The Implications of 'Soft' Requirements

Alistair Sutcliffe[1], Pete Sawyer[2] & Nelly Bencomo [1]

[1]*Dept of Computer Science, Durham University, UK*

[2]*School of Engineering and Applied Science, Aston University, UK*

alistair.g.sutcliffe@durham.ac.uk, p.sawyer@aston.ac.uk, nelly.bencomo@durham.ac.uk

***Abstract*—A new focus for RE is investigated as 'soft' requirements which extends non-functional requirements / soft goals with a collection of people-oriented phenomena: values, motivations, emotions, and other socio-political issues that may influence the requirements specification. The convergence of RE with user experience (HCI) and technology acceptance from the information systems literature is reviewed from a temporal perspective: pre-use, through initial to longer-term use. A taxonomy of soft requirements is proposed that extends non-functional requirements and soft-goal concepts to direct attention towards user characteristics and beliefs that may have implications for functional as well as system support requirements, such as training, help, explanation and trust in software. A timeline model of soft and hard (functional) requirements is presented with a focus on customization, adaptation and other soft requirements to, improve product acceptance and persuade users to take appropriate action. The paper concludes with a research agenda for soft requirements to improve the probability of system acceptance and the effectiveness of applications that aim to influence people's decisions and behaviour in internet apps and other discretionary-use applications.**

***Keywords*—soft requirements, NFRs, elicitation, technology acceptance, social issues, values.**

I Introduction

'Soft' socio-political issues have a long history in requirements engineering [1], having been acknowledged as important influences in shaping requirements; furthermore, socio-political issues have frequently frustrated the best endeavours of requirements engineers as they can lead to emergent problems culminating in system failure [1-3]. In this paper we propose the concept of 'soft requirements' as an umbrella to cover socio-political and other people-oriented issues such as values and emotions which may influence system level requirements (i.e. not only software requirements by also organizational and operational support). Soft requirements (SR) are related to non-functional requirements (NFRs) [4-5] and soft goals [6] in RE but have a wider connotation including the 'socio' part of socio-technical systems where user resistance can lead to systems being resented or rejected [2-4]. In many public-facing systems use is discretionary, and success depends on requirements matching users' interests; for example, in healthcare, e-government, e-commerce, social media and games. The success of these systems depends not only on 'hard' functional requirements (FRs) but also 'soft' requirements such as training, help, and persuading people that the benefits of using an application outweigh the costs, hence ensuring adoption and continued use. Exploring the perspective of SR analysis to facilitate system/product adoption and user persuasion is the first motivation for this paper.

Software engineering manifestos have drawn attention to values and related phenomena [7,8], although there is little evidence about how values relate to requirements and how they actually influence users' goals and decisions to accept IT products. Values have also attracted considerable attention in AI as a consequence of bias in classifiers and ethical issues in robotics [9,10]. Values may emerge from ethnographic analysis [11,12]; however, interpreting their design implications relies on the intuition and experience of the analyst. Value-Sensitive Design (VSD) [13] proposed scenario-based probing of values in the requirements phase to sensitize designs to stakeholder values; while Value-Based Requirements Engineering (VBRE) [14] also followed a scenario-based analysis with a taxonomy of values and motivations as an elicitation guide, supplemented with hints on potential design implications. In spite of these initiatives, the state of the art has not progressed beyond case study exemplars investigating value implications in a small number of domain/application contexts [15-17].

In information systems, SR related concepts have been the focus of technology adoption research over many years [18,19] which aimed to analyse and eventually predict the critical success factors that influence people to use rather than reject a system. The Technology Acceptance Model (TAM) [18] has been progressively elaborated to encompass social and contextual influences on users' decisions [19]. However, TAM studies are survey-based analyses of existing technologies which have little predictive power to guide product designers. There is a gap between RE, which focuses on detailed functional and non-functional characteristics of a potential product, and TAM-style analysis of general qualities that may contribute to a product's success. Although product-line adaptation and COTS procurement RE [20,21] have provided some criteria for product adoption and requirements, no model has emerged that integrates the TAM perspective with RE practice to predict critical success factors for technology adoption. Investigating the convergence of technology acceptance and modelling of SR and values is a second motivation for this paper.

