracts have an average of 200 words and are also used as a classification corpus in search engines such as SCOPUS, among others. Within Carrot2, we chose the Lingo algorithm, given its good performance for topic clustering [38], and we also tested it on a subset of our corpus with promising results. We used the default configuration of Carrot2 for the Lingo algorithm.

The corpus consisted of 1023 pairs of titles and abstracts, 890 from CSEET and 133 from ICSE. The elements of the corpus are heterogeneous in their creation dates. Since the clustering algorithm does not use synonyms, it does not accumulate in the same cluster if the elements use different terms for the same concept. This may become an important bias since there are certain trends in the use of the language which change with time. To minimize this, the articles were processed in subsets taking the closeness of the publication date as a grouping criterion. The election of the subsets was arbitrary although there was an attempt to make sets of similar sizes and number of years. This additional step had the purpose of collecting vocabulary in different periods of history and enables, in some way, to extend the literary warrant through time. The following seven subsets were created: 2012-2014 (158 articles), 2009-2011 (170), 2006-2008 (157), 2003-2005 (139), 2000-2002 (125), 1995-1999 (122) and 1988-1994 (152).

Once we obtained the list of clusters, we manually studied the composition of each cluster found. This involves the reading of the titles and abstracts of the articles of each cluster to determine if it is a concept relevant to the taxonomy or not (some clusters obtained by the tool could correspond to terms like ‘paper’ or ‘conference’ that are not relevant concepts). If the concept is relevant, the term is added to the list of terms to be included in the controlled vocabulary.

### 5.3 Identification of concepts and relationships

In this phase, we reviewed the list of terms obtained before, that is, the terms associated with the clusters not discarded in the previous stage, to identify concepts and establish preferred terms for each of them. For this, we reviewed the content of the clusters again to determine the possible concept to which they referred and their synonyms. We also used the principles of content analysis [39] to detect similar clusters and group them when they referred to the same concept. This involved reviewing the content of both clusters and highlighting the segments that identified each one, then comparing the selected segments and verifying that both clusters dealt with the same concept. For example, we found that the cluster ‘software process improvement’ (2005-2003) dealt with the homonymous concept as well as the clusters ‘process improvement’ (1999-1995) and ‘improve’ (1994-1988).

In addition, we obtained several concepts that seemed to belong to very different topics (e.g. ‘software design’, ‘master degree in software engineering’, ‘problem-based learning’, ‘communication and collaboration skills’). This was disconcerting at first and it was necessary to read several classification works in order to find a way to group them at a high level that was not against the literary warrant. In particular, we carried out a first classification in the following facets that have been adapted from the work of Nie et al. [40]: ‘what to teach’, ‘how to teach’ and ‘where to teach’. This classification allows a more detailed subsequent study of each group of concepts without affecting the results obtained before.

#### 5.4 Validation of the controlled vocabulary

In this first stage, we decided to use benchmarking validation and to compare the taxonomy with similar classification schemes [20]. As it was not possible to find taxonomies from the field, we made comparisons with the body of knowledge of software engineering - SWEBOK [2]. To do this we also used content analysis. The relevant results from these comparisons are shown in the next section, along with the results of the stage.

#### 5.5 Results

The result (see Figure 2) is a controlled vocabulary in the form of a taxonomy with 43 terms that correspond to the concepts that were identified as the most used in the research publications from the period covered. The concept classification is presented in three main facets: ‘what to teach’ (24 terms), ‘how to teach’ (13 terms) and ‘where to teach’ (6 terms).

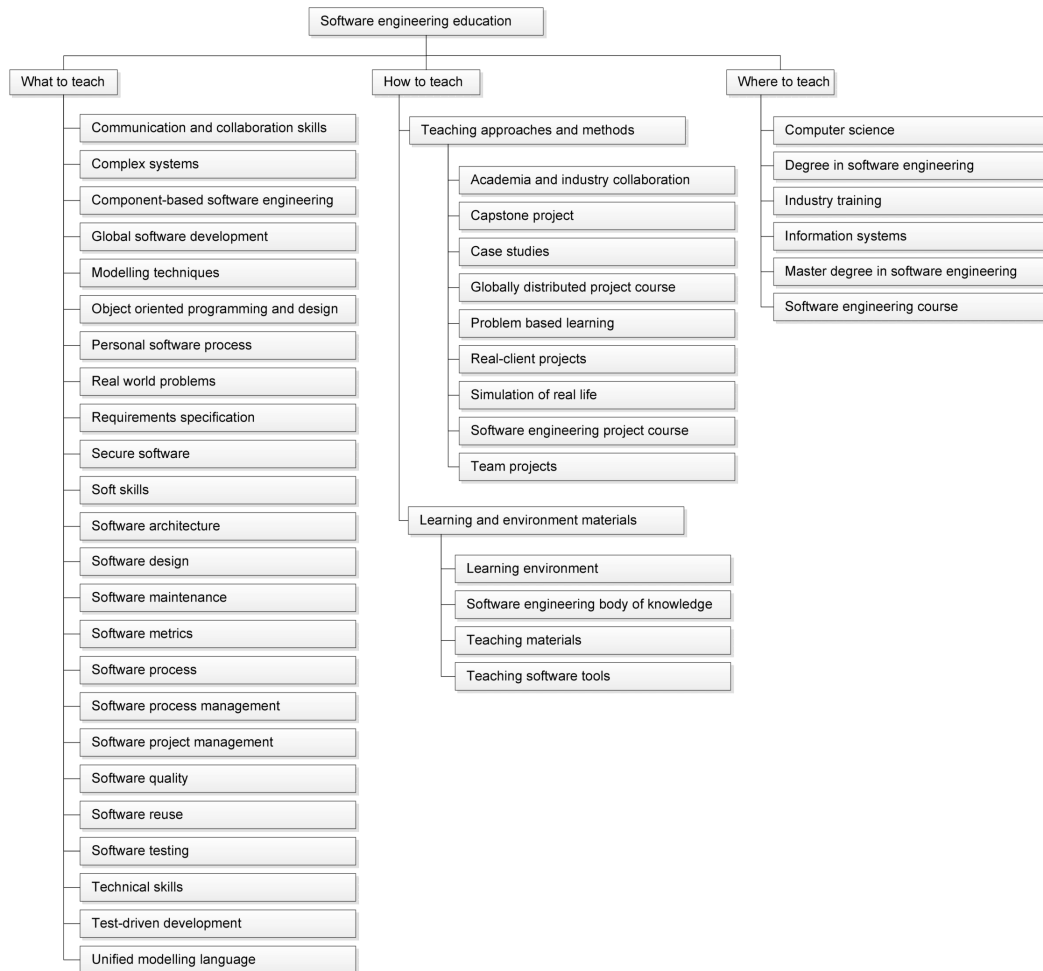


Figure 2: Software engineering education initial taxonomy as a result of the First Stage of our work.

The ‘what to teach’ facet includes terms that correspond to concepts related to topics, skills, or knowledge taught to the students. The identified concepts comprise sub-disciplines of software engineering (‘software design’, ‘software maintenance’, etc.) development approaches (‘object-oriented programming and design’, ‘test driven development’, ‘personal software process’) and skills (‘communication and collaboration skills’, ‘soft skills’, ‘technical skills’). We carried out an exhaustive comparison and concluded that there is nothing that indicates that the SWEBOK [2] cannot be used as reference to expand this facet.

