End-User Highlighted: featuring tailorable systems development

Claiton Marques Correa
claiton.correa@edu.pucrs.br
Farroupilha Federal Institute - IFFar
School of Technology - PUCRS

Milene Selbach Silveira
milene.silveira@pucrs.br
School of Technology - PUCRS

ABSTRACT

Software varies in the domain and has different target users. Also, it is not possible to predict all users' requirements at the system development stage. This situation sheds light on the need to support end-users to adapt their applications to cope with new demands. End-User Development (EUD) is a research area that intends to permit end-users to perform activities beyond application usage. Such activities may involve software tailoring. Considering this, we aim to map EUD towards supporting software with tailoring capabilities. To achieve this goal, we conducted a literature review using a snowballing procedure on the subject, and we found 42 studies in the time range 1997-2020. We analyzed the topic's development in terms of concepts, methodologies to building tailorable software, tools, and we treat issues and implications of tailorable systems development. Finally, we present future works to research and develop tailorable systems based on gaps and opportunities identified.

CCS CONCEPTS

• **Human-centered computing → Interaction design theory, concepts and paradigms**.

KEYWORDS

end-user development, tailoring, adapting, customization

ACM Reference Format:
Claiton Marques Correa and Milene Selbach Silveira. 2021. End-User Highlighted: featuring tailorable systems development . In *XX Brazilian Symposium on Human Factors in Computing Systems (IHC'21), October 18–22, 2021, Virtual Event, Brazil.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3472301.3484332

1 INTRODUCTION

Software artifacts vary in the domain and have different target users[1]. As a general goal, the software attempts to meet the user's needs. However, during its use, as the users gain proficiency in system usage, unexpected necessities might unfold. This scenario points out the need to enable users to adapt their applications to cope with new demands.

Considering the perspective above, researchers have studied approaches to empower end-users to tailor the software to their needs (*tailoring* is the term employed to describe the activity of modifying an application to meet some using context of the user [35]). These studies refer to End-User Development (EUD) topic, and most of them have been developed within the Human-Computer Interaction (HCI) field [5].

EUD is formally defined as a set of methods, techniques, and tools to allow users of software artifacts that are not professional developers, at some point to create, modify, or extend the software [37]. The seminal studies of this research field date to the nineties. Nardi [43] coined the end-user programming term to describe a set of advanced activities that software users can get involved with. These activities may encompass simple customization actions such as changing the color or font size or more sophisticated ones as creating new features during the software use.

Empowering users to perform the activities mentioned above involves letting them do this at design time and during software use [20]. At this point, it helps to distinguish between design time and use time, following the definition stated by Fischer and Giaccardi [24]: at design time, system developers create environments and tools to meet the users' requirements. At use time, in turn, users use the system.

The studies in EUD topic raised some issues concerning the development of software with tailoring capabilities. For instance, the trade-off between flexibility and usability. Another concern is to make users aware of the parts of their applications that can be modified in use or how to enhance communication between system developers and end-users.

Advances in technology make it possible to develop more dynamic applications in which users can be provided with such tailorable resources. Nowadays, specific web programming frameworks or libraries, as ReactJS[2], and VueJS[3], can be employed in this context. These new possibilities ground the interest to investigate how the research community has approached tailorable systems development.

To delimitate our research to EUD towards developing software with tailoring capabilities, we defined our research in terms of *How the research field has been evolving to support tailoring software by end-users*. Additionally, we are interested in identifying concepts, methods, opportunities, and difficulties within tailoring-featured

[1] In this paper we consider the words user and end-user as synonymous

[2] https://reactjs.org/

[3] https://vuejs.org/

IHC'21, October 18–22, 2021, Virtual Event, Brazil
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8617-3/21/10…$15.00
https://doi.org/10.1145/3472301.3484332

applications. We then conducted a literature review using a snowballing procedure on this subject, and we found and analyzed 42 papers dealing with tailoring software within EUD topic.

As part of our results, we analyzed the selected studies' contributions. Then, after a textual analysis, we organized the research topics into four periods encompassing concepts, approaches, and requirements for developing tailorable software, and associated technology. This contribution can aid novice researchers in recognizing the field in terms of authors, approaches, and concepts. Another contribution of this study reflects future works for software development with tailoring features research raised from the studies' outcomes observed.

To present the research done, the remainder of this paper is organized as follows. Section 2 briefly presents the fundamental concepts that anchor this research. Section 3 details the review strategy followed in our snowballing procedure. Section 4 presents the analysis of the results found during the study. Section 5 presents a discussion about our results key points and presents gaps and opportunities observed, and the study's limitations. Section 6 discusses works most related to ours. Finally, Section 7 presents our conclusions and future work.

2 BACKGROUND

At design time, system developers aim to gather information to develop applications that meet the users' needs. The task of collecting information cannot handle all of them. The requirements for new demands emerge as users get familiar with their daily applications, and because of this, the system must evolve [25].

EUD is a research field focused on such studies and for empowering users to create and adapt their applications according to their needs [37]. EUD intends to make the users able to customize, modify, or extend software artifacts. Therefore, it is essential to differentiate customization and modification. Lieberman et al. [37] defined these terms as follow.

- *Customization*: this category encompasses the activities that allow users to choose predefined systems behaviors. Actions such as change the color font, font type, or other properties are examples of customization. These changes can occur by an explicit user action or as a system response to a user action, such as in responsive applications.
- *Modification*: modifications are actions that imply in create or modify the software. Visual programming, programming by demonstration, macros, and script languages are examples of approaches in this category.

Another aspect within EUD involves the learning cost of a programming language and its scope of use. Fig. 1 shows a mapping of this cost proposed by Fischer et al. [25]. In this mapping, the cost and scope range from high to low. Programming languages such as Java and C++ are set with a higher scope and cost. The EUD ideal context is the one in which a language combines the high scope and low cost.

Over time, as End-User (EU) researches evolved, two more EU-related topics were defined to distinguish the different concerns in activities performed by end-users. Hence, End-User Programming (EUP) and End-User Software Engineering (EUSE) were introduced.