While soft requirements are related to non functional requirements and soft goals in RE, SRs include several other attributes of people (i.e. values, beliefs, motivations, personality, emotions) and socio-political issues. Soft requirements are imprecise knowledge since they relate to attitudes, beliefs and intentions of people; hence they might be considered as 'partially known' within the 'unknown unknowns' framework [22]. A key question is how RE can investigate these unknowns and progress towards more precise specification by a combination of methods and taxonomies that encourage elicitation and understanding influences on users' preferences and their reaction to prototypes and socio-technical systems that embody their values and concerns. Trying to predict the critical success factors of a future system implies emergent requirements since the future context of use can never be completely predicted *a priori*. Mapping a research pathway to help requirements engineers elicit and discover the implications of soft and emergent requirements forms a third motivation for this paper.

In following sections, first we review related work in NFR and socio-political issues in RE research; this is followed by technology acceptance modelling in information systems and user experience in HCI (human computer interaction) where users' attitudes to technology have also been investigated. The third section describes a framework for analyzing SRs. In the fourth section we introduce, requirements engineering for technology adoption, which integrates SRs and TAM models with a temporal perspective for acceptance with a cost-benefit analysis to understand the critical success factors for IT product adoption. The final section proposes a research road map for elicitation techniques, models and techniques for SRs to improve the success of discretionary-use systems.

II Socio-political issues Research

A. Socio-Political Requirements Engineering

Soft issues relating to people, social groups and economics have traditionally been related to NFRs in RE. Many taxonomies of NFRs have been produced [5]; however, since NFRs are goals relating to performance and quality issues [23], they do not describe many non-goal-related issues. Soft goals evolved as a more flexible construct to model qualities in i* [6] that are satisficed by hard goals or FRs. Studies of COTS procurement have suggested some socio-economic attributes of suppliers to guide requirements-product selection [20,21], and consider adaptation costs. Some user-oriented issues have been addressed in RE, for example the goals, skills and preferences framework for matching user capabilities to tasks [24]. Furthermore, relationships for responsibility, trust and authority can be modelled in i* [6] and extensions thereof [25], but as Milne and Maiden [26] pointed out, requirements are socially constructed in a political context; hence they advocated development of techniques for social power modelling. Soft requirements need to be explicitly analysed in RE rather than being represented in agent attributes in i* or as *ad hoc* components of other RE models.

A taxonomy of user-oriented values, motivations and emotions, described by Thew and Sutcliffe [14], included limited explanation of possible implications for each value and motivation in specification of high-level user goals. Fuentes-Fernandez [27] applied Activity Theory to RE, elaborating UML schema with social issues and proposing patterns for recognizing stakeholder conflicts; however, these did not give specific advice about eliciting or analysing users' values and emotions. RE methods have been proposed for analysing values in the sense of economic worth for services [28], although only general advice about considering political and social issues is given in most RE methods [1,26]. Goal-oriented RE has been extended by proposing goals for individual users and cultural considerations in requirements elicitation [29]. A more detailed taxonomy of social and political RE issues with guidelines for recognizing affective reactions among stakeholders was proposed by Ramos et al. [4], who applied their approach in analysing requirements for ERP applications. Zradkovic et al. [17] surveyed values among undergraduate and postgraduate students in distance education applications, reporting that universalism (ethics), benevolence, self-determination (play), achievement and stimulation (play-fun) were important influences.

Garde and Knaup [30] argued for a grounded theory approach to RE to deal with the complexity of the domain and socio-political issues; while Cysenieros [31] reviewed a variety of requirements elicitation techniques suggesting that technique combination might be more effective. Technique combination (scenarios, prototypes and linguistic corpus analysis) has been applied to user requirements with different stakeholder viewpoints in a decision support system [32]. Apart from the merit of combining RE techniques and noting the complexity of socio-political issues inherent in RE, advice on how to elicit and deal with such issues is still fragmented and incomplete.

B. Technology Acceptance

The original Technology Acceptance Model [18] asserted that Perceived Usefulness (PU) and Perceived Ease of Use (PEoU) influenced users' behavioural intention and hence product acceptance. Subsequent studies have elaborated the original model, leading to TAM3 [18, 19] which includes many antecedent variables organized in four main areas: individual differences, social influence, product characteristics and facilitating conditions (context). Antecedents cover social factors such as subjective norm (group conformance), image (self-esteem), job relevance; outcomes, e.g. output quality (goal-related), result demonstrability (visibility of outcomes); and user attributes such as IT efficacy (self-control), playfulness (curiosity), enjoyment, and anxiety. In addition objective usability and experience (hands-on use) were proposed as important moderating variables.

