Ethics in Software Engineering: A Systematic Literature Review

Alidoosti, Razieh; Lago, Patricia; Razavian, Maryam; Tang, Antony

2022

Ethics in Software Engineering: a Systematic Literature Review

Razieh Alidoosti[a,b], Patricia Lago[a], Maryam Razavian[c] and Antony Tang[a,d]

[a]Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
[b]Gran Sasso Science Institute, L'Aquila, Italy
[c]Eindhoven University of Technology, Eindhoven, The Netherlands
[d]Swinburne University of Technology, Melbourne, Australia

ARTICLE INFO

Keywords:
Ethics
Moral
Ethical value
Value sensitive design
Software engineering
Systematic literature review

ABSTRACT

Context: Ethics has attracted considerable attention in the field of software engineering. The prevalence of software-intensive systems and how they become an integral part of people's lives makes it essential for developers to care about the potential impacts and injustice of software systems on people. The influence of systems' social and ethical implications on the values of individuals and society plays an important role in the study of software engineering ethics.
Objective: The overall aim of this paper is to study the state of the art concerning ethics in software engineering and create an understanding of ethical considerations in software design and development.
Method: We conducted a systematic literature review with an initial sample of 478 articles, of which 57 articles were selected as primary studies.
Result: According to the identified research gaps in this review, we created a body of knowledge on stakeholders and SE ethical values. More specifically, we provided a stakeholders map to identify different system stakeholders and presented a value model to categorize ethical values and recognize the value relations.
Conclusion: This SLR provides an understanding of the state of the art in software engineering ethics. The results, such as different categories of stakeholders, various ethical values, and the relations among the values, can inspire future research and practice to consider ethical aspects in software engineering.

1. Introduction

Today, the study of ethics can be found in a variety of fields like medicine and business management [66]; ethics as a branch of philosophy resolves the questions related to human value and morality by defining concepts like right[1]/good[2] and wrong[3]/bad[4] [63, 110, 90, 105]. Over the last decade, ethics has received the attention of the software engineering field because of the high dependency of people's lives on software-intensive systems and the related social and ethical implications on individuals and society. In this context, ethics refers to a set of principles and rules providing guidance for members of the software engineering community. The principles help software engineers adhere to the welfare, justice, and safety of users and society and guide them in their activities and decisions during the software design and development process [79].

Software-intensive systems, or *systems* for short, can promote, support, undermine, or corrupt common values of individuals and society. Failures in these systems can potentially harm individuals and society, such as financial loss and reputation damage, and in some cases can even go as far as risking human life. Consider the Corona-Warn App (a COVID-19 contact-tracing app used in Germany [65]) as an example to show potential harm towards the individuals and their values. Despite the role of this app in improving the effectiveness of contact-tracing efforts, it could cause violations of privacy and data security by disclosing the users' identity and place. As another example, consider robotics and autonomous systems. They could raise ethical concerns by displacing specific workers and, in doing so, harm individuals' employment; or introduce bias in their decision making (*e.g.,* because of their ethnically biased training) and cause risks for certain groups of individuals [107]. The impacts of systems on people and society are high and *ethics in software engineering* (SE ethics) must receive due attention.

When addressing ethical considerations, there are issues that one needs to take into account: (i) ethics is context-dependent, (ii) ethics is not a clear matter of right or wrong, good or bad, or white or black, and (iii) ethical issues are not easy to recognize and deal with. Whilst ethical principles can be used as a guideline in software engineering, ethics cannot be achieved by simply following a set of rules and standards (*e.g.,* ACM/IEEE software engineering code of ethics [78]) because ethical aspects are not rules or requirements that can be easily tested. As such, software engineers miss approaches that address in an explicit and systematic way human- and non-technical aspects during software design and development; *i.e.,* there is the need to consider ethical considerations and embed ethical values (such as well-being, welfare, and human rights) already in the early phases of creating software, like software design [100, 61, 103]. Identifying ethical considerations requires a continuous process of analysis that considers social life, seeks complexities instead of clarity, needs collective responsibility instead of individual, and focuses on dialogue instead of solutions [92]. In software design, decisions (both technical and non-technical) that are meant to address ethical considerations are complex and challenging. In the software design and development

ORCID(s): 0000-0002-2234-0845 (P. Lago)

[1]It is derived from "rectus" and means straight or according to rule.
[2]It implies something desirable and acceptable.
[3]It means not in accordance with what is morally right or good.
[4]It is the opposite of good and implies something undesirable, unacceptable, and not pleasing in every manner.

process, some trade-offs for satisfying constraints like clients' needs, environmental considerations, societal effects, and many other difficult and intangible ethical issues should be made. In order to address such complexities, software engineers need to analyze the non-technical and ethical aspects of the systems (*e.g.,* factors related to humanity, society, the environment) and, in particular, consider the implications of their decisions [100, 104, 61].

Understanding ethical considerations and aspects of systems depends on knowing the systems and their stakeholders. In this regard, there is a need to foster the habit of reflecting on all the relevant stakeholders and assessing what is at stake for each stakeholder during the software design and development process [104]. In particular, identifying stakeholders who affect or are affected by the systems, determining how stakeholders, their rights and well-being will be affected, and looking for ethical values in design decisions, can help software engineers. For example, they would be able to identify potential ethical issues, gain a better understanding of the ethical implications of their decisions, and find a way to balance the impacts of their decisions.

In spite of the research in the area of ethics and technology, we have insufficient knowledge on how to identify the relevant stakeholders, how to extract ethical values and how to translate them into software requirements. This Systematic Literature Review (SLR) aims to identify what has been studied to date in the context of SE ethics. To support ethical considerations in the software design and development process, we aim to identify the existing gaps in current research and uncover interesting trends for future research.

The remainder of this paper is structured as follows. Section 2 discusses background information, including fundamental concepts and the need for conducting a systematic review on SE ethics. Section 3 describes research methodology, which consists of research questions, search strategy, research process, selection criteria, data extraction strategy, and data synthesis. The results of the SLR are presented in Section 4, and discussion of the SLR is presented in Section 5. Section 6 describes the threats to validity and Section 7 concludes the paper.

2. Background

In this section, we describe the background knowledge of ethics in software design and development. We first define each fundamental concept that underlies our work, then summarize the need for a systematic review on SE ethics.

2.1. Concepts and definitions

2.1.1. Ethics

The word "ethics" comes from the Greek word "ethos", meaning character [99]. Ethics as an academic field belongs primarily to the discipline of philosophy. It is about the study of some concepts such as good, bad, right, and wrong applied to human conduct in a moral sense [101]. Ethics refers to the rules for and the study of moral beliefs and also understanding and adopting moral values within a context or place that should be defined [72].

To better understand the relation between ethics and technology, we should first understand what ethics could mean in the context of technology. In Greek philosophy, ethics was studied and considered as what is specific to human beings. Also, from the Aristotelian vision, ethics is the capacity to guide oneself by using reason. In today's world, this capacity is seen as the foundation of human dignity and human rights, which based on, one could say, certain behavior or choice is ethical if it is in accordance with the dignity of human nature. Another definition is given by J.L. Lorda [70]: "Ethics is the art to live well" which suggests that ethics is the art to make good use of technology.

Further, the study of ethics can be found in many different fields like mathematics, social sciences, and computer. Computer ethics is one of the most important research fields of ethics which discusses integrating ethical principles into the practice of software design. The term "computer ethics", first coined by Walter Maner, refers to the research field that studies the ethical problems created by computer technology [87]. It is worth noting that the distinction between the terms "ethics" and "morality" is not very clear, and sometimes they are used interchangeably. As we explained, ethics refers to the right and wrong conduct of humans in a specific situation, while morals are concerned with principles of right and wrong. In the following part, the moral/morality details will be explained thoroughly.

2.1.2. Morality

The word "Morality" is derived from "moralis", which means customs or manners [99]. Morality consists of notions such as rightness and wrongness, guilt and shame. In philosophy, there are varied definitions of morality in different situations. For Immanuel Kant, Morality is good or evil of voluntary action which does not depend on factors external to the person but on the person himself. Kierkegaard has considered morality as the rapport of the sinner's action with the Faith that saves. Jean Paul Sartre has also considered authenticity as the foundation of morality. Morality can be seen as a system of behaviour according to right or wrong behaviour. To better understand morality, it is helpful to become familiar with the meaning of some related words in this area, which helps avoid misunderstandings. Below we define each of these words.

- *Moral:* it means a proper and acceptable way of behaving.
- *Immoral:* it is the opposite of moral and means an unacceptable way of behaving.
- *Amoral:* it means having no sense or being indifferent to right and wrong.
- *Non-moral:* it means being out of the realm of morality.

In general, morality concerns humans and their relations with others (both human and non-human) and explores these relations to promote mutual welfare, growth, and creativity. These human relations and their moral concerns are extended through ubiquitous software systems in the modern world.

2.1.3. Ethical values

Values have been a fundamental concept in the social sciences, playing an important role in the other scientific fields. There are different definitions for values, and here, we explain some of the widely-used ones. Kluckhohn [85] defines value as "a conception, explicit or implicit, distinctive of an individual, or characteristic of a group, of the desirable which influences the selection from available modes, means, and ends of action". Braithwaite [62] considers value as "principles for action encompassing abstract goals in life and modes of conduct that an individual or a collective considers preferable across contexts and situations". Also, Schwartz [96] considers value as "a belief about desirable end states or modes of conduct that transcend specific situations; guides selection or evaluation of behavior, people, and events; and is ordered by the importance relative to other values to form a system of value priorities". Furthermore, Hutcheon [81] claims "values are not the same as ideals, norms, desired objects, or espoused beliefs about the "good", but are, instead, operating criteria for action".

In a narrow sense, value is an indicator to show the economic worth of an object. However, in this study, we consider a broader meaning of this word which refers to what a person or group of people consider important in life, and each of them can hold numerous values with different degrees of importance [77]. Some examples of values include human welfare, ownership and property, privacy, trust, and autonomy.

2.1.4. Value sensitive design

Value Sensitive Design (VSD) was first introduced by Batya Friedman and Peter Kahn and is considered the most comprehensive method to account for human values throughout the design process [108, 86, 68, 77]. VSD has concerns about a variety of values such as democracy, cooperation, and especially values that are morally important [74].

VSD provides an integrated methodology consisting of three investigations: (i) conceptual, (ii) empirical, and (iii) technical. *Conceptual investigations* provide philosophically-informed analyses of the central issues under investigation and focus on identifying values and stakeholders. More precisely, this investigation focuses on questions like: What are values? Whose values should be considered in the design process? How can technological designs support or diminish values? How should trade-offs among competing values (*e.g.,* anonymity vs. trust) in the design be done? Should moral values (*e.g.,* right to privacy) have greater weight than non-moral values (*e.g.,* aesthetic preferences)? Conceptual investigations are informed by empirical investigations of technology. *Empirical investigations* have a role in the understanding and experiences of the people affected by technological designs. To do these kinds of investigations, the quantitative and qualitative methods in social science research are applicable. These investigations concentrate, for example, on the following questions: How are values perceived by stakeholders in a given context? How do stakeholders prioritize values in design trade-offs? *Technical investigations* in the context of technology can be used in two ways. First, these investigations evaluate the technologies to see how they support or hinder human values. Second, they can design a system to support the identified human values in both conceptual and empirical investigations.

It is worth mentioning that each of these three investigations (conceptual, empirical, and technical) contains some activities to support the values in the design process of software systems (see Figure. 1).

2.1.5. Schwartz value structure

Schwartz value structure recognizes 59 human values in 10 value categories in which each value category has different motivational goals [95, 98]. This structure shows both the distinct value categories and the relations among the categories. These relations can be any kind of conflict or congruity. In other words, sometimes, the pursuit of a value may conflict or may be congruent with the pursuit of other values in terms of practical or social implications. Based on the proposed circular arrangement for this structure, the closer any two values are, the more similarity in terms of their underlying motivations, and conversely. This structure has two orthogonal dimensions that show oppositions among competing values (Figure 3): (i) self-enhancement vs. self-transcendence and (ii) openness to change vs. conservation. The first orthogonal dimension shows conflict among the values that focus on social superiority and esteem (power and achievement) and the values that focus on the enhancement of others and transcendence of selfish interests (universalism and benevolence). The second orthogonal dimension shows conflict among the values that focus on intrinsic interest in novelty and mastery (self-direction and stimulation) and those that focus on preserving existing social arrangements and harmony in relations (security, conformity, and tradition). It is worth mentioning that based on the structure, hedonism has been shared among two adjacent dimensions (openness to change and self-enhancement). This means that hedonism focuses on the broad motivational goal, *i.e.,* intrinsic interest in novelty and mastery, and social superiority and esteem [98].

2.2. Identification of the need for a systematic review on SE ethics

Over the last decade, ethics has received increasing attention from the software engineering field. This new paradigm has led to the emergence of new issues concerning software systems, most of which have been neglected by previous research. After exploring much relevant literature, we found only a few review papers on these issues, especially on human values in software engineering. Perera *et al.* [91] used the Schwartz values structure, and investigated the extent to which human values have been considered in SE research. They found that only a small portion of papers (*i.e.,* 216 out of 1350 papers) have investigated human values, and the values of security and privacy are the main focus of SE research. Friedman *et al.* [76] investigated the existing value sensitive design methods in the literature by applying three criteria: (i) the method has undergone substantial adaptation or develop-

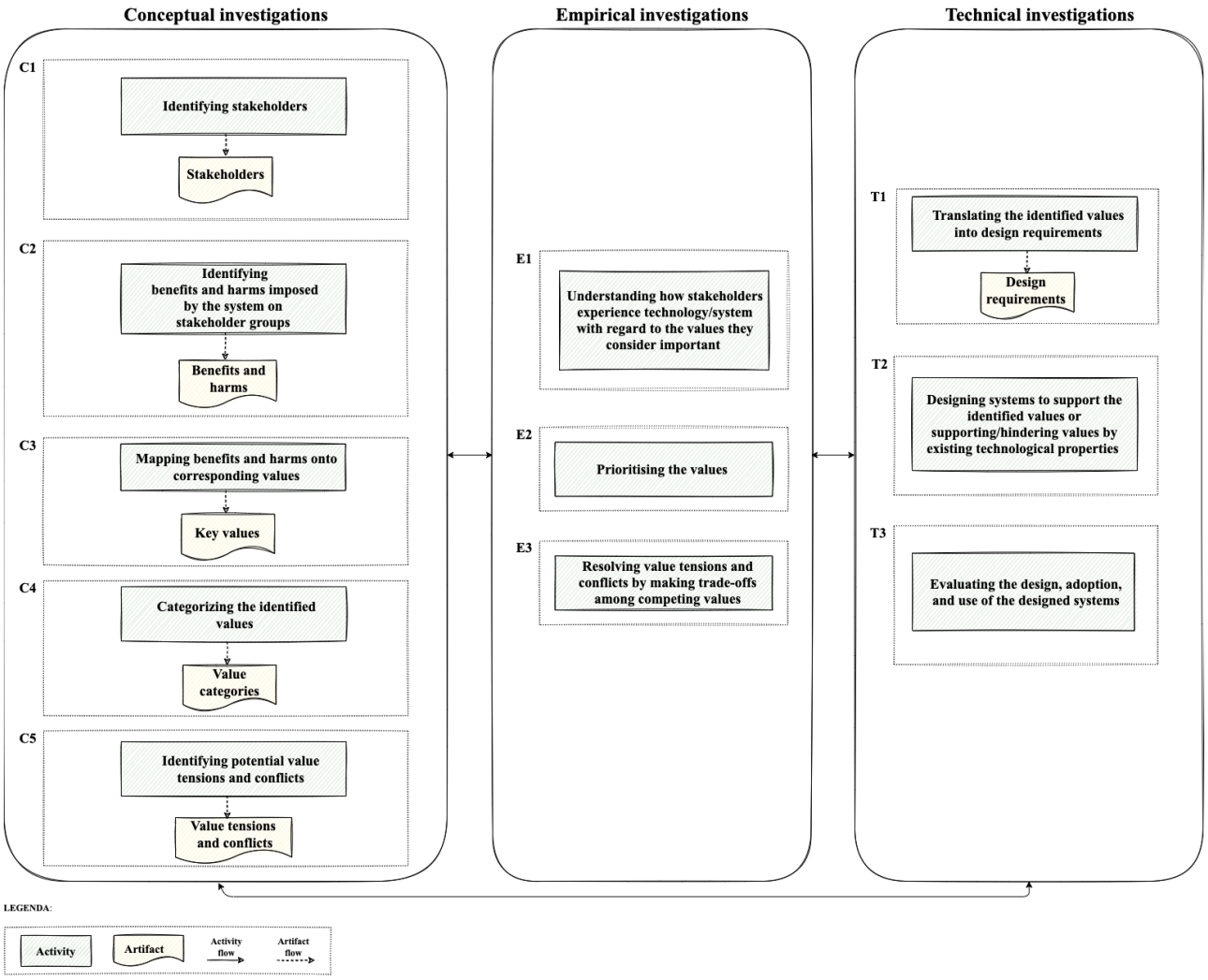

Figure 1: An overview of the investigations and activities in VSD methods

ment, (ii) the method should be relatively self-contained, and (iii) the method that covers a broad range of values and areas. Using these three criteria, they discovered 14 VSD methods that support human values during software design, like *value source analysis* and *envisioning cards*. They realized that although the definition of VSD focuses on the design process, it can cover the development of any technology.

In this work, we carry out an SLR to understand different aspects of ethics in relation to software-intensive systems. This review aims to help researchers gain an overview of the state of the art in SE ethics and a summary of the current evidence relevant to our research questions. These have a focus on (i) key stakeholders; (ii) ethical values; and (iii) prominent methods and approaches. Also, this review provides an opportunity to have a broad picture including focus, benefits, and limitations of the existing research related to ethics to develop methods, models, approaches, *etc.*, to manage ethical issues in software engineering. In addition to identifying the main issues and problems relevant to research on ethics in this field, the review also raises numerous questions and directions for future research.

3. Methodology

In this section, we describe the methodology we used for conducting our SLR (following the guidelines by Kitchenham [84]). Sections 3.1 through 3.6 present, respectively, the research goal and questions, search strategy, research process, selection criteria, data extraction strategy and process, and data synthesis.

