

Article

A Collaborative Method for Scoping Software Product Lines: A Case Study in a Small Software Company [†]

Marta Cecilia Camacho ^{1,*}, Francisco Álvarez ², César Collazos ³, Paul Leger ⁴,
Julián Dario Bermúdez ⁵ and Julio Ariel Hurtado ³

¹ Institución Universitaria Colegio Mayor del Cauca, Popayán 190003, Colombia

² Universidad Autónoma de Aguascalientes, Aguascalientes 20100, Mexico; fjalvar@correo.uaa.mx

³ Universidad del Cauca, Popayán 190003, Colombia; ccollazo@unicauca.edu.co (C.C.);
ahurtado@unicauca.edu.co (J.A.H.)

⁴ Escuela de Ingeniería, Universidad Católica del Norte, Coquimbo 1781421, Chile; pleger@ucn.cl

⁵ Sunset Software House S.A.S, Popayán 190003, Colombia; julian.bermudez@sunsetsw.com

* Correspondence: cecamacho@unimayor.edu.co

[†] This paper is an extended version of Camacho M. C., Alvarez F., Collazos C., Identifying collaborative aspects during software product lines scoping. In Proceedings of the 23rd International Systems and Software Product Line Conference SPLC'19, Paris, France, September 2019; pp. 98–105.

[‡] These authors contributed equally to this work.

Abstract: SPL scoping is the activity for bounding Software Product Lines (SPL), gathering heterogeneous knowledge from diverse sources. For achieving an agreement among different stakeholders, a commonality scope must be understood and committed to. However, gathering this knowledge from stakeholders with individual interests is a complex task. This paper reports the experience of scoping the SPL of a small Colombian software company, applying and evaluating a collaborative method called CoMeS-SPL. The company was looking to develop a set of products from a product previously developed with great potential to be adapted and sold to different customers. From a collaborative relationship university–enterprise model, the research groups that developed CoMeS-SPL proposed to use it answering to the company needs for defining an organization-suitable reuse scope around its platform called CORA. Both parties joined in the scoping co-production of the first SPL of the company. This method implied that the company would perform new tasks and involve other roles different for those who are used to defining the scope of a single product. The company actors considered that they obtained a useful scope and perceived the collaboration as valuable because they shared different knowledge and perspectives. The researchers were able to provide feedback on their proposed model, identifying successes and aspects to improve. The experience allowed strengthening the ties of cooperation with the company, and new projects and consultancies are being carried out.

Keywords: software product lines; SPL scoping; collaborative work; collaboration engineering; empirical study; experience report; empirical software engineering



Citation: Camacho, M.C.; Álvarez, F.; Collazos, C.; Leger, P.; Bermúdez, J.D.; Hurtado, J.A. A Collaborative Method for Scoping Software Product Lines: A Case Study in a Small Software Company. *Appl. Sci.* **2021**, *11*, 6820. <https://doi.org/10.3390/app11156820>

Academic Editor: Paolo Renna

Received: 31 March 2021

Accepted: 14 June 2021

Published: 24 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Software Product Line Engineering (SPLE) is a production strategy based on planned reuse of the assets in the development of a set product that shares a set of common characteristics and enough variability to be different products focused on target market, known as Software Product Lines (SPL) [1]. Adopting a software production strategy, SPLE based on asset reuse has some benefits for software companies, for example, in production cost reduction, improvements in product quality, and reduction in time development [2].

An essential activity in the development of an SPL is defining its reuse scope (SPL scoping) [3], which specifies the application domain and identifies the product portfolio with the variations among them and plans the reuse infrastructure [4]. However, scoping

an SPL is a difficult and risky activity because of the complexity and the variety of different factors to be considered, and some of these normally are unknown by the technical team [4]. As a consequence, SPL scoping requires the participation of non-technical experts who usually do not take part in the development team [5].

SPL scoping depends on the knowledge and experience of roles involved in gathering the information of the target domain (context), the possibilities of development, and market conditions [5,6]. Therefore, this activity requires the interaction of participants who have partial and different knowledge because none of the scoping participants has all the necessary knowledge and experience to obtain a complete and useful scope [7]. The correct scope of an SPL depends on balanced decision-making by the participants [8] and, thus, the diversity of participants is a critical factor. It is because this activity involves marketing issues [8] as well as technical management practices of the SPL [1]. The implications of this duality have been analyzed by different authors [9–12], considering that communication and collaboration are fundamental aspects to achieving the participation of people from a variety of organizational areas, each one defending his interests [9,11,12]. However, to the best of our knowledge and understanding, SPL scoping approaches that have considered collaborative practices have not been sufficiently formalized as organized collaborative practices or do not specify unequivocal shared artifacts. There is a lack of available methods to help answer the question: How could a software organization collaboratively perform SPL scoping activity?

Although there are approaches for guiding decision-making software engineering activities in a collaborative way [13], these approaches are general proposals that do not describe elements of the method engineering (work products, roles, and tasks) and none of these approaches is specific or focuses on scoping an SPL. Although scoping an SPL is a decision-making activity, this involves analyzing, discussing, modeling, and validating the SPL scope, one of the most relevant artifacts from business, management, and technical viewpoints. Moreover, there are diverse ways for representing and building the scope, and its use in the subsequent steps lacks clarity [14,15]. These limitations mean that the communication and collaboration of stakeholders are not supporting, in an expected and replicable way, the scope definition; that is to say, they do not know who, where, when, and how collaboratively to build the scope. For achieving real collaboration, it is necessary to structure activities that convey communication among different participants in a group [16], including concepts and relationships of the SPL scope.

In this paper, we present a Collaborative Method for Scoping an SPL (CoMeS-SPL) and report the experience of its application in a software company. This method includes an ordered set of tasks and artifacts guiding, step by step, the scope definition to the domain engineering team. At the same time, it promotes and emphasizes collaboration among the participants. This experience of applying the CoMeS-SPL method was planned and executed following the case study research method, evaluating the team collaboration and resulting scope usefulness in industrial settings, combining qualitative and quantitative approaches. The study was carried out in Sunset Software House, a small Colombian software company, where software engineers were considering to produce a group of products from a reference product previously developed by the company. This project was a new experience for the company because stakeholders must have thought about a family product instead of a single product. Additionally, the method implied changes in the activities and roles for those that usually participate in scoping a single product.

The rest of this paper is organized as follows. Section 2 discusses similar proposals related to SPL approaches that consider collaborative practices. Section 3 presents our proposal, CoMeS-SPL. Section 4 presents the methodology used to carry out CoMeS-SPL in a company and Section 5 and describes this experience. Sections 6 and 7 present results of this experience and the company evaluation, respectively. Section 8 concludes this paper.

2. SPL Scoping Approaches That Considering Collaborative Practices

Collaborative Engineering (CE) is an approach that supports the design and implementation of collaborative processes from patterns in order to enhance communication, cooperation, and team awareness. The collaboration implies teamwork, meaning that multiple individuals combine their knowledge and efforts to achieve a team goal [17,18]. CE uses patterns of collaboration to classify group activities [17–19] and also proposes design blocks, named thinkLets [20]. ThinkLets are used for the design or re-design of tasks to structure them in such a way that communication and collaboration is achieved among the different participants based on collaboration patterns [20]. A thinkLet is a scripted technique where the specification task of process describes inputs, steps, and outputs [20]. ThinkLets are construction bricks used by process designers to build processes and each task composing it. ThinkLets have become reusable units to define predictable and repeatable tasks that involve teamwork [18,19].

Several literature reviews have reported, to date, different approaches for SPL scoping in journals or closed to SPL conferences [6,15,21–23]. Many of the approaches for SPL scoping highlight the importance of involving participants with diverse knowledge and different work experiences [21,24]. Although many of the proposals and studies about SPL scoping highlight the importance of involving participants with diverse knowledge and expertise [21,24], most of the approaches only mentioned participants in a general way or only named the roles or provided brief descriptions [11,12]. Some authors mentioned that the benefits obtained by including collaborative practices in scoping activity [11,12,25,26].

Some proposals have analyzed the effect of communicative factors on SPL scoping, and some of these have focused on improving communication and collaboration among participants [11,12,25]. However, one of the limitations that has been reported in the different scoping reviews is the lack of formality in this activity, evidenced by little clarity and disagreement on how the scope is created and represented, and that causes the obtained scope not to be so used as input artifact in the following activities the development process of a SPL [21,23]. If the participants do not know how the is composed and represented of a product line scope, the communication and collaboration among them will be difficult.

Table 1 presents a comparison among SPL scoping approaches that include scoping type, collaborative practices, considering additional indicating the level with which the proposal addresses elements such as practices, roles, and artifacts, as well as their availability. Product portfolio and functional domains refer to the set of products of the SPL, early identifying commonalities and variabilities. The last approach included in the table corresponds to the CoMeS-SPL method that we propose to define the scope of an SPL in a collaborative way and which is described in the next section of this article.

One of the items considered in Table 1 is the scope type or types covered by SPL scoping approaches that use collaborative elements. The SPL scope has been classified into three types of scopes: product portfolio scoping, domain scoping, and asset scoping [21,27]. Product portfolio scoping identifies the particular products that belong to the line as well as the features that each of these products must provide. In domain scoping, the identified features are grouped by functionalities and, finally, in asset scoping, particular and reusable components (functional parts of the product line) are identified [15,21,27].