Other studies of system acceptance have proposed personality factors [33] and design attributes such as interactivity as antecedents [34,35] that influence users' preferences and behaviour, operationalized as e-loyalty, trust or website return visits in e-commerce. Wu et al. [36] showed that a predisposition towards innovative use of technology promoted perceived ease of use and intention to use, while Magni et al. [33] also demonstrated the association between

users' attitudes to innovation and increased tendency to explore/adopt technology. However, TAM recommendations tend to be explanatory rather than prescriptive design advice; furthermore, reviews of TAM predictions and actual use have shown that the core variables PU and PEoU are only weakly predictive [37]. In an extensive review of health-related TAM research, Holden and Karsh [38] concluded that TAM models were moderately accurate in predicting IT acceptance, but more research on contextualization was necessary. Venkatesh et al. [18] recommended that further research is necessary to unpack the TAM variables to give more specific recommendations for organizational management, and peer support, training, incentives, user participation and design characteristics.

C. User Experience and Values in HCI

In HCI, the VSD (value-sensitive design) method [13] proposed a process for eliciting user feelings and attitudes to potential systems. Scenarios and storyboarding techniques are used to elicit stakeholder responses, but value-based design does not focus directly on requirements; instead, it aims to elicit users' attitudes and feelings about products and prototypes as an aid towards refining product designs with human-centred values. Values and affective responses were investigated by Cockton [39] in worth maps, which document stakeholders' views about products or prototypes as informal descriptions, expressed in stakeholders' language as feelings, values and attitudes.

In user experience (UX), value-related components that contribute to users' overall judgement of IT products have been identified as usability, service quality (similar to utility) and classic and expressive aesthetics [40], while pragmatics (an amalgam of utility and usability) and hedonics were proposed by Hassenzahl [41] as antecedents to judgement of general product qualities of goodness and beauty. While no overall consensus of variables influencing overall judgement has emerged in UX research, the more important components appear to be utility/pragmatics and aesthetics/hedonics [42-44]. Kujala et al. [44] reported that both pragmatic and hedonic qualities contributed to attractiveness over a 6-12 month period; this is supported by Mendoza et al.'s [45] finding of decreased frustration over time, as users overcome initial usability problems. However, overall preferences for websites with similar content but different designs can be swayed by framing effects of tasks and users' characteristics [46]. Interactive system features may also influence user judgement of effectiveness, efficiency and overall attitude [34-35,47]. In a study of product acceptance among medical students, Hart and Sutcliffe [48] found that functionality of the device (iPAD) and apps was the most important influence on acceptance, with useful functions overcoming poor perceived usability. However, contextual factors such as no perceived need, lack of training and poor fit with working practices also influenced rejection of the iPAD.

D. Methods/Techniques for Socio-Political RE

Apart from ethnographic analysis and scenario-based techniques, little explicit guidance has emerged to shape analysis of socio-political requirements. Value-Based Requirements Engineering (VBRE, [14]) proposed a scenario-based analysis with a taxonomy of values and motivations as an elicitation guide, supplemented with hints on potential design implications. Use of values in agile development and participatory design using the Schwartz [49] taxonomy has been described with a framework of roles for value champions, etc. and elaboration of user stories [50]. The Schwartz taxonomy was applied in a case study that deconstructed the European data privacy regulations (GDPR) [51], showing the dependencies between rights, principles, and values such as privacy, trust, transparency, accuracy and legality. Further case studies have illustrated how the impact of values depends on organizational culture [52] and interpretation of values depends on the domain context, with proposals for value-oriented personas in e-health applications [53].

III TAXONOMY OF SOFT REQUIREMENTS

Soft requirements can be informally defined as issues attributable to people, organizations and society which may have direct or indirect implications for RE. Drawing a precise boundary between SRs and the related RE concepts of NFRs and soft goals is not productive, since SRs are a linguistic concept that embraces a wide range of human-oriented phenomena which have received increasing attention in RE, e.g. values, motivations, socio-political and systems issues [14,26,54].