Within the ‘how to teach’ facet, we grouped terms that correspond to concepts related to approaches, techniques or teaching methods, i.e. all the concepts that explain how to teach software engineering. Based on the work of Nascimento et al. [41], we determined that two subcategories proposed in that work could be incorporated into the taxonomy with very good results in the hierarchy of concepts, these subcategories were: ‘teaching approaches and methods’ and ‘learning environment and materials’.

The ‘where to teach’ facet shapes the context in which research on software engineering education is



reported. This includes courses related to other disciplines (computer science or information systems), training for workers of the industry, as well as courses, or software engineering programs.

## 5.6 Limitations

The results of this stage should be interpreted taking into account the following limitations: (a) only the names and summaries of the collected publications were studied, (b) criteria used to group the articles in order to process them was arbitrary, in spite of the attempt to obtain subsets of similar size and publication date, (c) the main aim of this stage is to obtain an initial controlled vocabulary using the most used terms, and therefore, the result is not a comprehensive vocabulary; and (d) the body of the studied literature in this work is limited and it corresponds to a sample of research articles, since the objective of the created taxonomy is the classification of research on software engineering education. There are other sources used for the teaching of software engineering, such as text books and message forums on the web, which have not been taken into consideration due to lack of resources and prioritizing.

## 6 Second Stage: Expanding techniques and methods of teaching software engineering

After the first stage, we decided to expand the taxonomy to include more terms. For the ‘how to teach’ facet, the low number of terms has a greater impact as there is no reference material available in the literature, such as the SWEBOK [2] for the ‘what to teach’ facet. This is the reason why we carried out a second stage of the process with the aim of expanding the ‘teaching approaches and methods’ category of the ‘how to teach’ facet. This stage and its results are presented in detail below. Each subsection corresponds to a phase of the process presented in section 4 and describes the details of its execution in this case.

### 6.1 Definition of the objectives and work plan

The objective of this stage is the expansion of the ‘teaching approaches and methods’ category. This implies increasing the number of terms of the controlled vocabulary included in the category and identifying a concise and unambiguous definition for each one.

We decided to maintain the high priority given to the literary warrant in the previous stage. Since in this stage we were studying teaching methods and approaches that are supposed to be used by other disciplines and not only in software engineering, we decided to expand the literature review to engineering and technology education.

### 6.2 Vocabulary gathering

In this stage, as in the previous one, vocabulary gathering is done through the content auditing technique. In this case in particular, there is an initial list of terms in the category, therefore, the goal is to validate and extend this set, so that it will include a greater number of concepts. Besides, specific information is collected that will later be synthesized and incorporated into each entry of the taxonomy (i.e., for each term: a definition, possible synonyms and relevant references). The steps involved in the vocabulary gathering phase of the process are shown in Figure 3 and are explained below.

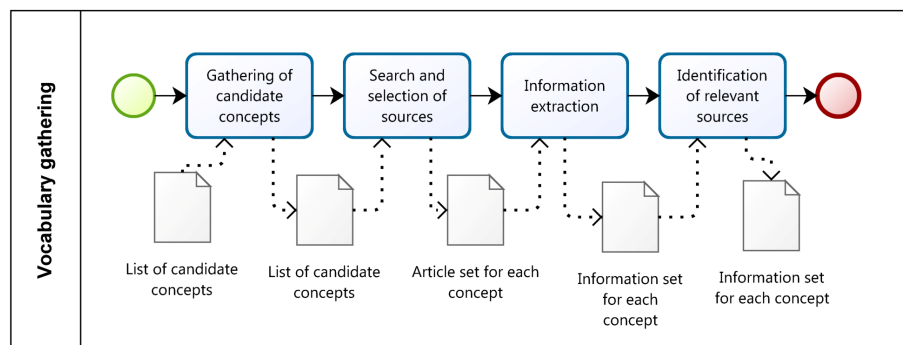


Figure 3: Detail of activities of the Vocabulary Gathering Phase for the ‘teaching approaches and methods’ expansion

### 6.2.1 Gathering of candidate concepts

In order to expand the list of candidate concepts to be included in the taxonomy, we requested help from one expert in Engineering Education from the Teaching Department in the Faculty of Engineering at our University. With her help, we identified the following three sources of concepts related to the teaching of engineering:

- S1 The website on didactic techniques of the Monterrey, Instituto Tecnológico y de Estudios Superiores de Monterrey [42]. This space is linked to the initiative to redesign the teaching practice that the Tecnológico de Monterrey has undertaken since 1997. The site contains sections alluding to the process of implementation of teaching techniques carried out in this institution.
- S2 The book “Teaching Engineering” by Professor Peter Goodhew of the University of Liverpool [43]. In this book the author presents a review of learning and teaching techniques used in engineering education.
- S3 The Student-Centered Teaching Methods list prepared by the Council on Science and Technology at Princeton University [44].

Teaching techniques and methods are presented in each of these sources. The concepts related to these techniques were collected manually and added to the concepts obtained in the previous stage for the ‘teaching approaches and methods’ category. Table 2 shows all the candidate concepts to the ‘teaching approaches and methods’ category including ones from the initial taxonomy (2nd column) and new candidates from sources recommended by an engineering education expert (columns 3rd to 5th).

Table 2: Extended list of Candidate concepts to the ‘teaching approaches and methods’ category.

Candidate concept	Initial tax.	S1	S2	S3
Lectures		✓		
Tutorials		✓		
Problem-based learning	✓	✓	✓	✓
Project-based learning		✓	✓	
Student-centered learning			✓	
Active learning			✓	
Cooperative learning		✓	✓	
E-learning			✓	
Simulation and games			✓	✓
Distance learning			✓	
Personal development planning			✓	
Case studies	✓	✓		✓
Service-learning		✓		
Research-based learning		✓		
Think-pair-share				✓
Clickers				✓
One-minute papers				✓
Interactive lecture demonstrations				✓
Concept mapping				✓
Tutorial worksheets				✓
Just-in-time teaching				✓
Invention activities				✓
Group-tests				✓
Problem sets in groups				✓
Random calling				✓
Writing with peer review				✓
Technology enhanced learning			✓	
Academia and industry collaboration	✓			
Capstone project	✓			
Globally distributed project course	✓			
Team projects	✓			
Simulation of real life	✓			
Software engineering project course	✓			
Real-clients projects	✓			

### 6.2.2 Search and selection of sources

To validate that each candidate concept is used in education research, in this step, we look for research articles that mention that concept and provide, for example, definitions or references. In practice, this involves collecting material that allows us to determine, in steps that follow, if the term will be included in the taxonomy, and to develop a definition for that term as well as a list of relevant synonyms and references.

In order to avoid a potential specialization bias that corresponds to the distortion of the technique because of its application to the field of software engineering education, or to the education of engineering in general, we considered articles on teaching (without restrictions). Then, for each concept listed in Table 2, we developed a search string, which included the following terms: Teaching approach, Teaching method, Teaching technique, Learning approach, Learning method. As an example the search string for the ‘Lectures’ concept is shown in Table 3.