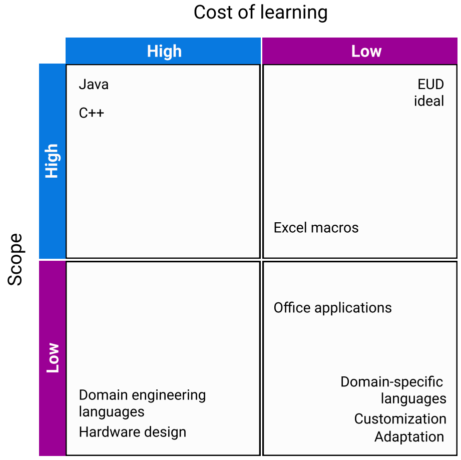

Figure 1: Learning cost of programming languages. Adapted from Fischer et al. [25].

EUP relied on approaches that involve end-users in programming activities, so it is roughly related to ideas proposed by Nardi [43]. EUSE, in its turn, is focused on activities carried out to guarantee the quality of the code created by end-users [36]. EUD has taken a broader meaning, covering methods, situations, and social-technical environments to empower users to craft software artifacts and being independent of high-tech scribes [23].

In this work, we sought EUD studies covering concepts, approaches, tools related to the situations in which end-users can tailor their applications. It may include theoretical investigations or investigations aiming to build a tool, for example. Therefore, it is essential to consider that tailoring is referred to in the literature in different ways; as Costabile et al. [9] observed, actions of adaptation, customization, personalization, extension, etc., are regarded as tailoring activities.

3 METHODOLOGY

This study was carried out by following the Snowballing procedure for the literature review proposed by Wohlin [49]. Snowballing involves defining a set of initial studies and rounds or iterations of backward and forward steps. In the backward step, the paper's reference list is considered. The forward step encompasses the analysis of articles citing the paper under evaluation. Both steps are unrolled to seek additional papers. Each new paper found is pushed in a pile for the next round of forward and backward steps. The process ends when no papers are added to this pile.

The choice for the snowballing procedure relies on the fact that we found a recent literature review encompassing EU topics. Barricelli and colleagues' [5] employed a literature review, and their study encompasses EUD, EUP, and EUSE studies. Therefore, we decided not to replicate its methodology because we were explicitly interested in EUD studies. Hence, we choose snowballing to seek additional papers on the topic.

Before starting the snowballing procedure, we defined our study's research question and topics to analyze during the papers' scrutiny.

3.1 Research Question

This study's primary goal is to provide an overview of tailoring software by end-users within End-User Development (EUD) topic. Thus, we defined the following research question:

> *How EUD has been evolving to support tailoring software by end-users?*

By answering this question, we aimed to identify how studies have been evolved in terms of concepts, methods, tools, opportunities, and difficulties researchers have been going through to build applications with tailoring features.

3.2 Snowballing procedure

The snowballing procedure's first step is to define the initial set of studies to start the procedure. Following the rules stated by Wohlin [49], to avoid bias in favor of any specific publisher, we used Google Scholar to define the initial set of studies. We searched for EUD studies on Google Scholar, and we found a secondary study on the End-User Development topic conducted by Barricelli et al. [5]. We analyzed this study's reference list to determine the initial set of papers to start our literature review, employing snowballing. According to Wohlin, although there is no recipe to define the initial set of studies, a helpful approach is to select highly cited papers in the area. Hence, based on this characteristic, we selected the following studies to compound our initial set:

> P1: Fischer, G., & Giaccardi, E. (2006). Meta-design: A framework for the future of end-user development. In End user development (pp. 427-457). Springer, Dordrecht.
> P2: Lieberman, H., Paternò, F., Klann, M., & Wulf. (2006). End-user development: An emerging paradigm. In End user development (pp. 1-8). Springer, Dordrecht.
> P3: Fisher, G., Fogli, D., & Piccinno, A. (2017). Revisiting and broadening the meta-design framework for end-user development. In New perspectives in end-user development (pp. 61-97). Springer, Cham.

The third study has not a high citation score. However, we decided to include it in the starting set due to its scope. This study broads the discussion addressed by the first high cited study.

3.2.1 *Selection Strategy.*

To conduct our literature review, we defined the exclusion and inclusion criteria as follows.

Publications that met any of the following criteria were excluded from the review:

- Duplicated papers.
- Studies that are not written in English.
- Ph.D. or Master dissertations.

The inclusion criteria are:

- Studies must be published in the Computer Science area.
- Studies must address EUD and/or terms related to tailoring software by end-user.
- The time frame considered is 1997 - 2020.

We defined the beginning of the time frame range, taking into account the fundamental work in tailoring activities published by Mørch [41] and ending the time frame range by the year in which the review was carried out.

3.2.2 *Conducting.*

After defining the starting set, we conducted the backward and forward steps seeking additional papers related to the research question. Hence, considering the set of papers being analyzed, we checked their reference list in the backward step. We excluded papers that do not fulfill our inclusion criteria. Following this, we removed papers from the list that have already been examined in previous iterations. We then analyzed the title, publication venue, authors, and the most relevant parts of each paper to decide whether they should be included or not in a pile for the next iteration.

In the forward step, we identified new papers based on the papers citing the paper being examined. We considered the information provided by Google Scholar, and we checked the papers similarly as approached in the backward step. Each selected paper was included in a pile for the next iteration of backward and forward steps. The process ended when no new papers are included.

To address our research question and select studies adhering to it, we only selected studies that met at least one of the following criteria: (i) present systems architectures that allow customization or modification by end-users; (ii) address significant concepts for tailoring software development; and (iii) present methodologies or frameworks for tailoring software development.

We started to carry out our review at the end of 2019. During the time required to conduct all the analyses, we also decided to include papers published in 2020. From the starting set, a total of four iterations were performed until no new papers were found. In the first, second, and third iterations, we added 31, 7, and one study, respectively. In the fourth iteration, no papers were added, and the snowballing procedure was concluded. Thus, 42 publications were selected. Fig. 2 illustrates the performed iterations from the snowballing process. Note that considering the three papers in the starting set in the first iteration, we found 13 new studies in the backward step and 18 new ones in the forward step. Thus, 31 publications compounded the set for the subsequent iteration. The process lasted until no papers were found.