Soft requirements can extend NFRs/soft goals from the perspective of the product adoption life cycle. Some soft goals become satisficed by hard goals/FRs, e.g. privacy, adaptability; however, SRs also cover attributes of people as values and motivations and social support, which do not involve software. SRs may be emergent because their influences are subtle and can be difficult to anticipate at design time, as illustrated by AI systems [10]. Emergence may be a consequence of the complexity of the social system within which the IT product is embedded, i.e. multiple stakeholders, groups of people, society level influences. It may be also caused by change in location and the operational context in mobile systems. A challenge for RE is to anticipate possible SR influences early so either stakeholder goals can be modified or social support can be provided as explanation, documentation, advice and training.

The taxonomy we propose considers SRs in five categories, as shown in Table 1. *Individual user SRs*: attitudes, feelings and characteristics of people which may influence their behaviour when using IT products. These include emotions which may influence their reactions to systems [4], values and motivations that may have an indirect influence on design and hence FRs [14]. User intent and goals, which are familiar FRs, are also attributable to individual users.

Table 1. Soft requirements taxonomy including FRs and NFRs where appropriate

Attribute	Category	Req. types	Implications & examples
	Values (value for money)	SRs, NFRs, emergent	Social responsibility hedonism, helpfulness, fairness

Individual users	Motivation, emotions	NFRs, SRs	Belonging: social media, reaction to products
	Experience, skills	NFRs, SRs emergent	Adaptation, customization
	Goals Usefulness	FRs	Traditional RE, also personal user goals
Context of use	Facilitating conditions	SRs, emergent	Training, help, explanation, persuasion
	Trust	SRs, emergent	Reputation, transparency, feedback
	Values-location, mobility	emergent context FRs	Localization, contextual adaption, convenience, culture
Product qualities	Pragmatics utility	FRs	Traditional RE, user goals, task fit
	Hedonics aesthetics	NFRs	Aesthetics, attractiveness design
Organiz-ations	Values	SRs, emergent	Sustainability, social responsibility, recommendations
Society	Socio-political values	SRs emergent	Equality, diversity, inclusiveness, fairness

User skills, experience and preferences [24] have implications for personal goals and adaptation/customization requirements, such as monitors, editors and change functions that improve the fit between users' goals and product functionality by adaptation, configuration and customization. Other SR examples such as value for money [28] may be a critical concern in market-related RE (e.g. internet applications with subscription, free apps and alternatives). Value SRs may depend on contextual SRs such as trust.

Context of use: these include aspects of the social system which facilitate system use such as training, installation and help desks, thereby reducing the costs of using a product. Promotions, advertising and packaging fall into this category as they may increase the perceived reward of the product. Trust in IT products, and by implication their supplier, is an important part of this category which may have direct implications for FRs in the form of reputation badges (e.g. membership of trusted organizations) and transparency (contact details and feedback facilities). However, trust is also an emergent SR which depends on experience in use and an organization's reputation. Trust as a relationship can be measured from assessment of reputation, experience and attitude of users to products, suppliers, etc. Other contextual SRs are influences on users' choice from product reviews, experience with similar products, recommendations from friends and social media.

Product features: include FRs and NFRs, with product line features; however this category also includes higher-level attributes from HCI: pragmatics/usefulness, hedonics, beauty and aesthetics. The fit between users' values, characteristics and preferences, and product design qualities such as aesthetics and attractive products, complements task fit between users' operational procedures and how FRs operate in the product user interface. Another product SR is compatibility between the software and its intended operating system or hosting platform, often a market-place consideration in product choice.

Organizations: values may be attributes of organizations enshrined in their ethos, while goals become aims and objectives at the organizational level, realized as policies and procedures. Organizational SRs may become apparent when values are reflected in company recommendations or standards for application choice such as security, sustainability or operating system compatibility.

Society: this category encompasses social, political and economic values and issues, the implications of which are frequently emergent. Some implications may be implemented in law where legal-oriented requirements have been extensively researched [55]. Society values also impinge on individual choice via social media and cultural influences.

We argue that requirements relating to users' motivations, needs and values are important additions to RE. In psychology, motivations are considered as long-term user goals related to personality, expressed as constructs ranging from basic needs such as safety and satisfying hunger and thirst, to achieving and possession of objects, to more social-personally related constructs of self-actualization (achieving own ambitions), self-esteem, altruism and belonging (to groups) [56]. This category has been partially addressed in the concept of personal user goals [29] where goals are individual targets for achievement, which the system may support and monitor. Thew and Sutcliffe [14] have also drawn attention to the importance of users' values which extend soft goals/NFRs beyond privacy, security, etc. to people-oriented constructs of trust and cooperation.