Table 3: Search string for the ‘Lectures’ concept

((“lecture” OR “lecture-based”) AND (“teaching approach” OR “teaching method” OR “teaching technique” OR “learning approach” OR “learning method” OR “learning technique”))
---

The search was conducted in SCOPUS and a number of articles was chosen from the list ordered by number of quotes using the following criteria. We selected the articles whose title and abstract were closely related to the concept under study. The selection process involved reading the titles and abstracts of the articles trying to find those which refer to the foundations of the studied concept, to its application or its comparison with other techniques. Those articles that refer to the concept indirectly were not taken into account. For this analysis we also used the content analysis principles. We chose in order up to the first 20 articles that met our requirements. This number is arbitrary, even though we believe that it respects Hedden’s recommendations because it allows us to have a certain literary warrant [9]. We also did several previous tests, in which we built definitions for some candidate concepts, with satisfactory results. As an example, 12 articles were selected for the ‘Lectures’ concept, the only ones that met our criteria on the title and abstract, out of the 204 that returned the search. In other cases, for concepts with more publications that met our criteria, 20 articles were selected, for example, for the ‘E-learning’ concept.

### 6.2.3 Information extraction

In the previous step, we obtained a set of articles for each candidate concept; the aim of this third step is the extraction of relevant information on each concept of the gathered articles. We read the collected articles and identified fragments in which the concept studied was presented or discussed. From these fragments, we extracted descriptions, synonyms and related references used in relation to that concept.

Figure 4 shows the extraction information for the ‘Lectures’ concept. In the upper part the extraction of the articles is presented: in the 1st column basic paper information is included, 2nd column presents the description given in the paper to the concept studied (in this case ‘Lectures’), in the 3rd column possible synonyms used in the paper for that concept (in this example the papers did not include any), and the 4th column includes possible references of interest on that concept. The same type of extraction is shown at the bottom part but for the relevant sources identified from the references of interest (4th column from the upper part).

### 6.2.4 Identification of relevant sources

The aim of this step is to detect possible relevant sources for each of the candidate concepts and extract their information to incorporate it into the result of the previous phase. We consider relevant sources to be articles, books or other bibliographic references that are frequently quoted in the collected articles in the previous phases for a candidate concept.

In practice, this involves the following activities. For each concept, we reviewed the list of references registered in the previous phase, in order to detect publications quoted in three or more articles. If publications that fulfill this condition were detected, we searched the complete text and added it to the set of sources to be analyzed for that concept. Then, information was extracted in the same way as in step 6.2.3.

Following the example, for the ‘Lectures’ concept we detected that a relevant reference was the book *Bligh, Donald A. 1972. What’s the Use of Lectures? 3rd ed., Penguin Education. Harmondsworth: Penguin.* We analyzed this book and added the relevant information to that collected in the previous phase. This is shown in the bottom part of Figure 4.

Article	Description	Synonyms	References
P. Goodhew, «Teaching engineering», UKCME, Liverpool, 2014.	The authors name the technique but do not give a definition.	-	· Bligh, Donald, A., What's the Use of Lectures? 3rd Edition, Penguin Books 1972 · Sutherland, P. and Badger, R., (2004) Lecturers' Perceptions of Lectures; J Further & Higher Education 28, 3, 277-28
J. Biggs y C. Tang, Teaching For Quality Learning At University. McGraw-Hill Education, 2011.	"A lecture is where the subject matter expert tells the students about the major topics that make up the discipline or professional area, and what the latest thinking is on a topic or discipline."	-	· Bligh, Donald, A., What's the Use of Lectures? 3rd Edition, Penguin Books 1972
Butler J.A. Use of teaching methods within the lecture format	"The continuous exposition by one person for approximately 50 minutes to a largely passive recipient audience."	-	· Bligh, Donald, A., What's the Use of Lectures? 3rd Edition, Penguin Books 1972 · H. HASEKURA, H. FUKUSHIMA, y K. HIRAIDE, «Evaluation of lectures on medical ethics through students' drawings», Medical Education, vol. 24, n.º 1, pp. 42-45, 1990.
Van Dijk L.A., Jochems W.M.G. Changing a traditional lecturing approach into an interactive approach: Effects of interrupting the monologue in lectures	"Typically consisting of a monologue delivered by the lecturer."	-	· C. C. Bonwell and J. A. Eison, Active Learning: Creating Excitement in the Classroom, The George Washington University, School of Education and Human Development, Washington, DC, ASHE ERIC Higher Education ReportNo.1(1991). · A. Saroyan and L. S. Snell, Variations in lecturing styles, Higher Education, 33(1997) pp. 85 104
Omelicheva M.Y., Avdeyeva O. Teaching with lecture or debate? Testing the effectiveness of traditional versus active learning methods of instruction	"The lecture format is conducive to direct reproduction of information laid out by the instructor. Since the lecturer has maximum control over the flow of information, and is able to provide students with precise, up-to-date, and not otherwise available information more effectively than debaters, who may deliberately or inadvertently misrepresent or omit data, we expect that lecture will be more effective than educational debate in transmitting factual knowledge to students.  In a lecture class, students are passive learners, engaged in extensive note-taking."	-	· Beard, R. Research into Teaching Methods in Higher Education, mainly in British Universities. 4th Ed. Guildford Society for Research into Higher Education, ; 1978. · Bligh, D. What's the Use of Lectures?. Harmondsworth, Penguin; 1972. · Goolkasian, P. Memory for Lectures. Journal of Educational Psychology. 1979;71. · Green, E.B. Relative effectiveness of the lecture and individual readings as methods of college teaching. Genetic Psychology Monographs. 1928;4. · Guyot, Y. Teacher location and teacher-student relationships. International Review of Applied Psychology. 1970;19:2. · Kintch, W. Recognition Memory. Journal of Experimental Psychology. 1977;3. · Lloyd, D.H. A concept of improvement of learning responses in the taught lesson. Visual Education. 1968; (Oct 1968). · Thomas, E.J. The variation of memory with time of information appearing during a lecture. in: Studies in Adult Education. 1. University of Bristol, ; 1972.
<b>Relevant Sources</b>			
Bligh, Donald, A., What's the Use of Lectures? 3rd Edition, Penguin Books 1972	"[lectures] are in fact more or less continuous expositions by a speaker who wants the audience to learn something."		

Figure 4: Example of extracted information for 'Lectures' concept.

### 6.3 Identification of concepts and relationships

During this phase, we analyzed the extracted information and identified the concepts and relationships that would be included in the taxonomy. This activity involved reviewing the material and, first of all, dismissing the candidate concepts (see section 6.2.1) whose searches did not retrieve research articles (see section 6.2.2).

To carry out the synthesis of the collected definitions, we used an adaptation of the thematic synthesis process [45, 46], as shown in Figure 5. The process - a type of qualitative synthesis - makes it possible to reach a concise definition of each concept using the key information segments (codes) from other definitions, so that the definition obtained has the fundamental ideas of the rest.

In Figure 6, the application of the synthesis process to the 'Lectures' concept is shown. The upper part shows the coding process: the 1st column presents some paper fragments with descriptions of the concept 'Lectures' where the relevant parts have been highlighted, the 2nd column shows codes that tag the highlighted parts. The definition obtained by using the most frequent codes is shown at the bottom. All the phases of the thematic synthesis, as well as all the activities presented in this section, were conducted manually.