These studies address End-User Development towards tailoring software or related terms to this subject. Table 1 shows the selected papers ordered by year. Studies from Barricelli et al. [5] and Maceli [38] are also systematic reviews. We found and decided to maintain them during the snowballing iterations due to their contribution and because they differ from this study in scope. The section 6 discusses these works.

After identifying and collecting, papers were independently assessed to identify the work's overall purpose and main contributions. We employed a textual analysis used in the open codification phase from the Grounded Theory [8] for each paper's full-reading. This approach helped us to define and group concepts and patterns we found during the process. Thus, we realized that it was possible to organize the studies into four periods due to similar patterns found. This analysis allowed us to define each period's label (see Section 4).

In this sense, the studies in the first period present terms most related to tailoring and show the researchers' effort to (re)think methodologies or develop new ones to involve end-users in the software planning. The second period has studies discussing issues related to system performance, users involved in the use of such systems, and their activities and approaches that may be helpful in

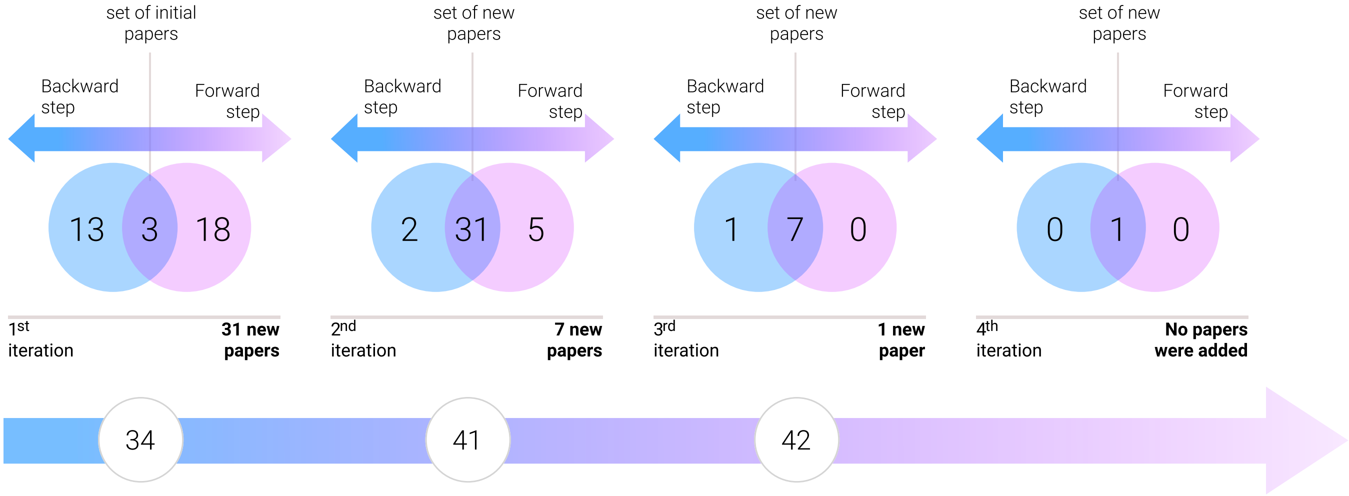

Figure 2: Amount of selected studies per performed iteration

tailorable systems design. The same is true for the two subsequent periods. The period's label tries to catch the central aspects observed in the studies within the timeframe.

Table 1: Selected papers per year

Year	Selected papers	Year	Selected papers
2000	[42]	2011	[7]
2001	[50]	2012	[14] [29] [31] [40]
2002	[17]	2013	[30] [39] [44] [45]
2003	[9]	2014	[2] [6] [47]
2004	[25]	2015	[16]
2005	-	2016	-
2006	[10] [18] [24] [37]	2017	[3] [23] [38]
2007	[4] [11] [12] [19] [20]	2018	[1]
2008	[13] [28]	2019	[5]
2009	[27]	2020	[15][22][32][33][34]
2010	[21] [46]		

4 RESULTS

In this section, we discuss the analysis carried out on the 42 selected papers. Afterward, as mentioned before, from our textual analysis, we decided to organize our results as a four periods featuring the studies' outcomes. Then, we labeled the periods based on the contribution recognized from the group of studies belonging to each trend.

Fig. 3 illustrates these periods of EUD studies. We named the trends as Emerging needs for tailoring, Designing of tailorable systems, EUD environments, and Broadening EUD concepts. Below each period in the image, we summarize terms from the studies in the period. In each subsection, we present figures to illustrate central concepts from the periods.

4.1 Emerging needs for tailoring (2000-2004)

A glance at the studies that belong to this trend shows the researcher's effort to define the purpose of tailoring actions, build approaches to enable users to adapt their applications, and make them aware of tailoring functions.

The intent of developing applications with tailorable capabilities raised from reasons such as the impossibility of gathering all user needs at design time. Technology evolution also brought new opportunities for users to get engaged and for designers to develop their applications. Thus, technologies that emerged in the so-called web 2.0 enabled users to interact with others or take place as content producers.

Taking advantage of all the new opportunities implied a change towards new models, methodologies, and programming languages. In this context, this trend encompasses studies that reveal the author's endeavor to foster the understanding of tailoring activities definition and involve users in meaningful actions, such as designing or tailoring their daily software. This scenario also reveals some issues and implications to accomplish such development.

A shared vision from this trend is that the distinction between users' and developers' roles started to get blurred due to ongoing changes in the relationship between users and applications, and users and content. However, the authors noticed that although users could perform meaningful activities, they were not likely to engage in programming activities. Hence, tailoring was said to encompass actions to modify the system during its use by means that do not involve writing codes [42].