3.1. Research goal and questions

The objective of our SLR is to characterize the state of the art concerning research on ethics and ethical aspects in the design and development of software systems. This leads to the following research questions (RQs), each provided with the underlying rationale:

- **RQ1:** Who are the key stakeholders in software-intensive systems that affect or are affected by ethical values? **Rationale:** Identifying stakeholders (*e.g.,* who can be negatively affected by software systems) is important for the design decision makers, but their identification is not always easy. In order to analyze stake-

holders' concerns, we need first to identify them.

- **RQ2:** What are ethical values and the relations between the values? **Rationale:** Ethical values as a design aspiration can be embodied in the software design and development process for accounting ethical considerations and resolving ethical issues. Ethical values can be interrelated, and they can conflict with each other. To address the ethical considerations and issues, designers need to discover ethical values from stakeholders and related communities to understand their impacts on individuals and society [73]. By answering this research question, we aim to identify a list of ethical values and the relations between ethical values.
- **RQ3:** How can stakeholders' requirements and their values be addressed in the design and development of software systems? And how can ethical values be translated into software requirements? **Rationale:** To support ethical values in the design and development of software systems, they need to be translated into concrete terms and requirements. By answering this research question, we aim to recognize the approaches, methods, and techniques that support and embed the stakeholders' values in software design and development, and the ways to translate values into software requirements.

3.2. Search strategy

One of the main motivations of this SLR is to identify primary studies addressing the specified research questions by using a well-defined search strategy. We present the selected source and search strings for our research in the following.

3.2.1. Source

We used Google Scholar as our digital library because it encompasses other libraries, such as IEEE Xplore, Scopus, *etc.*, and covers a broad range of sources (*e.g.,* articles, theses, books).

3.2.2. Search string

We formulated our search strings based on the identified keywords and the main intent of the research questions (*i.e.,* focusing on key stakeholders, ethical values, and value sensitive design); we also took synonyms of these terms into account. We considered "ethics" and "ethical" in our search string to find as many as relevant studies. Also, we embedded "moral" and "morality" in our search query. Ethics and morality are two sides of the same coin, and many studies used them as synonyms. In contrast, ethics and morals/morality are different in nature (morality refers to individuals' principles while ethics refer to society or organizations' rules). Moreover, because the study of ethics is multidisciplinary, we should specify the domain. Regarding the objective of this SLR which is analyzing ethical aspects and issues in software-intensive systems, our main focus is on the generic level related to these systems (*i.e.,* software architecture and software design in general, rather than any specific phase in particular, like requirement engineering or testing). Therefore, the search string should include phrases such as "software engineering", "software architecture", "software development", "software design", or "software system" to discover more relevant literature. To select a representative yet feasible search query, we proceeded as follows. First, we conducted a pilot search with the different combinations of defined terms in a specified time period (from 1980 to 2020). Some of them are elaborated on below.

Query 1 ("ethics" OR "ethical" OR "moral" OR "morality") AND "software" AND ("engineering" OR "architecture" OR "development" OR "design" OR "system")
This set of terms was discarded as it resulted in too many hits (approximately 3.3 million). To support search feasibility, we decided to limit the search to the title of the studies. So, the reasonable combination with these terms was the following form with 193 hits, of which 190 were available.

Query 2 (intitle:"ethics" OR intitle:"ethical" OR intitle:"moral" OR intitle:"morality") AND intitle:"software" AND (intitle:"engineering" OR intitle:"architecture" OR intitle:"development" OR intitle:"design" OR intitle:"system")
Also, as "value" is a motivation or foundation for one's morality, we need to apply it in our research query. However, combining this term even with Query 2 can result in a large number of irrelevant studies that chiefly refer to the economic worth. Therefore, based on the definition of value used in this study (*i.e.,* as something that is important to a group or an individual), we defined a new query including phrases like "value-sensitive" and "value-centered". This query led to 294 hits, of which 288 were available. Based on the similar reason in creating Query 2, *i.e.,* avoiding producing an enormous number of irrelevant studies, these phrases were only applied in the title of the studies as below.

Query 3 intitle:value AND (intitle:sensitive OR intitle: centered) AND (intitle:design OR intitle:software)

In total, we considered both Queries 2 and 3 as the search strings for our research (with a total of 478 hits).

3.3. Research process

For this research, we carried out an SLR based on the guidelines proposed by Keele *et al.* [82]. The process of this review shows the methodology that we used to conduct the SLR and includes three phases: (i) planning the review, (ii) conducting the review, and (iii) reporting the review (see Figure 2). For each phase, we explain the outcomes in detail.

Planning the review In this phase, we do some activities such as identifying the need for SLR, specifying the main goal and research questions, discussing inclusion and exclusion criteria, and specifying the strategy for data extraction and coding the data.

Conducting the review In this phase, we use Google Scholar as the database for conducting our search. The time period for selecting the studies is three decades. The two specified search strings defined in Subsection 3.2.2, were properly executed on this database by adapting to the search engine settings and constraints. The first search string (focusing on ethics and morality in software design and development) led to 190 studies. The second search string (focusing on value in software design and development) led to 288 studies. In

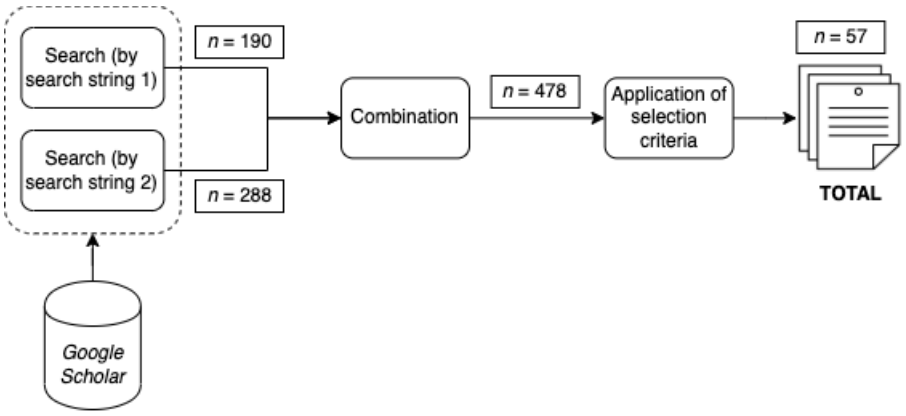

Figure 2: Overview of the review process

total, we found 478 studies. However, after applying the application of selection criteria that considered each study's title, abstract, and keywords, we deleted duplicated and irrelevant papers and got 143 studies. Then, after a complete reading of these studies, we finally found 57 studies. Afterward, the needed information of each included study was extracted to meet the primary purpose of the SLR, and then we summarized the information and results for the studies.

Reporting the review In this phase, we found the main result of the studies. Then, we discussed some potential drawbacks and limitations of the study, and finally, we documented the final result.

3.4. Selection criteria

Based on the defined guidelines in [83] to perform SLR in the context of software engineering, we determined the following explicit inclusion and exclusion criteria for the inclusiveness and evaluation of primary studies. In particular, the inclusion and exclusion of studies in the selected set are dependent on their capability to satisfy the criteria (*i.e.,* considering studies that meet all the inclusion criteria and eliminating studies that satisfy at least one of the exclusion criteria).

3.4.1. Inclusion criteria

The established inclusion criteria were:

IC1: Studies published in a peer-reviewed journal or conference. With this inclusion criterion, we ensure that only papers with enough level of quality are included.
IC2: Studies were available through digital libraries.
IC3: Studies written in English.
IC4: Studies that focused on ethics and values in software engineering, software architecture, software design, software development, and software systems. This inclusion criterion is applied to select studies whose main concerns are ethical aspects in the design and development process of software systems or technologies.

3.4.2. Exclusion criteria

The considered exclusion criteria were:

EC1: Studies that were not subject to peer review and not original research or/and unpublished. Considering this criterion, discard studies that lack a high level of quality.
EC2: Studies in the form of books/book sections, editorials, tutorials, short papers, reports, thesis, posters, systematic reviews, and tertiary studies. This exclusion criterion is adopted to exclude studies that do not provide the desired level of information in terms of details and elaborations about ethics in software engineering.
EC3: Studies that we could not find a full version of them.
EC4: Studies that were not written in English, as they could be challenging to analyze and also took considerable time to translate.
EC5: Studies that did not focus on ethics or ethical aspects (*i.e.,* they just concentrated on the context of software design or development process).
EC6: Studies that did not focus on software design or development process (*i.e.,* the studies whose primary attention was on ethics or ethical aspects). For example, studies focused on the ethical principles or code of ethics in software engineering. Also, studies were about ethics in the curricula context.
EC7: Studies that were generic and did not have case studies or information related to our research questions.

3.5. Data extraction strategy and process

In this section, we explain the way of defining attributes of the data extraction sheet and also how the studies were analyzed and coded (see Table 9 in Appendix A for information about the list of codes in our research).

3.5.1. Coding for data extraction

There are three categories of the codes in this list: (i) $RQ1$, (ii) $RQ2$, and (iii) $RQ3$, each of which corresponds to the research questions in this study. In the following, we describe the way of defining attributes of the data extraction sheet in detail.

First category (RQ1): The code related to stakeholders (*i.e.,* "stakeholder types") is intended to answer the $RQ1$. More precisely, the collected data from this code can be used by software engineers to answer some detailed questions in this regard, such as (i) who are the individuals that affect or be affected by software-intensive systems (both positively

or negatively)?, (ii) what are their concerns from an ethical point of view? And so on.

Second category (RQ2): Furthermore, by defining codes such as "values" and "value relations", we aim to answer the *RQ*2 that focuses on the values in this context. Extracting data through these codes and analyzing them can help software engineers to gain insights into (ethical) values and enable them to answer some questions in this regard, such as (i) what are the important ethical values that need to focus on during the software design and development process?, (ii) are there criteria for categorizing the values?, (iii) what are the relations among the values? And so on.

Third category (RQ3): Lastly, we define the codes of "the goals of approaches, methods, techniques, *etc.*" and "mapping values to requirements" to identify how requirements and values of different stakeholders can be addressed in software design and development. We need to know what efforts have been made in this regard so far. Also, we need to gain insight into the goals and the activities that these approaches, methods, techniques, *etc.*, conduct or do neglect. The collected data through these codes help software engineers answer the *RQ*3 and also some related questions in this regard, for example: (i) what are the goals of these approaches, methods, techniques, *etc.*?, (ii) what are their important activities?, (iii) what are the critical issues they focus on?, (iv) how can values be converted into software requirements? And so on.

3.5.2. Coding Procedure

When the list of primary studies was selected, their data was extracted. To do so, we carried out the coding for extracting data by following the proposed method by Miles and Huberman [89]. This procedure was started by analyzing 20 selected primary studies for answering each research question. Accordingly, we formed a data extraction sheet that contains attributes such as stakeholder types, values, and so on (the complete list of these attributes is explained in the previous subsection). To verify the reliability of the extraction process, we piloted the data extraction sheet for this set of studies before applying it to all the primary studies. Also, to make a more homogeneous set of studies and avoid bias, we extracted more qualitative details from the primary studies. The attributes of this sheet have evolved as the research proceeded and as the knowledge in the area refined. It is worth noting that we considered VSD methods as the theoretical underpinning of the codification for our review. Based on these methods, we mainly focused on and coded the stakeholders, values, and other related issues, as illustrated below.

This is the manner in which the studies were analyzed and coded:

Step 1: With considering the main objective of the studies, which is to address the ethical considerations and resolve the potential ethical issues in the context of software systems, we coded the studies. First, each study is coded with respect to the involved stakeholders in relation to software-intensive systems in different domains. This coding is completed based on three types of stakeholders: (i) stakeholders that use the system and receive direct benefit/harm from the system, (ii) stakeholders that have indirect benefits/harms in interaction with the system, and (iii) stakeholders that do not use or interact with the system, they may or may not receive benefit/harm from the system. If the selected studies explicitly mentioned stakeholders, they were immediately classified in those categories; otherwise, we implicitly interpreted and classified them based on their relations with the system (*i.e.,* using or interacting with the system). Second, the concerns of stakeholders (who affect or be affected by the system), which were discussed in the studies, were coded. To do this, during the coding procedure, which was conducted as an iterative process, we specified some indicators such as issue, harm, concern, challenge, problem, risk, *etc.*, to extract and code the stakeholders' concerns.

Step 2: We coded the values that were mentioned in the studies. Each value elicited from the studies was listed with its definition(s) and also the methods and techniques through which the values were extracted. Moreover, any relations among values that were explicitly mentioned in the studies were coded as value relations.

It is worth mentioning that to classify and code the values, we borrowed the human value structure of Schwartz as the underpinning theoretical framework [98]. Indeed, we considered the generic value categories and sub-values of this structure as the start list to categorize the values. In our analysis, we first extracted the values from the studies and analyzed their meaning. Second, to minimize the ambiguity of the value categories, we utilized a meta-inventory of human values in [64]. Based on this value model, we considered a label for those with similar meanings (*e.g.,* considering the label of responsibility for the values of accountability and responsibility). Then, we started to map the values found in a bottom-up way based on the list of values of Schwartz. In our iteration, some extracted values directly were mapped to the values in this list (shown with red in Figure 3), while some values could not directly be mapped (shown with blue in Figure 3). Also, some values in this structure were not mentioned in the primary studies (shown with black in Figure 3). This shows how we came up with Figure 3.

Step 3: We coded the approaches, methods, techniques, *etc.*, utilized in the studies to address the ethical considerations and embed the stakeholders' requirements and values during the software design and development process. Also, we coded the methods in the studies used to translate the values into concrete constructs (like software requirements) as mapping values to requirements.

3.6. Data synthesis

After extracting the data from the primary studies, we carried out data synthesis, which enabled us to understand the role of ethics in software engineering through our findings. The results of the synthesis are provided in Section 4.

4. Results

The search and selection process resulted in 57 primary studies that we analyzed as described in Section 3. In the

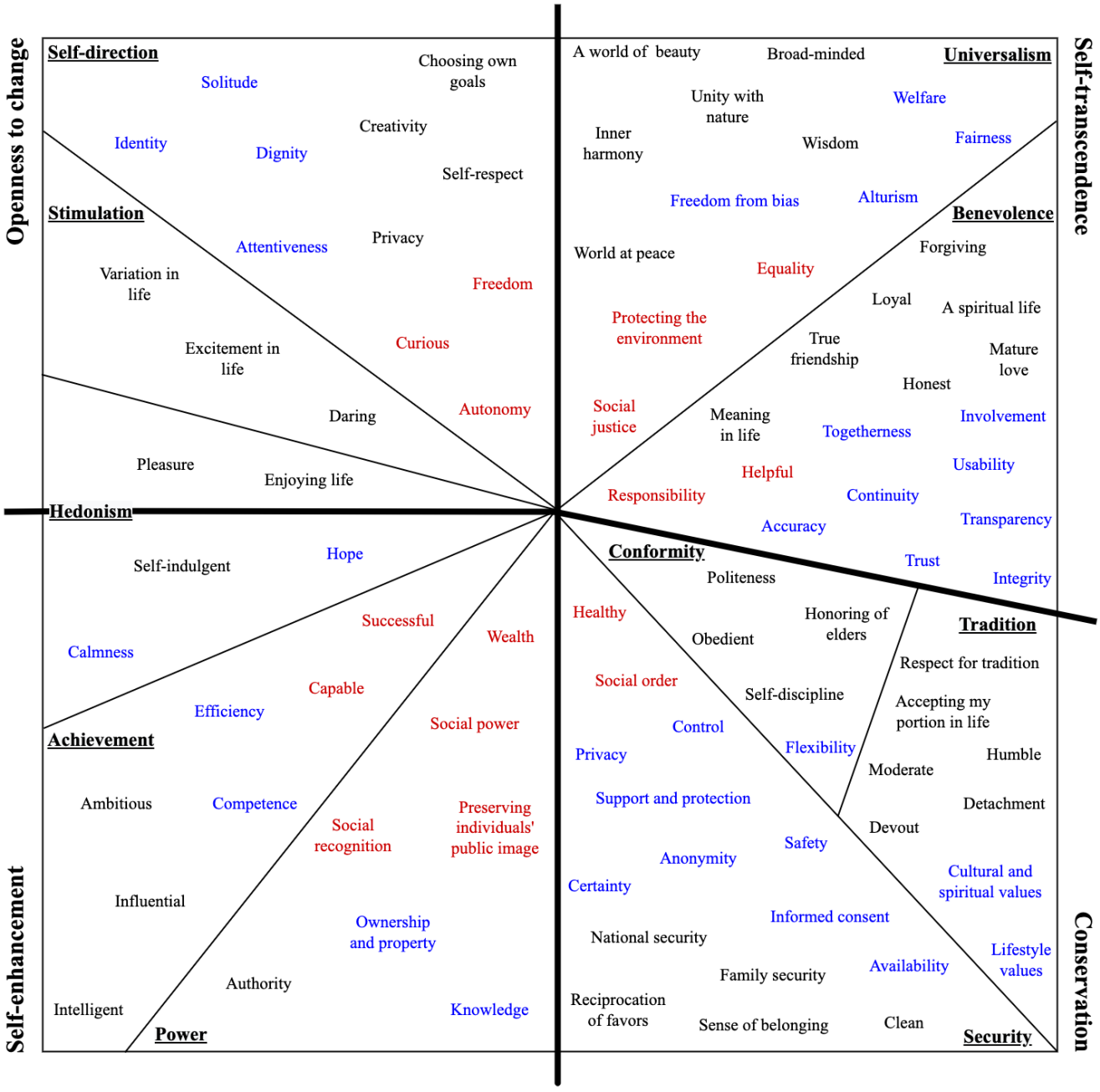

Figure 3: Categorizing the extracted values from the primary studies based on Schwartz value structure [97, 95]. The values shown in **red** indicate the extracted values from the primary studies that already exist in Schwartz's structure, and we found examples in our SLR. The values in **blue** are the values that we added from our SLR to the values of Schwartz's structure, and the values in **black** are values that were in this structure and we did not find any in our primary studies. Thick lines shown in the structure specify the orthogonal dimensions: (i) self-enhancement (including hedonism, achievement, and power) vs. self-transcendence (including universalism and benevolence) and (ii) openness to change (including self-direction, stimulation, and hedonism) vs. conservation (including tradition, conformity, and security).

following, we present a general overview of the publication trends and the results addressing our research questions.