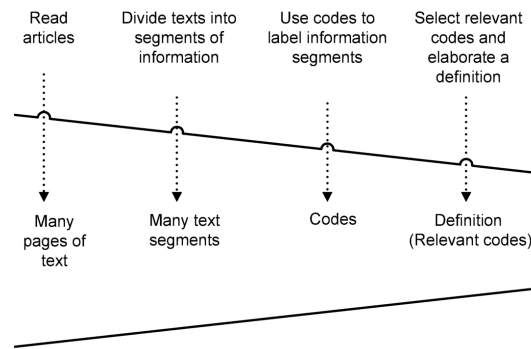


Figure 5: Thematic synthesis process adapted from Creswell [46]

Articles	Codes
<i>J. Biggs y C. Tang. Teaching For Quality Learning At University. McGraw-Hill Education, 2011.</i> A lecture is where the subject matter expert tells the students about the major topics that make up the discipline or professional area, and what the latest thinking is on a topic or discipline.	expert tells the students
<i>Bligh, Donald, A., What's the Use of Lectures? 3rd Edition, Penguin Books 1972</i> [lectures] are in fact more or less continuous expositions by a speaker who wants the audience to learn something.	continuous exposition speaker audience
<i>Butler J.A. Use of teaching methods within the lecture format</i> The continuous exposition by one person for approximately 50 minutes to a largely passive recipient audience.	continuous exposition one person passive audience
<i>Van Dijk L.A., Jochems W.M.G. Changing a traditional lecturing approach into an interactive approach: Effects of interrupting the monologue in lectures</i> Typically consisting of a monologue delivered by the lecturer.	monologue lecturer
<i>Omelicheva M.Y., Avdeyeva O. Teaching with lecture or debate? Testing the effectiveness of traditional versus active learning methods of instruction</i> The lecture format is conducive to direct reproduction of information laid out by the instructor. Since the lecturer has maximum control over the flow of information, and is able to provide students with precise, up-to-date, and not otherwise available information more effectively than debaters, who may deliberately or inadvertently misrepresent or omit data, we expect that lecture will be more effective than educational debate in transmitting factual knowledge to students. In a lecture class, students are passive learners, engaged in extensive note-taking.	direct reproduction instructor  passive learners
<b>Definition</b>	
Lectures are (more or less) continuous expositions of a theme by a speaker to a passive recipient audience.	

Figure 6: Synthesis process applied to the 'Lectures' concept.

#### 6.4 Validation of the controlled vocabulary

Up until now, we have not carried out activities to validate the taxonomy resulting from the expansion achieved in this stage. We wish to carry out activities that correspond to utility demonstration in order to validate that the taxonomy is suitable for use. In this sense, a case study can be a very good alternative. This is a relevant pending activity, but we thought it important to disseminate our progress in the software engineering education and engineering education communities before exploring an empirical validation -that will surely be somewhat extensive in time-, in order to obtain feedback from expert researchers who will enrich the work done.

#### 6.5 Results

In this stage, we reviewed 34 candidate concepts and studied over 250 definitions from several authors. The result comprises the expansion and formalization of the 'teaching approaches and methods' category of the 'how to teach' facet, which now covers 26 terms with their definitions and most relevant references. A concept is considered to be really used in education research if it has associated publications. The candidate concepts that were not finally included in the taxonomy did not have that bibliographic support or were synonyms of other concepts already included in the vocabulary.

We include a template for the presentation of each taxonomy entry that contains preferred term, defini-

tion, synonyms, facet, related terms and comments on relevant publications (in the field of Related content). An example entry is shown in Table 4 (see Appendix A for all the detail of the entries of the ‘teaching approaches and methods’ category).

Table 4: Example entry with the term ‘Lectures’

Term	Lectures
Definition note	Lectures are (more or less) continuous expositions of a theme by a speaker to a largely passive recipient audience.
Synonyms	
Related terms	
Related content	The definition included here is based on Bligh’s classic work [47].

## 6.6 Discussion

The complete taxonomy that includes the category ‘teaching approaches and methods’ resulting from our work with all its preferred terms is presented in Figure 7 while the detail of the entries of the category can be consulted in Appendix A. As a strength, we can point out the literary warrant with which the taxonomy was built as well as the transparency and rigor of the procedures.

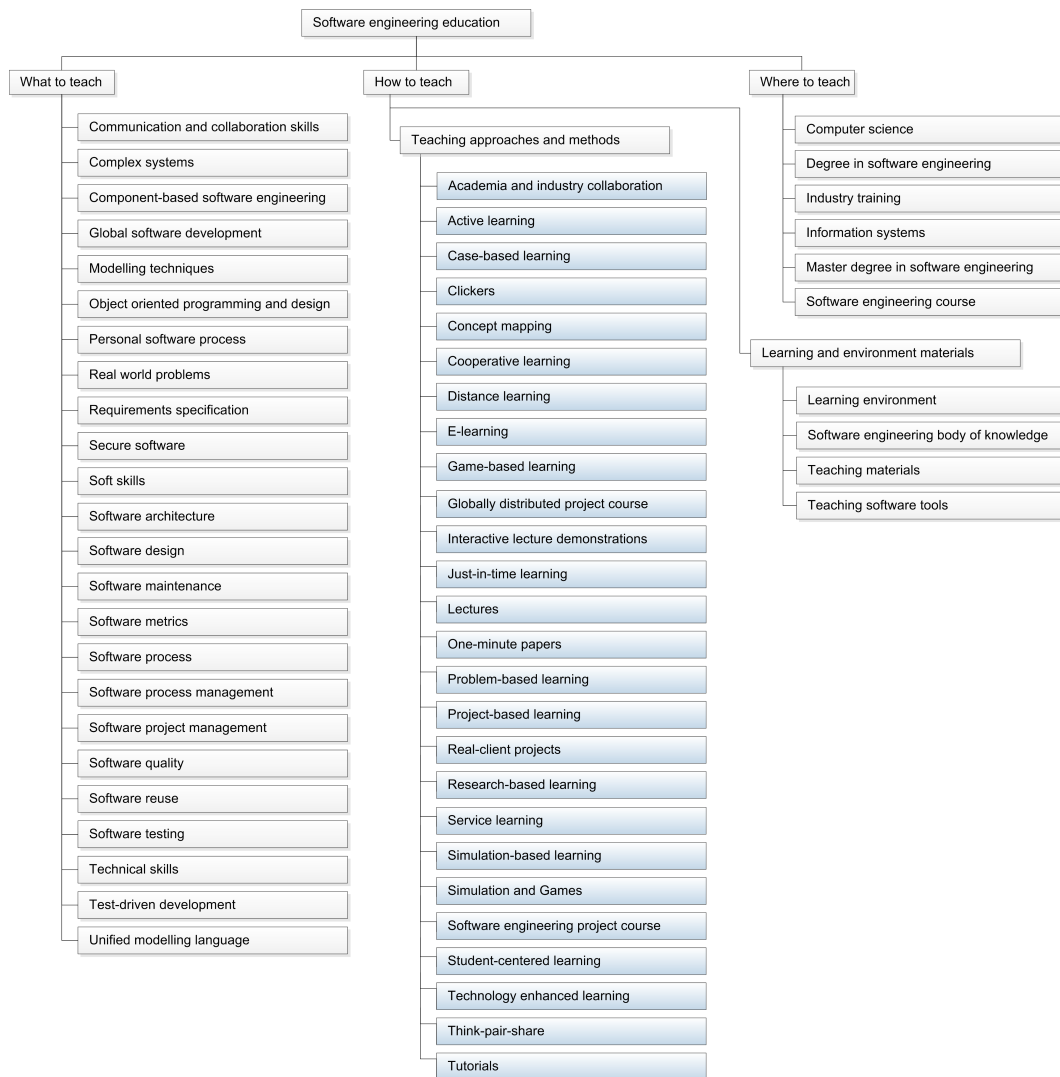


Figure 7: Initial taxonomy of software engineering education, as result of the two stages of our work, with ‘teaching approaches and methods’ category highlighted.