Mørch and Mehandjiev [42] set tailoring as a long-term collaboration between the system's developers and users. This collaboration is mediated by the application's interface, which tells the users which action they can take. Thus, to enable tailoring, the authors argued that the model from which an application is developed should be closer to users' tasks. Hence, they proposed systems multiple representations and applications units to fill the

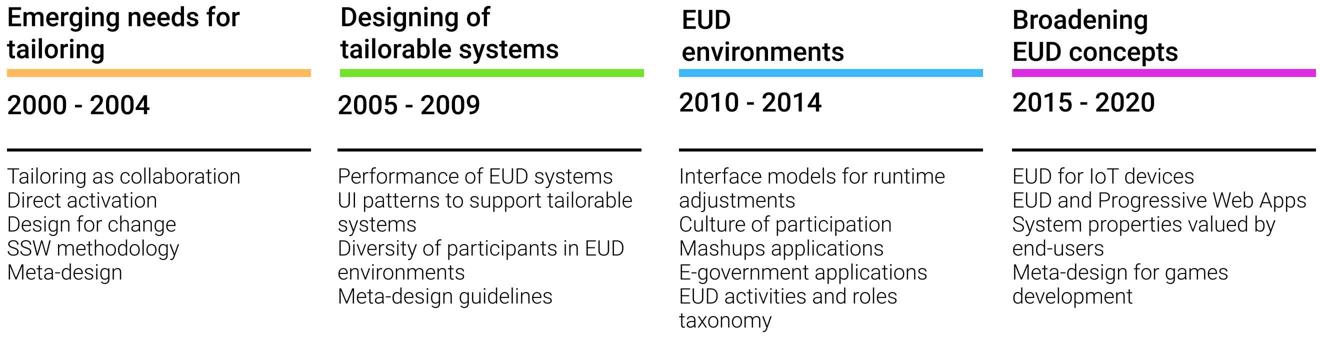
Emerging needs for tailoring
2000 - 2004
Tailoring as collaboration
Direct activation
Design for change
SSW methodology
Meta-design
Designing of tailorable systems
2005 - 2009
Performance of EUD systems
UI patterns to support tailorable systems
Diversity of participants in EUD environments
Meta-design guidelines
EUD environments
2010 - 2014
Interface models for runtime adjustments
Culture of participation
Mashups applications
E-government applications
EUD activities and roles taxonomy
Broadening EUD concepts
2015 - 2020
EUD for IoT devices
EUD and Progressive Web Apps
System properties valued by end-users
Meta-design for games development

Figure 3: The four periods of EUD studies

gap between user-oriented representations and the application's underlying code.

These representations are part of the software that users may match in order to tailor their applications. So, they represent communication paths from where the users can realize which actions they may consider for modifying the software. One important issue, as the authors reported, is to maintain the system congruence between different parts of the system.

One fundamental work in this trend addressed the changes mentioned previously to reconsider software development practices. Regarding to design and use of tailorable software, Dittrich et al. [17] claimed that they should not be regarded as two separate and sequential activities but rather as overlapping and intertwined perspectives. So, they advocated for researchers to take the participatory design beyond stakeholders' involvement at design time. They called it to design for change. From this perspective, the software design must be an ongoing collaboration among the developers and stakeholders involved in the software use. In this sense, this work highlighted implications for the participatory design and its methods and tools for tailoring software during its use. It also highlighted the importance of the continuous interaction among the actors involved in the development, whether they are users or professional developers.

As essential as having tailoring functionalities is making them available for users. An additional concern raised in the studies is designing the application from which users can be aware of tailoring functions. To cope with this demand, Wulf and Golombek [50] established direct activation to allow the user to access tailorable systems functions. In this concept, a function's tailorable parameters are displayed close to the function's visual representation. Thus, users can note the tailorable function and recognize which results achieve from it.

Two works approached means to aid users' involvement in significant activities in light of the increased need for tailoring software. Costabile et al. [9] reported the Software Shaping Workshops (SSW) to support tailoring driven by users and then led to the concept of system co-evolution. The SSW allows users to collaborate with design teams to build applications to their needs.

Fischer et al. [25] presented the meta-design framework. This framework stands for an important proposal for EUD environments. Indeed, we realized its importance based on the number of studies that consider it a baseline to create environments for users to perform design activities. Meta-design characterizes objectives, techniques, and process to users act as designer of their systems [25].

4.2 Designing of tailorable systems (2005-2009)

Systems with tailoring capabilities require attention to issues that vary from tech aspects to communication among the different stakeholders. The goal of a EUD-like interface is to let the users adapt systems at a level of complexity appropriate to their individual skills thus, taking the systems from easy to use to easy to develop [37]. The studies from this trend discussed such issues. They also provided implications and employed methodologies to cope with them.

Regarding tech aspects, Dittrich et al. [18] reported five issues that system developers must be aware of in this subject. They are usability, decide what parts of the system must be adaptable (and how to design for it), system performance, software engineering for end-users, and how the software development changes with tailoring.

The tailoring interface's usability plays an essential role once it represents the higher abstraction of the actions. The tailorable parts of the system have to match a user perspective on the domain. Here, we can establish a possible approach to cope with this issue that may rely on Baranauskas and Neris' study [4]. Regarding the different people, interests, domains, and contexts diversity, the authors proposed using design patterns to support the design of tailorable User Interfaces (UI). This work presents an interesting approach regarding its contribution. The authors explored a set of interaction patterns suitable for the design of flexible applications. As a result, they classified these patterns according to the semiotic ladder. This classification may help developers to construct applications with tailoring features regarding different domains and contexts.

The second issue addressed by Dittrich and colleagues is related to what should be adaptable or not in the software. According to the authors, take this type of definition requires the full understanding of the user's domain. Also, user feedback is important to define the tailorable parts of the software. For this issue, we can also relate another study from this trend that may represent a probable approach to handle it. The study published by Costabile et al. [11]

employed a methodology that, among other goals, aims to improve the communication and feedback exchange between users and the developers' team in the context of tailoring software development.

Another issue about tailorable systems is their performance. Many design techniques that provide flexibility and tailorability reduce system performance [18]. Unfortunately, the studies do not deepen this issue. From our perspective, it requires more investigation.

The fourth and fifth issues concern the software documentation and the changes in software development methodologies that the tailorable applications imply on developers teams. The former intends to cope with the users' fear to perform tailoring actions. From documented software, users' may feel safer in conducting such activities. The latter addresses how the industry changed in terms of methodologies to support the development of tailorable software.