Some of the artifacts proposed as work products in the scope approaches in SPL are the product map, product portfolio, or matrix of features and products, which begins to be carried out in the first tasks of the portfolio scope and is validated as it does refining in the tasks of the other two types of scope. In this artifact, it can be observed which features are common to all or most of the products belonging to the line and which are specific or particular features. This classification makes it possible to identify the functionality domains, the particular and reusable components (functional parts of the product line) as well as they will be the basis for later raising the points of variability [15,27].

Table 1. Comparison among SPL scoping (Software Product Line Scoping) approaches that considering collaborative practices.

Approaches		Characteristics
People Oriented Approach	Type of scope	Scope of product portfolio describes in which aspects of the scope it influences
	Roles	Description, it does not include template
	Artifacts	Collaborative practices applied to a specific artifact
	Additional practices	Partial in a published article
Collaborative Approach to Scoping	Availability	
	Type of scope	Scope of product portfolio Scope of functionality domains
	Roles	general description of their interests and participation
	Artifacts	Only names
RiPLE-ASC (RiPLE-The RiSE Process for Product Line Engineering) (Rise-Reuse in Software Engineering) (ASC-Agile SCoping)	Additional practices	Collaborative practices Agile practices
	Availability	Partial in a published article
	Type of scope	Scope of product portfolio Scope of functionality domains
	Roles	description of what they contribute to scoping
RiPLE-SC (RiPLE-The RiSE Process for Product Line Engineering) (Rise-Reuse in Software Engineering) (SC-Agile SCoping)	Artifacts	Description, templates for some artifacts
	Additional practices	Agile practices, Scrum
	Availability	Partial in a published article Doctoral thesis in web repository http://repositorio.ufpe.br Accessed on 10 July 2019
	Type of scope	Pre-scoping Scope of product portfolio Scope of functionality domains Scope of reusable assets
CoMeS-SPL (Collaborative Method for Scoping an SPL)	Roles	Only mentioned
	Artifacts	Brief description
	Additional practices	Agile practices Partial in a published article Doctoral thesis in web repository http://repositorio.ufpe.br Accessed on 10 July 2019
	Availability	
CoMeS-SPL (Collaborative Method for Scoping an SPL)	Type of scope	Scope of product portfolio Scope of functionality domains
	Roles	description of what they contribute to scoping
	Artifacts	Description and included template
	Additional practices	Collaborative practices Partial in a published article http://repositorio.unicauca.edu.co Accessed on 17 March 2021 Website http://comesspl.com Accessed on 12 May 2021
CoMeS-SPL (Collaborative Method for Scoping an SPL)	Availability	

3. CoMeS-SPL Method

This section presents CoMeS-SPL, Collaborative Method for Scoping of Software Product Lines (CoMeS-SPL), built using Method Engineering guidelines and Collaboration Engineering practices in its specification.

The CoMeS-SPL method guides the participants involved in scoping through steps, roles, and well-defined artifacts. The steps encourage and focus on collaborative tasks to obtain a set of tangible, descriptive, and multiview SPL scope artifacts. Figure 1 shows the hierarchical structure of the CoMeS-SPL method. The top node represents the main activity goal: “To define the software product line scope”, and this node is broken down into six sub-objectives, where four of them correspond to the three scope types defined [27]. From left to right, the children’s nodes that come from “To identify product and features” until “To define the assets for reuse” correspond to activities of the scope types. The children nodes that come from the two nodes located at the extreme left and right correspond to communicative objectives. The lower level represented with orange nodes corresponds to the tasks and sub-tasks of the proposed method.

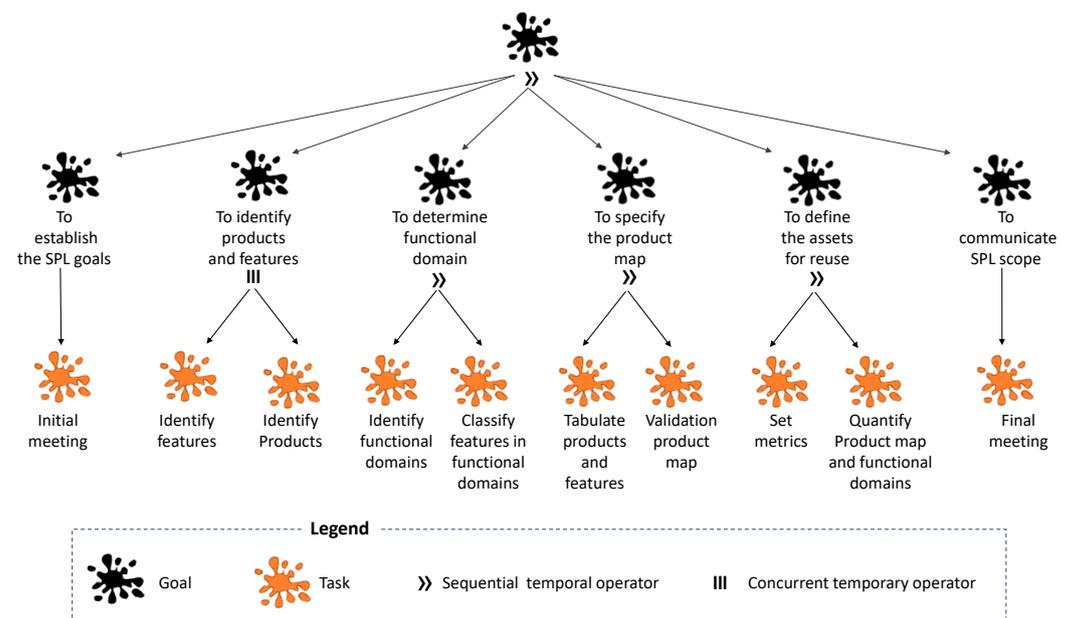


Figure 1. Hierarchical structure of CoMeS-SPL (Collaborative Method for Scoping an SPL).

Table 2 presents the scope type, goals, and tasks of the CoMeS-SPL method, as also indicated each task or sub-task, the collaborative patterns, and the applied thinkLets. The CoMeS-SPL method is formed by tasks (units of work), roles, and artifacts (work products). The CoMeS-SPL method is composed of 10 tasks, and the first two tasks have been divided into sub-tasks in such a way that each task or sub-task is associated with a collaborative pattern and a thinkLet, a one-to-one relationship. The thinkLets associated with each task or sub-task were selected from analyzing and identifying the type of necessary interaction among the participants, strengthening the communication among them and combining their efforts to achieve the objective of the task. For this reason, some tasks of the scope were divided into sub-tasks in such a way that for each type of necessary interaction among the participating people, the appropriate thinkLet will be used considering a one-to-one relationship among sub-tasks and thinkLets.

CoMeS-SPL was modeled using the notation for the modeling of collaborative processes proposed by Solano et al. [28,29], which is an extended proposal of the notation HAMSTERS (Human-centered Assessment and Modeling to Support Task Engineering for Resilient Systems) [30]. The HAMSTERS notation offers a set of appropriate elements to complement the graphical representation of the FPM (Facilitation Process Model) which complements the graphical representation including elements that allow representing

the concurrence and interaction between activities or tasks to achieve an objective [31]. This notation provides graphic representation of different elements of a process, as they are such as tasks, relationships among tasks, input or output artifacts, and the workflow. Additionally, this notation also provides graphic representation of collaborative elements as participating roles, collaboration patterns, the used thinkLet, and the task steps.

There are different modeling languages that can be used in the specification of a process or method, and one of the options is SPEM (Software Process Engineering Metamodel) that facilitates the elements and concepts necessary for the modeling, documentation, presentation, administration and exchange of processes. and software development methods [32]. SPEM is a language proposed specifically to model software engineering processes and methods and consider the basic elements of the specification such as role, work product, and task; however, SPEM does not allow modeling specific elements of collaborative engineering such as which is the collaborative pattern in a task or the participating roles [32]. De Vreede and Briggs propose a Facilitation Process Model (FPM) that uses three graphic symbols to represent the flow of a process or method. This notation allows identifying the activity or task, the collaborative pattern, and the thinkLet instantiated in the activity, and FPM also allows modeling the flow of the process or method [33]. However, this notation proposal falls short in the inclusion of other important elements in a collaborative method such as the roles involved in an activity or task. This notation was used in the proposal "Collaborative Approach to Scoping" [12]).

Figure 2 presents the CoMeS-SPL workflow, complementing the general vision of the method provided by Table 2, allowing us to observe the order in which the proposed tasks and sub-tasks should be performed.