Another alternative for the expansion of the ‘how to teach’ facet could have been to borrow concepts and definitions directly from pedagogy texts, e.g., [42, 43, 44], where terms such as ‘problem-based learning’

already exist. This does not prioritize the literary warrant, which can lead to having a large number of terms of which very few are used for the classification of the specific area, and it can also lead to the absence of specific terms of the area. Considering these limitations, we decided to prioritize the literary warrant and study terms resulting from published papers.

In comparison with the project by Finelli et al. [5], our work is modest and incipient. This is probably due to the fact that Engineering Education has a more advanced context (this can be seen in the number of previous work cited by the authors), and to the financing fund that surely allowed authors to attend different venues to achieve the involvement of different stakeholders. The latter is an important aspect of their work because they have managed not only to create an inclusive construction process but also to validate the obtained taxonomy. While the taxonomy of Finelli et al. covers engineering education, the taxonomy that we present here tries to cover software engineering education, being, therefore, more specific.

The scope of the engineering education research (EER) taxonomy is broader than that of software engineering education (SEE taxonomy). While the former also refers to research concepts (for example, the ‘research approaches’ branch), the latter currently only applies to the field of software engineering education. What is more, the ‘instruction’ branch of the EER taxonomy that includes a sub-branch called ‘instructional methods’ with terms like ‘lecture’, ‘team-based learning’ and ‘project-based learning’, is similar to the ‘how to teach’ facet of the SEE taxonomy. The ‘outcomes’ branch of the EER taxonomy covers terms on learning outcomes that may be related to the content of the ‘what to teach’ facet of the SEE taxonomy, although the latter currently covers topics of the area and not outcomes as ‘critical thinking’ or ‘problem solving’ as the EER taxonomy does. It would be interesting to take the EER taxonomy as a reference for future work on the SEE taxonomy both regarding its working method and the structure and content of the taxonomy. The SEE taxonomy could also be thought of as a specialization or extension of the EER Taxonomy for a particular area.

Although neither the work by Usman et al, nor that of Bayona-Oré et al. are based on the Z39.19-2005 guide, they share almost the same activities as our process. The four phases defined have the same objectives as ours: definition of objectives and planning, collection of vocabulary, taxonomy design and construction, and finally validation. There is only one activity that is not contemplated in our method: the definition of guides to use and update the taxonomy, which we consider adequate and we will surely add in future updates of the method, after reviewing other similar work and analyzing the subject in the aforementioned guide.

## 6.7 Limitations

The results must be interpreted taking into consideration the following limitations: (a) the list of concepts was certainly biased by the initial sources of candidate concepts, i.e., [42, 43, 44]; it is quite likely that there are concepts related to software engineering approaches and teaching methods that have not yet been included in the presented taxonomy, (b) the criterion to select the articles to be considered in the search for definitions and information about each concept is based on the reading of the title and the abstract only, (c) publications in a language other than English have not been taken into account for the searches related to each concept, (d) no activities have been carried out yet to validate the taxonomy.

## 7 Conclusions

Software engineering education is a relatively new discipline and it is the object of wide research. It suffers constant demands associated to the critical role that software plays in the current world. Although there are bodies of knowledge and textbooks on software engineering, the literature on the area of software engineering education consists mainly of articles published in journals and specialized conferences. This material has not been classified and does not have a standard terminology.

The importance of organizing the knowledge and publications is associated to the benefits obtained by the different stakeholders involved, i.e., allowing researchers to place their initiatives in a suitable context, achieving a better induction to the terms and concepts of the thematic area, improving the management of scientific journals and conferences through a better thematic organization of authors and research, and also substantially improving the accumulation of evidence (systematic reviews of literature and mapping studies).

Starting from the guidelines and activities proposed in the Z39.19-2005 guide provided by ANSI/NISO [8] and the recommendation by Hedden [9], we adapted a process for the creation and update of controlled vocabularies.

We applied this process in two stages: firstly, we explored the existing literature and created an initial taxonomy of software engineering education, and secondly, we used the process to expand the number of terms on teaching approaches and methods. Although in both stages the work was done with the same process, it is important to highlight the fact that the objectives and the techniques used were different.

The taxonomy has 60 terms organized in three facets at its highest level. To the best of our knowledge, this taxonomy is the first in the software engineering education field. It can be used to create keywords for the labelling of articles, to understand a concept and go even further consulting the suggested sources, and it can also be used as a basis for future work within the terminology standardization in this area. Because of the number of concepts it includes, we believe it is a taxonomy that should be considered initial or in the process of being constructed.

It would be good to continue working in the expansion of the taxonomy maintaining the rigor and support in the literary warrant. Possible activities along this line could be: the expansion of the ‘what to teach’ facet using the SWEBOK as reference material or continue the expansion of the ‘teaching approaches and methods’ category or exploring specific teaching methods and techniques used for software engineering education.

Activities that allow the evaluation of their suitability for use, which in the proposed process are known as validation activities, must also be carried out. It is possible and it seems reasonable to combine some empirical validation activities with the experts’ approach, including for example, surveys for experts on their perceptions of the taxonomy and evaluation of its use in tagging scientific articles by independent researchers.

## Appendix A “Teaching approaches and methods’ category

This appendix presents, as shown in Table 5, the entries’ details of the ‘teaching approaches and methods’ category resulting from the second stage of our work (see Section 6).

Table 5: “Teaching approaches and methods’ category of the taxonomy on software engineering education.

Term	Academia and industry collaboration
Definition note	It refers to any effort made together between academia (one or more universities) and industry (one or more companies) for the education of software engineering concepts and practices.
Synonyms	
Related terms	
Related content	
Term	Active learning
Definition note	Active learning is generally defined as any instructional method that engages students in the learning process. In short, active learning requires students to do meaningful learning activities and think about what they are doing.
Synonyms	
Related terms	
Related content	The definition included here is based on Bonwell and Eison [48].
Term	Case-based learning
Definition note	Students analyze, draw inferences and make decisions given a case study - a detailed specific situation with a open-ended question or problem (often based on a real life scenario). Case studies can be presented to individuals or groups.
Synonyms	Case studies, CBL, case study approach
Related terms	
Related content	
Term	Clickers
Definition note	A clicker (or a audience response system) is a combination of hardware and software that enables the instructor to pose real-time questions to students during a lecture. The students usually register their responses using handheld clickers, although other input such as laptops may be used. After receivers transmit the responses to the instructor’s workstation, the software compiles and displays the results.
Synonyms	Audience response system, personal response system, classroom response system, student response system
Related terms	
Related content	In one of the most cited paper [49], Caldwell includes a comprehensive overview of the technique and best-practice tips.
Term	Concept mapping
Definition note	Concepts maps can be defined as a knowledge representation language. In short, the students create or use graphic structures that arrange key ideas or concepts in a hierarchical set of nodes with lines or arrows that indicate linkages and relationships between them.
Synonyms	
Related terms	
Related content	Concept mapping were developed by Joseph Novak. In two often cited works he presents the basis of concept mapping and Vee diagrams [50, 51].