Software development usually involves stakeholders from different skills and knowledge. Software engineers, HCI experts, and end-users are examples. Naturally, each one of them has an understanding of the system. This situation raises some communication gaps. From this trend, we can relate two more studies due to their contributions.

Regarding the diversity of users involved in EUD environments, Fischer et al. [28] and Costabile et al. [13] established frameworks that describe the different types of participants developing and designing the tools they use. In a similar intent, the study mentioned previously, conducted by Costabile et al. [11], addressed the communication among different stakeholders engaged in such environments. In another study, Costabile et al. [10] and Costabile et al. [12] presented an approach to cope with users' diversity by employing a HCI model to develop a visual interactive system for medical domain. These papers are complementary in terms of coping with users' diversity and communication.

By conducting a case study, Dörner et al. [19] explored the development of an EUD tool for small and medium entreprises (SME). The authors interviewed several employees aiming to understand the dynamic of adapting software by end-users. As a result, they presented a set of problem types (e.g. usability problems, functional problems) identified during the study, and the strategies applied to solve them.

Three studies adressed the meta-design and the Seeding, Evolutionary Growth, and Reseeding (SER) model. This model considers system design as a three-phase activity: Seeding, Evolutionary Growth, and Reseeding framework in this trend. Fischer et al. [27] present guidelines for using meta-design to build systems. Fischer [20] and Fischer and Giaccardi [24] reflected on the socio-technical environments and the meta-design adoption through challenges and opportunities for the framework.

4.3 EUD environments (2010-2014)

Many solutions have been proposed with the intent to empower end-users to craft their applications. As the main feature, this trend encompasses studies approaching the development of solutions to make software usage more flexible. This trend also contains studies reflecting EUD environments for web and mobile solutions and types of roles and activities performed by end-users.

In general, the works concern mashups and domain-specific solutions, e.g., e-government services. Regarding the issues dealt with, the tools try to approach communication gaps pointed out by Costabile et al. [11] or the system congruence in tailorable tools addressed in the second trend.

Schwartze et al. [46] explored the concept of interface models to describe users' interfaces and their constraints. The models with solver support enable the interface to be adaptable in response to users' actions. These constraints' goal is to ensure the system congruence when the users are performing tailoring actions. Similarly, to construct more flexible interfaces, Neris and Baranauskas [14] employed a participatory design approach to explore rules and constraints to develop a social network system solution. In their study, a framework for the design of a flexible interface (PluRaL) was employed. The users' participation in design time helped define the possible interfaces, and their flexible behavior was formalized in norm-based structures.

A set of works also investigated EUD applications using mashups or other techniques that enable end-users to perform their customization actions. Cappiello et al. [7] explored mashups' development, i.e., applications based on components' compositions. In the pursuit of this, the authors developed the DashMash platform, an application aiming to demonstrate mashup potential. This study's main issue is to make the mashup process easy for end-users to compound the application.

Two studies employed a meta-design approach to enhance e-government services. Fogli [29] and Fogli and Provenza [31] developed a solution to enable domain-experts to build employees' and end-users' applications. Both studies are useful examples of how to create and apply a meta-design approach. Although the studies' target was specific e-government services, the methodology employed can be derived for other domains.

One more work published in this time frame concerned e-government applications. Fogli and Piccinno [30] analyzed the usability issue inherent to EUD activities towards software development to end-users. As a case study, the authors used the application developed by Fogli and Provenza [31].

Pantazos et al. [44] explored EUD features from Information Visualization (InfoVis) applications. They evaluated a set of the most popular InfoVis tools in predefined visualizations and the offering resources for visualization customization. Other studies presented guidelines to support EUD's environments [39, 40].

Regarding the actions in which a user can commit when conducting tailoring activities, Cabitza et al. [6] built an interesting taxonomy. This classification allows us to discern situations in which usability is most pursued.

The authors also specialized public EUD into inward EUD and outward EUD. Inward and outward specializations encompass situations in which the users tailor artifacts for a community they also belong to, or external communities, respectively. This classification is useful to understand the usability issue when outward EUD is the focus. Indeed, when performing individual EUD, one may concern with functionality rather than usability. Conversely, usability is likely desirable when performing outward EUD.

The remaining papers in this trend addressed EUD environment topics, such as cultures of participation [21], participatory design

[14], guidelines to support end-users customization, a study presenting EUD approaches for web, and mobile solutions [45].

4.4 Broadening EUD concepts (2015-2020)

The EUD research field has increased once the evolution of technology tools introduced new possibilities for users to shape systems in use. This trend's primary outcomes are expanding concepts related to EUD, addressing EUD-like approaches for Internet of Things (IoT) and web applications, and presenting some improvements in mashup applications development.

Ardito et al. [1] reflected on system properties valued by end-user when performing EUD activities and showed that the users are focused on supporting the tasks they are performing. By evaluating an application with users, their study has reported significant findings addressing tailorable systems' development. An important outcome is that the users are more focused on how the application supported their tasks than its aesthetic features.

Fischer et al. [23] reviewed studies employing the meta-design framework in the range of ten years. They also investigated meta-design implications supporting different domains, such as e-government, medical, and virtual worlds. As Cabitza et al. [6], they stressed the trade-off between utility and usability of tailorable tools. In respect of technologies employed to develop these tools, Maceli [38] showed that they did not change over the years, although the increasing number of publications observed. Also, she observed the lack of solutions for interfaces such as voice and tangible.

Concerning other mashup tools, Desolda et al. [16] presented the EFESTO platform that distinguishes from other proposals by considering the end-user mental model for user composition reported in the literature. This approach tries to overcome some mashup difficulties (e.g. complexity of composition paradigm) identified in previous studies. Ardito et al. [3] discussed a three-layer meta-design model to develop a mashup tool that enables any domain customizations.

Naturally, as EUD research evolves, the researchers also considered the IoT domain. Johnsson and Magnusson [34] presented a Graphical User Interface (GUI) development for IoT devices. The GUI was developed concerning situations in which users input are required to interact with IoT devices. The authors focused on developing an approach that empowers end-users to build GUIs without typing any piece of code. The users are provided with an editor in which they define the graphical components of their applications. The interface behavior is stored in an eXtensible Markup Language (XML) file while performing these actions.