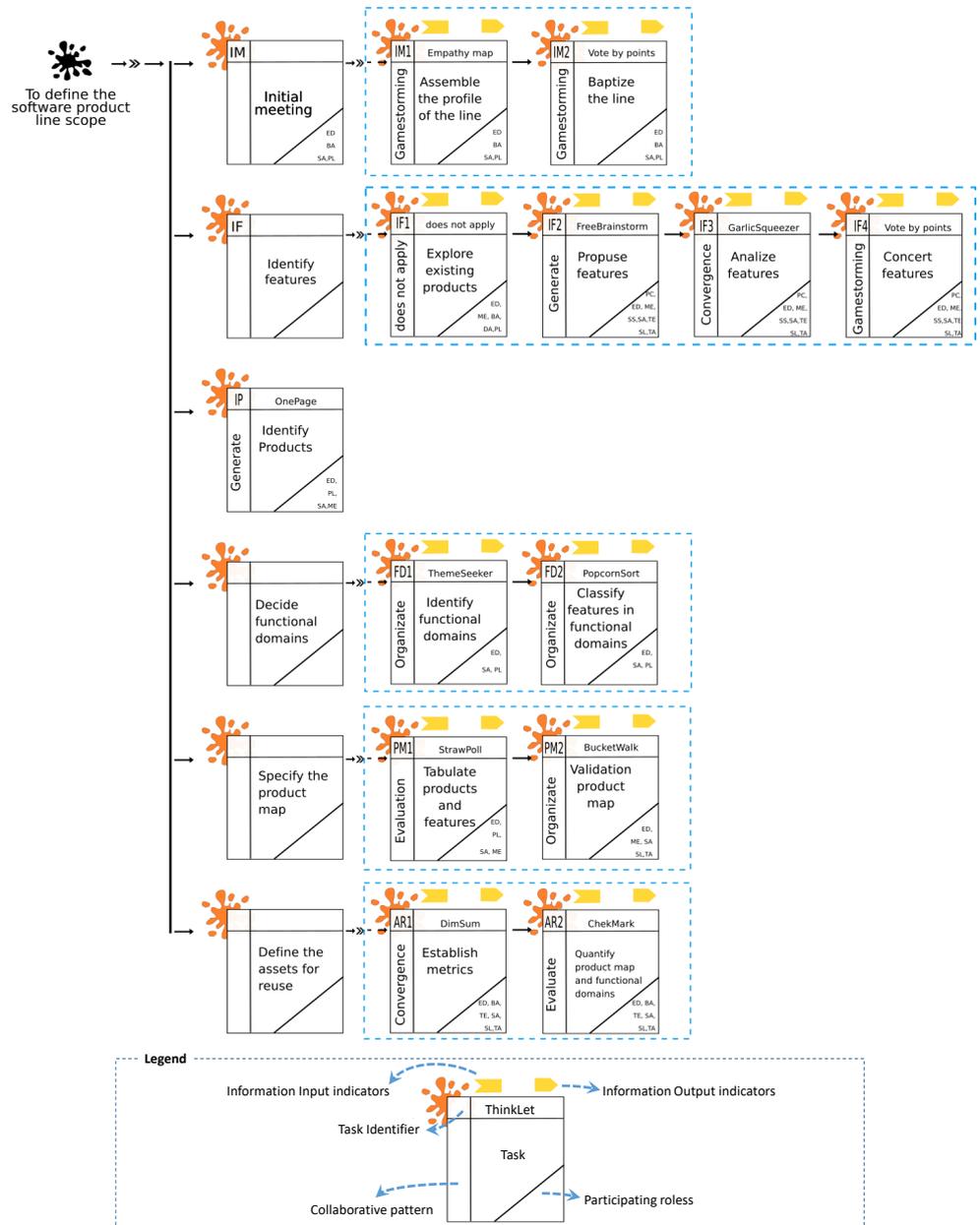


Figure 2. CoMeS-SPL (Collaborative Method for Scoping an SPL) workflow.

Table 2. CoMeS-SPL (Collaborative Method for Scoping an SPL) tasks.

Sub-Goal	Scope Type	Task	Sub-Task	Collaborative Pattern	ThinkLets
To establish the SPL goals		Initial meeting	Assemble the profile of the line	Gamestorming	Empathy map
			Baptize the line	Gamestorming	Voting by points
To identify products and features	Product Portfolio Scoping	Identify features	Explore existing products	Does not apply	does not apply
			Propose features	Generate	Free Brainstorm
			Analyze features	Convergence	Garlic Squeezer
			Agree on features	Gamestorming	Voting by points
To determine functional domain	Domain Scoping	Identify functional domains		Organize	Theme Seeker
			Classify features in functional domains	Organize	Popcorn Sort
To specify the product map	Product Portfolio Scoping	Tabulate products and features		Evaluation	StrawPoll
			Validation product map	Evaluation	Bucket Walk
To define the assets for reuse	Asset Scoping	Set metrics		Convergence	DimSum
			Quantify product map and functional domains	Gamestorming	Voting by points
To communicate SPL scope		Final meeting		Gamestorming	Matrix who-what-when

The complete description of the CoMeS-SPL method is available at the website (<https://comesspl.com>, accessed on 12 May 2021), where we can find its elements such as tasks, roles, artifacts, and artifact templates. For each task, the method describes the objective, the participating roles, the input artifacts, the output artifacts, and the steps or sub-tasks, as well as the collaboration patterns and associated thinkLets.

This article presents one of the sub-tasks from the CoMeS-SPL method. The second task of our proposed method, called “Identify features”, belongs to the scope type product portfolio. This task is broken down into four sub-tasks shown in Figure 3. The used modeling language of the third sub-task “Analyze features” is an extension of HAMSTERS language (see Figure 4). In this modeling language, a task, represented by a rectangle, is divided into five sections: in the middle is the task’s name, in the upper part is the task code (IF3), on the right slot the associated thinkLet (GarlicSqueezer), on the left and vertically the used collaborative pattern (Convergence), and on the lower right corner the abbreviations of the participating roles. Above the rectangle are the input and output artifacts of the sub-task. At the right side of the rectangle, the steps to be carried out are found, each one indicating the corresponding type of collaborative action. Table 3 illustrates the specification of this sub-task.

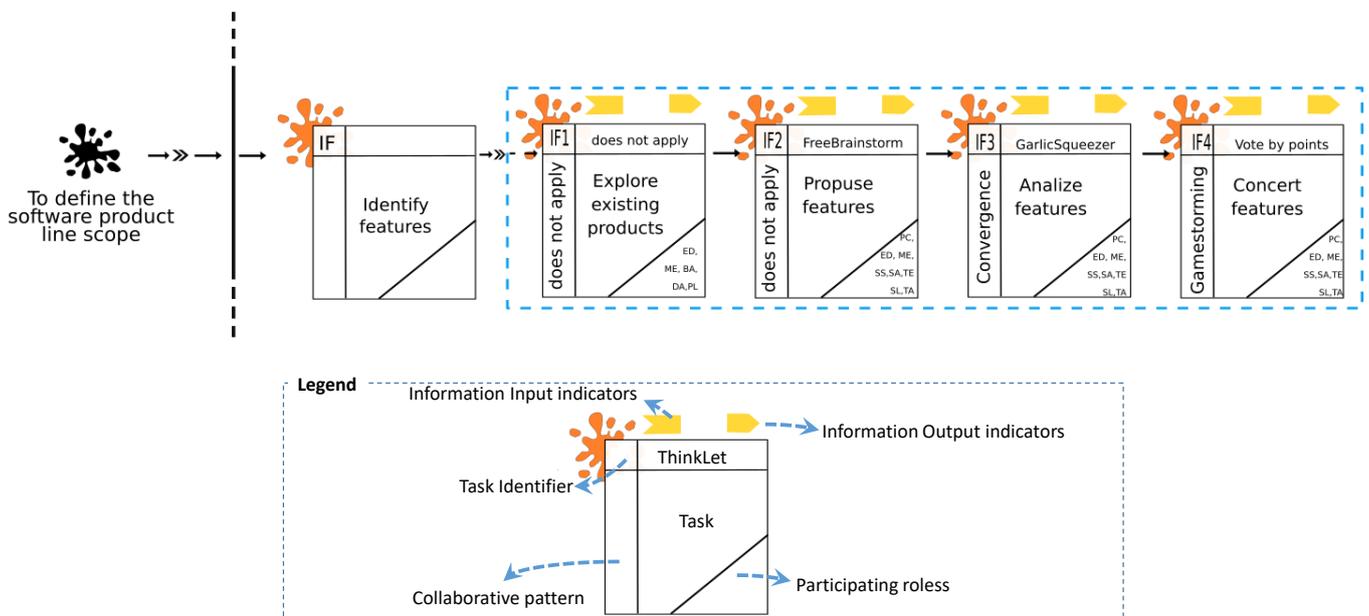


Figure 3. Task identify features.