4.1. Publication Trends

In this section, we aim to discover the published efforts and trends in SE ethics based on publication year, publication venue, and venue type.

Over the last decades, there has been a modest yet

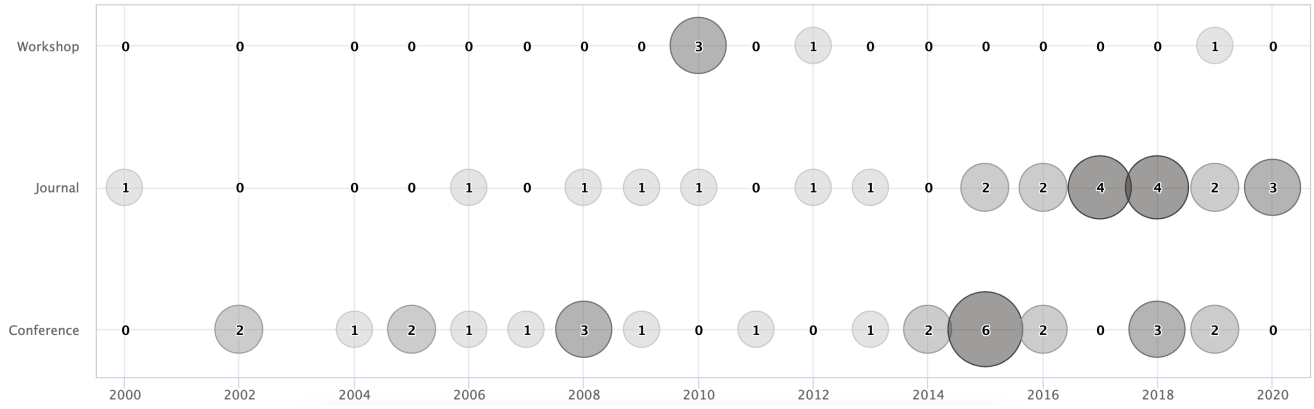

Figure 4: Number of primary studies based on publication year and venue type

visible growth of publications in the context of SE ethics. Figure 4 shows the relationship between publication year and venue type of the selected studies, and the number in bubbles represents the corresponding number of studies. As shown, the primary studies span from 2000 to 2020. Also, we observe a modest yet visible growth in recent years.

Conference and journal are the most targeted venue types by the studies. Conference and journal are the two main venue types for the selected studies, which account for 49% (28 out of 57 studies) and 42% (24 out of 57 studies), respectively.

Some publication venues publish more articles on SE ethics. Ethics has a multidisciplinary nature which has attracted attention from various fields like social sciences and software engineering. Table 11 in Appendix B, gives an overview of the studies in SE ethics with respect to publication venues and venue types. We can observe that the 57 primary studies were distributed over 44 different publication venues based on the results. We notice various publication venues are active in the context of SE ethics. Most venues ($37 \approx 65\%$) published only one paper. The preferred venues are "Science and Engineering Ethics" with five papers and "Conference on Human Factors in Computing Systems (CHI)" with four papers.

4.2. RQ1: Identifying Stakeholders

One of the important objectives in designing software systems is satisfying stakeholders through supporting their values and meeting their needs. The first step in achieving this goal is to identify the stakeholders who affect or are affected by a system and identify their concerns.

4.2.1. Categories of Stakeholders

In software engineering, stakeholders are the people who use the system, build the system, or are affected by the system. From the primary studies, we identified individuals, groups, and institutions reported to use, build, and be affected by the software system. We further categorized those stakeholders based on the way they interact with the system (*i.e.,* use or build) and are affected by the system. Our results suggest three overarching categories of stakeholders, shown in Figure 5: (i) *system users*—those who directly use the system, (ii) *system development organisation*—those that design and develop the system, and (iii) *indirect stakeholders*—those who are indirectly affected by the system. These three stakeholder categories are summarized in Table 1.

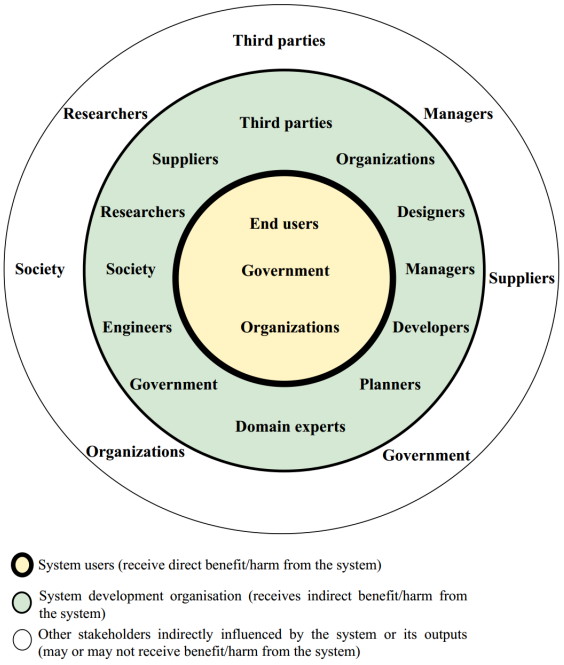

Figure 5: Stakeholders in terms of receiving benefit/harm from the system

System users are individuals, groups, or institutions that use the system and receive direct benefit/harm from the system. Consider Electronic Health Record (EHR) system for use in hospitals. Hospital personnel such as doctors, in-take receptionists as well as insurance companies directly use the system; hence, they are 'system users'. The most prevalent stakeholder in this category, identified by primary studies, is *end users* (mentioned in 50 out of 57 studies). Some primary studies also recognize institutions as stakeholders, *e.g., organizations* or *government*.

Table 1: Three general categories of stakeholders

Stakeholder category	Category definition	Stakeholder	Description	Studies
System users	Stakeholders that use the system and receive direct benefit/harm from the system.	End users	Individuals who systems are designed for them and ultimately use systems.	[1, 34, 6, 2, 51, 56, 36, 57, 54, 20, 23, 45, 31, 55, 9, 41, 42, 44, 4, 17, 8, 38, 24, 15, 33, 5, 16, 22, 7, 40, 19, 37, 50, 35, 32, 25, 26, 53, 10, 21, 43, 39, 30, 11, 28, 27, 29, 52, 18, 46]
		Organizations	A group of individuals who work together to achieve a common goal.	[49, 38, 33, 11, 52]
		Government	A group of individuals that govern society through legislation and regulatory rules.	[33]
System development organisation	Stakeholders that are involved in software design and development, they receive indirect benefit/harm from the system.	Designers	Individuals who use the designing techniques and methods to create a foundation for developing software systems.	[36, 57, 45, 55, 44, 4, 8, 49, 13, 40, 30, 29]
		Developers	Individuals like programmers and coders who are involved in the design and development process of software systems.	[4, 49, 24, 15, 22, 40, 21, 30, 46]
		Domain experts	Individuals who have knowledge and expertise in a specific domain like software development.	[36, 57, 47, 28]
		Organizations	A group of individuals who work together to achieve a common goal.	[36, 57, 55, 37]
		Managers	Individuals who coordinate human resources, activities, and processes in organizations to *e.g.,* build software systems.	[36, 57, 30]
		Suppliers	Individuals who provide necessary resources for organizations to meet a goal or to take action *e.g.,* hardware and software for software development teams to build software systems.	[2, 17, 11]
		Engineers	Individuals who analyze, design, and build systems and structures to accomplish objectives with considerations of limitations, *e.g.,* software engineers who are responsible for applying software engineering principles in the design and development of software systems.	[53, 30]
		Planners	Individuals who create plans and evaluate individuals' requirements (like systems' stakeholders) to ensure meet their needs.	[53, 30]
		Researchers	Individuals who collect, analyze, and interpret data to explore and solve issues in a specific context like software systems.	[17, 46]
		Government	A group of individuals that govern society through legislation and regulatory rules.	[29]
Other stakeholders indirectly influenced by the system or its outputs	Stakeholders that do not interact with the system (*i.e.,* use or build), they may or may not receive benefit/harm from the system.	Society	A group of individuals who live together and have continuous social interaction.	[8, 38, 12, 19, 35, 43, 11, 27, 52]
		Third parties	Organizations or individuals that play an important role in projects (like software projects) but are not the ones that directly involved in.	[34, 2, 17, 24, 52, 46]
		Suppliers	Individuals who provide necessary resources for organizations to meet a goal or to take action *e.g.,* hardware and software for software development teams to build software systems.	[2, 42, 17, 35, 18]
		Organizations	A group of individuals who work together to achieve a common goal.	[2, 42, 8, 46]
		Government	A group of individuals that govern society through legislation and regulatory rules.	[42, 19, 11]
		Researchers	Individuals who collect, analyze, and interpret data to explore and solve issues in a specific context like software systems.	[17, 19]
		Managers	Individuals who coordinate human resources, activities, and processes in organizations to *e.g.,* build software systems.	[34, 6]

System development organisation are those individuals, groups, or institutions that make decisions, design, develop, fund, and support the system, and although they may not use the system, they affect and are affected by it. When designing and developing the software, this group of stakeholders, explicitly or implicitly, impose their own individual values and the organizational values in the system. They may also be affected by the system, for instance, through the success and failure of the project. Continuing with the example of the EHR system, the personal and organizational values of product owners, software designers, project managers, and the software organization as a whole affect the design of EHR. Invariably, the success or failure of the EHR system affects these stakeholders as well. Figure 5 shows the stakeholders that belong to this category and were recognized in the primary studies. Among them, *designers* and *developers* were the most prevalent ones (mentioned in 12/57 and 9/57 studies, respectively).

Indirect stakeholders are the individuals, groups, or institutions that are indirectly affected by a software system but never interact with the system. In the example of the EHR system, indirect stakeholders may be the patients and their families who are dependent upon information in the software system but do not use it directly. The most prevalent indirect stakeholders identified in the primary studies are *society* and *third parties* (mentioned in 9/57 and 6/57 studies, respectively).

4.2.2. Stakeholders' Concerns

The concerns of stakeholders are an expression of specific interest about some topics related to a particular context that stakeholders care about [80]. To incorporate the stakeholders' values in software design, their concerns should be understood and considered [106]. Some of their concerns may have importance from the ethical point of view that could affect ethical values. For example, stakeholders may have concerns about privacy violations that could affect the ethical values of privacy and security. So, there is a need to address the stakeholders' concerns to support and embed the ethical values and their corresponding software requirements in the design and development of software systems.

Different stakeholders have different ethical concerns regarding the relationship with the system, and thus different values may need to be embedded in the software design. Table 2 shows the concerns of the three mentioned stakeholder categories surrounding software systems along with their relevant values.

Stakeholders are mainly concerned with the issues related to safety and privacy in software systems. All stakeholders (who affect or are affected by the systems) have different kinds of concerns about software systems. These concerns are about the values at stake or the harms of the system towards stakeholders. By drawing on these concerns, we found that they are most frequently related to the values of *safety* and *privacy*, mentioned in 14 out of 57 studies and 13 out of 57 studies, respectively. These two values are in the value category of security which consists 45.5% of the studies. We have also found other stakeholders' concerns in the primary studies relevant to the value categories of benevolence (14%), power (12%), universalism (10.5%), self-direction (9%), and achievement (2%). We describe them in detail in the next section.

4.3. RQ2: Values

Human values are tacit in the design and development of software systems, and their identification is often over looked. The primary studies characterize and define human values in philosophical and social sciences' terms that we use for conceptualization. There are different conceptualisations of values across the primary studies. For example, Maathuis *et al.* [30] refer to the value of anonymity in capturing data so that it cannot be associated with a specific person.

4.3.1. Identifying and Conceptualizing Values

We looked at how values are expressed and embedded in the software systems reported in the primary studies. As argued by Rescher [93], consideration of different aspects of value categorizations can help understand further the concept of values. We adopted the Schwartz value structure, a classification widely used in disciplines like social sciences and ethics [98, 94], and categorized the values extracted from the primary studies (see Figure 3). Table 3 provides an overview of the values identified in the primary studies and their classification based on this structure (see more details about the values including explanations and examples in Tables 12-15 in Appendix C). Also, Table 4 summarizes the coverage by the primary studies of the value categories of Schwartz [97, 95], and Table 10 (in Appendix A) shows the categorized values and their codification at a finer detail.

In what follows, we first provide a description of each value category covered in the primary studies as well as examples from the primary studies to elucidate how such categories of values can be accounted for in software systems. We then summarize our findings with respect to each value category.

- *Security* is about caring for and keeping individuals and data safe against potential harm. Security features protect individuals' health (both physically and mentally), their personal information, identity, and safety against unintended hazards, physical risks, and threats. They also concern with controlling and securing data and keeping their flow secure against system failure and data loss. Consider the case of software enabled by an affective car assistance system for traffic safety. This software captures the driver's face to calculate the driver's emotional state (like anger, happiness, or fear) and gives situation-specific warnings. Security in this example can be incorporated by protecting the individuals' data and their *anonymity* by not using cloud-based solutions and personal data that are unnecessary for the system's functioning and also concealing all information about the driver [30]. We found that security has attracted considerable attention in the context of software systems, as mentioned in 52 out of 57 studies (see Table 4). The most prevalent values in this

category are *privacy* and *safety* (see Table 3).

- *Conformity* is about adapting the systems to meet the stakeholders' needs. Consider the case of software enabled by a smart grid system (called smart metering) that records electricity, gas, or water consumption and communicates that information for monitoring and billing. Conformity in this example can be accounted for by modifying generation or consumption patterns in reaction to an external signal like a price change to meet the consumers' needs, *e.g.,* using water at an affordable cost [47]. One primary study reported the value of *flexibility* in the conformity value category (see Table 3).
- *Tradition* is about caring to respect the behaviors, culture, and habits of individuals in the social environment. Consider the case of communication media in an Arabic cultural context. The media can provide interactions between men and women with respect to certain conventions. For example, by providing some features such as friend suggestions that can prioritize females as friends for women or using kin relationships to suggest appropriate male friends, the media technologies care for *cultural values* (in this case, Islamic values) [1]. There are 3 primary studies expressed values such as *cultural values* and *lifestyle values* that belong to tradition value category (see Table 3).
- *Self-direction* is about caring for the rights of individuals in determining how they should be treated; have autonomy in their choices and control over activities. Consider the case of pervasive software for elderly care in a residential setting. This software provides various services for elders at home like queries, alarms, detecting falls, recognizing specific behaviors, and monitoring the health status. By allowing the elderly to stay at home as they get older (*i.e.,* aging-in-place) without depending on their families, this software supports the *autonomy* and the *independence* [18] and allows *self-direction*. Self-direction is one of the most recurring value categories expressed and incorporated in the primary studies, mentioned in 26 out of 57 studies (see Table 4).
- *Hedonism* can be incorporated into software systems by caring for pleasure or sensual gratification when using the system. Consider the case of healthcare drones software that allows delivery of blood donations, vaccinations, medications, and other medical supplies [11, 12]. Hedonism, in this example, can be incorporated by enhancing the *calmness* of local communities and authorities by, for instance, replacing loud and disruptive medical helicopters. Five (5) out of 57 studies expressed values such as *calmness* and *hope* that belong to hedonism value category (see Table 3).
- *Achievement* can be embedded in a software system by supporting users to carry out the activities successfully, leading to an increase in individuals' personal satisfaction. Consider the case of software enabled by a sensor-based physiotherapeutic assistance system, used for the patients' treatment in a home setting [30]. Achievement, in this example, can be accounted for by offering physiotherapeutic exercises at home through live video coaching via the software system. As shown in Table 3, we found four values belonging to achievement category: *capable*, *successful*, *efficiency*, and *competence*. In total, 11 primary studies reported values in this category.
- *Power* can be incorporated into software systems by caring for dominance over people and resources, and by maintaining social status. Consider the case of an EHR system. By allowing patients to control their information (*i.e.,* resource), an EHR system cares for the value of the *ownership* and *property*, *i.e.,* the patient has the ability to manage and own their health information [26]. In total, 15 primary studies reported values that we mapped to power value category (see Table 4).
- *Benevolence* can be incorporated into software systems by being kind, like helping individuals to do their tasks. Consider the case of software enabled by a virtual assistant that supports workload harmonization in the context of train traffic control. By sharing workload information to support operators to help each other when needed and yielding a more even distribution of workload over the team members, this software cares for the value of *helpful* [28]. For instance, if a virtual assistant detects a high and low workload level for operators A and B, respectively, sharing this information allows operator B to assist operator A in doing its tasks in a timely way. We found values like *usability* and *transparency* that belong to the benevolence category that are not part of Schwartz value structure (see Table 3). The focus of these values on the helpfulness of the system is a common aspect among them. We also found that benevolence is one of the most recurring value categories expressed and incorporated in the primary studies, mentioned in 46 out of 57 studies (see Table 4).
- *Universalism* is about preserving the welfare of people and protecting the environment through software systems (*e.g.,* safeguarding the individuals' physical and mental health, overcoming unfairness perpetrated on individuals). Consider the case of software to simulate and predict urban development patterns over time. For instance, by not discriminating unfairly against any group of individuals to utilize specific infrastructural facilities like transport and communication, this software cares for the value of *freedom from bias* [8]. We found that universalism is one of the most recurring value categories, expressed in 26 out of 57 primary studies (see Table 4).

Table 4 shows that the value category of *stimulation*, involving values such as *excitement* and *novelty*, which is not addressed by any of the primary studies. Naturally, none of the primary studies covers all the value categories.

Based on the Schwartz value structure [97, 95], each value category has specific motivational goals:

- *Security* has the goals of safety, harmony, and stability of society, relationships, and self.
- *Conformity* has the goals of restraint of actions, inclinations, and impulses likely to upset or harm others and violate social expectations or norms.
- *Tradition* has the goals of respect, commitment, and acceptance of the customs and ideas that one's culture or religion provides.
- *Self-direction* has the goals of independent thought and action–choosing, creating, exploring.
- *Hedonism* has the goals of pleasure or sensuous gratification for oneself.
- *Achievement* has the goals of personal success through demonstrating competence according to social standards.
- *Power* has the goals of social status and prestige, control or dominance over people and resources.
- *Benevolence* has the goals of preserving and enhancing the welfare of those with whom one is in frequent personal contact.
- *Universalism* has the goals of understanding, appreciation, tolerance, and protection for the welfare of all people and for nature.
- *Stimulation* has the goals of excitement, novelty, and challenge in life.