Addressing EUD research on other domains or technologies, Cardieri and Zaina [15] investigated the combination between EUD and Progressive Web App (PWA). The authors' goal was to identify communication breakdowns that emerge when users play the role of co-designers. The choice for combining PWA and EUD relies on the scant of studies addressing it. Cardieri and Zaina contribuitons are twofolded. They present communication breakdowns that can affect the end-user experience when they act as co-designers. Also, they explored a promising technology for web development under a EUD-like approach.

This period also contains studies investigating EUD and meta-design framework employment to game development [32], website (re)design activities [33], and the trade-offs of enabling users to perform meaningful activities, i.e., customizing, adapting, or creating new digital artifacts [22].

5 DISCUSSION

In this work, the primary goal was to understand how EUD has evolved to support tailoring software by end-users. In this pursuit, we conducted a literature review and analyzed the reported results of selected studies. We recognized patterns that allowed us to organize the outcomes in a timeline of four periods from our analysis. The studies first addressed tailoring fundamental concepts, then going to tailorable systems development issues, and finally developing applications (e.g., mashups). We claimed that our timeline supports the reader to reasoning on fundamental perspectives of tailorable software development.

The first trend represents how and why the researchers considered empowering end-users to perform meaningful activities in their daily applications. In this trend, Fischer et al. [25] published the work presenting the meta-design as a framework to cope with this goal. We also observed that concepts, such as tailoring, have been handled differently by the authors. Indeed, Costabile et al. [9] have already pointed out this.

The second trend encompasses studies addressing the designing of tailorable systems. Studies from this trend present requirements, issues, and they concern the performance and usability of tailorable systems. Also, two methodologies supporting the meta-design are presented, the SER [26] and the SSW methodology [11].

Our third trend presented a taxonomy, proposed by Cabitza et al. [6], concerning the classification of EUD activities in which a user may commit. This classification showed a trade-off when performing such activities, i.e., the relation between functionality and usability. The former is related to situations in which users are committed to tailoring for their purpose. The latter is desirable when they are performing tailoring activities for the use of other users. Another trade-off is related to the relationship between flexibility and usability. Although these trade-offs are fundamental issues when one considers software development with tailoring capabilities, Fischer and Piccinno [23] argued that they are more related to End-User Software Engineering (EUSE).

In the last trend, Ardito et al. [1] showed that users engaged in tailoring activities focus on properties related to the system effectiveness in support of their actions rather than in software aesthetic features. Of course, it does not mean that aesthetic aspects of an application should be disregard. It enhances that the system must be designed to best fit the user's goals and support than in their daily tasks. This situation may imply further studies addressing the trade-offs of empowering end-users in this context.

We identified some gaps and opportunities for research during our review. For instance, we did not find studies measuring the performance of tailorable applications. One hypothesis for this lack is that specific research communities, such as parallel and distributed computing, not studied here, may conduct this type of study. Regarding opportunities for additional investigation are the implications of building software with tailorable features for development methodologies. We believe that this discussion may be worthy of addressing in the face of current methodologies.

Some limitations may affect our review of the literature. We defined our review time frame considering the moment in which we carried out the research. Due to this and the time required to conduct all analyses, at the date of submission, there are probably other published papers not considered here that could be of interest. Also, we did not consider Master or Ph.D. dissertations. Thus, some original ideas concerning tailorable systems may not have been considered.

Moreover, papers describing methods and tools for EUD tailoring actions may not cite these terms explicitly. Our study could not reach these works.

6 RELATED WORK

EUD is an active research topic in the HCI community. Prior studies addressed reviews on the subject to classify technologies, target users, and other topic features.

Maceli [38] focused on technologies proposed in EUD studies. To accomplish this, the author analyzed 73 papers from 2004 to 2016. The selected papers are derived from EUD-related conferences and journals. The author considered the categories discussed by Tettero and Markopoulos [48], i.e., *engineering* and *re-engineering*. Of 73 studies, 48 were also considered in the survey conducted by Tettero and Markopoulos [48]. Thus, Macelis' studies added 25 papers to the analysis. Thirteen categories have been obtained from the studies with the dominance of programming environments and web and information authoring tools.

Considering the list of papers analyzed by Maceli, three are also examined in our systematic review, i.e., we examined 39 different studies. Besides, the analysis conducted in our review has a different goal over investigating EUD evolution. We did not focus only on technologies but also on other aspects, such as concepts, methodologies for developing tailorable systems, issues, and implications.

Barricelli et al. [5] also conducted a systematic review of EUD and related fields. The authors have classified scientific literature about EUD, EUP, and EUSE. They analyzed 165 papers ranging from January 2000 to May 2017. Regarding the studies' selection, the authors employed a manual and automatic search. The manual search considered proceedings, special issues, and books related to EUD, EUP, or EUSE themes. The automatic search considered the following digital libraries: ACM, IEEE Xplore, Springer, and ScienceDirect.

The selected papers were classified into seven dimensions: type of approach, interaction technique, phase in which the approach is adopted, application domain, target use, class of user, and type of evaluation.

Considering the list of papers analyzed by Barricelli et al. [5], four are also examined in our systematic review, i.e., we examined 38 different studies. By considering the three related fields that aim to empower end-users, i.e., EUP, EUD, and EUSE, Barricelli and colleagues conducted a broader review that maps the studies in the sevens dimensions aforementioned. Our review tried to show the topic's evolution by catching the studies' common goals or concerns at each trend.

Regarding the results, the reviews do not overlap. The study conducted by Maceli [38] is the one most similar to ours. However, she focused only on the technological evolution of EUD studies.

7 CONCLUSION

EUD studies have increased and spread to research fields beyond HCI due to the technology's evolution that enabled the development of more powerful interactive systems [4]. This evolution also directed to a shift from consumer cultures to participation [21]. Furthermore, the impossibility to collect all users' needs at design time led to the interest of providing them the means to perform more meaningful activities.