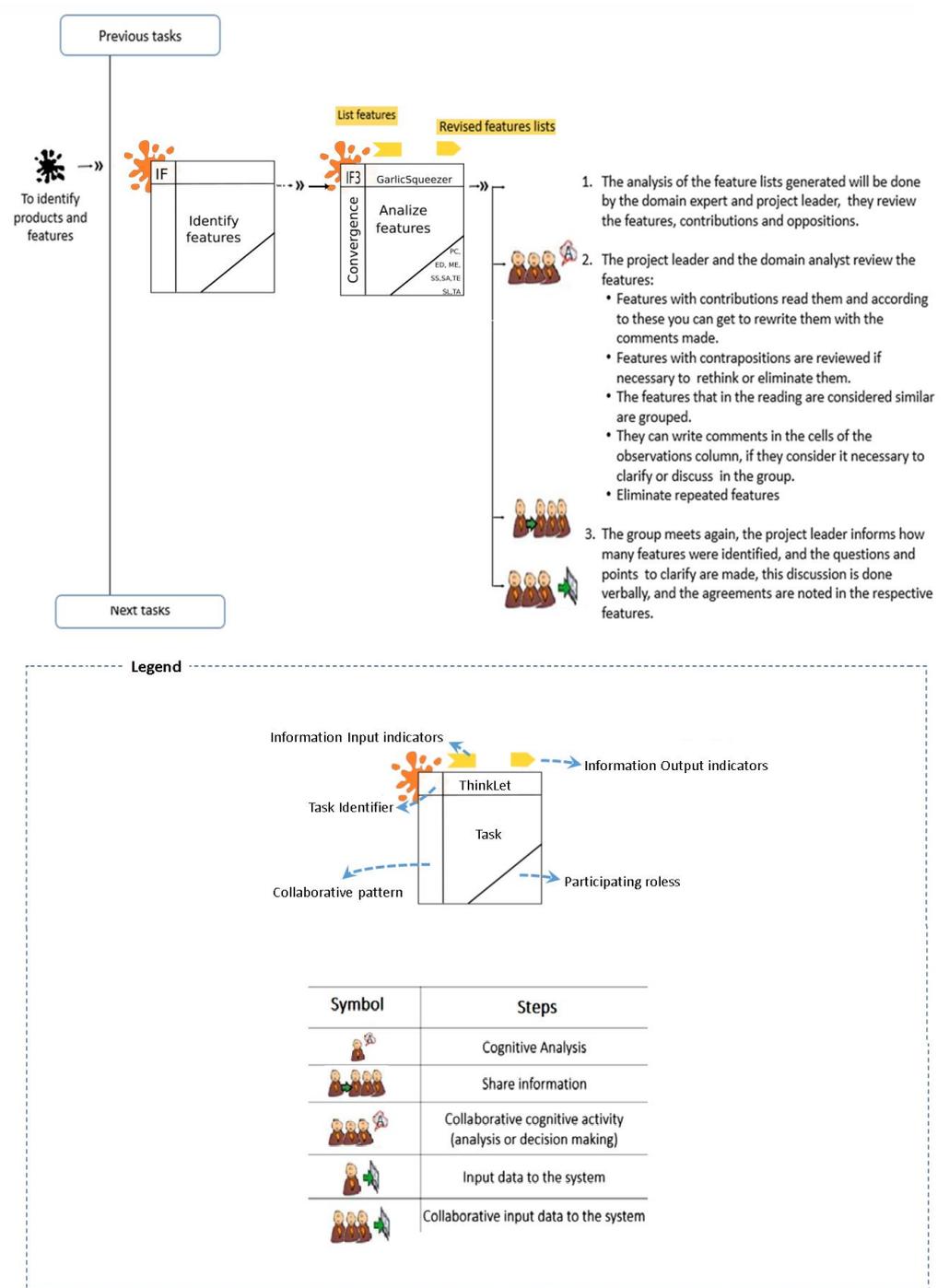


Figure 4. Sub-task analyze features.

Table 3. Analyze features.

Sub-Task	Analyze Features
Task	Identify features
ID	IF3
Description	This task seeks to filter features lists, contributions, and contrapositions to achieve a clean list and an agreement by the team
Collaborative pattern	Convergence
ThinkLet	GarlicSqueezer
Mandatory roles	Expert domain of application (ED) SPL project leader (PL) Software architect (SA) Marketing expert (ME)
Optional roles	Business administrator (BA) Potential customers (PC) Sales staff (SS) Technical expert (TE) SPL expert (LE) Teamwork advisor (TA) Domain analyst (DA)
Input artifact	List features
Output artifacts	Revised features lists
Steps	<ol style="list-style-type: none"> The analysis of the feature generated lists will be conducted by the domain expert and project leader, they review the features, contributions and oppositions. The project leader and domain analyst review: <ul style="list-style-type: none"> Features with contributions, read them, and according to these they can rewrite them with the made comments. Features with contrapositions are reviewed if necessary to rethink or eliminate them. The features that are considered as similar are grouped. They can write comments in the cells of the observations if they consider it necessary to clarify or discuss in group. Eliminate repeated features. The group meets again, the project leader informs how many features were identified, and the questions and points to clarify are made, where this discussion is done verbally, and the agreements are noted in the respective features.
Rules	During step 2, only the project leader and the domain expert remain in the space in order to make a quick analysis; the more people involved, the longer the discussion becomes.

3.1. Participating Roles in CoMeS-SPL

Collaborative work requires joining the efforts of a team of people to achieve a common goal, so it is important that the roles are well defined so that each participant knows their functions and how they contribute to the activity. The identified roles in CoMeS-SPL are:

- **Domain expert:** This participant is usually an external advisor from the companies or organizations of the target domain. He/she provides knowledge about the application domains, context information, customers' needs, related products, and associated regulations from the companies or organizations of the target domain. It is a mandatory role.
- **Business administrator:** This role belongs to the administrative unit of the software producing company (a manager or administrator). It is a mandatory role.
- **Marketing expert:** This role represents the business concerns of the company against the production of an SPL. This person can belong to the unit of marketing and sales of the production company. It is a mandatory role.
- **Software architect:** A software architect is responsible for the high-level design and strategic planning of software. He/she is usually a person with the greatest experience of the development team with good communications and negotiation skills. It is a mandatory role.
- **SPL project leader:** The SPL leader is the person who manages and controls the resources assigned to the SPL project, with the purpose that the plans may correctly be fulfilled in the estimated time. This person provides a strategic vision, a business approach that considers aspects such as cost, time, and resources of the project. It is a mandatory role.
- **Potential customers representative:** This role represents the possible customers of the products to be developed as part of SPL. The objective of the customer's participation is to identify real needs. It is an optional role.
- **Sales staff:** The person in this role knows and collects all possible information about the sales of the company's products, competitors, and aspects of the customer's purchasing processes. This is an optional role.
- **Domain analyst:** This role involves interaction with potential users and experts in domains to establish the SPL scope. The person in this role gives an overview of the target domain and determines the common and variable features from the products that belong to the SPL and their restrictions. This role is optional.
- **Technical expert:** He/she knows different tools, techniques, environments, and programming languages. He/she provides technical knowledge of the products and helps the team evaluate the available technical options. This role is optional.
- **SPL expert:** This role is necessary if the leader of SPL has no experience in the development of SPL, he/she provides knowledge in the planning and development of an SPL initiative. This role is optional.
- **Teamwork advisor:** He/she knows about collaboration techniques and practices, which will help in the execution and management of the method, in the solution of impediments, and in the interaction among the participants. He/she supports and encourages communication and collaboration among the participants. This role is optional.

3.2. CoMeS-SPL Artifacts

CoMeS-SPL specifies the artifacts that composed the scope of an SPL. The artifacts were identified and compared in the base approaches used in the definition of the proposed method. The task's description includes the steps of performing transformations from input to output artifacts, the roles, and the type of required collaboration. A CoMeS-SPL task defines a systematic transformation from one or more input artifacts into a defined output artifact (output). The traceability and consistency among input and output artifacts were reviewed using a manual check and carried out in the empirical studies. The scope of

an SPL includes seven artifacts: an SPL profile, a features list, a products list, a product map, a functional domain list, a metrics list, and quantified a product map.

3.3. Formulation and Evolution of the CoMeS-SPL Method

The CoMeS-SPL method formulation combines software product lines engineering and collaborative engineering using method engineering for description of its elements and the traceability among them. The formulation considers iterations that would allow increasing the specification of the method, and at the end of each iteration an empirical evaluation was performed. The empirical experiences allowed for contrasting the theoretical elements obtained from the review of the literature and the method components with the practical elements obtained from the observation and to measure the results obtained in each case. The construction of method was iterative and incremental, and each of the empirical studies carried out corresponds to an evaluation of a version of the method, the findings that constituted the starting points for the new versions, indicating the shortcomings to be improved for which it was necessary to resume the construction cycle of the method.

Four iterations were carried out in the formulation of the CoMeS method. In the first iteration, an exploratory study was carried out that allowed to identify which tasks of the SPL scope require collaborative practices to improve communication and interaction between the participants.

The second iteration was an comparative exploratory study in which one of the methods used as a reference was compared with the proposed method in which collaborative practices were included.

The evaluation carried out in the third cycle of the formulation of the method was a workshop with SPL experts where the usefulness of the method to define the scope of an SPL was evaluated from the point of view of the participants as well as the level of collaboration reached by the work team. Finally, the evaluation carried out in the fourth iteration is the one presented in this paper.

4. Materials and Methods

Empirical research methods have been used in the SPLE to evaluate how feasible or efficient a new technique, method, or tool is. These methods allow identifying weaknesses and strengths of a new proposal in a specific time and a real context. Studies in industrial settings allow researchers to evaluate new proposals and give them realistic feedback, making new proposals in SPLE less distant and alien to the practice [34]. Nowadays, there is great attention on the interaction among researchers and the software industry because this relationship is crucial for the production and communication of knowledge [35]. The interaction among the academic-investigative sector and production sectors should be a permanent activity in which they communicate the findings of the research initiatives and proposals to all those interested [35].

Jahn et al. [36,37] propose a model that facilitates collaboration and sharing knowledge among researchers and extra-scientific actors. This transdisciplinary model relates societal problems with scientific problems; it integrates different visions to produce new knowledge and contributes to both societal and scientific progress.

From the viewpoint of academia and research groups, there is a great interest to exchange knowledge and experiences with the industry that allow enriching research ideas. Similarly, companies have an interest in improving their processes and products by using innovative ideas. However, this interaction is complex since it requires one to be able to identify common points that lead to shared goals, and it is also necessary to coordinate their schedules, efforts, and costs when an investigative initiative is going to be carried out. A collaborative approach among researchers and people from the productive sector enables spaces to validate research initiatives as long as enterprises can innovate improving aspects such as the productivity and competitiveness of software. The experience presented in this paper is an empirical model about the collaboration among researchers and industrial

actors from past experiences of our research groups of the last ten years but applied from the last four years. This model is depicted in Figure 5.