The implications of SRs depend on the domain context [57]. Values may have direct implications for FRs, e.g. sustainability and green values could suggest content for a politically oriented website. Social responsibility in the Covid-19 pandemic led to requirements in the Zoe app [57] to encourage symptom reporting and data sharing to track infection for research purposes. Trust, in contrast, depends on the provenance of the software and its supplier/ organizational owner; however, it may have FR implications for explanation to assuage user concerns and the need to display trust-engendering information, e.g. trust authenticators and feedback facilities in e-commerce websites.

Persuasive technology [58] is a SR-laden domain, familiar from recommender applications present in many leading internet systems (e.g. Amazon, Facebook); however, it also covers a wide range of applications where human behaviour change is the high-level goal. These applications have several SRs which often conflict: for example, system owner stakeholders aim to sell more products by tracking users' purchases and internet behaviour. This creates a privacy value conflict with an economic SR (maximize profit). In healthcare, recommenders attempt to persuade people to adopt healthy life styles (e.g. give up smoking, improve diet), which may conflict with privacy values. When recommendations are based on big data and ML classifiers, value conflicts with equality and fairness may be the consequence [10]. SRs provide a perspective for requirements engineers to anticipate such

conflicts and resolve them, if possible, through stakeholder negotiation.

Interactions between SR categories are shown in Figure 1. The bounding box, including user, context of use and product SRs, illustrates the close coupling between these categories where requirements, both FRs and SRs, are connected in terms of goodness of fit between the product to be designed (FRs plus product qualities), that matches not only the goals of users but also their SRs in skills, experience and preferences. Context of use implies fit of FRs and product qualities in different contexts, a key concern in mobile apps and the need to localize products in different countries, cultures and languages. Context of use SRs modifies user requirements, both hard and soft, via training, explanation and user support. Product features should map to user requirements via adaptation and customization to individual users; thereby improving the fit of product features to individual user skills and preferences. Adaptation may be implemented either through software facilities or via training manuals as system support.

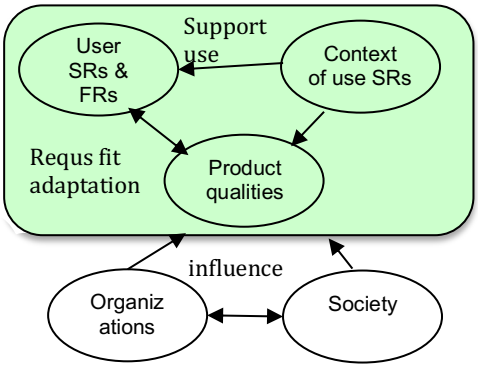

Fig. 1. Interaction between categories of soft requirements

Adaptation may be realized either by human intervention or automated by monitoring system operation; cf. awareness requirements [59] applied to the product, which then invoke adaptive change to software automatically. Organizational and society-level SRs influence both individual user and context of use requirements and *inter alia* product features; however, these requirements tend to be emergent. An important direction for future research is to identify these SRs before implementation by anticipating their influences. The values category of SRs are a pervasive influence across all categories since individual users may be influenced by values of organizations they belong to and the society in which they live. Context of use reflects the local, national/society context; however, the connection between values and product qualities is not always immediately apparent [57]. SR categories because they are linguistic constructs show degrees of overlap which have to be resolved via ontologies and semantic lexicons (e.g. WORDNet), and Table 1 suggests relationships to unpack SRs which can not be formally design, as is the value with values and similar cognitive constructs [13, 49].

The relationship between software requirements, NFRs and FRs is shown in figure 2, illustrating the pathways by which requirements may be refined during specification. Both NFRs and SRs belong to the social part of socio-technical systems, i.e. they are attributes of people, organisations and societies. Functional requirements become specifications of software components, data structures and algorithms; while non functional requirements may be refined into quality criteria by which system performance can be measured. However, several NFRs are refined into functional requirements which satisfy the desired quality, e.g. privacy and security delivered by encryption, secure access and communication protocols. The refinement pathway for SRs shares the quality criteria destination with NFRs, e.g. values for sustainability might be measured by power consumption. SRs may also progress towards FRs which are necessary to implement information display and decision support functions. However specification of SRs may results in decision criteria for critical success factors for system acceptance or system/user support facilities such as training, help desks, etc.