Some values extracted from the primary studies express motivational goals of more than one value category. Some values may have multiple meanings and express motivational goals of more than one value category. To reduce the complexity of value relations and clarify the meaning of values in this structure, we only consider these values in one specific value category. This is possible by analyzing the precise meaning of values and their accordance with the primary goals of value categories. For example, the value of usability, which refers to the level of comfort in using the system, could belong to the two value categories (*i.e.,* benevolence and universalism). These two categories are good candidates for this value in terms of the goal. The benevolence category emphasizes preserving welfare in a specific group and helping others, whereas the universalism category emphasizes protecting the welfare and well-being of individuals in society. Regarding the meaning of this value and the focus of these two categories, we consider usability having primary goal that aligns with the benevolence category.

4.3.2. Eliciting Values from Stakeholders

Values are expressions of what humans and organisations consider good, bad, right, and wrong, so extracting them is not easy because the meaning of good, bad, right, and wrong differs when the perspectives change. In our analysis, we elicited methods and techniques used in the primary studies to support the extraction of the values and the reflection upon ethical issues. These methods and techniques, together with examples from the primary studies, are presented in Table 5. It is worth noting that some studies used a mixed-method approach during the design process, *e.g.,* methods like sketches and prototypes were used in combination with other methods like interviews.

Methods and techniques to elicit values from the stakeholders. According to the primary studies, we found that although there are various methods and techniques for the extraction of values from the stakeholders, there is a need to understand which methods and techniques or which combination thereof is preferable or more suitable for value extraction in a specific ethical purpose or situation. In the following, each method and technique is explained in more detail.

Interviews. Interviews were mainly used to collect individuals' opinions, views, and values to understand their judgments about a use context [5], an existing technology, or a proposed design.

Surveys. Surveys were used to extract views and values from the individuals, measure individuals' needs, solicit their views on values, discover value-related issues and their implications, and rank values.

Workshops. Workshops provided an opportunity for the individuals to collaboratively discuss, analyze, and share the value-related issues and ethical considerations for a specific system. In particular, workshops allowed the individuals to identify values and value tensions, expose ethical issues, and recognize the value implications.

Case studies. Case studies were used to examine a real-life situation. The focus was to have a zoomed-in detailed look at real-world complexities like how ethical values can be satisfied in the system or to what extent ethical considerations can be considered during software design. Case studies could help assess the strengths and challenges of relevant methods like the value sensitive design.

Focus groups. Focus groups allowed the individuals to cooperatively express their opinions and attitudes in discussions about the key values, value-related issues, and implications. In contrast to workshops, focus groups were not used to develop a consensus for a decision among all participants. Focus groups were utilized to concentrate on a specific topic or issue relevant to the discussion.

Sketches. Sketches were mainly used to provide non-verbal understandings of a system using visual expressions and focused on the participants' views and values. In particular, sketches assisted the individuals in making value-related decisions by understanding what is important for them in relation to a system, how a system is situated in a context, and how values are implicated by system functioning.

Mock-ups and prototypes. Mock-ups and prototypes were used to make software designs more concrete for the individuals and provide a visual way to refer to systems. They allowed the participants to investigate the system's functions, interactions among stakeholders and system, implications of systems on stakeholders, value tensions, and the system's suitability in a specific context.

[5] It is the conditions (including the technical, social, and organizational environments) under which the intended users utilize an artifact (like software products) to achieve their desired goals.

Table 2: Concerns of systems' stakeholders.

Related value category and value	End users' concerns	System development organisation' concerns	Other stakeholders' concerns	Studies
Security (privacy)	• Privacy violations related to identity data. • Privacy issues related to data security.	• Monitoring data in the system. • Privacy violations of personal information. • Unauthorized use of data, disclosure to unauthorized parties, or unauthorized inference with unexpected external information. • Privacy concern during delivering personal data of the consumers to the providers.	• Hindering privacy of individuals by monitoring.	[34, 2, 54, 55, 9, 42, 24, 33, 12, 25, 28, 27, 18]
Security (anonymity)	• An anonymous reply to a question or trusting an anonymous post. • Anonymity makes it difficult to assess reviewers' expertise and damage the quality of discussion in the conference management system.	• Confidentiality of data in the system.	-	[34, 2, 20, 19]
Security (safety)	• Negative impacts of system on individuals' health. • Danger of third parties for hacking the personal information.	• Concern about IP safety. • Ensuring about using the system by pre-approved individuals. • Safety-oriented issues about using system by individuals. • Addressing parental concerns for children's online safety. • Mitigating safety risks such as the possibility of hitting or injuring individuals by the system.	• Negative impacts of system on individuals' health. • Vulnerability and risk in the use of the technology by children. • Safety concerns of parents about teens in the context of mobile platforms.	[34, 2, 56, 36, 57, 45, 55, 4, 33, 12, 13, 37, 25, 11]
Security (informed consent)	• The lack of agreement or consent of users in access to their information.	-	-	[9, 22, 21]
Security (availability)	• Unauthorized or unjustifiable access to personal data in the system.	• The availability of the database across regions, across providers and manufacturers, in case of disaster. • The lack of access to knowledge which makes it incredibly difficult for users to participate in conversations about decision-making.	-	[9, 17, 40]
Security (family security)	• Concerns about security such as family security, healthy and social order in the use of technology.	• Concerns about security of system.	-	[17, 10]
Power (preserving individuals' public image)	-	• Issues related to reputation (*e.g.,* gaining a poor public ranking through contributing incorrect or buggy code).	-	[34]
Power (knowledge)	• Having not enough technical knowledge for doing the activities in the context of system in an informed manner.	-	-	[31, 32]

Table 2 – Concerns of systems' stakeholders.

Related value category and value	End users' concerns	System development organisation' concerns	Other stakeholders' concerns	Studies
Power (ownership and property)	• Concerns about owning the data by the users. • Concerns about property rights in the context of systems.	• Concerns regarding data collection and storage.	-	[42, 8, 33, 40]
Benevolence (trust)	• Reliability is hindered because the system does not work in a simple way. • Hindering trust by monitoring.	• Reliability is hindered by difficulty in properly understanding and interpreting the users' information.	-	[34, 9, 13, 27, 18]
Benevolence (transparency)	• Low transparency of the system and its functions.	-	-	[20, 13, 19]
Benevolence (responsibility)	-	• Being accountable for keeping participants' data from misuse.	-	[20, 40, 19]
Benevolence (usability)	• Using the system by illiterate users or unsuitability of the system for some groups of users like individuals with specific mental conditions.	• Concerns about balancing usability and other values of users in the system.	-	[15, 33]
Benevolence (togetherness)	• Issues related to being in contact with family through technology to manage the stress and disorientation.	-	-	[31, 32]
Universalism (protecting the environment)	• Concerns about sustainability, environmental protection, value of nature, energy saving, and resource consumption. • Environmental risks of the system.	-	• Concerns about sustainability, environmental protection, value of nature, energy saving, and resource consumption.	[8, 12, 53, 10, 21]
Universalism (freedom from bias)	• Unfairly discrimination among individuals.	-	-	[3]
Self-direction (dignity)	-	• Concerns about ethical considerations related to software systems (*e.g.,* ethical issues related to different subjects and different threatened objects). • Issues related to human dignity (*e.g.,* lack of perception of human and humanoids).	-	[45, 13]
Self-direction (autonomy)	• Undermining autonomy and freedom of choice for individuals due to increased information, procedural controls, *etc.* • Motivating users to do something that they do not want.	-	-	[42, 3, 18]
Achievement (competence)	• Concerns about the competency of software design.	-	-	[9]

Table 3: Extracted values from the primary studies based on the Schwartz value structure. The values shown in "red" indicate the extracted values from the primary studies that already exist in Schwartz's structure, and the values in "blue" indicate the values that we added from our SLR to the values of Schwartz's structure.

Schwartz's dimension	**Schwartz's value category**	**Extracted values from the studies**	**Studies**
Conservation	Security (SE)	Social order	[39]
		Healthy	[23, 24, 12, 14, 11, 46]
		Privacy	[1, 34, 6, 2, 54, 20, 23, 31, 55, 42, 44, 4, 17, 24, 15, 33, 12, 47, 5, 22, 19, 26, 53, 10, 28, 18]
		Availability	[54, 9, 44, 17, 24, 53, 21, 27, 52, 46]
		Safety	[34, 2, 56, 36, 57, 45, 55, 9, 4, 17, 38, 12, 47, 14, 13, 37, 25, 53]
		Certainty	[30]
		Control	[36, 54, 44, 33, 40, 37, 27]
		Informed consent	[34, 23, 31, 44, 24, 22, 14, 21]
		Anonymity	[34, 20, 31, 19, 5, 30]
		Support and protection	[1, 4, 27]
	Conformity (CO)	Flexibility	[47]
	Tradition (TR)	Cultural and spiritual values	[1, 13]
		Lifestyle values	[10]
Openness to change	Self-direction (SD)	Autonomy	[36, 31, 41, 4, 15, 33, 47, 5, 16, 22, 37, 25, 26, 53, 43, 39, 30, 11, 52, 18, 46]
		Freedom	[32, 39]
		Curious	[39, 27]
		Solitude	[24]
		Attentiveness	[48, 24]
		Dignity	[36, 31, 13, 37, 30, 18]
		Identity	[31, 32, 11, 27]
	Hedonism (HE)	Calmness	[47, 53, 11]
		Hope	[15, 16]
Self-enhancement	Achievement (AC)	Capable	[45, 12]
		Successful	[36, 37]
		Efficiency	[54, 45, 44, 38, 33, 47]
		Competence	[48, 30]
	Power (PO)	Preserving individuals' public image	[34, 6, 24, 19]
		Social power	[54, 7, 39]
		Recognition	[34, 54]
		Wealth	[42, 19]
		Ownership and property	[57, 23, 31, 42, 8, 24, 47, 26, 53]
		Knowledge	[31, 42, 19]
	Hedonism (HE)	Calmness	[47, 53, 11]
		Hope	[15, 16]
Self-transcendence	Benevolence (BE)	Responsibility	[20, 31, 44, 8, 48, 33, 7, 14, 19, 21, 39, 11, 52, 3, 18]
		Helpful	[28, 27]
		Togetherness	[1, 31, 33]
		Transparency	[51, 20, 44, 12, 13, 19, 50, 43, 30, 7, 3, 18, 40, 49, 21, 27, 39]
		Involvement	[34, 36, 31, 41, 4, 40, 37, 32, 53, 10]
		Trust	[34, 6, 36, 23, 9, 49, 15, 33, 12, 47, 5, 16, 22, 7, 13, 19, 37, 26, 53, 43, 30, 11, 27, 52, 18, 46]
		Accuracy	[30]
		Integrity	[54, 19]
		Usability	[2, 54, 15, 33, 47, 16, 30, 11]
		Continuity	[30]
	Universalism (UN)	Justice	[8, 47]
		Equality	[20, 39]
		Protecting the environment	[31, 8, 12, 47, 53, 21, 11]
		Fairness	[51, 31, 8, 12, 7, 19, 50, 53, 43]
		Freedom from bias	[57, 31, 7, 21]
		Altruism	[41]
		Welfare	[51, 23, 24, 15, 12, 47, 16, 14, 50, 26, 53, 30, 11, 29, 52, 18]

Table 4: Coverage of the primary studies related to the value categories of Schwartz value structure. “SD” stands for self-direction, “ST” stimulation, “HE” hedonism, “AC” achievement, “PO” power, “SE” security, “CO” conformity, “TR” tradition, “BE” benevolence, and “UN” universalism.

Studies	*SD*	*ST*	*HE*	*AC*	*PO*	*SE*	*CO*	*TR*	*BE*	*UN*
[1]	□	□	□	□	□	■	□	■	■	□
[34, 6]	□	□	□	□	■	■	□	□	■	□
[2]	□	□	■	□	□	■	□	□	■	■
[56]	□	□	□	□	□	■	□	□	□	□
[36, 9, 49, 40, 28, 3]	□	□	□	□	□	■	□	□	■	□
[57]	□	□	□	□	■	■	□	□	□	■
[54]	□	□	■	■	■	■	□	□	■	■
[45, 38]	□	□	□	■	□	■	□	□	□	□
[31]	■	□	□	□	■	■	□	□	■	■
[55]	□	□	□	□	□	■	□	□	□	□
[41]	■	□	□	□	□	□	□	□	■	■
[42]	□	□	□	□	■	■	□	□	□	□
[44]	□	□	□	■	□	■	□	□	■	□
[4]	■	□	□	□	□	■	□	□	■	□
[17]	□	□	□	□	■	■	□	□	□	□
[8]	□	□	□	□	■	□	□	□	■	■
[48]	■	□	□	■	□	□	□	□	■	□
[24]	■	□	□	□	■	■	□	□	■	■
[15]	■	□	■	□	□	■	□	□	■	■
[33]	■	□	■	■	■	■	□	□	■	■
[12]	□	□	□	■	□	■	□	□	■	■
[47]	■	□	■	■	■	■	■	□	■	■
[5]	■	□	□	□	□	■	□	□	■	□
[16]	■	□	■	□	□	■	□	□	■	■
[51, 20, 14, 50, 21]	■	□	■	□	□	■	□	□	■	■
[22]	■	□	□	□	□	■	□	□	■	□
[23, 7, 19]	□	□	□	□	■	■	□	□	■	■
[13]	■	□	□	□	□	■	□	■	■	□
[37]	■	□	□	■	□	■	□	□	■	□
[35]	□	□	□	□	□	■	□	□	□	□
[32]	■	□	□	□	□	□	□	□	■	□
[25]	■	□	□	□	□	■	□	□	□	□
[26]	■	□	□	□	■	■	□	□	■	■
[53]	■	□	■	□	■	■	■	□	■	■
[10]	□	□	□	□	□	■	□	■	■	□
[43]	■	□	□	□	□	■	□	■	■	■
[39]	■	□	□	■	■	■	■	□	■	■
[30]	■	□	■	■	□	■	□	□	■	■
[11]	■	□	■	□	□	■	□	□	■	■
[27]	■	□	□	□	■	■	■	□	■	□
[29]	□	□	□	□	□	□	□	□	■	■
[52]	■	□	□	□	■	■	■	□	■	■
[18, 46]	■	□	□	□	□	■	□	□	■	■

Table 5
Methods and techniques to elicit values from stakeholders in the primary studies.

Methods and techniques	Example	Studies
Interviews	Through interviews in [34], the participants indicated their views on values-related issues and how those issues can be handled.	[34, 19, 20, 41, 7, 54, 37, 30, 36, 57, 17, 9, 28, 33, 5, 51, 1, 22, 38, 52, 47]
Surveys	Through a survey in [46], the participants indicated their interpretations, views, and comments about the relevant values of the system and rated the values in order of their importance.	[34, 35, 46, 5, 51, 25, 50]
Workshops	Through a workshop in [37, 36, 57], some personas were created to help participants to become familiar with the subject (*i.e.,* parental mediation). Then according to the personas and the extracted values from the stakeholders, the other stakeholder group (*i.e.,* system designers) was asked to build a value hierarchy to make value alignment and discuss the conflicting stakeholder values.	[37, 36, 57, 17, 9, 27, 28, 33, 5, 38, 40]
Case studies	In a case study in [19], the participants' values were captured, and the extent to which values could be addressed in the system (in accordance with ethical principles) was discovered.	[19, 20, 30, 57, 38, 25, 31, 32, 11, 14, 12, 26, 44]
Focus groups	In a focus group in [33], the participants discussed their experience with the use of the system (*i.e.,* web-based technologies) in their professional lives and the relevant ethical values in the design of the system.	[30, 28, 33, 10, 5, 52]
Sketches	The value sketches in [56] were about the understanding of participants on mobile phones and safety. The sketches represented the participants' perceptions about where and when homeless young people might feel unsafe.	[4, 56]
Mock-ups and prototypes	In a co-design activity in [56], the participants developed some prototypes for the phone to express their ideas for using mobile phones to provide safety for homeless young people.	[7, 54, 17, 28, 33, 5, 22, 56, 11, 14]
Value scenarios and story-boards	The presented scenarios in [28] aimed to explain the functioning and envision the values of the virtual assistant in a train traffic control context for different operators.	[28, 56]
Photo-elicitation	The photo-elicitation in [31, 32] allowed an understanding of the perceptions and use of technology by the homeless. Using this method, homeless people could express their values concerning technology, its use, and its relationship to their daily lives in the homeless community.	[31, 32]

Value scenarios and story-boards. Value scenarios and story-boards were used to help the individuals discuss and communicate about different system features, especially humanistic and societal considerations. In particular, value scenarios and story-boards assisted participants to focus on implications for stakeholders, key values, use contexts, and longer-term and potential impacts.

Photo-elicitation. Photo-elicitation was used to elicit comments and views from the individuals. Visual images (such as photographs and paintings) allowed the participants to direct the discussion about a system and enabled them to speak in greater detail about their situation, concerns, and key values from their perspective.

4.3.3. Relationships among Values

Values have relationships among them [98]. They can conflict or reinforce one another. For example, consider a conflict between two values of *well-being* and *autonomy* in a sensor-based assistance system that supports physiotherapeutic activities at home [30]. This system that aims to support the well-being of patients can conflict with the value of autonomy since this value requires the system to support the ideal ways of executing the treatment for the patients.

Value conflicts and value tensions. In our study, we found two types of relationships: *value conflict* and *value tension*. Value conflict refers to situations in which technology supports one value while undermines another one. Value tension, in turn, refers to instances in which two values are in conflict in a given situation [75]. Although value conflict and value tension theoretically are different concepts, *i.e.,* value tension is a general term mentioning a value conflict; they were used quite similarly in the primary studies. Most studies generally discussed these two concepts without mentioning any example: only 13 out of 57 studies explained them with example(s) (see Table 6). Based on the identified relations among the values in this table, the relations among their corresponding categories are represented in Figure 6. According to the figure, the most conflicting value categories with other categories are *security*– contradicts with six categories (*i.e.,* benevolence, self-direction, universalism, achievement, power, and security) and *self-direction*– contradicts with four categories (*i.e.,* benevolence, security, power, and universalism).