*Continue in next page*



Table 5 – Continued from previous page

Term	Cooperative learning
Definition note	Refers to any of a variety of teaching methods in which students work in small groups to help one another learn academic content. Most experts agree that cooperative learning has several components that distinguish it from other small group learning methods. These components may include: positive interdependence (a positive correlation between the gains of individuals and the gains of teams), individual accountability (although learning activities rely on cooperative efforts, individuals are ultimately responsible of their own learning), group processing (group members discuss their progress towards the achievement of their goals and the maintenance of effective working relations), face-to-face interaction (the size of the groups must be small), social and cooperative skills (that must be taught and motivated by the instructor) and appropriate grouping (some authors recommend heterogeneous teams, reflecting varied learning abilities, ethnic and linguistic diversity and a mixture of the sexes).
Synonyms	
Related terms	
Related content	Three important and cited books about this topic are those from Slavin [52]; Kagan [53]; and Johnson et al.[54].
Term	Distance Learning
Definition note	Any educational or learning process or system in which the teacher and instructor are separated geographically or in time from his or her students; or in which students are separated from each other or educational resources. Contemporary distance learning is affected through the implementation of computer and electronics technology to connect teacher and student in either real or delayed time or on an as-needed basis.
Synonyms	
Related terms	
Related content	The description included here is the broader found, taken from Illyefalvi-Vitez and Gordon [55].
Term	E-learning
Definition note	E-learning can be defined as instruction delivered electronically via the internet, intranet, or multimedia platforms such as cd-rom or dvd. E-learning is used to describe a wide set of applications and processes, such as web-based learning, virtual classrooms, and digital collaboration.
Synonyms	
Related terms	
Related content	The definition included here is based on the work of Smart and Cappel [56] and in the glossary of Kaplan-Leiserson [57], cited by Derouin et al. [58], although it is no longer available online.
Term	Game-based learning
Definition note	This technique deals with games that have defined learning outcomes. A game can be defined as an activity that is voluntary and enjoyable, separate from the real world, uncertain, unproductive in that the activity does not produce any goods of external value, and governed by rules.
Synonyms	Games
Related terms	Simulation and Games
Related content	The definition of a game is taken from Caillois and Barash [59].
Term	Globally distributed project course
Definition note	It refers to some initiatives within Project-Based Learning. In these cases, students face projects distributed between two or more universities which work collaboratively on this initiative.
Synonyms	
Related terms	Project-Based Learning
Related content	
Term	Interactive lecture demonstrations
Definition note	Students are asked to predict individually the outcome of a classroom demonstration. Later the students interact in small groups, discussing their predictions and explaining their reasoning. Finally, the demonstration is performed and the students discuss and reflect on the results.
Synonyms	
Related terms	
Related content	The approach was developed by Sokoloff and Thornton [60, 61].
Term	Just-in-time teaching
Definition note	This approach is based on the feedback loop between the students and the instructor. The instructor uses the internet to post course materials and warm up assignments before class, and the students use those materials to prepare themselves for each class. The instructor uses students responses to enhance the classroom component.
Synonyms	Just-in-time, JiTT

*Continue in next page*

Table 5 – Continued from previous page

Related terms	
Related content	JiT'T was first presented by Novak [62].

Term	Lectures
Definition note	Lectures are (more or less) continuous expositions of a theme by a speaker to a largely passive recipient audience.
Synonyms	
Related terms	
Related content	The definition included here is based on Bligh's classic work [47].

Term	One-minute papers
Definition note	The instructor asks the students (often in the last minutes of class) to write a quick response to one or more questions regarding the content of the class (typically a lecture). Questions might include: what is the most important thing you learned today? what is the muddiest point still remaining at the conclusion of today's class? After collecting the responses, the instructor reads the questions and ideally responds to them in the next class, or privately on an individual basis.
Synonyms	
Related terms	
Related content	More information of this technique can be found in two often cited works by Cross and Angelo [63] and Chizmar and Ostrosky [64].

Term	Problem-based learning
Definition note	Students work in groups to analyze and solve a problematic situation, usually a realistic scenario without a single correct answer, under the supervision of a tutor.
Synonyms	PBL
Related terms	
Related content	Two relevant and cited articles about this topic are those by Barrows [65] and Schmidt [66].

Term	Project-based learning
Definition note	Students, typically organized in groups, face open multidisciplinary projects with the instructor playing the role of facilitator or coach. The projects engage students in authentic real-world problems and usually lead to the production of a final product (a design, a model, a software product, etc).
Synonyms	PjBL
Related terms	
Related content	

Term	Real-client projects
Definition note	It refers to instances of Project-Based Learning with real clients (this means the clients are not teachers or other students, they are usually industry members).
Synonyms	
Related terms	Project-based learning
Related content	

Term	Research-based learning
Definition note	It refers to initiatives that connect teaching with research, which allow partial or full inclusion of students in an investigation based on scientific methods, under the supervision of an instructor.
Synonyms	
Related terms	
Related content	

Term	Service learning
Definition note	A form of experiential learning in which students engage in activities that address human and community needs while allowing students to reflect on their service to gain further understanding of course concepts.
Synonyms	
Related terms	
Related content	

Term	Simulation-based learning
Definition note	This technique deals with simulations that have defined learning outcomes. Simulation is a technique to replace or amplify real experiences with guided experiences, often immersive in nature, that evoke or replicate substantial aspects of the real world in a fully interactive fashion.
Synonyms	
Related terms	Simulation and Games
Related content	The selected description is the broader found, taken from Gaba [67]

Term	Simulation and Games
------	----------------------

*Continue in next page*

Table 5 – Continued from previous page

Definition note	Both techniques seeks the instruction by guided experiences in ruled environments (usually immersive). In other words, both have some underlying model, allowable actions that the learner can take, and constraints under which these actions should occur. Additionally, learners observe their actions' consequences. The key distinction is that simulations propose to represent reality and games do not.
Synonyms	
Related terms	
Related content	The definition included here is based on the work of Garris et al. [68].
Term	Software engineering project course
Definition note	It refers to initiatives within Project-Based Learning related to specific courses in which students work on a software engineering project (generally involving software development).
Synonyms	
Related terms	Project-Based Learning
Related content	
Term	Student-centered learning
Definition note	Ways of thinking about teaching and learning that emphasise student responsibility and activity in learning rather than what teachers are doing. The students exert a degree of influence over both the content of the course and the learning methods.
Synonyms	
Related terms	
Related content	The first part of the definition included here is taken from the work of Cannon and Newble [69]. Two relevant and cited articles about this topic are those from Lea et al. [70] and Hannafin et al. [71].
Term	Technology enhanced learning
Definition note	All approaches in which technology is used to support the learning or teaching process.
Synonyms	
Related terms	
Related content	The definition included here is taken from the work by Schweighofer and Ebner [72].
Term	Think-pair-share
Definition note	Students are taught to listen to a question, think about the question, to discuss the question in pairs, and finally to share with the total group.
Synonyms	
Related terms	
Related content	This technique was first proposed by Lyman [73].
Term	Tutorials
Definition note	It is an activity in which the instructor works with one or a small groups of students and that is characterized as a space for discussion. Usually it served to complement other teaching techniques (e.g. lectures) and can be enhanced if the students have done some relevant prior work. It is considered a technique within the student-centered learning approach.
Synonyms	
Related terms	
Related content	Tutorial teaching is part of the learning system at the University of Oxford and involves some particular features. Palfreyman provide a good review of this technique [74] .