This research reports on the results of a literature review assessing EUD evolution towards tailorable systems development. For clarity sakes, we organized our results in four trends that tried to catch this evolution. Each period from our timeline supports the reader to reasoning on fundamental perspectives of tailorable software development. This chronological arrangement helped observe how each trend connected the previous trend contribution to evolve the researches. For instance, the SSW methodology approaches the end-user active involvement in system development in a way not supported by other methodologies (e.g., participatory design or User-Centered Design - UCD). Moreover, it enhances communication between the stakeholders.

Another relation we may establish relates to the composition paradigm. The employment of users' mental model from the fourth trend tries to handle system component's suitability discussed in the third trend and previously, by Mørch and Mehandjiev [42] in the first one. Also, it attempts to make the processes easy for users.

Generally speaking, we observed that through the years, the studies first addressed tailoring fundamental concepts, then went to tailorable systems development issues, and finally, to developing applications (e.g., mashups).

New technologies and approaches for system development have been released. They may represent a means to advance in this research topic. In this context, we believe that this work can help settle the knowledge basis for EUD research studies.

Finally, as future work, we may envision addressing the following opportunities identified after our analysis.

- Reasoning about the implications of tailorable systems development over modern software development methodologies (e.g., agile methodologies).
- Further investigation about the UI patterns for tailorable parts of software presented by Baranauskas and Neris [4] in context of tailorable system development.
- Exploring how new script frameworks and their resources can be employed to build such systems minimizing the trade-off between flexibility and usability.

ACKNOWLEDGMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior – Brasil (CAPES) – Finance Code 001.

REFERENCES

[1] Carmelo Ardito, Maria Francesca Costabile, Giuseppe Desolda, Rosa Lanzilotti, and Maristella Matera. 2017. Reflections on System Properties Valued by End Users in Designing End-User Development Systems. In *IFIP Conference on Human-Computer Interaction*. Springer, 106–113.

[2] Carmelo Ardito, Maria Francesca Costabile, Giuseppe Desolda, Rosa Lanzilotti, Maristella Matera, and Matteo Picozzi. 2014. Visual composition of data sources

by end users. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*. 257–260.

[3] Carmelo Ardito, Maria Francesca Costabile, Giuseppe Desolda, and Maristella Matera. 2017. A three-layer meta-design model for addressing domain-specific customizations. In *New Perspectives in End-User Development*. Springer, 99–120.

[4] Maria Cecília Calani Baranauskas and Vania Paula de Almeida Neris. 2007. Using patterns to support the design of flexible user interaction. In *International Conference on Human-Computer Interaction*. Springer, 1033–1042.

[5] Barbara Rita Barricelli, Fabio Cassano, Daniela Fogli, and Antonio Piccinno. 2019. End-user development, end-user programming and end-user software engineering: A systematic mapping study. *Journal of Systems and Software* 149 (2019), 101–137.

[6] Federico Cabitza, Daniela Fogli, and Antonio Piccinno. 2014. "Each to His Own": Distinguishing activities, roles and artifacts in EUD practices. In *Smart organizations and smart artifacts*. Springer, 193–205.

[7] Cinzia Cappiello, Florian Daniel, Maristella Matera, Matteo Picozzi, and Michael Weiss. 2011. Enabling end user development through mashups: requirements, abstractions and innovation toolkits. In *International Symposium on End User Development*. Springer, 9–24.

[8] Juliet Corbin and Anselm Straus. 2008. *Basics of Qualitative research: techniques and procedures for developing grounded theory* (3 ed.). Sage publications, Ins. 400 pages.

[9] Maria Francesca Costabile, Daniela Fogli, Giuseppe Fresta, Piero Mussio, and Antonio Piccinno. 2003. Building environments for end-user development and tailoring. In *IEEE Symposium on Human Centric Computing Languages and Environments, 2003. Proceedings. 2003*. IEEE, 31–38.

[10] Maria Francesca Costabile, Daniela Fogli, and Rosa Lanzilotti. 2006. Supporting work practice through end-user development environments. *Journal of Organizational and End User Computing (JOEUC)* 18, 4 (2006), 43–65.

[11] Maria Francesca Costabile, Daniela Fogli, Rosa Lanzilotti, Andrea Marcante, Piero Mussio, Loredana Parasiliti Provenza, and Antonio Piccinno. 2007. Meta-design to face co-evolution and communication gaps between users and designers. In *International Conference on Universal Access in Human-Computer Interaction*. Springer, 46–55.

[12] Maria Francesca Costabile, Daniela Fogli, Piero Mussio, and Antonio Piccinno. 2007. Visual interactive systems for end-user development: a model-based design methodology. *IEEE transactions on systems, man, and cybernetics-part a: systems and humans* 37, 6 (2007), 1029–1046.

[13] Maria Francesca Costabile, Piero Mussio, Loredana Parasiliti Provenza, and Antonio Piccinno. 2008. Advanced visual systems supporting unwitting EUD. In *Proceedings of the working conference on Advanced visual interfaces*. 313–316.

[14] Vânia Paula de Almeida Neris and Maria Cecília Calani Baranauskas. 2012. Designing tailorable software systems with the users' participation. *Journal of the Brazilian Computer Society* 18, 3 (2012), 213–227.

[15] Giulia de Andrade Cardieri and Luciana Aparecida Martinez Zaina. 2020. Playing the Role of Co-designers on Mobile PWAs: An Investigation of End-Users Interaction.. In *ICEIS (2)*. 476–486.

[16] Giuseppe Desolda, Carmelo Ardito, and Maristella Matera. 2015. EFESTO: a platform for the end-user development of interactive workspaces for data exploration. In *International Rapid Mashup Challenge*. Springer, 63–81.

[17] Yvonne Dittrich, Sara Erikseji, and Christina Hanssoni. 2002. PD in the Wild; Evolving practices of Design in Use. In *Proceedings of the participatory design conference*. 23–25.

[18] Yvonne Dittrich, Olle Lindeberg, and Lars Lundberg. 2006. End-user development as adaptive maintenance. In *End user development*. Springer, 295–313.

[19] Christian Dörner, Jan Hess, and Volkmar Pipek. 2007. Improving information systems by end user development: A case study. In *Proceedings of the 15th European Conference on Information Systems (ECIS2007),*. 783–794.