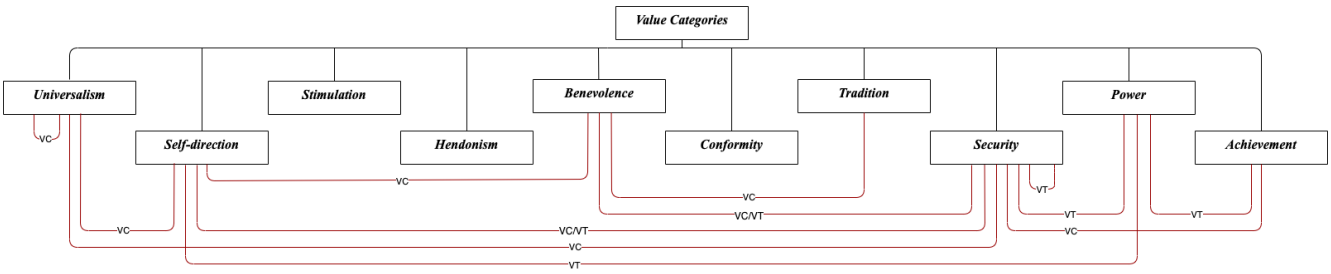

Figure 6: The relationships among the value categories.

Privacy, autonomy, safety, welfare, and involvement are the most conflicting values. According to Table 6, Values can be conflicting in the same category or it can be conflicting with another value category. An example of the former case, the safety of a person may conflict with the safety of society. An example of cross-value conflict is privacy of a suspect may conflict with welfare of society. We found that the most conflicting values are: (i) *privacy* conflicts with ten values, (ii) *autonomy* conflicts with six values, (iii) *safety* conflicts with five values, (iv) *welfare* conflicts with five values, and (v) *involvement* conflicts with four values.

Resolving value conflicts and tensions. To resolve the value conflicts and value tensions in software design, a few primary studies ([6, 34, 36, 37]) focused on addressing conflicts and tensions among the values and proposed the methods of *value dams and value flows* and *value alignment*.

Value dams and value flows[6] by keeping the design options supporting ethical values and ignoring those undermining values, helps to resolve the value tensions. For example, consider the use of this method for resolving value tensions among *privacy* and *awareness* in [34]. In this example, value dams are design options like *logging searches* and *logging queries* which concern the value of privacy, and many stakeholders agreed that these options could violate their privacy. Value flows are design options such as *reporting the frequency of use of the participants' codes* and *showing the ranking of participants' posts by their peers* which are related to the value of awareness. A majority of stakeholders agreed that they would like to have these two options. Thus, to reduce the privacy-related value dams and benefit from the awareness-related value flows, it was decided not to log or report who searches or queries but to log and report the frequency of code use and implement content ranking.

Value alignment[7] by prioritizing and aligning the values of different stakeholder groups helps to make trade-offs among the values. For example, in [36], there are two stakeholder groups: media company employees and children's parents. The media company employees built a hierarchy of their values in relation to the media without considering the parents' values. Then, the employees tried to map the parents' values onto their values and find the value conflicts.

4.4. RQ3: How stakeholders' requirements and values can be addressed in software design and development?

VSD is one of the most comprehensive and commonly used methods to account for human values in software design [77, 108], we used it to analyze the primary studies (see Figure 1). Table 7 shows the three investigations and conducted activities of VSD in the primary studies that support the ethical values in software design and development. Table 8 also shows the coverage of the conducted activities in the studies.

Most studies focus on the identification of stakeholders, ethical values, and the benefits and harms from the systems. According to the Tables 7 and 8, among all undertaken activities in three investigations of VSD (*i.e.,* conceptual, empirical, and technical), $C1$, $C2$, and $C3$ have more frequency than other activities in the primary studies. It means they focus more on identifying stakeholders (both explicitly and implicitly), identifying benefits and harms imposed by the systems on stakeholders, and mapping these benefits and harms to their corresponding values.

A few studies focus on prioritising values and translating them into software requirements. Based on the Tables 7 and 8, a few studies refer to the activities of $E2$ and $T1$ (1/57 and 2/57 studies, respectively). It means they put less effort into prioritising values and translating them into more concrete[8] constructs (*e.g.,* software requirements).

None of the primary studies covers all the depicted activities in regard to embedding values in software design and development. Based on Tables 7 and 8, none of the primary studies focuses on all the activities to embed the values in software design and development.

4.4.1. How can ethical values be represented as software requirements?

As values are intrinsically ambiguous and supporting them in the design process of software systems is not trivial, there is a need to translate values into more tangible con-

[6] In this method, when a small percentage of stakeholders strongly object to a design option, it should be removed from the design space (known as the value dams). Also, when a design option that a good percentage of stakeholders find appealing, should be remained in the design space (known as the value flows).

[7] In this method, one stakeholder group should prioritize the values based on their goals in the system without knowing the others' values. Afterward, they should align the other stakeholder groups' values to their values through discussions and find the value conflicts.

[8] It refers to making the values simple and clear enough (instead of abstract) to allow design decision makers to look at them and figure out what they are.

structs (like software requirements) and make them concrete enough to operationalize them. For example, in [33], the users of mental health (mHealth) apps must be confident and have trust in mental health care. However, the ambiguous nature of trust in mHealth forces developers to translate this value into requirements that foster and make it concrete, such as security, privacy, and good access to information and communication technologies infrastructure.

Translation of values into software requirements. Translation of values into software requirements is a relatively neglected aspect in designing systems which has been conducted in 2 out of 57 studies (*i.e.,* [44, 12]), using the method of the *value hierarchy*. By translating values into more concrete and tangible constructs, this method ensures that the design sufficiently reflects the ethical values.

A value hierarchy is a structure that is proposed by value sensitive design methods and consists of three layers: (i) values, (ii) norms, and (iii) software requirements. Consider an example of using this structure in the context of blood sample transportation drones (cf. [12]). The *values layer* (*i.e.,* upper layer) consists of values that are independent from the context in which the system is situated, *e.g., safety*. The *norms layer* (*i.e.,* middle layer) consists of norms that are context-dependent in contrast to values and could be any prescription or restriction for action. In this example, this layer includes the norm relevant to safety, *e.g., using drones is safer than driving with respect to fatalities per round-trip*. The *software requirements layer* (*i.e.,* lower layer) consists of software requirements that are obtained by further investigations of the norms. In this example, this layer includes the requirements: *minimizing weight to less than 1.5 kg*, *having components with less than 300 grams*, and *minimizing impact speed to less than 60 km/hr*.

Table 6: Value relations extracted from the primary studies.

Value relations (VR)	Studies	Value categories	Examples
Value conflict (VC) and value tension (VT)	[34, 6, 42, 4, 33, 19, 37, 53, 43, 30, 11]	Security	• **VCs with privacy:** privacy vs. involvement, privacy vs. responsibility, privacy vs. transparency, privacy vs. welfare, privacy vs. creativity • **VTs with privacy:** privacy vs. reputation, privacy vs. trust, privacy vs. safety, privacy vs. autonomy, privacy vs. awareness • **VCs with safety:** safety vs. efficiency, safety vs. welfare • **VTs with safety:** safety vs. autonomy, safety vs. privacy, safety vs. involvement • **VCs with security:** security vs. creativity, security vs. welfare • **VCs with certainty:** certainty vs. competence • **VCs with control:** control vs. involvement • **VTs with control:** control vs. autonomy • **VCs with availability:** availability vs. equity • **VCs with anonymity:** anonymity vs. responsibility, anonymity vs. transparency, anonymity vs. identity
	[1, 6, 42, 33, 19, 37, 30, 40]	Benevolence	• **VCs of accuracy:** accuracy vs. dignity • **VCs of involvement:** involvement vs. cultural values, involvement vs. privacy, involvement vs. control • **VTs of involvement:** involvement vs. safety • **VCs of transparency:** transparency vs. privacy, transparency vs. anonymity, transparency vs. opacity • **VTs of trust:** trust vs. privacy
	[4, 33, 19, 53, 30]	Self-direction	• **VCs with autonomy:** autonomy vs. community, autonomy vs. welfare • **VTs with autonomy:** autonomy vs. safety, autonomy vs. control, autonomy vs. privacy, autonomy vs. power • **VCs with creativity:** creativity vs. privacy, creativity vs. security • **VCs with dignity:** dignity vs. accuracy • **VCs with identity:** identity vs. anonymity
	[33, 53, 43]	Achievement	• **VCs with efficiency:** efficiency vs. safety • **VTs with efficiency:** efficiency vs. power
	[11, 30]	Universalism	• **VCs with welfare:** welfare vs. protecting the environment, welfare vs. safety, welfare vs. privacy, welfare vs. security, welfare vs. autonomy

Table 7
An overview of the conducted activities in the primary studies to embed the (ethical) values in design and development of software systems which is based on VSD methods (Figure. 1).

VSD investigations	Activities	Description	Studies
Conceptual investigations	$C1$	Identifying stakeholders (explicitly)	[34, 6, 2, 51, 36, 57, 31, 55, 9, 42, 17, 8, 49, 38, 33, 50, 35, 32, 26, 43, 27, 52, 18]
		Identifying stakeholders (implicitly)	[1, 56, 54, 20, 45, 41, 44, 4, 24, 15, 12, 47, 5, 16, 22, 7, 13, 40, 19, 37, 25, 53, 10, 21, 39, 30, 11, 28, 29, 46]
	$C2$	Identifying benefits and harms imposed by the system on stakeholder groups (explicitly)	[34, 20, 45, 12, 19, 39, 11, 27, 18, 46]
		Identifying benefits and harms imposed by the system on stakeholder groups (implicitly)	[1, 6, 2, 51, 56, 36, 57, 54, 23, 31, 55, 9, 41, 42, 44, 4, 17, 8, 49, 48, 38, 24, 15, 33, 47, 5, 16, 22, 7, 14, 13, 40, 37, 50, 35, 32, 25, 26, 53, 10, 21, 43, 30, 28, 29, 52, 3]
	$C3$	Mapping benefits and harms onto corresponding values	[1, 34, 6, 2, 51, 56, 36, 57, 54, 20, 23, 45, 31, 55, 9, 41, 42, 44, 4, 17, 8, 49, 48, 38, 24, 15, 33, 12, 47, 5, 16, 22, 7, 14, 13, 40, 19, 37, 50, 35, 32, 25, 26, 53, 10, 21, 43, 39, 30, 11, 28, 27, 29, 52, 3, 18, 46]
	$C4$	Categorizing the identified values	[51, 8, 7, 50, 21, 3]
	$C5$	Identifying potential value tensions and conflicts	[1, 34, 6, 36, 57, 45, 42, 44, 4, 33, 40, 19, 37, 53, 43, 39, 30, 11]
Empirical investigations	$E1$	Understanding how stakeholders experience a technology with regard to the values they consider important	[1, 34, 56, 36, 57, 54, 20, 31, 9, 41, 4, 17, 38, 15, 33, 47, 5, 7, 13, 19, 37, 35, 32, 25, 10, 30, 28, 27, 46]
	$E2$	Prioritising the values	[46]
	$E3$	Resolving value tensions and conflicts through making trade-offs among competing values	[34, 6, 36, 57, 37]
Technical investigations	$T1$	Translating the identified values into software requirements	[44, 12]
	$T2$	Designing systems to support the identified values or supporting/hindering values by existing technological properties	[34, 6, 56, 54, 45, 55, 4, 8, 49, 38, 24, 12, 5, 22, 7, 14, 13, 25, 21, 43, 30, 11, 28, 52, 18, 46]
	$T3$	Evaluating the design, adoption, and use of the designed systems	[34, 54, 22, 7, 28]

Table 8: Coverage of the primary studies with respect to the conducted activities to embed the (ethical) values in design and development of software systems.

Studies	$C1$	$C2$	$C3$	$C4$	$C5$	$E1$	$E2$	$E3$	$T1$	$T2$	$T3$
[1]	■	■	■	□	■	■	□	□	□	□	□
[34]	■	■	■	□	■	■	□	■	□	■	■
[6]	■	■	■	□	■	□	□	■	□	■	□
[2]	■	■	■	□	□	□	□	□	□	□	□
[51]	■	■	■	■	□	□	□	□	□	□	□
[56]	■	■	■	□	□	■	□	□	□	■	□
[36]	■	■	■	□	■	■	□	■	□	□	□
[57]	■	■	■	□	■	■	□	■	□	□	□
[54]	■	■	■	□	□	■	□	□	□	■	■
[20]	■	■	■	□	□	■	□	□	□	□	□
[23]	□	■	■	□	□	□	□	□	□	□	□
[45]	■	■	■	□	■	□	□	□	□	■	□
[31]	■	■	■	□	□	■	□	□	□	□	□
[55]	■	■	■	□	□	□	□	□	□	■	□
[9]	■	■	■	□	□	■	□	□	□	□	□
[41]	■	■	■	□	□	■	□	□	□	□	□
[42]	■	■	■	□	■	□	□	□	□	□	□
[44]	■	■	■	□	■	□	□	□	■	□	□
[4]	■	■	■	□	■	■	□	□	□	■	□
[17]	■	■	■	□	□	■	□	□	□	□	□
[8]	■	■	■	■	□	□	□	□	□	■	□
[49]	■	■	■	□	□	□	□	□	□	■	□
[48]	■	■	■	□	□	□	□	□	□	□	□
[38]	■	■	■	□	□	■	□	□	□	■	□
[24]	■	■	■	□	□	□	□	□	□	■	□
[15]	■	■	■	□	□	■	□	□	□	□	□
[33]	■	■	■	□	■	■	□	□	□	□	□
[12]	■	■	■	□	□	□	□	□	■	■	□
[47]	■	■	■	□	□	■	□	□	□	□	□
[5]	■	■	■	□	□	■	□	□	□	■	□
[16]	■	■	■	□	□	□	□	□	□	□	□
[22]	■	■	■	□	□	□	□	□	□	■	■
[7]	■	■	■	■	□	■	□	□	□	■	■
[14]	■	■	■	□	□	□	□	□	□	■	□
[13]	■	■	■	□	□	■	□	□	□	■	□
[40]	■	■	■	□	■	□	□	□	□	□	□
[19]	■	■	■	□	■	■	□	□	□	□	□
[37]	■	■	■	□	■	■	□	■	□	□	□
[50]	■	■	■	■	□	□	□	□	□	□	□
[35]	■	■	■	□	□	■	□	□	□	□	□
[32]	■	■	■	□	□	□	□	□	□	□	□
[25]	■	■	■	□	□	■	□	□	□	■	□
[26]	■	■	■	□	□	□	□	□	□	□	□
[53]	■	■	■	□	■	□	□	□	□	□	□
[10]	■	■	■	□	□	■	□	□	□	□	□
[21]	■	■	■	■	□	□	□	□	□	■	□
[43]	■	■	■	□	■	□	□	□	□	■	□
[39]	■	■	■	□	■	□	□	□	□	□	□
[30]	■	■	■	□	■	■	□	□	□	■	□
[11]	■	■	■	□	■	□	□	□	□	■	□
[28]	■	■	■	□	□	■	□	□	□	■	■
[27]	■	■	■	□	□	■	□	□	□	□	□
[29]	■	■	■	□	□	□	□	□	□	□	□
[52]	■	■	■	□	□	□	□	□	□	■	□
[3]	□	■	■	■	□	□	□	□	□	□	□
[18]	■	■	■	□	□	□	□	□	□	■	□
[46]	■	■	■	□	□	■	■	□	□	■	□

5. Discussion

This section discusses the implications of the results of this SLR, and we explain the future research directions per research question.

5.1. Identification of key stakeholders (RQ1)

In Section 4.2, we proposed a classification of the stakeholders in terms of receiving benefit or harm from the system. It includes both the stakeholders who directly interact with the system (*i.e.,* use or build the system) and the passively involved stakeholders who are indirectly affected by the system.

Indirect stakeholders are invisible in software design. Our primary studies identified two categories of stakeholders: (i) those who interact directly with the system (*i.e., visible stakeholders*[9]) and (ii) those who do not directly interact with the system but may be affected by it (*i.e., invisible stakeholders*[10]). While direct stakeholders are identifiable, indirect stakeholders are often ignored as they depend on the different contexts in which the system will be used, therefore they are invisible to developers. Not making explicit the invisible stakeholders during software design implies ignoring potential ethical issues and harms that systems might cause to them. For example, in the context of blood sample transportation drones in [11], ignoring society as a stakeholder during the design process could lead to building a system that causes ethical risks and may endanger the individuals' physical welfare, such as the possibility for the drones to strike an aircraft or hit someone on the ground. In this example, "the aircraft" and "someone on the ground" are the invisible stakeholders.

As invisible stakeholders might indirectly be harmed or negatively affected by the system, it is essential to consider the identification of such stakeholders as an integral part of software design. As such, there is a need to anticipate the system implications on individuals and society, and become aware of the potential ethical issues that the system might create. To anticipate the implications of the system at hand, designers can utilize methods such as the *social implication design method* [102]. By focusing on the hidden influence of design on individuals and the consequential social implications, this method may help designers address potential social problems when designing a product or a service.

Identification of invisible stakeholders is non-trivial. Whilst the identification of invisible stakeholders is crucial, it is however not trivial due to the following three reasons: (i) existing methods (such as VSD) do not provide any systematic way to identify invisible stakeholders or the far-reaching impacts of systems. (ii) conventional scoping activities in software engineering primarily focus on identifying stakeholders that directly interact with the system, *e.g.,* through use cases [69]. Indirect stakeholders, the ones that do not interact with the system, are deemed out of scope. Naturally, the resulting system does not account for its effects on these stakeholders who become invisible. (iii) Anticipating who might be indirectly affected by the system requires anticipating human-system interactions as well as the contexts in which those interactions occur, *beyond* the immediate scope of the system and in a creative manner. As such, there is a need to develop ways to identify important indirect stakeholders and their concerns.

Given the above-mentioned reasons, we distinguish the following research directions: (i) proposing methods to anticipate the overarching effects of the system on individuals and society, (ii) developing techniques to explicitly include into the scope of a system indirect stakeholders, and (iii) proposing systematic yet creative ways to anticipate the human-system interactions and the use contexts of the system.

Stakeholder classification can help identify direct and indirect stakeholders. In general, an explicit stakeholder classification helps in identifying the relevant stakeholders of a software system [58, 60]. In particular for ethics, however, our analysis of the primary studies revealed that such classification is lacking. We conjecture that our proposed stakeholder classification (see Figure 5 and Table 1) can be a step towards filling this gap. This classification divides the stakeholders into three categories based on how they interact with the system (*i.e.,* system users, system development organisation, and indirect stakeholders). By making explicit both direct and indirect stakeholders, it enables design decision makers to consider a wider range of stakeholders and be cognisant of indirect stakeholders that would otherwise become invisible.