The first stage is confidence close up among the parts to know interests, problems, and research opportunities. In this stage, researchers use divergent and convergent collaborative patterns and techniques such as brainstorming, prioritization; actors select the more valuable initiatives. The research groups joined three software enterprises prioritizing some ideas related to software architecture, maintenance, testing, and reuse. The research groups started two software product/process line projects answering the three companies because the common difficulty was the lack of clarity about their reuse scope and strategies. In the second stage, the research groups and companies performed several joint activities. For instance, expert talks (architecture, software product lines, and software process lines), collaborative working meetings (using several collaboration patterns), controlled experiments, and workshops about collaboration and software product lines. In the third stage, the CoMeS-SPL case study took place in the Sunset Software House company. It was planned and executed as a pilot project allowing research groups and the company to work concretely around a mutual beneficial activity. Thus, the research groups evaluated CoMeS-SPL in industrial settings. As the main result, Sunset Software House company understood, from multiple viewpoints, its reuse opportunities scoping its first software product line, re-engineering from their product named CORA platform.

Among the main research results are papers published in related conferences and workshops [15,38–41]. From the Sunset Software House company viewpoint, the CoMeS-SPL method was adopted, particularly the the scope practices, not just for SPL development but for software development in general. Currently, the research group keeps the collaboration with these two companies. We are working on two new goals: the first goal is to build a predictive model for software testing, and the second one is to define a decision-making model to offer maintenance as a service.

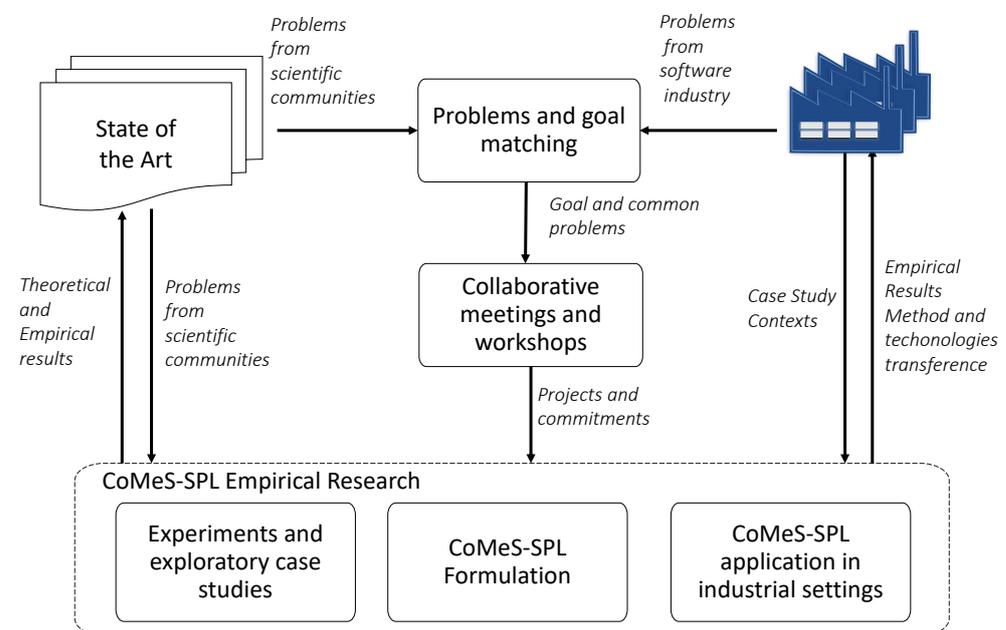


Figure 5. Empirical research approach.

A case study is an empirical inquiry that allows observing the proposed scoping method within its real-life context [42,43]. We used this research method for building (exploratory cases) and evaluating the CoMeS-SPL method (confirmatory case). Our study objective was to measure the effectiveness of the CoMeS-SPL method scoping the first SPL in the context of a small software company and, additionally, to evaluate the utility of the obtained scope from an industrial perspective.

The company goal was to how to build an SPL from an existing product. In other words, a product family reusing a technical infrastructure but also considering a potential market. Therefore, the company analyzed different aspects from previous projects of a single product or customer development. The study design asks the following research question:

Can CoMeS-SPL enable suitable stakeholder collaboration to build a usefulness scope of the first SPL of the Sunset Software House company?

We performed the case study to find an answer to our research question. A development team of the company worked to characterize the first SPL scope using the CoMeS-SPL method. The study focused on analyzing team communication, interaction, and collaboration. In addition, it focused on evaluating the usefulness of the obtained scope.

4.1. Selection of the Case

We selected Sunset Software House company because it fulfills the following characteristics:

- It is a software company
- The number of employees is greater than 10
- It has more than three years of experience in software production
- It has one or more groups of employees dedicated to software development
- It has a person with specific roles as a marketing expert and/or a sales person that belongs to the employee group
- It has a person with responsibilities related to software architecture that belongs to the employees' group
- It has a project in execution or about to start related to the production of a set of products with common elements
- The most important is it has the availability and interest to participate

4.2. Company Context

Sunset Software House S.A.S (<https://sunsetswh.com>, accessed on 12 May 2021) is a technological solution company with nine years of experience in the market, during which they have developed approximately 53 projects. This company has twenty employees and is located in the city of Popayán.

The Sunset Software House company offers three types of services: outsourcing services in the development of software components to other software producing companies; custom development of web and mobile applications; and the company implements e-commerce stores.

The Sunset Software House company developed a software tool for organizational management (CORA), which was initially conceived as a unique in-house product. However, during its development, the responsible team and the company managers observed that some CORA modules could integrate new products for various potential customers.

Therefore, CORA becomes a reusable platform. However, the company's experience focuses on software outsourcing for other companies. The company had many doubts about starting and leading the reuse initiative because it focused on business customization. The Sunset Software House company is interested in establishing chances to work on the software product line approach, by identifying possible products, customers, and functionalities. In this context, the company considered defining the scope of its potential SPL. The CORA platform was developed in a modular way, which facilitates its maintainability and adaptability. Hence, CORA platform can be a potential base to develop other products. The CORA platform is composed by the modules contracts, projects, activities, human and strategic management, and notifications.

A first product called IAS was developed on the basis of CORA platform and uses two of its modules of managing the users. The products derived from the CORA platform are focused on private Colombian companies (Mypimes) that manage public or private projects and that need to organize their processes (it is not oriented to startups).

CORA has always been called CORA, but with different connotations. Initially, when it was developed as an internal product, the product looked for the Control and Activity Register, and the name corresponds to an acronym using the first letters of those words. When the company realized that this product could be the heart or the basis for other products, the name was preserved because it was the heart of other products, since heart in Spanish is written corazón. The names of CORA derivative products are decided with each of the customers. To differentiate the initial CORA product from the platform or core for the derivation of other products, when we refer to this second one, we will call it CORA platform.

One of the approaches for the adoption of the SPL is an extractive approach, in which the products of the line are defined from existing products developed by the company previously, the commercial objectives of the organization, and the target market of the line [44]. The extractive SPL adoption strategy for companies is one of the most used in practice, which reuses one or more existing software products so the product line could reduce the risks, costs, and efforts involved in starting from scratch to raise an SPL [45]. This was the strategy that was carried out with the sunset company, who started CORA as an internal product and then used it as a basis to develop a product for another company, and realized that this first internal product could be used as a basis to derive a set of specific products. That was the common point that was found between the company and the researchers.

4.3. Measurement Design

To evaluate the CoMeS-SPL method through the application of a case study, in the designing of it we proposed indicators, metrics, and instruments to help understand and analyze the proposed method. The metrics target three measurement entities: CoMeS-SPL method, team, and the resultant scope.

The case study allows us to evaluate CoMeS-SPL in the software industry context, defining several metrics to evaluate the method effectiveness and its easy understanding and application by the participants. This measurement includes the task description clarity, completeness, and traceability from the perspective of collaboration engineering. While measuring the stakeholder interaction, we obtained the collaboration level achieved by the team. Moreover, the evaluation establishes the correctness and usefulness of the SPL scope.

4.3.1. Metrics to Evaluate the Proposed Method

Perceived method effectiveness is understood as the ability of a scoping method to produce useful outputs in relation to the expectations and needs of the company and, therefore, method effectiveness implies that the tasks being performed in the method are adequate to produce the desired and useful results [46].

The effectiveness of a scoping method refers to the usefulness of the method outputs in relation to the expectations and needs of the company and, therefore, method effectiveness implies that the tasks being performed in the method are adequate to produce the desired results [46]. In order to measure the effectiveness of CoMeS-SPL, three qualitative metrics were considered: perceived ease of use, perceived usefulness, and intention to use.

- Perceived ease of use (PUE): This variable represents a perceptual judgment of a participant about the required effort to learn and use the CoMeS-SPL [47].
- Perceived usefulness of the method (PUM): The degree to which a participant considers the description of the CoMeS-SPL components has enough clarity and facilitates the completions of tasks and the construction of the scope artifacts. The perceived usefulness is a dependent variable, measured with a survey instrument after applying the method. The instrument design is based on the proposals of [47,48].
- Intention to use (ITU): The extent to which a person intends to use the CoMeS-SPL method the next time performing scoping activities is required. This variable is used to predict the likelihood of acceptance of a CoMeS-SPL in practice [49].

4.3.2. Collaboration and Participation Achieved

The term effective interaction or effective participation is used in collaboration processes to refer to the interactions among participants where they cooperate and have a well social atmosphere, group involvement, opinions and efforts of the other members are considered [50]. We measured the level of the effective participation in CoMeS-SPL according to the participation and cohesion degrees proposed by [50].