## References

- [1] I. Sommerville, *Software engineering*, 10th ed. Harlow, England: Pearson, 2015.
- [2] P. Bourque and R. E. Fairley, Eds., *SWEBOK: Guide to the software engineering body of knowledge*, version 3.0 ed. IEEE Computer Society, 2014.
- [3] P. Freeman, A. I. Wasserman, and R. E. Fairley, “Essential elements of software engineering education,” in *2nd International Conference on Software Engineering*, 1976, pp. 116–122.
- [4] T. C. Lethbridge, “What knowledge is important to a software professional?” *Computer*, vol. 33, no. 5, pp. 44–50, 2000.
- [5] C. J. Finelli, M. Borrego, and G. Rasoulifar, “Development of a taxonomy of keywords for engineering education research,” *Journal of Engineering Education*, vol. 104, no. 4, pp. 365–387, 2015.
- [6] B. Kitchenham, D. Budgen, and P. Brereton, *Evidence-based software engineering and systematic reviews*, ser. Chapman & Hall/CRC Innovations in Software Engineering and Software Development Series. CRC Press, 2015.
- [7] S. Pizard and D. Vallespir, “Towards a controlled vocabulary on software engineering education,” *European Journal of Engineering Education*, vol. 42, no. 6, pp. 927–943, 2017.
- [8] ANSI/NISO, *Z39.19-2005 - Guidelines for the construction, format, and management of monolingual controlled vocabularies*. Baltimore, USA: National information standards organization, May 2005.
- [9] H. Hedden, *The accidental taxonomist*. New Jersey, USA: Information Today Inc, May 2010.
- [10] *ISO/IEC/IEEE International Standard - Systems and software engineering – Vocabulary*, Dec 2010.
- [11] T. Levy, “Standard taxonomies offer flexibility,” *Seybold Report: Analyzing Publishing Technologies*, vol. 4, no. 8, pp. 9 – 11, 2004.
- [12] P. Lambe, *Organising knowledge: Taxonomies, knowledge and organisational effectiveness*, ser. Chandos knowledge management series. Oxford, UK: Chandos, 2007.
- [13] L. N. Cassel, A. Hacquebard, A. McGettrick, G. Davies, R. LeBlanc, C. Riedesel, Y. L. Varol, G. T. Finley, S. Mann, and R. H. Sloan, “A synthesis of computing concepts,” *ACM SIGCSE Bulletin*, vol. 37, no. 4, pp. 162–172, 2005.
- [14] L. N. Cassel, G. Davies, W. Fone, A. Hacquebard, J. Impagliazzo, R. LeBlanc, J. C. Little, A. McGettrick, and M. Pedrona, “The computing ontology: application in education,” *ACM SIGCSE Bulletin*, vol. 39, no. 4, pp. 171–183, 2007.
- [15] L. N. Cassel, S. K. Inguva, and J. Buzydowski, “An ontology of all of computing: An update on challenges and approaches,” in *14th International Conference on Artificial Intelligence: Methodology, Systems, Applications*, D. Dicheva and D. Dochev, Eds., 2010, pp. 283–284.
- [16] M. A. Musen, “The protégé project: A look back and a look forward,” *AI Matters*, vol. 1, no. 4, pp. 4–12, 2015.
- [17] L. N. Cassel, “Computing ontology,” accessed: 2018-11-06. [Online]. Available: <http://www.distributedexpertise.org/computingontology/>
- [18] Joint Task Force on Computing Curricula, “Computing curricula 2001,” *J. Educ. Resour. Comput.*, vol. 1, Sep. 2001.
- [19] Simon, A. Clear, J. Carter, G. Cross, A. Radenski, L. Tudor, and E. Tõnisson, “What’s in a name?: International interpretations of computing education terminology,” in *Integrating Technology into Computer Science Education*, 2015, pp. 173–186.
- [20] M. Usman, R. Britto, J. Börstler, and E. Mendes, “Taxonomies in software engineering: A systematic mapping study and a revised taxonomy development method,” *Information and Software Technology*, vol. 85, pp. 43 – 59, 2017.
- [21] P. C. Wankat, “An analysis of the articles in the journal of engineering education,” *Journal of Engineering Education*, vol. 88, no. 1, pp. 37–42, 1999.

- [22] —, “Analysis of the first ten years of the journal of engineering education,” *Journal of Engineering Education*, vol. 93, no. 1, pp. 13 – 21, 2004.
- [23] K. Whittin and S. Sheppard, “Taking stock: An analysis of the publishing record as represented by the journal of engineering education,” *Journal of Engineering Education*, vol. 93, no. 1, pp. 5–12, 2004.
- [24] H. Xian and K. Madhavan, “Anatomy of scholarly collaboration in engineering education: A big-data bibliometric analysis,” *Journal of Engineering Education*, vol. 103, no. 3, pp. 486–514, 2014.
- [25] B. K. Jesiek, M. Borrego, K. Beddoes, M. Hurtado, P. Rajendran, and D. Sangam, “Mapping global trends in engineering education research, 2005-2008,” *International Journal of Engineering Education*, vol. 27, pp. 77–90, 2011.
- [26] L. Malmi, T. Adawi, R. Curmi, E. De Graaff, G. Duffy, C. Kautz, P. Kinnunen, and B. Williams, “Methodological analysis of SEFI EER papers,” in *41st SEFI (European Society for Engineering Education) Conference*, 2013.
- [27] S. Bayona-Oré, J. A. Calvo-Manzano, G. Cuevas, and T. San-Feliu, “Critical success factors taxonomy for software process deployment,” *Software Quality Journal*, vol. 22, no. 1, pp. 21–48, 2014.
- [28] R. Britto and M. Usman, “Bloom’s taxonomy in software engineering education: A systematic mapping study,” in *IEEE Frontiers in Education Conference*, 2015, pp. 1–8.
- [29] D. Šmite, C. Wohlin, Z. Galviņa, and R. Prikladnicki, “An empirically based terminology and taxonomy for global software engineering,” *Empirical Software Engineering*, vol. 19, no. 1, pp. 105–153, 2014.
- [30] G. R. Wheaton, “Development of a taxonomy of human performance: A review of classificatory systems relating to tasks and performance,” DTIC Document, Tech. Rep., 1968.
- [31] S. Michie, C. E. Wood, M. Johnston, C. Abraham, J. J. Francis, and W. Hardeman, “Behaviour change techniques: The development and evaluation of a taxonomic method for reporting and describing behaviour change interventions (a suite of five studies involving consensus methods, randomised controlled trials and analysis of qualitative data),” *Health Technology Assessment*, vol. 19, no. 99, pp. 1–188, 2015.
- [32] E. Hulme, “Principles of book classification,” *Library Association Records*, vol. 13-14, 1911.
- [33] M. Shaw, “Writing good software engineering research papers: Minitutorial,” in *25th International Conference on Software Engineering*, 2003, pp. 726–736.
- [34] K. M. Hammouda and M. S. Kamel, “Efficient phrase-based document indexing for web document clustering,” *IEEE Trans. on Knowl. and Data Eng.*, vol. 16, no. 10, p. 1279–1296, Oct. 2004.
- [35] W. B. Frakes and R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms*. Prentice Hall, 1992.
- [36] D. Weiss and S. Osinski, “Carrot2 - Open source search results clustering engine,” <http://project.carrot2.org/>, 2015.
- [37] D. Weiss, “Descriptive clustering as a method for exploring text collections,” *Unpublished doctoral dissertation, Poznan University of Technology, Poznan, Poland*, 2006.
- [38] C. Carpineto, S. Osiski, G. Romano, and D. Weiss, “A survey of web clustering engines,” *ACM Computing Surveys*, vol. 41, no. 3, jul 2009.
- [39] S. Elo and H. Kyngäs, “The qualitative content analysis process,” *Journal of Advanced Nursing*, vol. 62, no. 1, pp. 107–115, 2008.
- [40] K. Nie, T. Ma, and Y. Nakamori, “Building a taxonomy for understanding knowledge management,” *Electronic Journal of Knowledge Management*, vol. 5, no. 4, pp. 453–466, 2007.
- [41] D. Nascimento, K. Cox, T. Almeida, W. Sampaio, R. Almeida Bittencourt, R. Souza, and C. Chavez, “Using open source projects in software engineering education: A systematic mapping study,” in *Frontiers in Education Conference*, 2013, pp. 1837–1843.
- [42] Monterrey, Instituto Tecnológico y de Estudios Superiores de Monterrey, “Técnicas didácticas,” <http://sitios.itesm.mx/va/diie/tecnicasdidacticas/> (last accessed date: 04/2018), 2013.