5.2. Identification of ethical values (RQ2)

Different stakeholders have different views and values in relation to the system. Software engineers may focus primarily on the ethical values related to system implementation, Product managers may refer to the values relevant to organization profits, whereas end users may refer to the values in terms of satisfying their needs and reaching their goals. In the following, we discuss the implications related to identifying and supporting ethical values in software systems.

Stakeholders' concerns provide context to discover ethical values. Concerns of stakeholders in a specific context are related to perceived risks and issues that can potentially affect their values. Based on the primary studies, we found that the identification of stakeholders' concerns could provide context to help discover the values at stake (see Table 2). Discovering these links between concerns and values could help take ethical values into account as well as uncover conflicting concerns and values for design space exploration. For example, in [37], parents have concerns about privacy violations for the use of digital media by their young children. In such a case, concerns can lead to discovering the ethical values of *privacy* and *security*.

Moreover, as illustrated in Table 2, different stakeholder concerns might correspond to common values. In such cases, however, although the types of values may be the same for different stakeholders, they may differ with respect to the specific interests of each stakeholder category. For example,

[9] They usually can be identified by the stakeholder analysis methods when designing systems.

[10] They are difficult to be identified by existing stakeholder identification methods when designing systems.

the value of *privacy* identified from the concern of privacy violation could have different meanings for the two stakeholder categories of end users (*e.g.,* protect access to identity data of young children) and system development organisation (*e.g.,* comply with the "integrity and confidentiality" principle of the GDPR). To satisfy this value for each category, we may need to make different ethics-related decisions and select different design options.

Methods and techniques for eliciting ethical values. We found different methods and techniques used for value extraction from the system stakeholders (see Table 5), each suitable for certain generic purpose and situation. For example, *workshops* can help to collaboratively analyze values-related issues and ethical values, while *sketches* can provide a non-verbal understanding of the elicited values. Despite the usefulness of each method and technique in extracting values from stakeholders, there is very little understanding of the nature and appropriateness of these methods and techniques specifically with respect to different ethical situations. This is a gap requiring further research. For example, by introducing a classification that makes explicit invisible stakeholders (discussed in Section 5.1), we need to identify and customize methods and techniques to actively involve them in the design decision making process. Current methods are not yet tailored for it [59].

The value model helps define ethical values and recognize relations among them. Our classification of values is a first attempt to model values systematically, and it is a prerequisite for explaining and interpreting software ethical values. We suggest using the proposed value model (see Figure 3) for value classification. This model provides an opportunity to define software ethical values according to the value categories and their motivational goals (see Tables 12-15 in Appendix C). Also, the circular arrangement of the model could help recognize the relations among the values. More precisely, the location of a specific value in the model with respect to the four orthogonal dimensions could help discover the relation of that value with other values. Accordingly, the closer any two values are, the more similarity they have in terms of underlying motivations and vice versa. For example, in Figure 3, consider the value of *flexibility* in the dimension of *conservation*. According to the value model, it can be recognized that the values located in the orthogonal dimension of this value (*i.e., openness to change*) are likely to have contradictory relation with it, and the values located in its adjacent dimensions (*i.e., self-transcendence* and *self-enhancement*) can be congruent with it.

Some ethical values have not received much attention. While some values (*i.e., privacy*, *trust*, *autonomy*, and *safety*) have received considerable attention in the literature, some others (*e.g., social order* and *flexibility*) are understudied. Ignoring the unexplored values during software design implies ignoring unknown ethical risks and harms which surround the stakeholders and, as such, not providing any solutions addressing such risks. This may potentially affect adversely the individuals or even endanger their lives.

The value model resulting from our study could help in raising awareness of design decision makers about unexplored ethical values and in attracting research that will address them whenever risky to do otherwise.

Inadequacy in value conflict resolution. When designers design for a range of values (instead of a single one), it is possible that a design option that supports one value would compromise another value. Trade-offs amongst ethical values can be revealed by considering different design options.

Based on the primary studies, we found that only a few studies ([6, 34, 36, 37]) propose solutions to resolve conflicts among the values in the software design process: the methods value alignment, and value dams and value flows. These methods aim to resolve value conflicts; however, in spite of their efforts, they still need further research to best suit this purpose. For example, the value dams and value flows method has to determine "threshold values for value dams and value flows"[11] in a project [34], which is challenging and needs further research to determine the best threshold values.

Also, our results show that there is little focus on value conflict resolution. Further research is needed, for example, to understand how value conflicts can be dealt with and hence embedded in methods and techniques, or if and what qualitative and quantitative techniques can be used to make trade-offs among values and the associated possible design options.

5.3. Embedding requirements and values (RQ3)

A new class of requirements to address stakeholders' ethical values will require software researchers and developers to revisit and create software engineering methods and techniques for software design and development.

VSD, as a widely used method to support values in the software design process, has been supported to a varying degree in the primary studies. According to the results, VSD was the only method used to address ethical values during the software design process. However, the primary studies cover only a proper subset of the VSD activities (see Table 8), and in particular the conceptual ones addressing design.

This could be due to (i) the narrow focus of the study itself, or (ii) the intrinsic challenge in addressing ethical considerations in all VSD activities.

As an example of the studies with a narrow focus, [18] emphasizes design patterns to help designers incorporate value considerations in the design process without resolving value conflicts. In this example, the focus is limited, by design, to conceptual and design-related VSD activities, *i.e.,* $C1$-$C3$ and $T2$.

As an example of the studies with challenges in accounting for ethical considerations, [57] discusses the difficulties in identifying and applying human values in design, *e.g.,* finding suitable ways to talk about values among different stakeholders, or accounting for the voices of stakeholders and considering their values with respect to their power in the

[11] The value dams threshold would account for, *e.g.,* how strongly the concerns are held and the severity of the potential harm; similar aspects of benefits can be relevant for the value flows.

organization. In this example, supporting the above poses such a great challenge that some VSD activities are left out, *i.e.,* $C4$, $E2$ and $T1$-$T3$.

Operationalize ethical values in design. Embedding intrinsically ambiguous ethical values into software systems requires them to be operationalized in such a way that they could be easy to understand for design decision makers. In doing so, values have to be made as concrete as possible in the software design process.

We found only two primary studies ([44, 12]) discussing the translation of values into software requirements. This shows little focus on converting values into software requirements in order to support values in the software design. Further research is needed to support at least the disambiguation of ethical values, and their translation into software requirements that are ready for the design process.

6. Threats to validity

In this section, we discuss the main threats that may have affected the validity of our study. We organize them according to the classification by Wohlin *et al.* [109].

Construct validity refers to the correct data collection and the correct measurement of the theoretical concepts [71].

- One typical threat in SLRs can be related to selecting unsuitable search engines. To mitigate this threat, Google Scholar was used as a tool to collect papers (as explained in Section 3.2.1) since it is a meta-engine and is commonly used in SLRs. In addition, Martin-Martin *et al.* [88] provide empirical evidence that Google Scholar citation data is essentially a super-set of other online bibliographic databases.
- Ill-defined search queries can result in a large number of irrelevant studies. In order to mitigate this threat, we defined two search queries to ensure covering the potentially relevant studies (as discussed in Section 3.2.2). By using the first query, we covered a feasible number of studies, but there was a risk of overlooking the studies containing the term "value". So, to cover the studies including this term and the relevant phrases like "value-sensitive" and "value-centered" that focus on SE ethics, we defined another query (*i.e.,* search query 2).
- Choosing an improper classification for the values can introduce some bias to the analysis. We have mitigated this threat by selecting the Schwartz value structure, a widely adopted classification scheme in social sciences. This value classification was led to reducing researchers' bias since there was no need to develop our own classification.

Internal validity aims to ensure that the collected data enables researchers to draw valid conclusions [67].

- One typical threat to internal validity in SLRs is bias in extracting enough data from the studies, influencing the accuracy of the extracted data and the result. To mitigate this threat, we extracted also descriptive data concerning our research questions.
- There is a threat to internal validity due to improperly defining and conceptualizing the extracted values from the studies. This threat might result from the experience level of researchers in relation to philosophy and social science for value conceptualization. In our study, this threat was mitigated by involving two researchers in defining values, and in the presence of disagreements, by consulting a third researcher.
- Classifying values can introduce some bias to the analysis as values are intrinsically ambiguous and can be related to more than one value category. To mitigate this threat and facilitate the value classification, we took advantage of the presented meta-inventory of human values in [64]. Also, we involved three researchers in providing an objective classification.

7. Conclusion

Software systems are ubiquitously employed in society and increasingly integrated into all aspects of people's lives. The proper functioning of software systems in both technical and ethical dimensions is essential, as individuals and society at large can be affected by these systems in many facets, from healthcare to communication, from education to economic welfare. These systems can affect individuals by undermining their values and causing ethical issues, like causing harm to the environment, violating privacy, and discriminating amongst people.

In this study, we conducted an SLR addressing the identification of stakeholders, their values and value relationships. By analyzing 57 primary studies, we identified three categories of system stakeholders who are affected, from an ethical value perspective, by software systems (*i.e.,* system users, system development organisation, and indirect stakeholders). We also identified a list of ethical values commonly used in designing software systems. Using the Schwartz value structure, we categorized the extracted values from the studies. This value classification can be utilized in identifying values and recognizing their relations during software design and development.

Our analysis resulted in a number of future research directions. These include: (i) providing a systematic way to identify the influential stakeholders (and especially invisible stakeholders) in software design, (ii) proposing methods and techniques to elicit ethical values from the system stakeholders that suit different ethical purposes and situations, (iii) providing a comprehensive list of ethical values used in software design along with their definitions, (iv) proposing a model to recognize the relations among values during the design process, (v) proposing suitable methods to resolve value conflicts, and (vi) developing methods to operationalize ethical values during software design.

By giving insights into existing issues in SE ethics and by providing promising future research directions in this area, we hope to have provided food for thought for researchers and practitioners to make ethical considerations an integral part of software design and development.

List of Primary Studies

[1] Tamara Alsheikh, Jennifer A Rode, and Siân E Lindley. Whose value-sensitive design: a study of long-distance relationships in an arabic cultural context. In *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, pages 75–84, 2011.

[2] Aida Alvarenga and George Tanev. A cybersecurity risk assessment framework that integrates value-sensitive design. *Technology Innovation Management Review*, 7(4), 2017.

[3] Markus Ast and Martin Gaedke. Value-sensitive design in hyper-connected societies. *Informatik 2016*, 2016.

[4] Karla Badillo-Urquiola, Chhaya Chouhan, Stevie Chancellor, Munmun De Choudhary, and Pamela Wisniewski. Beyond parental control: designing adolescent online safety apps using value sensitive design. *Journal of adolescent research*, 35(1):147–175, 2020.

[5] Balbir S Barn and Ravinder Barn. Human and value sensitive aspects of mobile app design: a foucauldian perspective. In *International Conference on Advanced Information Systems Engineering*, pages 103–118. Springer, 2018.

[6] Stefanie Betz and Andreas Fritsch. A comparison of value sensitive design and sustainability design. *Informatik 2016*, 2016.

[7] Alan Borning, Batya Friedman, Janet Davis, and Peyina Lin. Informing public deliberation: Value sensitive design of indicators for a large-scale urban simulation. In *ECSCW 2005*, pages 449–468. Springer, 2005.

[8] Alan Borning, Batya Friedman, and P Kahn. Designing for human values in an urban simulation system: Value sensitive design and participatory design. In *Proceedings From the Eighth Biennial Participatory Design Conference*. Citeseer, 2004.

[9] Pam Briggs and Lisa Thomas. An inclusive, value sensitive design perspective on future identity technologies. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 22(5):1–28, 2015.

[10] Rachel Burrows, Johnson Peter, and Johnson Hilary. Value sensitive design approach to influence energy-use behaviour. pages 657–658, 2015.

[11] Dylan Cawthorne and Alessandra Cenci. Value sensitive design of a humanitarian cargo drone. In *2019 International conference on unmanned aircraft systems (ICUAS)*, pages 1117–1125. IEEE, 2019.

[12] Dylan Cawthorne and Aimee Robbins-van Wynsberghe. From health-drone to frugaldrone: Value-sensitive design of a blood sample transportation drone. In *2019 IEEE International Symposium on Technology and Society (ISTAS)*, pages 1–7. IEEE, 2019.

[13] EunJeong Cheon and Norman Makoto Su. Integrating roboticist values into a value sensitive design framework for humanoid robots. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 375–382. IEEE, 2016.

[14] Mary L Cummings. Integrating ethics in design through the value-sensitive design approach. *Science and Engineering Ethics*, 12(4):701–715, 2006.

[15] Majid Dadgar and KD Joshi. Diabetes self-management using mobile apps: An empirical investigation based on app reviews and through value sensitive design perspective. In *2015 International Conference on Mobile Business. Paper*, volume 3, 2015.

[16] Majid Dadgar and KD Joshi. Ict-enabled self-management of chronic diseases: Literature review and analysis using value-sensitive design. In *2015 48th Hawaii International Conference on System Sciences*, pages 3217–3226. IEEE, 2015.

[17] Tamara Denning, Daniel B Kramer, Batya Friedman, Matthew R Reynolds, Brian Gill, and Tadayoshi Kohno. Cps: Beyond usability: Applying value sensitive design based methods to investigate domain characteristics for security for implantable cardiac devices. In *Proceedings of the 30th Annual Computer Security Applications Conference*, pages 426–435, 2014.

[18] Christian Detweiler and Koen Hindriks. Value-sensitive design patterns for pervasive health care. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 908–913. IEEE, 2012.

[19] Christian Detweiler, Koen Hindriks, and Catholijn Jonker. Principles for value-sensitive agent-oriented software engineering. In *International Workshop on Agent-Oriented Software Engineering*, pages 1–16. Springer, 2010.

[20] Christian Detweiler, Koen Hindriks, and Catholijn Jonker. A value-sensitive approach to agent-oriented software engineering. In *Workshop 8 The Eleventh International Workshop on Agent Oriented Software Engineering AOSE 2010*, page 1, 2010.

[21] Batya Friedman and Alan Borning. Value sensitive design as a pattern: Examples from informed consent in web browsers and from urban simulation. In *Proceedings of the DIAC 2002 Directions and Implications of Advanced Computing Symposium, Palo Alto, CA*, pages 109–113. Citeseer, 2002.

[22] Batya Friedman, Daniel C Howe, and Edward Felten. Informed consent in the mozilla browser: Implementing value-sensitive design. In *Proceedings of the 35th annual hawaii international conference on system sciences*, pages 10–pp. IEEE, 2002.

[23] Batya Friedman and Peter H Kahn Jr. A value-sensitive design approach to augmented reality. *Design of Augmented Reality Environments, The MIT Press, Cambridge, MA, in press-a*, 2000.

[24] Batya Friedman, Ian Smith, Peter H Kahn, Sunny Consolvo, and Jaina Selawski. Development of a privacy addendum for open source licenses: Value sensitive design in industry. In *International Conference on Ubiquitous Computing*, pages 194–211. Springer, 2006.

[25] Arup Kumar Ghosh. Using a value sensitive design approach to promote adolescent online safety on mobile platforms. In *2016 International Conference on Collaboration Technologies and Systems (CTS)*, pages 593–596. IEEE, 2016.

[26] Christiane Grünloh. Using technological frames as an analytic tool in value sensitive design. *Ethics and Information Technology*, pages 1–5, 2018.

[27] Maaike Harbers and Mark A Neerincx. Value sensitive design of automated workload distribution support for traffic control teams. In *International Conference on Engineering Psychology and Cognitive Ergonomics*, pages 12–21. Springer, 2014.

[28] Maaike Harbers and Mark A Neerincx. Value sensitive design of a virtual assistant for workload harmonization in teams. *Cognition, Technology & Work*, 19(2):329–343, 2017.

[29] Maaike Harbers, Peter Van Waart, and Eva Visser. Value sensitive design of smart cities. In *Charting the Next Decade for Value Sensitive Design Workshop. Denmark: Aarhus*, 2010.

[30] Oliver Heger. Value sensitive design in design science research projects: The cases of affective technology and healthcare technology. In *EMoWI@ Wirtschaftsinformatik*, pages 17–26, 2019.

[31] Christopher A Le Dantec, Erika Shehan Poole, and Susan P Wyche. Values as lived experience: evolving value sensitive design in support of value discovery. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1141–1150, 2009.

[32] Christopher A Le Dantec and ES Poole. The value of pictures: Photo elicitation techniques for value sensitive design. *CHI, April*, 10, 2008.

[33] Ivo Maathuis, Maartje Niezen, David Buitenweg, Ilja L Bongers, and Chijs van Nieuwenhuizen. Exploring human values in the design of a web-based qol-instrument for people with mental health problems: A value sensitive design approach. *Science and engineering ethics*, pages 1–28, 2019.

[34] Jessica K Miller, Batya Friedman, Gavin Jancke, and Brian Gill. Value tensions in design: the value sensitive design, development, and appropriation of a corporation's groupware system. In *Proceedings of the 2007 international ACM conference on Supporting group work*, pages 281–290, 2007.

[35] Ahamed M Mithun, Z Abu Bakar, and Wael MS Yafooz. The realism of value sensitive design on user interface development. In *2018 IEEE Conference on Open Systems (ICOS)*, pages 86–91. IEEE, 2018.

[36] Marije Nouwen, Maarten Van Mechelen, and Bieke Zaman. A value sensitive design approach to parental software for young children. In *Proceedings of the 14th International Conference on Interaction Design and Children*, pages 363–366, 2015.

[37] Marije Nouwen and Bieke Zaman. Redefining the role of parents in young children's online interactions. A value-sensitive design case study. *Int. J. Child Comput. Interact.*, 18:22–26, 2018.

[38] Lambèr Royakkers and Marc Steen. Developing tools to counteract and prevent suicide bomber incidents: A case study in value sensitive design. *Science and engineering ethics*, 23(4):1041–1058, 2017.

[39] Agnieszka Rychwalska and Magdalena Roszczynska-Kurasinska. Value sensitive design for peer production systems: mediating social interactions. *IEEE Technology and Society Magazine*, 36(3):48–55, 2017.