- Participation degree (PD): The intensity with which the individual members participate in the group's activities and their proactive commitment [50]. For this study, there are no previous values that allow comparing whether or not the degree of obtained participation is greater when applying the CoMeS-SPL method compared to another.
- Factor artifact history (HF_{ar}): This factor indicates the balance in the number of contributions to an artifact made by the different participants. It is calculated with the standard deviation and the average value of the contributions of participants in the completion of an artifact. This value is in the range [0, 1]: values close to 0 indicate that the contributions were made only by some of the actors, while values close to 1 express a higher degree of balance in the contributions of the participants [51]. The following formula is used to calculate the (HF_{ar}).

$$HF_{ar} = 1 - \frac{stdev(C_{ar})}{med(C_{ar})}$$

The proposal [50] of this indicator is known as equal participation degree (EPD), and in an 'effective' collaborative group all members should participate to a similar degree without any participant having monopolizing behavior. However, we considered the same weight for each role involved in the construction of an artifact. Similarly, we used the same weight for each artifact in the scope.

- Collaboration factor (CF): This describes the level of relative participation of the members of a collaborative activity [51]. It is based on the type and size of contributions or events made by the participants in the completion of the artifacts that constitute the scope activity. The result allows inferring elements of cooperation, such as the evolution of group performance of time, as well as the effectiveness of the collaboration [51]. The factor of collaboration (CF) is calculated as the average value of the collaboration factors of the artifacts.

$$CF = \frac{\sum_{i=1}^n HF_{ar_i}}{n}$$

- Perceived collaboration (PC): This is the degree to which participants in SPL scoping activity believe that using CoMeS-SPL encourages collaboration and effective communication among them.

4.3.3. Range of Scope Correctness

We evaluated the CoMeS-SPL method in terms of the correctness and usefulness of obtained scope artifacts.

- Perceived usefulness of scope obtained (PUSO): The degree to which a person thinks the defined scope will be used in the following activities of the product line development and will allow (or not) decision making about the production of the product line. The perceived usefulness is a subjective and dependent variable, and for measuring this variable an instrument was used that was supported in the proposals of [47,48].
- Scope usefulness (SU): The degree to which the scope is used for decision making about SPL adoption and cost estimation.
- Range of well-defined scope (RWS): We assessed the scoping artifacts of the CoMeS-SPL method in terms of completeness and correspondence with the requirements. Furthermore, we evaluated the value of these artifacts for decision-making regarding the SPL's production and the subsequent development stages.

Table 4 summarizes the indicators, metrics, and methods for data collection and instruments considered in the design of the case study in order to evaluate the CoMeS-SPL method. This study case combined qualitative and quantitative data to achieve a better understanding of the studied phenomenon.

Three data collection methods were applied in this study, namely documentation analysis, observation, and survey. Additionally, the documents related to the artifacts produced in the scoping were analyzed. To measure the perceived ease of use (PEOU), perceived utility (PU), intention to use (ITU), perceived collaboration (CP), and perceived usefulness of scope obtained (PUSO) variables, we designed an instrument (survey) from the one proposed for the evaluation of the requirements modeling methods [49] and the Technology Acceptance Model [47]. The instrument has defined a set of items to measure each of the perception variables. The evaluation of these elements used a Likert scale of 6 points (the lowest value or disagreement 1, and the highest value or agreement 6). The items were randomly organized in the questionnaire to prevent the participants from giving systematic answers. Four open questions were included in the instrument to obtain suggestions and observations regarding the proposed method. In order to measure the variables participation degree (PD) and factor artifact history (HFar), these were measured by counting the contributions of each of the participants in the construction of the artifacts that make up the scope according to the CoMeS-SPL method. The metrics' scope usefulness (SU) and range of well-defined scope (RWS) were evaluated by the architect and the manager of the company.

Table 4. Parameters of the evaluation.

Indicators	Metrics	Methods for Data Collection	Instruments
Perception of the method effectiveness	Perceived use ease (PUE)	Direct	Survey
	Perceived usefulness of method (PUM)	Direct	Survey
	Intention to use (ITU)	Direct	Survey
Collaboration and Participation achieved	Participation degree (PD)	Indirect	Scope artifacts produced
	Factor artifact history (H Far)	Indirect	Scope artifacts produced
	Collaboration factor (CF)	Indirect	Scope artifacts produced
	Perceived collaboration (PC)	Direct	Survey
Range of scope correctness	Perceived usefulness of scope obtained (PUSO)	Direct	Survey Survey
	Scope usefulness (SU)	Direct	Scope artifacts produced survey
	Range of well-defined scope (RWS)	Independent	Scope artifacts produced

5. Execution of the Study

Initially, it is necessary to know the company, its production process (some of its activities and participating roles), the products they develop, its customers, its projects and expectations, and for this we conducted a survey of the CEO of Sunset Software

House company. The case study was framed in a company project. Usually, the company determines the scope of a new project with the participation of the project manager and technology director, two computer engineers were dedicated to specifying and detailing a product for a specific customer and, usually, the product belongs to a different domain of the knowledge area engineers. A similar process occurred when CORA was considered as a product for internal use. When the company decides that the CORA project is a platform for software reuse and the development of a set of products and defines the scope using the CoMeS-SPL method, one of the changes is the linking of other roles. In this study case, six company people participated: the CEO, the project manager/product manager, the innovation manager, the marketing expert, the technology manager, and an analyst/developer of CORA project. There was also the external participation of an expert in product lines. Table 5 establishes the relationship between the participants of the company (positions) and the roles of the method.

Table 5. Relation of company positions with the method roles.

Position in the Company	Role Proposed by CoMeS-SPL	
CEO	Business administrator	Mandatory
Project manager	Expert domain of application	Mandatory
Product manager		
Marketing manager	Marketing expert	Mandatory
	Sales staff	Optional
Technology manager	Software architect	Mandatory
	SPL project leader	Mandatory
Innovation manager	Technical expert	Optional
Analyst and developer	Domain analyst	Optional
SPL expert (external)	SPL expert	Optional
Not included	Teamwork advisor	Optional
Not included	Potential customers representative	Optional

The study was carried out for three months in the frame of one of the company's innovation projects. During these months, dedication time was approximately 65 h. Training on SPL was given to the entire company in two 3-h sections, and then section on the CoMeS-SPL method. Table 6 presents the times used in scoping tasks.

Table 6. Execution schedule of the study.

Task	Duration (Hours)
Previos rapprochement	7
Training in SPL	6
Training in the CoMeS-SPL method	3
Initial meeting	3
Specify product portfolio	24
Identify functional domains	6
Definition of the assets scoping	12
Final meeting of the scoping	4
Total time	65

6. Results and Analysis of Results

In this section, we present the findings of the case study describing the scoping activity performed within the company. The product scoping was performed to identify and review features, identify products, construct and validate the product map.

The perception metrics perceived use ease (PUE), perceived usefulness of the method (PUM), intention to use (ITU), perceived collaboration (PC) and perceived usefulness of

scope obtained (PUSO) are defined in Table 4. For this study, an instrument was designed with twenty-six items that included the five metrics proposed, each item was measured using a Likert scale from 1 to 6 points, where 6 was the highest value that participants could assign to in their assessment. The survey was answered by the participants after applying the CoMeS-SPL method.

Table 7 presents the average results that were obtained through the survey applied to the participants in the case study. These results allow affirming that for this case study:

- CoMeS-SPL was perceived as an easy and useful method, and this appreciation was reaffirmed with the intention of the participants to apply the method in future projects.
- The participants perceived the CoMeS-SPL method as useful and usable. Additionally, the participants considered that the artifact formats facilitate their use in future projects.
- CoMeS-SPL encourages and achieves the participation of multiple actors and allows them to understand the importance of each other's contribution.

Table 7. Results of perception metrics.

Category	Acronym	Metrics	Median	Standard Deviation
Perception of the effectiveness	PEOU	Perceived ease of use	5.60	0.48
	PUM	Perceived usefulness of method	5.40	0.50
	ITU	Intention to use	5.70	0.40
Correctness range of scope	PUSO	Perceived usefulness of obtained scope	5.40	0.64
Collaboration	CP	Collaboration perception	5.70	0.55

We measured the collaboration using the defined metrics: participation degree (PD), factor artifact history (HFar), collaboration factor (CF), and range of well-defined scope (RWS) through the study and analysis of the scope artifacts during their development and after this. The used tool to process the artifacts enabled teamwork as well as measuring the indicators planned. The obtained results can be seen in Table 8, where all the factor artifact history (HFar) reached values closer to 1 than 0, which allows concluding that there was a degree of balance in the participation or contributions of the different actors.

Table 8. Results of quantitative indicators of the collaboration.

Task	Artifact	Participation Degree (PD)	Factor Artifact History (HFar)
Initial meeting	SPL profile	75	0.70
Identify features	List features	1520	0.84
Identify products	List products	7	0.75
Identify functional domains	Functional domain list	15	0.69
Classify features in functional domains	List of categorized features	211	0.68
Tabulate products and features	Product map	1340	0.87
Validation product map	Product map	29	0.85

The collaboration factor achieved in the scoping activity following the CoMeS-SPL method corresponds to the value of 0.77, achieving homogeneous participation of the participating roles, indicating suitable role interaction. The survey also included some open questions as well as the conducted interviews with the participants. Regarding the

method and the tasks, they consider that CoMeS-SPL facilitates the communication of ideas, contributions, and knowledge, evidencing the importance of the different roles. Furthermore, the company participants concluded that the role of the method named “representative of potential customers” must be a mandatory role and not an optional role as has been proposed. They consider this role to be imperative to get a correct scope. The team valued the software tool used to facilitate the exchange of contributions because it helped increase the number of proposals. This tool made it possible for the participants to know and consider the opinions of the other participants. The team indicated one improvement opportunity, that it is necessary to make an agreement on the wording and description granularity of the features before starting to propose and document them. In the study carried out, not agreeing in the level description of the features increased the time and effort that were required the after task identify features.