- [43] P. Goodhew, *Teaching Engineering*. Liverpool: UKCME - <http://teachingengineering.liv.ac.uk/book/> (last accessed date: 04/2018), 2014.
- [44] Princeton, "Student-Centered Teaching Methods - Council on Science and Technology - Princeton University," <http://www.princeton.edu/cst/teaching-resources/methods/ModifiedPCASTTable.pdf>, 2014, accessed in 2018 via [http://i.wayne.edu/otl/backward\\_course\\_design/student-centered\\_teaching\\_methods.pdf](http://i.wayne.edu/otl/backward_course_design/student-centered_teaching_methods.pdf).
- [45] D. Cruzes and T. Dyba, "Recommended steps for thematic synthesis in software engineering," in *International Symposium on Empirical Software Engineering and Measurement*, 2011, pp. 275–284.
- [46] J. Creswell, *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research*. Boston, USA: Pearson/Merrill Prentice Hall, 2008.
- [47] D. A. Bligh, *What's the Use of Lectures?*, 3rd ed., ser. Penguin education. Harmondsworth: Penguin, 1972.
- [48] C. C. Bonwell and J. A. Eison, *Active Learning: Creating Excitement in the Classroom (J-B ASHE Higher Education Report Series (AEHE))*. Jossey-Bass, 1991.
- [49] J. E. Caldwell, "Clickers in the Large Classroom: Current Research and Best-Practice Tips," *CBE-Life Sciences Education*, vol. 6, no. 1, pp. 9–20, 2007.
- [50] J. Novak and D. Gowin, *Learning How to Learn*. Cambridge University Press, 1984.
- [51] J. D. Novak, "Concept maps and Vee diagrams: Two metacognitive tools to facilitate meaningful learning," *Instructional Science*, vol. 19, no. 1, pp. 29–52, Jan. 1990.
- [52] R. Slavin, *Cooperative Learning: Theory, Research, and Practice*. Allyn and Bacon, 1995.
- [53] S. Kagan, *Cooperative Learning*. San Clemente, CA: Kagan, 1994.
- [54] D. W. Johnson, R. T. Johnson, and E. J. Holubec, *Circles of Learning: Cooperation in the Classroom*, 4th ed. Edina, Minn: Interaction Book Co, 1993.
- [55] Z. Illyefalvi-Vitez and P. Gordon, "Distance learning - how to use this new didactic method in education of electronics engineering?" in *Electronic Components and Technology Conference, 2004. Proceedings. 54th*, vol. 2, Jun. 2004, pp. 1725–1730 Vol.2.
- [56] K. L. Smart and J. J. Cappel, "Students' Perceptions of Online Learning: A Comparative Study," *Journal of Information Technology Education: Research*, vol. 5, pp. 201–219, Jan. 2006.
- [57] E. Kaplan-Leiserson, "E-learning glossary," <http://www.learningcircuits.org/glossary.html>.
- [58] R. E. Derouin, B. A. Fritzsche, and E. Salas, "E-Learning in Organizations," *Journal of Management*, vol. 31, no. 6, pp. 920–940, 2005.
- [59] R. Caillois and M. Barash, *Man, Play, and Games*, ser. Sociology/Sport. University of Illinois Press, 1961.
- [60] D. R. Sokoloff and R. K. Thornton, "Using interactive lecture demonstrations to create an active learning environment," *The Physics Teacher*, vol. 35, pp. 340–347, 1997.
- [61] D. Sokoloff and R. Thornton, *Interactive Lecture Demonstrations, Active Learning in Introductory Physics*. Wiley, 2006.
- [62] G. Novak, *Just-in-Time Teaching: Blending Active Learning with Web Technology*, ser. Ellis Horwood Series in Environmental Management, Science an. Prentice Hall, 1999.
- [63] K. Cross and T. Angelo, *Classroom Assessment Techniques: A Handbook for Faculty*. National Center for Research to Improve Postsecondary Teaching and Learning, 1988.
- [64] J. F. Chizmar and A. L. Ostrosky, "The One-Minute Paper: Some Empirical Findings," *The Journal of Economic Education*, vol. 29, no. 1, pp. 3–10, 1998.
- [65] H. S. BARROWS, "A taxonomy of problem-based learning methods," *Medical Education*, vol. 20, no. 6, pp. 481–486, 1986.

- [66] H. G. Schmidt, "Foundations of problem-based learning: Some explanatory notes," *Medical Education*, vol. 27, no. 5, pp. 422–432, 1993.
- [67] D. M. Gaba, "The future vision of simulation in health care," *Quality and Safety in Health Care*, vol. 13, no. suppl 1, pp. i2–i10, 2004.
- [68] R. Garris, R. Ahlers, and J. E. Driskell, "Games, Motivation, and Learning: A Research and Practice Model," *Simulation & Gaming*, vol. 33, no. 4, pp. 441–467, 2002.
- [69] R. Cannon and D. Newble, *A Handbook for Teachers in Universities and Colleges: A Guide to Improving Teaching Methods*. Kogan Page, 2000.
- [70] S. J. Lea, D. Stephenson, and J. Troy, "Higher Education Students' Attitudes to Student-centred Learning: Beyond 'educational bulimia'?" *Studies in Higher Education*, vol. 28, no. 3, pp. 321–334, 2003.
- [71] M. J. Hannafin, J. R. Hill, and S. M. Land, "Student Centered Learning and Interactive Multimedia: Status, Issues, and Implications," *Contemporary Education*, vol. 68, no. 2, p. 94, 1997.
- [72] P. Schweighofer and M. Ebner, "Aspects to Be Considered when Implementing Technology-Enhanced Learning Approaches: A Literature Review," *Future Internet*, vol. 7, no. 1, pp. 26–49, 2015.
- [73] F. T. Lyman, "The responsive classroom discussion: The inclusion of all students," *Mainstreaming digest*, vol. 109, p. 113, 1981.
- [74] D. Palfreyman, *The Oxford Tutorial: 'Thanks, You Taught Me How to Think'*. Oxford Centre for Higher Education Policy Studies - OxCHEPS, 2001.