[40] Donnie Johnson Sackey. One-size-fits-none: A heuristic for proactive value sensitive environmental design. *Technical Communication Quarterly*, 29(1):33–48, 2020.

[41] Nithya Sambasivan and Melody Moore Jackson. Applying value sensitive design in pervasive brain-computer interfaces. In *CHI: ACM Conference on Human Factors in Computing Systems. Extended Abstracts; Florence, Italy*, 2008.

[42] Luuk PA Simons and W Pieter Verhagen. Applying value-sensitive design and quality function deployment to healthcare ict: the case of dutch primary care unit dossiers. *Journal of Design Research*, 7(2):155–176, 2008.

[43] Sarah M Thornton, Francis E Lewis, Vivian Zhang, Mykel J Kochenderfer, and J Christian Gerdes. Value sensitive design for autonomous vehicle motion planning. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1157–1162. IEEE, 2018.

[44] Steven Umbrello. Beneficial artificial intelligence coordination by means of a value sensitive design approach. *Big Data and Cognitive Computing*, 3(1):5, 2019.

[45] Steven Umbrello and Angelo Frank De Bellis. A value-sensitive design approach to intelligent agents. *Artificial Intelligence Safety and Security (2018) CRC Press (. ed) Roman Yampolskiy*, 2018.

[46] Judith Van Andel, Frans Leijten, Hans Van Delden, and Ghislaine van Thiel. What makes a good home-based nocturnal seizure detector? a value sensitive design. *PloS one*, 10(4):e0121446, 2015.

[47] Geerten van de Kaa, Jafar Rezaei, Behnam Taebi, Ibo van de Poel, and Abhilash Kizhakenath. How to weigh values in value sensitive design: A best worst method approach for the case of smart metering. *Science and engineering ethics*, 26(1):475–494, 2020.

[48] Aimee Van Wynsberghe. Designing robots for care: Care centered value-sensitive design. *Science and engineering ethics*, 19(2):407–433, 2013.

[49] Pieter E Vermaas, Yao-Hua Tan, Jeroen van den Hoven, Brigitte Burgemeestre, and Joris Hulstijn. Designing for trust: A case of value-sensitive design. *Knowledge, Technology & Policy*, 23(3-4):491–505, 2010.

[50] Åke Walldius, Jan Gulliksen, and Yngve Sundblad. Revisiting the usersaward programme from a value sensitive design perspective. In *Proceedings of The Fifth Decennial Aarhus Conference on Critical Alternatives*, pages 1–4, 2015.

[51] Åke Walldius, Yngve Sundblad, and Alan Borning. A first analysis of the usersaward programme from a value sensitive design perspective. In *Proceedings of the 4th decennial conference on Critical computing: between sense and sensibility*, pages 199–202, 2005.

[52] Rebecca Walton and Brian DeRenzi. Value-sensitive design and health care in africa. *IEEE Transactions on Professional Communication*, 52(4):346–358, 2009.

[53] Kari Edison Watkins. Using value sensitive design to understand transportation choices and envision a future transportation system. *Ethics and Information Technology*, pages 1–4, 2018.

[54] Heng Xu, Robert E Crossler, and France Bélanger. A value sensitive design investigation of privacy enhancing tools in web browsers. *Decision support systems*, 54(1):424–433, 2012.

[55] Heng Xu, Nazneen Irani, Sencun Zhu, and Wei Xu. Alleviating parental concerns for children's online privacy: a value sensitive design investigation. 2008.

[56] Daisy Yoo, Alina Huldtgren, Jill Palzkill Woelfer, David G Hendry, and Batya Friedman. A value sensitive action-reflection model: evolving a co-design space with stakeholder and designer prompts. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 419–428, 2013.

[57] Bieke Zaman and Nassim JafariNaimi. A value sensitive design case study: Why values do (not) design. In *Workshop Charting the Next Decade for Value Sensitive Design, Critical Alternatives Conference*, 2015.

References

[58] Stakeholder Discovery and Classification Based on Systems Science Principles. In *Proceedings Second Asia-Pacific Conference on Quality Software*.

[59] Razieh Alidoosti, Patricia Lago, Eltjo Poort, Maryam Razavian, and Antony Tang. Incorporating Ethical Values into Software Architecture Design Practices. In *19th International Conference on Software Architecture Companion (ICSA-C)*, pages 124–127. IEEE, March 2022.

[60] Adriatik Bedjeti, Patricia Lago, Grace Lewis, Remco C De Boer, and Rich Hilliard. Modeling Context with an Architecture Viewpoint. In *IEEE International Conference on Software Architecture (ICSA)*. IEEE, April 2017.

[61] Brian Berenbach and Manfred Broy. Professional and ethical dilemmas in software engineering. *Computer*, 42(1):74–80, 2009.

[62] Valerie Braithwaite. Consensus, stability and meaning in abstract social values. *Australian Journal of Political Science*, 33(3):363–380, 1998.

[63] Rafael Capurro. Information ethos and information ethics-ideas to take responsible action in the field of information. *Nachrichten Fur Dokumentation*, 39(1):1–4, 1988.

[64] An-Shou Cheng and Kenneth R Fleischmann. Developing a meta-inventory of human values. *Proceedings of the American Society for Information Science and Technology*, 47(1):1–10, 2010.

[65] Hyunghoon Cho, Daphne Ippolito, and Yun William Yu. Contact tracing mobile apps for covid-19: Privacy considerations and related trade-offs. *arXiv preprint arXiv:2003.11511*, 2020.

[66] H Michael Chung and MB Khan. Classification of unethical behaviors in the management of information systems: The use of behaviorally anchored rating scale procedures. *International Journal of management*, 25(2):262, 2008.

[67] John W Creswell and J David Creswell. *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications, 2017.

[68] Janet Davis and Lisa P Nathan. Value sensitive design: Applications, adaptations, and critiques. *Handbook of ethics, values, and technological design: Sources, theory, values and application domains*, pages 11–40, 2015.

[69] Marcela Balbino Santos de Moraes, Eduardo Santana de Almeida, and Silvio Romero. A systematic review on software product lines scoping. In *Proceedings of 6th Experimental Software Engineering Latin American Workshop (ESELAW 2009)*, page 63. Citeseer, 2009.

[70] Alfred Driessen. Ethical aspects of research in ultrafast communication. In *Evaluating New Technologies*, pages 11–19. Springer, 2009.

[71] Steve Easterbrook, Janice Singer, Margaret-Anne Storey, and Daniela Damian. Selecting empirical methods for software engineering research. In *Guide to advanced empirical software engineering*, pages 285–311. Springer, 2008.

[72] Guy Cromwell Field. *Moral theory: an introduction to ethics*. Routledge, 2020.

[73] Mary Flanagan, Daniel C Howe, and Helen Nissenbaum. Embodying values in technology: Theory and practice. *Information technology and moral philosophy*, 322, 2008.

[74] Batya Friedman. Value-sensitive design: A research agenda for information technology. *Contract No: SBR-9729633). National Science Foundation, Arlington, VA*, 1999.

[75] Batya Friedman and David Hendry. The envisioning cards: a toolkit for catalyzing humanistic and technical imaginations. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1145–1148, 2012.

[76] Batya Friedman, David G Hendry, and Alan Borning. A survey of value sensitive design methods. *Foundations and Trends in Human-Computer Interaction*, 11(2):63–125, 2017.

[77] Batya Friedman, Peter H Kahn, Alan Borning, and Alina Huldtgren. Value sensitive design and information systems. In *Early engage-*

ment and new technologies: Opening up the laboratory, pages 55–95. Springer, 2013.

[78] Don Gotterbarn, Keith Miller, and Simon Rogerson. Software engineering code of ethics. *Communications of the ACM*, 40(11):110–118, 1997.

[79] Donald Gotterbarn. Software engineering ethics. *Encyclopedia of Software Engineering*, 2002.

[80] Rich Hilliard. Aspects, concerns, subjects, views. In *First Workshop on Multi-Dimensional Separation of Concerns in Object-oriented Systems (at OOPSLA'99)*, page 59. Citeseer, 1999.

[81] Pat Duffy Hutcheon. Value theory: Towards conceptual clarification. *The British Journal of Sociology*, 23(2):172–187, 1972.

[82] Staffs Keele et al. Guidelines for performing systematic literature reviews in software engineering. Technical report, Technical report, Ver. 2.3 EBSE Technical Report. EBSE, 2007.

[83] Staffs Keele et al. Guidelines for performing systematic literature reviews in software engineering. Technical report, Technical report, Ver. 2.3 EBSE Technical Report. EBSE, 2007.

[84] Barbara Kitchenham. Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26, 2004.

[85] Clyde Kluckhohn. *2. Values and value-orientations in the theory of action: An exploration in definition and classification*. Harvard University Press, 2013.

[86] Noëmi Manders-Huits. What values in design? the challenge of incorporating moral values into design. *Science and engineering ethics*, 17(2):271–287, 2011.

[87] Walter Maner. Unique ethical problems in information technology. *Science and Engineering Ethics*, 2(2):137–154, 1996.

[88] Alberto Martín-Martín, Enrique Orduna-Malea, Mike Thelwall, and Emilio Delgado López-Cózar. Google Scholar, Web of Science, and Scopus: A systematic comparison of citations in 252 subject categories. *Journal of informetrics*, 12(4):1160–1177, November 2018.

[89] Matthew B Miles and A Michael Huberman. *Qualitative data analysis: An expanded sourcebook*. sage, 1994.

[90] Richard W Paul and Linda Elder. *The Thinker's Guide to Understanding the Foundations of Ethical Reasoning: Based on" Critical Thinking Concepts Et Principles"*. Foundation for Critical Thinking Press, 2013.

[91] Harsha Perera, Waqar Hussain, Jon Whittle, Arif Nurwidyantoro, Davoud Mougouei, Rifat Ara Shams, and Gillian Oliver. A study on the prevalence of human values in software engineering publications, 2015-2018. In *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*, pages 409–420. IEEE, 2020.

[92] Awais Rashid, Karenza Moore, Corinne May-Chahal, and Ruzanna Chitchyan. Managing emergent ethical concerns for software engineering in society. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, volume 2, pages 523–526. IEEE, 2015.

[93] Nicholas Rescher. *Introduction to value theory*. Prentice Hall, 1969.

[94] González-Rodríguez Ma Rosario, Díaz-Fernández Ma Carmen, and Simonetti Biagio. Values and corporate social initiative: An approach through schwartz theory. *International Journal of Business and Society*, 15(1):19, 2014.

[95] Shalom H Schwartz. Universals in the content and structure of values: Theoretical advances and empirical tests in 20 countries. In *Advances in experimental social psychology*, volume 25, pages 1–65. Elsevier, 1992.

[96] Shalom H Schwartz. Are there universal aspects in the structure and contents of human values? *Journal of social issues*, 50(4):19–45, 1994.

[97] Shalom H Schwartz. Basic human values: Theory, measurement, and applications. *Revue française de sociologie*, 47(4):929, 2007.

[98] Shalom H Schwartz. An overview of the schwartz theory of basic values. *Online readings in Psychology and Culture*, 2(1):2307–0919, 2012.

[99] Jacques P Thiroux and Keith W Krasemann. *Ethics: Theory and practice*. Prentice Hall Upper Saddle River, NJ, 2001.

[100] Alan J Thomson and Daniel L Schmoldt. Ethics in computer software design and development. *Computers and Electronics in Agriculture*, 30(1-3):85–102, 2001.

[101] Alan J Thomson and Daniel L Schmoldt. Ethics in computer software design and development. *Computers and Electronics in Agriculture*, 30(1-3):85–102, 2001.

[102] Nynke Tromp and Paul Hekkert. Social implication design (sid)–a design method to exploit the unique value of the artefact to counteract social problems. 2014.

[103] Matteo Turilli. Ethics and the practice of software design. In *Proceedings of the 2008 conference on Current Issues in Computing and Philosophy*, pages 171–183. IOS Press, 2008.

[104] Shannon Vallor and Arvind Narayanan. An introduction to software engineering ethics. http://tiny.cc/vgyvtz, 2015. Markkula Center for Applied Ethics.

[105] Robert M Veatch. *Case studies in medical ethics*. Harvard University Press, 1977.

[106] Pieter E Vermaas, Paul Hekkert, Noëmi Manders-Huits, and Nynke Tromp. Design methods in design for values. *Handbook of Ethics, Values and Technological Design*, pages 179–202, 2015.

[107] Alan Winfield, John McDermid, Vincent C Müller, Zoë Porter, and Tony Pipe. Ethical issues for robotics and autonomous systems. 2019.

[108] Till Winkler and Sarah Spiekermann. Twenty years of value sensitive design: a review of methodological practices in vsd projects. *Ethics and Information Technology*, pages 1–5, 2018.

[109] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. *Experimentation in software engineering*. Springer Science & Business Media, 2012.

[110] R Yücel, H Elibol, and O Dağdelen. Globalization and international marketing ethics problems. 2009.

Appendix A. List of codes related to the research questions of the research and the values.

Table 9: List of codes related to the research questions of the research.

Category	Sub-category	Code	Name	Sub-subcategory	Code	Name	Description
RQ1	**RQ1.a:** Identifying influential stakeholders.	ST	Stakeholder types	Stakeholder types in terms of relation with the system	ST_ D_ BH	Stakeholders who use the system with direct benefit/harm	Stakeholders are people who could either affect, or be affected by the software systems, for example: end users of the system, designers, society, *etc.* The type of people in terms of their relationships with the system which can be divided into two categories: (i) people who have direct relation with the system; use or interact with the system and (ii) people who do not use or interact with the system, but they are affected by the system or its output through others.
					ST_ IN_ BH	Stakeholders who interact with the system with indirect benefit/harm	
					ST_ N_ BH	Stakeholders who not engage in the system's development or use and may have or not have benefit/harm from the system	
	RQ1.b: Identify concerns of stakeholders.	SC	Stakeholder concerns	-	-	-	The issues that stakeholders care about and the issues that can be addressed through analyzing and embedding the ethical values throughout the software design and development process, for example: privacy violation, *etc.*
RQ2	**RQ2.a:** Identification of ethical values.	V	Values	-	-	-	Values are expressions of what humans, organisations, *etc.*, find important and some conception of what they consider good, bad, right, and wrong. Values can be context-dependent or can be instantiated in specific situations, but often are formulated abstract, for example: privacy, justice, *etc.*
	RQ2.b: Eliciting values from stakeholders.	VE	Value elicitation	-	-	-	There are methods and techniques to extract values from the stakeholders, for example: interview, survey, *etc.*
	RQ2.c: Identification of relations among ethical values.	VR	Value relations	-	-	-	The position and relation of values towards each other, for example: conflict/tension, congruity, *etc.*

Table 9 – List of codes related to the research questions of the research.

Category	Sub-category	Code	Name	Sub-subcategory	Code	Name	Description
RQ3	**RQ3.a:** Recognizing how the requirements and values of different stakeholders in the software design process can be embedded.	EVS	Embedding values of stakeholders in the software design process	-	-	-	Recognizing the approaches, methods, techniques, *etc.*, which support and embed the stakeholders' values in the software design process.
	RQ3.b: Translating ethical values to design requirements (functional and quality requirements).	MVR	Mapping values to requirements	-	-	-	The approaches to translate values into design requirements, for example: value hierarchy.

Table 10: Codification of the extracted values.

Schwartz's value categories	Schwartz's values	Extracted values from the studies that were mapped to the Schwartz's values	Extracted values from the studies that were not mapped to the Schwartz's values
Self-Direction (SD)	Choosing own goals	-	-
	Privacy	-	-
	Self-respect	-	-
	Creativity	-	-
	Freedom	Freedom	-
	Curious	Curious	-
	Independent	Autonomy (autonomy, independent)	-
	-	-	Dignity (human dignity, humanness, work ethic, morality)
	-	-	Identity (identity, being normal, identity control, knowledgeable)
	-	-	Solitude
	-	-	Attentiveness
Hedonism (HE)	Pleasure	-	-
	Enjoying life	-	-
	Self-indulgent	-	-
	-	-	Calmness (calmness, comfort)
	-	-	Hope
Achievement (AC)	Ambitious	-	-
	Influential	-	-
	Intelligent	-	-
	Capable	Capable	-
	Successful	Successful	-
	-	-	Competence
	-	-	Efficiency (efficacy, utility, effectiveness, usefulness, productivity, efficiency)
Power (PO)	Authority	-	-
	Social recognition	Social recognition (awareness, social recognition)	-
	Social power	Social power (social influence, social power, legitimation)	-
	Wealth	Wealth	-
	Preserving individuals' public image	Preserving individuals' public image	-
	-	-	Ownership and property
	-	-	Knowledge
Security (SE)	National security	-	-
	Family security	-	-
	Reciprocation of favors	-	-
	Sense of belonging	-	-
	Clean	-	-
	Healthy	Healthy	-
	Social order	Social order	-
	-	-	Control (control for safety, surveillance, monitoring, control)
	-	-	Safety
	-	-	Privacy
	-	-	Support and protection
	-	-	Anonymity (confidentiality, anonymity)
	-	-	Certainty (assurance, certainty)
	-	-	Availability (accessibility, availability)
	-	-	Informed consent
Tradition (TR)	Respect for tradition	-	-
	Accepting my portion in life	-	-
	Moderate	-	-
	Humble	-	-
	Devout	-	-
	Detachment	-	-
	-	-	Cultural and spiritual values
	-	-	Lifestyle values
Conformity (CO)	Politeness	-	-
	Obedient	-	-
	Honoring of elders	-	-
	Self-discipline	-	-
	-	-	Flexibility
Benevolence (BE)	Forgiving	-	-
	A spiritual life	-	-

Table 10 – Codification of the extracted values.