7. Retrospective of Cooperation Experience

After the execution of the case study, we held a meeting with the Sunset Software House company to evaluate the obtained results from the viewpoint of four participants from the company: CEO of Sunset Software House company (business administrator—CoMeS-SPL role), the project manager/product manager (expert domain of application—CoMeS-SPL role), the technology manager (software architect and SPL project leader—CoMeS-SPL roles), and, the analyst/developer (domain analyst—CoMeS-SPL role). Two people did not participate in this retrospective because they were no longer linked to the company. The comments received were:

- The defined scope for a product line based on the CORA platform allowed for identifying the gaps and errors made in some of its modules. Therefore, it was evidence to continue working on this set of products.
- The scoping carried out to define CORA’s derivative product line has served in the definition of other products and projects.
- The CORA platform has significantly changed, and some of the changes were identified as necessary in the implementation of several modules. These variations caused changes in the scope, the participants think this was because of the lack of feature validation with a representative of customers, as the CoMeS-SPL method suggests.
- CoMeS-SPL is a useful method because it presents activities that help communication and collaboration in defining the scope of the project and future projects. This task must involve different roles, especially when no custom developments are being defined, but products set to offer a market segment, and not from a very technical point but from the context and the target market. Although the exercise involved roles that usually did not do so in the definition requirements, in the following stages of project development, they showed that other roles proposed by the method are necessary.
- The company actors consider that the interaction with the group of researchers was valuable, and that they would participate again in similar experiences. They suggested the use of short interactions in parallel with the other processes of the company. The efforts and times invested are manageable in the company, adapting, improving, and replicating the results.

Sunset’s CEO described the SPL as a fundamental initiative for his company from two viewpoints: engineering and business. From an engineering viewpoint, the development team learned about new development models motivating and challenging them. Furthermore, reusing resulting artifacts of other projects’ products, increasing productivity and quality, is a valuable practice. From the business viewpoint, the SPL was a contribution that allows better management of resources related to any development based on the CORA product. Thus, less effort, delivery on time, higher utility and quality of products will increase customer satisfaction, positioning the company in the Sunset’s target market.

8. Conclusions

The study, applying the CoMeS-SPL method in a small company, allowed us to work with an actual phenomenon with little control by researchers. One of the uncontrolled variables was the existence of the roles required by the method, although the company did not have some of the roles proposed by the method, and several were assumed by other employees. Potential customers, a role proposed as optional by the method, was not assumed by any person in the conducted study. The conclusion, according to the gathered results, it indicates that this role is not optional but mandatory. Another uncontrolled variable was the availability of the team people. The scoping activity was part of a project of the company planning some products for a long time, whereby urgent tasks from other projects delayed meeting dates or prevented any participants from being present in several work sections. These situations caused them to ramble from one meeting to another, forgetting the current step, the current activity, or a specific point to continue.

The use of an online tool facilitated asynchronous participation of the participants that were missing from a meeting. They made later contributions and allowed other participants to read the contributions. However, the evidence shows the importance of interactive and synchronous space to facilitate communication and collaboration, achieving a shared awareness.

This experience shows the importance of the exchange of knowledge among the participating diverse roles to define the scope of a product line that the participants concluded as an optional role proposed by the method as the potential customers representative is very important this absence was notorious to resolve doubts, so much that the group considered that it was not an optional role.

According to [52], achieving integration and communication between marketing and engineering departments is a challenge for most organizations because it requires cultural and organizational changes. Thus, defining systematic ways for functionally interconnecting these perspectives is a relevant research field. CoMeS could fulfill part of this gap. For instance, the CoMeS-SPL method could be applied together with the ISPMA (International Software Product Management Association) framework for supporting software product management, particularly for specifying the product scope and the product strategy under the SPL paradigm or any other paradigm requiring the collaboration between marketing and engineering teams.

Using a software tool facilitated the systematization of the obtained information, and it detected some stress in some of the sub-tasks. However, the research suggests creating an agreement about the feature descriptions and must describe the degree of granularity with which these descriptions will work. One of the identified future works is the development of a collaborative web tool that will facilitate the SPL scoping activity.

The discussions made by analyzing and organizing features sub-tasks reflect a high interaction of the roles, an achievement for a method that seeks collaboration among the participants. However, these sub-tasks required more time than estimated.

A collaborative relationship university–enterprise model allowed establishing the key elements to establish ties among research groups and the company by identify the meeting points that allow projects or experiences of co-production or collaboration. These meeting points cannot be forced. The activities must consider the resources and times of the company, so it may be necessary to adapt the proposals, approaches, or techniques to the inherent characteristics of the participating company. Interactions must be carried out in short times that cost little investment to the company (efforts, time and resources), and the results are not only documented in the research project, but these results are successful, and they can be adapted and adopted in the company processes.

Author Contributions: M.C.C., F.Á., and J.A.H. defined the CoMeS method and designed the case study. C.C. supported its collaborative aspects. Furthermore, M.C.C. refined the CoMeS detailing roles, tasks, and artifacts. M.C.C., J.D.B., and P.L. conducted the case study in Sunset Company. Authors collaboratively wrote the paper. Also, each author reviewed and improved it in content and style in advance. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially funded by the projects “Network of training of human talent for social and productive innovation in the department of Cauca” (INNOVACIÓN CAUCA), which is executed by the University of Cauca under code 3848, and the project “Domain Analysis for Software Product Lines of Serious Games” under code 4514 funded. Additionally, the work counted with funds from the internal call for research projects of the University Institution Colegio Mayor del Cauca and by calls for publications of the Vice-rectory for Research of the University of Cauca.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from the participating company and from all subjects involved in the study.

Data Availability Statement: The information of the method comes, and the study carried out can be found at <https://comesspl.com/>.

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Northrop, L.M.; Clements, P.C.; Bachmann, W.F.; Bergey, J.; Chastek, G.; Cohen, S.; Donohoe, P.; Jones, L.; Krut, R.; Little, R.; et al. *A Framework for Software Product Line Practice, Version 5.0*; Technical Report; Software Engineering Institute: Pittsburgh, PA, USA, 2012.
2. Faheem, A.; Luis Fernando, C. An organizational maturity model of software product line engineering. *Softw. Qual. J.* **2010**, *18*, 195–225.
3. Clements, P. *On the Importance of Product Line Scope*; Software Product-Family Engineering: Pittsburgh, PA, USA, 2002; pp. 102–113.
4. Schmid, K. Scoping software product lines, An Analysis of an Emerging Technology]. In *Software Product Lines*; The Springer International Series in Engineering and Computer Science; Donohoe, P., Ed.; Springer: Boston, MA, USA, 2000; pp. 513–532; Volume 576.
5. Chastek, G.; Donohoe, P.; Chul, K.; Pohang, K.; Thiel, S.; Bosch, R. *Product Line Analysis: A Practical Introduction*; Technical Report; Software Engineering Institute: Pittsburgh, PA, USA, 2001; CMU/SEI-2001-TR-001.
6. John, I.; Eisenbarth, M. A decade of scoping: A survey. In Proceedings of the 13th International Software Product Line Conference, San Francisco, CA, USA, 24–28 August 2009; pp. 31–40.
7. Ianzén, A.; Malucelli, A.; Reinehr, S. Scope definition in software product lines: A semi-automatic approach through linguistic annotation. In Proceedings of the 38th Latin America Conference on Informatics, CLEI 2012, Medellín, Colombia, 1–5 October 2012; pp. 1–9.
8. John, I.; Knodel, J.; Lehner, T.; Muthig, D. A Practical Guide to Product Line Scoping. In Proceedings of the 10th International Software Product Line Conference (SPLC’06), Baltimore, MD, USA, 21–24 August 2006; pp. 3–12.
9. Helferich, A.; Schmid, K.; Herzwurm, G. Reconciling marketed and engineered software product lines. In Proceedings of the 10th International Software Product Line Conference, SPLC 2006, Baltimore, MD, USA, 21–24 August 2006; pp. 23–30.
10. Carbon, R.; Knodel, J.; Muthig, D.; Meier, G. Providing Feedback from Application to Family Engineering—The Product Line Planning Game at the Testo AG. In Proceedings of the 2008 12th International Software Product Line Conference, Limerick, Ireland, 8–12 September 2008; Number 01, pp. 180–189.
11. Rommes, E.; Research, P. A People Oriented Approach to Product Line Scoping. In Proceedings of the PLEES 2003: International Workshop on Product Line Engineering: The Early Steps: Planning, Modeling, and Managing, Erfurt, Germany, 23 September 2003; Number 1.0, pp. 23–28.
12. Noor, M.; Rabiser, R.; Grünbacher, P. A collaborative approach for reengineering-based product line scoping. In Proceedings of the 1st International Workshop on Agile Product Line Engineering APLE’06 in the 10th International Conference, SPLC 2006, Baltimore, MD, USA, 21–24 August 2006.
13. Burge, J.E.; Carroll, J.M.; McCall, R.; Mistrk, I. Decision-Making in Software Engineering. In *Rationale-Based Software Engineering*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 67–76, doi:10.1007/978-3-540-77583-6_5.
14. Ullah, M.I.; Ruhe, G.; Garousi, V. Decision support for moving from a single product to a product portfolio in evolving software systems. *J. Syst. Softw.* **2010**, *83*, 2496–2512.
15. Camacho Ojeda, M.C. A Collaborative Method for Scoping Software Product Lines. Ph.D. Thesis, Universidad del Cauca, Popayán, Colombia, 2019.
16. Yousef, M.; Collazos, C.A. Collaborative strategies supporting knowledge management in organizations. *Rev. Colomb. Comput.* **2020**, *21*, 6–12.
17. Briggs, R.; Kolfschoten, G.; de Vreede, G.J. Defining Key Concepts for Collaboration Engineering. In Proceedings of the AMCIS 2006, Twelfth Americas Conference on Information Systems, Acapulco, Mexico, 4–6 August 2006; pp. 121–128.
18. de Vreede, G.J.; Briggs, R.O.; Massey, A.P. Collaboration Engineering: Foundations and Opportunities: Editorial to the Special Issue on the Journal of the Association of Information Systems. *J. Assoc. Inf. Syst.* **2009**, *10*, 121–137.