[20] Gerhard Fischer. 2007. Meta-design: expanding boundaries and redistributing control in design. In *IFIP Conference on Human-Computer Interaction*. Springer, 193–206.

[21] Gerhard Fischer. 2009. End-user development and meta-design: Foundations for cultures of participation. In *International Symposium on End User Development*. Springer, 3–14.

[22] Gerhard Fischer, Daniela Fogli, Anders Mørch, Antonio Piccinno, and Stefano Valtolina. 2020. Design trade-offs in cultures of participation: empowering end users to improve their quality of life. *Behaviour & Information Technology* 39, 1 (2020), 1–4.

[23] Gerhard Fischer, Daniela Fogli, and Antonio Piccinno. 2017. Revisiting and broadening the meta-design framework for end-user development. In *New perspectives in end-user development*. Springer, 61–97.

[24] Gerhard Fischer and Elisa Giaccardi. 2006. Meta-design: A framework for the future of end-user development. In *End user development*. Springer, 427–457.

[25] Gerhard Fischer, Elisa Giaccardi, Yunwen Ye, Alistair Sutcliffe, and Nikolay Mehandjiev. 2004. Meta-design: a manifesto for end-user development. *Commun. ACM* 47, 9 (2004), 33–37.

[26] Gerhard Fischer, Jonathan Grudin, Raymond McCall, Jonathan Ostwald, David Redmiles, Brent Reeves, and Frank Shipman. 2001. Seeding, evolutionary growth and reseeding: The incremental development of collaborative design environments. *Coordination theory and collaboration technology* 447 (2001), 472–472.

[27] Gerhard Fischer, Kumiyo Nakakoji, and Yunwen Ye. 2009. Metadesign: Guidelines for supporting domain experts in software development. *IEEE software* 26, 5 (2009), 37–44.

[28] Gerhard Fischer, Antonio Piccinno, and Yunwen Ye. 2008. The ecology of participants in co-evolving socio-technical environments. In *Engineering interactive systems*. Springer, 279–286.

[29] Daniela Fogli. 2012. Designing visual interactive systems in the e-government domain. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*. 46–49.

[30] Daniela Fogli and Antonio Piccinno. 2013. Enabling domain experts to develop usable software artifacts. In *Organizational change and information systems*. Springer, 419–428.

[31] Daniela Fogli and Loredana Parasiliti Provenza. 2012. A meta-design approach to the development of e-government services. *Journal of Visual Languages & Computing* 23, 2 (2012), 47–62.

[32] Franco Eusébio Garcia and Vânia Paula de Almeida Neris. 2020. A framework for tailorable games: toward inclusive end-user development of inclusive games. *Universal Access in the Information Society* (2020), 1–45.

[33] Henrik Hertel, Anke Dittmar, and Doritt Linke. 2020. Meta-level support for facilitating participation in website (re-) design activities. In *Companion Proceedings of the 12th ACM SIGCHI Symposium on Engineering Interactive Computing Systems*. 1–6.

[34] Björn A Johnsson and Boris Magnusson. 2020. Towards end-user development of graphical user interfaces for internet of things. *Future Generation Computer Systems* 107 (2020), 670–680.

[35] Helge Kahler, Anders Mørch, Oliver Stiemerling, and Volker Wulf. 2000. Computer supported cooperative work: the journal of collaborative computing. *Computer Supported Cooperative Work (CSCW)* 9, 1 (2000), 1–4.

[36] Andrew Ko, Robin Abraham, Laura Beckwith, Alan Blackwell, Margaret Burnett, Martin Erwig, Chris Scaffidi, Joseph Lawrance, Henry Lieberman, Brad Myers, Mary Beth Rosson, Gregg Rothermel, Mary Shaw, and Susan Wiedenbeck. 2011. The state of the art in end-user software engineering. *ACM Computing Surveys (CSUR)* 43, 3 (2011), 1–44.

[37] Henry Lieberman, Fabio Paternò, Markus Klann, and Volker Wulf. 2006. End-user development: An emerging paradigm. In *End user development*. Springer, 1–8.

[38] Monica Maceli. 2017. Tools of the trade: a survey of technologies in end-user development literature. In *International symposium on end user development*. Springer, 49–65.

[39] Monica Maceli and Michael Atwood. 2013. "Human Crafters" Once again: Supporting Users as Designers in Continuous Co-design. In *International symposium on end user development*. Springer, 9–24.

[40] José Macías. 2012. Development of end-user-centered EUD software. In *Proceedings of the 13th International Conference on Interacción Persona-Ordenador*. 1–2.

[41] Anders Mørch. 1997. *Three levels of end-user tailoring: Customization, integration, and extension*. The MIT Press, Cambridge, MA, 51–76.

[42] Anders Mørch and Nikolay Mehandjiev. 2000. Tailoring as collaboration: The mediating role of multiple representations and application units. *Computer Supported Cooperative Work (CSCW)* 9, 1 (2000), 75–100.

[43] Bonnie Nardi. 1993. *A small matter of programming: perspectives on end user computing*. MIT press. 162 pages.

[44] Kostas Pantazos, Soren Lauesen, and Ravi Vatrapu. 2013. End-user development of information visualization. In *International Symposium on End User Development*. Springer, 104–119.

[45] Fabio Paternò. 2013. End user development: Survey of an emerging field for empowering people. *International Scholarly Research Notices* 2013 (2013).

[46] Veit Schwartze, Marco Blumendorf, and Sahin Albayrak. 2010. Adjustable context adaptations for user interfaces at runtime. In *Proceedings of the International Conference on Advanced Visual Interfaces*. 321–324.

[47] Alistair Sutcliffe and George Papamargaritis. 2014. End-user development by application-domain configuration. *Journal of Systems and Software* 91 (2014), 85–99.

[48] Daniel Tetteroo and Panos Markopoulos. 2015. A review of research methods in end user development. In *International symposium on end user development*. Springer, 58–75.

[49] Claes Wohlin. 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering*. 1–10.

[50] Volker Wulf and Björn Golombek. 2001. Direct Activation: A Concept to Encourage Tailoring Activities. *Behaviour & IT* 20 (07 2001), 249–263.