Schwartz's value categories	Schwartz's values	Extracted values from the studies that mapped to the Schwartz's values	Extracted values from the studies that did not map to the Schwartz's values
	Loyal	-	-
	True friendship	-	-
	Mature love	-	-
	Honest	-	-
	Meaning in life	-	-
	Responsibility	Responsibility (accountability, responsibility)	-
	Helpful	Helpful (helping others, helpfulness)	-
	-	-	Togetherness (togetherness, intimacy, solidarity)
	-	-	Continuity
	-	-	Involvement (involvement, family-centricity, participation, communication, collaboration, connection, community)
	-	-	Usability (universal usability, design for all, ease of use)
	-	-	Accuracy
	-	-	Transparency
	-	-	Trust (reliability, trust, comprehensibility, articulateness, transparency, openness)
	-	-	Integrity
Universalism (*U N*)	A world of beauty	-	-
	Broad-minded	-	-
	Unity with nature	-	-
	Inner harmony	-	-
	Wisdom	-	-
	World at peace	-	-
	Equality	Equality	-
	Protecting the environment	Protecting the environment	-
	Social justice	Social justice (procedural justice, environmental justice, distributive justice)	-
	-	-	Welfare (human physical welfare, human psychological welfare, human material welfare, economic-cost, cost-effectiveness, emotional well-being, psychological well-being, physical health)
	-	-	Fairness
	-	-	Freedom from bias
	-	-	Altruism

Appendix B: List of primary studies over publication venues and venue types.

Table 11: Number of the selected studies over publication venues and venue types.

Publication venue (PV)	Venue type (VT)	Studies	No.
Science and Engineering Ethics	J	[48, 38, 33, 47, 14]	5
Ethics and Information Technology	J	[26, 53]	2
Informatik	J	[6, 3]	2
Technology Innovation Management Review (TIM Review)	J	[2]	1
International Journal of Decision Support Systems (IJDSS)	J	[54]	1
Designing of Augmented Reality Environments (DARE)	J	[23]	1
Artificial Intelligence Safety and Security	J	[45]	1
ACM Transactions on Computer-Human Interaction (TOCHI)	J	[9]	1
Journal of Design Research (JDR)	J	[42]	1
Big Data and Cognitive Computing (BDCC)	J	[44]	1
Journal of Adolescent Research (JAR)	J	[4]	1
Knowledge, Technology and Policy (KTP)	J	[49]	1
Technical Communication Quarterly (TCQ)	J	[40]	1
International Journal of Child-Computer Interaction (IJCCI)	J	[37]	1
IEEE Technology and Society Magazine	J	[39]	1
IEEE Transactions on Professional Communication (TPC)	J	[52]	1
Cognition, Technology and Work (CTW)	J	[28]	1
PLoS One	J	[46]	1
Conference on Human Factors in Computing Systems (CHI)	C	[56, 31, 41, 32]	4
Decennial Aarhus Conference on Critical Alternatives	C	[50, 57, 29]	3
Hawaii International Conference on System Sciences (HICSS)	C	[16, 22]	2
International Conference on Information Systems (ICIS)	C	[55]	1
ACM Conference On Computer Supported Cooperative Work (CSCW)	C	[1]	1
ACM International Conference on Supporting Group Work (GROUP)	C	[34]	1
Decennial Conference on Critical Computing: Between Sense and Sensibility	C	[51]	1
International Conference on Interaction Design and Children (IDC)	C	[36]	1
Annual Computer Security Applications Conference (ACSAC)	C	[17]	1
Participatory Design Conference (PDC)	C	[8]	1
International Conference on Ubiquitous Computing (UbiComp)	C	[24]	1
International Conference on Mobile Business (ICMB)	C	[15]	1
International Conference on Advanced Information Systems Engineering (CAiSE)	C	[5]	1
European Conference on Computer-Supported Cooperative Work (ECSCW)	C	[7]	1
ACM/IEEE International Conference on Human-Robot Interaction (HRI)	C	[13]	1
IEEE Conference on Open Systems (ICOS)	C	[35]	1
International Conference on Collaboration Technologies and Systems (CTS)	C	[25]	1
International Conference on Unmanned Aircraft Systems (ICUAS)	C	[11]	1
International Conference on Engineering Psychology and Cognitive Ergonomics (ICEPCE)	C	[27]	1
Directions and Implications of Advanced Computing Symposium (DIAC)	C	[21]	1
IEEE Intelligent Vehicles Symposium (IV)	C	[43]	1
International Symposium on Technology and Society (ISTAS)	C	[12]	1
International Workshop on Agent-Oriented Software Engineering (AOSE)	W	[20, 19]	2
Workshop on Fostering Smart Energy Applications (FSEA)	W	[10]	1
Pervasive Computing and Communications Workshops (PerCom)	W	[18]	1
Ethik und Moral in der Wirtschaftsinformatik (EMoWI)	W	[30]	1

Appendix C: List of ethical values according to the primary studies.

Table 12: Extracted values from the primary studies based on Schwartz's conservation dimension.

Schwartz's value categories	Extracted values from the studies	Explanation	Example
Security (SE)	Social order	It refers to the stable state of society, which is necessary for individuals to communicate, *e.g.,* communication through technology.	By hard-coding of some procedures in the platform or through running bots, *e.g.,* using bots that pre-screen edits to detect vandalism, or reverting changes, and sending a short message to the editor), Wikipedia cares for the value of social order [39].
	Healthy	It refers to individuals' well-being, both physical and mental, which should be protected by the systems and not cause harm like injury, sickness, or death.	By reducing waiting time for blood sample analysis and improving the physical welfare of patients, cargo drones (with the aim of blood sample transportation) care for the value of health [12].
	Privacy	It refers to individuals' claim or right to determine what information about themselves can be communicated to others.	By keeping the users' information safe, e.g., when they logged into the system, what they searched, and how they query the system, the groupware system cares for the value of privacy [34].
	Availability	It refers to the access of authorized users to the information in systems when needed (*e.g.,* information relevant to the tasks), regardless of their location.	By giving users access to the information relevant to their tasks, regardless of their location or organization of employment, the health information system (HIS) cares for the value of availability [52].
	Safety	It refers to the system avoiding unintended hazards (which result in potential harm to individuals either physically or mentally), detecting and mitigating physical risks, and protecting individuals, property, and the environment against the threats associated with the systems.	By protecting homeless young people and keeping them safe, Mobile phone technology cares for the value of safety [56].
	Certainty	It refers to assurance and lack of doubt in users to do their activities in the context of the systems.	By assuring parents about effectively carrying out the treatment steps at home, the sensor-based physiotherapeutic assistance system (for home therapy) cares for the value of certainty [30].
	Control	It refers to the ability of individuals (like designers) to keep the information and its flow safe through technologies.	By embedding some functionalities in the online entertainment platforms that enable parents to define time, content and activity restrictions, and monitor children online, these platforms care for the value of control (in this case, control for safety) [37].
	Informed consent	It refers to permission and agreement of individuals before conducting actions towards them in the context of the systems to protect the safety of data and individuals. It encompasses criteria of disclosure and comprehension (for *informed*) and voluntariness, competence, and agreement (for *consent*) of people.	By allowing users to become aware of when cookies occur or for what purposes they would be used, the Mozilla Browser cares for the value of informed consent [21].
	Anonymity	It refers to protecting stakeholders' identity in a system and assurance of privacy.	By concealing the identity of the reviewers to make judgments based on quality (not academic positions), conference management systems care for the value of anonymity [19].
	Support and protection	It refers to the preservation of individuals from threats or harms caused by the systems.	By allowing women to use missed calls to signal the men that they wish to communicate with them (but without financial burden), the media technology cares for the value of support and protection in the context of long-distance romantic relationships in Arabic culture [1].
Conformity (CO)	Flexibility	It refers to adaptation of systems to changes, *e.g.,* adaptation to changes in customer needs and new technological developments.	By modifying generation or consumption patterns in reaction to an external signal like a price change to meet the consumers' needs, the smart grid system cares for the value of flexibility [47].
Tradition (TR)	Cultural and spiritual values	It refers to behaviors of individuals in the social and public environments with respect to other individuals' expectations.	By embedding different components in the robots (like mind and emotions) according to cultural or religious differences existed among countries, the humanoid robots care for cultural and spiritual values [13].
	Lifestyle values	It refers to the habits, attitudes, tastes, economic level, *etc.*, that constitute the mode of living of an individual or group.	By considering the users' energy-saving strategies and sustainable behaviour changes in the design of structure and content of the website, the websites care for the value of lifestyle [10].

Table 13: Extracted values from the primary studies based on Schwartz's openness to change dimension.

Schwartz's value categories	**Extracted values from the studies**	**Explanation**	**Example**
Self-direction (*SD*)	Autonomy	It refers to individuals' ability to decide, plan, and act in ways they believe will help them achieve their goals, with or without the help of others (individuals or systems).	By allowing others to check on elders remotely, elders can live independently longer without their families' help, the CareNet Display (as a pervasive health care technology) cares for the value of autonomy [18].
	Freedom	It refers to the ability of individuals to have control over their activities or be free in their choice in the context of systems.	By embedding the implantable chip of Radio Frequency Identification (RFID) technology in the body of humans for unobtrusive sending and receiving of data like medical purposes or using in passports, this technology could undermine the value of freedom [32].
	Curious	It refers to individuals' interest to explore everything to become aware of different aspects about the systems.	By monitoring and measuring the cognitive load and the emotional state of train traffic controllers, the CLES monitor can satisfy the curiosity of team members in each other's functioning that can affect the value of curiosity [27].
	Solitude	It refers to the right of individuals to be left alone.	By providing a specific warning for vulnerable populations, the privacy addendum for an open source software can affect the value of solitude [24].
	Attentiveness	It refers to individuals for paying close attention and recognizing important things.	By perceiving the minute cues of the care-receivers (*i.e.,* patients) and being attentive to patients' frailty when lifting, the robots in the hospitals care for the value of attentiveness [48].
	Dignity	It refers to the rights of individuals to be respected and treated ethically in interaction with systems.	By enabling caregivers to interact with the elder in a more relaxed way and to treat them with more respect, the CareNet Display (as a pervasive health care technology) cares for the value of dignity [18].
	Identity	It refers to individuals' understanding of who they are over time (both continuity and discontinuity over time).	By developing the humanitarian cargo drones to transport blood samples, the vulnerable local communities (*e.g.,* with low-income) are able to develop their identity. So, drones can affect the value of identity [11].
Hedonism (*HE*)	Calmness	It refers to a peaceful psychological state in individuals.	By replacing loud and disruptive medical helicopters, the humanitarian cargo drones care for the value of calmness [11].
	Hope	It refers to a future-oriented expectation of attaining personal goals which are dependent on personal activities, characteristics, and external factors.	By considering the reminder feature that alerts patients to take the insulin, the diabetes apps care for the value of hope [15].

Table 14: Extracted values from the primary studies based on Schwartz's self-enhancement dimension.

Schwartz's value categories	Extracted values from the studies	Explanation	Example
Achievement (*AC*)	Capable	It refers to being able to achieve and under human control.	By being capable of providing a high standard of care (*e.g.,* by possessing strength and intelligence) to cater to the needs of patients, care robots consider the value of capability [45].
	Successful	It refers to accomplishing a desired aim for receiving benefits from the systems.	Design team by implementing functionalities (like a timer for parental involvement) in child platforms and parental controls to improve the quality of their platform cares for the value of success (in this case, commercial success) [37].
	Efficiency	It refers to individuals' ability to utilize the system in an optimal way.	By using mobile health (mHealth) technologies, Quality of Life (QoL) assessments might become more enjoyable, less time-consuming, and more efficient for people with severe mental health problems. So, mHealth technologies can affect the value of efficiency [33].
	Competence	It refers to individuals' abilities that help them properly carry out their tasks and be responsible for the results in the context of the systems.	By lifting the care-receivers (*i.e.,* patients) at the appropriate speed and angle without hurting or dropping them, the robots in the hospitals care for the value of competence [48].
Power (*PO*)	Preserving individuals' public image	It refers to beliefs and public attention about stakeholders based on their actions in relation to the systems.	By answering a question well or contributing to writing good code in the groupware system (through users), this system can affect the value of the reputation at a software engineering organization [34].
	Social power	It refers to attaining a dominant position or control over others in the context of the systems.	By emphasizing within-community achievements of editors within the Wikipedia community (*e.g.,* gaining high regard, or completing an immaculate history of interactions), this platform can affect the value of social power [39].
	Recognition	It refers to recognizing something or individuals based on previous knowledge in the context of the systems.	By monitoring train traffic controllers through the CLES monitor, the hardworking controllers can feel recognized when the monitor shows others how hard they work. So, the monitor can affect the value of recognition [27].
	Wealth	It refers to material possessions and financial benefits in the context of the systems.	By supporting the peer review process and publishing high-quality research through the conference management system, the publication's and the publisher's reputation potentially increase. So, the system can support the value of profit [19].
	Ownership and property	It refers to the right to a property including the right to possess it, use it, manage it, derive income from it, *etc.*	By giving patients online access to their health information in the context of Electronic Health Records (EHR) system, they may regard themselves as the owner. So, the system can affect the value of ownership and property [26].
	Knowledge	It refers to technical knowledge that individuals have in relation to the systems.	By publishing high-quality research and barring substandard level publications, the conference management system can support the value of knowledge [19].
Hedonism (*HE*)	Calmness	It refers to a peaceful psychological state in individuals.	By replacing loud and disruptive medical helicopters, the humanitarian cargo drones care for the value of calmness [11].
	Hope	It refers to a future-oriented expectation of attaining personal goals which are dependent on personal activities, characteristics, and external factors.	By considering the reminder feature that alerts patients to take the insulin, the diabetes apps care for the value of hope [15].

Table 15: Extracted values from the primary studies based on Schwartz's self-transcendence dimension.

Schwartz's value categories	Extracted values from the studies	Explanation	Example
Benevolence (*BE*)	Responsibility	It refers to the properties that ensure actions of individuals or organizations may be traced uniquely to the individuals or organizations. Also, stakeholders should be accountable for their activities in the system. For example, they should be able to confirm that their values are reflected in the simulation, evaluate and judge its validity, and develop an appropriate level of confidence in its output.	By ensuring the care-receivers (*i.e.,* patients) about the right direction for care or maintaining an accurate assessment of their needs through the care-givers (*i.e.,* robots), the robots in the hospitals care for the value of responsibility [48].
	Helpful	It refers to the ability of individuals to provide help or direction to others through the system.	By sharing workload information to support operators to help each other when needed, the virtual assistant in the context of train traffic control cares for the value of helpful [28].
	Togetherness	It refers to being close to other individuals for different reasons such as (financial) support, conveying emotion, having communication through technology.	By creating individual work out of group work and displacing some of the built-in social outlets in domestic work (*e.g.,* changing the way of doing laundry or washing dishes because of using electric clothes dryers and dishwashers), domestic technologies could affect the value of togetherness [31].
	Transparency	It refers to the openness, clarity, and understandability of the system, its functions, and data, which help reduce misunderstandings of the users.	By occupying multiple roles within conference management systems, reviewers who are also authors may see the ranking of their own papers. So, these systems need to pay attention to the value of transparency [19].
	Involvement	It refers to the cooperation of different individuals and organizations as equal partners at every decision-making level to produce systems, from assessment and planning to implementation and evaluation.	By involving parents in the child's activities, knowing what their children do, and showing the children that they care, the parental application cares for the value of involvement [36].
	Trust	It refers to expectations between people who can experience goodwill, extend goodwill toward others, feel vulnerable, and experience betrayal. Trust in software systems refers to having faith in systems to demonstrate honesty and predictable behavior and keep loyalty and trueness.	By supporting interactions among persons (especially interactions that may leave some persons vulnerable to the actions of other persons), augmented reality considers the value of trust [23].
	Accuracy	It refers to data that should be free of errors, and the proper and precise function of a system *e.g.,* accurate operation, measurement, and feedback provided by the system.	By providing highly accurate measurement and feedback of the therapy, the sensor-based physiotherapeutic assistance system for home therapy cares for the value of accuracy [30].
	Integrity	It refers to moral and ethical principles for doing the right things by individuals. Integrity in software systems refers to complete and uncorrupted data.	By protecting reviewers' anonymity, conference management systems care for the value of scientific integrity [19].
	Usability	It refers to level of comfort in the use of systems to make all people (non-technology users or everyone from all demographic groups) successful users.	In [15], the diabetes apps care for the value of usability through supporting some acts. For example, by supporting adjusting lifestyle and mitigating conflict with cultural norms; providing instructions, resources, and commands in non-English languages; supporting the needs of gender and age differences; providing culturally appropriate and adaptable self-management methods.
	Continuity	It refers to the maintenance of continuous operations in a system.	By providing continuous action for parents to be able to treat their children at home, the sensor-based physiotherapeutic assistance system for home therapy affects the value of continuity [30].
Universalism (*UN*)	Justice	It refers to the capability of doing what is just and also being just in action.	By ensuring transparency, timely, complete, and unbiased information in decision-making, the smart grid system cares for the value of procedural justice [47].
	Equality	It refers to behaving equally with users and the contribution of all of them in the system without considering hierarchical considerations and judgments.	By allowing the contribution of all users in the context of peer production systems (like Wikipedia), these systems care for the value of equality [39].
	Protecting the environment	It refers to sustaining environments to meet the needs of the present without compromising the future.	By using electric power instead of fossil fuels (62% of electricity in Denmark comes from renewable sources), cargo drones care for the value of environmental sustainability [12].

Table 15 – Extracted values from the primary studies based on Schwartz's self-transcendence dimension.

Schwartz's value categories	Extracted values from the studies	Explanation	Example
	Fairness	It refers to the system that should not discriminate unfairly against specific individuals or groups of individuals in favor of others or should not bias in order to make reasonable judgments.	By not taking biased or discriminatory actions based on information about the stakeholders and treating all individuals involved equally, autonomous vehicle technologies care for the value of fairness [43].
	Freedom from bias	It refers to overcoming systematic unfairness perpetrated on individuals and groups.	By not discriminating unfairly against any group of stakeholders or privileging one policy over another, the large-scale simulation system (UrbanSim) cares for the value of freedom from bias [8].
	Altruism	It refers to helping others through the systems without personal gain or benefit (for example, providing individuals with some services).	By helping others without personal gain through pervasive brain-computer interfaces, *e.g.,* answering a question about local doctors, these interfaces could affect the value of altruism [41].
	Welfare	In the context of software systems, welfare refers to the protection of the well-being of all people, which consists of (i) physical welfare that deals with bodily well-being, such as physical and mental health, (ii) material welfare that refers to physical circumstances, economics and employment, and (iii) psychological welfare that refers to psychological and emotional states of users like comfort, peace, and mental health.	In [15], the diabetes apps care for the value of welfare through supporting some acts. For example, by improving mental and physical health conditions, facilitating self-monitoring of disease symptoms, supporting and educating patients in overcoming depression.