19. Kolschoten, G.L.; Vreede, G.J. The Collaboration Engineering Approach for Designing. *Groupw. Des. Implement. Use* **2007**, *4715*, 95–110.
20. Vreede, G.J.D.; Kolschoten, G.L.; Briggs, R.O. ThinkLets: A collaboration engineering pattern language. *Int. J. Comput. Appl. Technol.* **2006**, *25*, 140–154.
21. Santos de Moraes, M.B.; Santana de Almeida, E.; Romero de Lemos Meira, S. A Systematic Review on Software Product Lines Scoping. In Proceedings of the VI Experimental Software Engineering Latin American Workshop, ESELAW'09, São Carlos, Brazil, 11–13 November, 2009; pp. 63–72.
22. Marques, M.; Simmonds, J.; Rossel, P.O.; Bastarrica, M.C. Software product line evolution: A systematic literature review. *Inf. Softw. Technol.* **2019**, *105*, 190–208.
23. Lee, J.; Kang, S.; Lee, D. A Comparison of Software Product Line Scoping Approaches. *Int. J. Softw. Eng. Knowl. Eng.* **2010**, *20*, 637–663.
24. John, I. Using documentation for product line scoping. *IEEE Softw.* **2010**, *27*, 42–47.
25. Balbino, M. RiPLE-SC: An Agile Scoping Process for Software Product Line, Universidade Federal de Pernambuco. Ph.D. Thesis, Federal University of Pernambuco, Recife, Brazil, 2010.
26. da Silva, I.F. An agile approach for software product lines scoping. In Proceedings of the 16th International Software Product Line Conference, SPLC '12, Salvador, Brazil, 2–7 September 2012; pp. 225–228; Volume 1.
27. Schmid, K. A comprehensive product line scoping approach and its validation. In Proceedings of the 24th International Conference on Software Engineering—ICSE '02, Orlando, FL, USA, 25 May 2002; pp. 593–603.
28. Solano, A.; Granollers, T.; Collazos, C.; Rusu, C. Proposing formal notation for modeling collaborative processes extending HAMSTERS notation. *Adv. Intell. Syst. Comput.* **2014**, *275*, 257–266.
29. Solano, A.; Granollers, T.; Collazos, C. Modelado de Procesos Colaborativos Extendiendo Elementos de la Notación. *Rev. Colomb. Comput.* **2015**, *16*, 144–161.
30. Martinie, C.; Palanque, P.; Winckler M. Designing and Assessing Interactive Systems Using Task Models. In Proceedings of IFIP Conference on Human-Computer Interaction, Bamberg, Germany, 14–18 September 2015; pp. 2–31. doi:10.1145/2851581.2856686
31. Koller, M.; Bogdan, C.; Meixner G. Collaborative Task Modeling: A First Prototype Integrated in HAMSTERS. In Proceedings of Human-Centered and Error-Resilient Systems Development, HESSD 2016, HCSE 2016, Lecture Notes in Computer Science, Stockholm, Sweden, 29–31 August 2016; pp. 366–373 doi:10.1007/978-3-319-44902-9-24
32. Menéndez Domínguez, V.H.; Castellanos Bolaños, M.E. Software Process Engineering Metamodel (SPEM). *Rev. Latinoam. Ing. Softw.* **2008**, *3*, 92–100, doi:10.18294/relais.2015.92-100.
33. De Vreede, G.J.; Briggs, R.O. Collaboration engineering: Designing repeatable processes for high-value collaborative tasks. In Proceedings of the Annual Hawaii International Conference on System Sciences, Big Island, HI, USA, 3–6 January 2005; p. 17, doi:10.1109/hicss.2005.144.
34. Chacón-Luna, A.E.; Gutiérrez, A.M.; Galindo, J.A.; Benavides, D. Empirical software product line engineering: A systematic literature review. *Inf. Softw. Technol.* **2020**, *128*, 106389.
35. Knaggard, A.; Slunge, D.; Ekbo, A.; Göthberg, M.; Sahlin, U. Researchers' approaches to stakeholders: Interaction or transfer of knowledge? *Environ. Sci. Policy* **2019**, *97*, 25–35.
36. Jahn, T. Transdisziplinarität in der Forschungspraxis. In *Transdisziplinäre Forschung. Integrative Forschungsprozesse verstehen und bewerten*; Bergmann, M., Engelbert, S., Eds.; Campus Verlag: Frankfurt, Germany; New York, NY, USA, 2008 ; pp. 21–37.
37. Jahn, T.; Bergmann, M.; Keil, F. Transdisciplinarity: Between mainstreaming and marginalization. *Ecol. Econ.* **2012**, *79*, 1–10.
38. Camacho, M.C.; Álvarez, F.J.; Ruiz, P.H.; Alegría Hurtado, J.A. *Review on Teamwork during the Scoping of Software Product Lines*; Technical Report; Universidad del Cauca: Popayán, Colombia, 2017.
39. Camacho Ojeda, M.C.; Hurtado Alegría, J.A.; Álvarez Rodríguez, F.J. An exploratory study for scoping software product lines in a collaborative way. In Proceedings of the 11th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE '18, 40th International Conference on Software Engineering ICSE'18, Gothenburg, Sweden, 27 May–3 June 2018; pp. 17–20, doi:10.1145/3195836.3195852.
40. Camacho Ojeda, M.C.; Hurtado Alegría, J.A.; Álvarez Rodríguez, F.J.; Ruiz Melenje, P.H. A Collaborative Method for a Tangible Software Product Line Scoping. In Proceedings of the International Conference on Applied Informatics Workshops, ICAIW 2018—Joint Proceedings of the Workshop on Data Engineering and Analytics, WDEA 2018, Workshop on Smart Sustainable Cities, WSSC 2018, Workshop on Intelligent Transportation Systems, WITS 2018 and Workshop on Empirical Experien, Bogota, Colombia, 1–3 November 2018.
41. Ruiz, P.H.; Camacho, C.; Hurtado, J.A. A Comparative Study for Scoping a Software Process Line. In Proceedings of the 2018 ICAI Workshops, ICAIW 2018—Joint Proceedings of the Workshop on Data Engineering and Analytics, WDEA 2018, Workshop on Smart Sustainable Cities, WSSC 2018, Workshop on Intelligent Transportation Systems, WITS 2018 and Workshop on Empirical Experien, Bogota, Colombia, 1–3 November 2018; pp. 1–6.
42. Wohlin, C.; Runeson, P.; Höst, M.; Ohlsson, M.; Regnell, B.; Wesslén, A. *Experimentation in Software Engineering*; Springer: Berlin/Heidelberg, Germany; Publishing Company, Incorporated: Singapore, 2012; pp. 1–236.
43. Runeson, P.; Höst, M. Guidelines for conducting and reporting case study research in software engineering. *Empir. Softw. Eng.* **2009**, *14*, 131–164.

44. de Medeiros, T.F.L.; Santana Almeida, E. CodeScoping: A Source Code Based Tool to Software Product Lines Scoping. In Proceedings of the 2012 38th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), Cesme, Turkey, 5–8 September 2012; pp. 101–104, doi:10.1109/SEAA.2012.70.
45. Krueger, C. *Eliminating the Adoption Barrier*; IEEE Computer Society Press: Washington, DC, USA, 2002; Number 4, Volume 19, pp. 23–31. doi:10.1109/MS.2002.1020284.
46. Reo, D.; Quintano, N.; Buglione, L. Measuring Software Process Improvement: There's more to it than just measuring processes. In Proceedings of the FESMA 99, 2nd European Software Measurement Conference, Amsterdam, The Netherlands, 4–8 September 1999.
47. Davis, F.D. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Q.* **1989**, *13*, 319.
48. Moody, D.L. The method evaluation model: A theoretical model for validating information systems design methods. In Proceedings of the Event European Conference on Information Systems 2003 (ECIS '03), Naples, Italy, 16–21 June 2003; pp. 1327–1336.
49. Abrahão, S.; Insfran, E.; Carsí, J.A.; Genero, M. Evaluating requirements modeling methods based on user perceptions: A family of experiments. *Inf. Sci.* **2011**, *181*, 3356–3378.
50. Calvani, A.; Fini, A.; Molino, M.; Ranieri, M. Visualizing and monitoring effective interactions in online collaborative groups. *Br. J. Educ. Technol.* **2010**, *41*, 213–226.
51. Avouris, N.; Margaritis, M.; Komis, V. Modelling interaction during small-group synchronous problem-solving activities: The Synergo approach. In Proceedings of the Intelligent Tutoring Systems 7th International Conference, ITS 2004, Maceio, Brazil, 30 August–3 September 2004; pp. 5–11.
52. Helferich, A.; Schmid, K.; Herzwurm, G. Product Management for Software Product Lines: An Unsolved Problem? *Commun. ACM* **2006**, *49*, 66–67, doi:10.1145/1183236.1183268