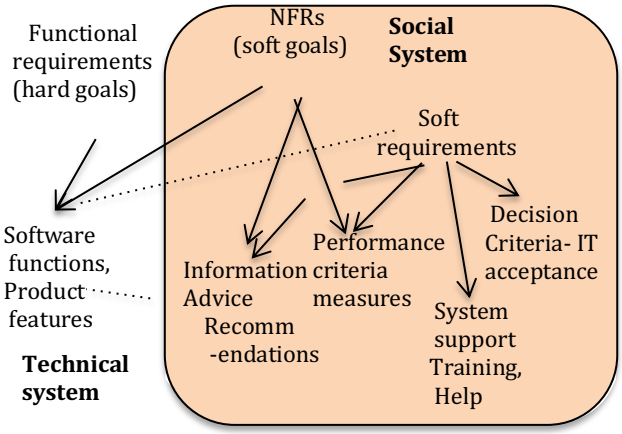

Fig. 2. The relationship of soft requirements, NFRs, and FRs with refinement pathways

In the following section we extend the concept of individual SRs towards critical success factors for system acceptance.

IV Soft Requirements & Technology Acceptance

Soft requirements need to be understood not only *a priori* during the design process, but also as emergent influences over time from the pre-use design phase to early and later stages of product adoption. We argue that a temporal perspective is essential because the implications of many soft requirements will become manifest in system support, such as installation, configuration, and user support for learning, explanation and help before and during use. The model we propose focuses on an important trade-off in SRs, between the perceived motivations for use balanced against costs, in terms of human effort in learning, configuring and using the system. The model, illustrated in Figure 3, presents a general abstraction of user-perceived and realized benefits contrasting usage costs over time. It is based in the HCI UX and TAM literature, where surveys of user experience have demonstrated that users'

acceptance and continued use of an application are determined by the balance of experienced utility benefits and usability costs with anticipated benefits from future use [43-45].

The figure illustrates three possible trade-off scenarios between benefits arising from effective use which increases users' motivation for continuing to use the application and pitted against costs of learning to use it. In scenario 1 learning costs increase initially but then decline, while the reward of effective use gradually increases. Rewards always exceed costs and the system is accepted. In scenario 2 rewards increase gradually as before; however, learning costs rise more steeply, due to poor design and/or poor training. Costs exceed the benefits and the user abandons the system. In scenario 3 the perceived benefits decline over time, possibly due to poor fit between users' needs and functionality, and/or usability problems. Costs again exceed benefits and the user gives up. Costs will vary between domains and be subject to contextual factors, for instance in application platforms (viz Facebook Apps) there is user and supplier 'lock in' which impose costs; in business domains company standards may impose learning costs for compliance. Figure 3 illustrated the general point about trade offs and change over time, naturally interpretation of instance models depends on the domain.

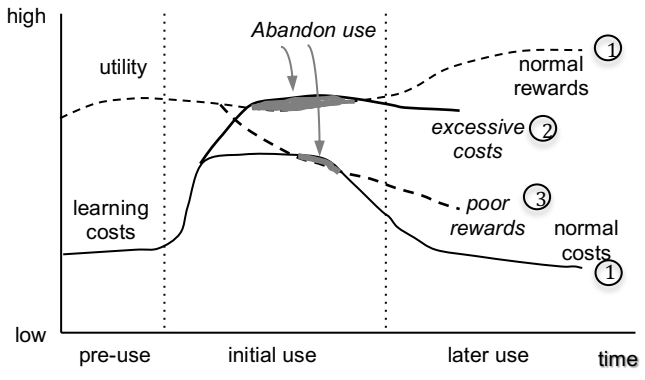

Fig. 3 Timeline view of the cost-benefit balance in system acceptance [44,45,48]

The figure illustrates a space of cost-benefit trade-offs with possible trajectories for products with different realizations of hard and soft system requirements with their implications for users' reactions as a balance between motivations and costs. Motivations can be fulfilled by users experiencing utility derived from FRs, although other motivations may be depend on the domain; for example, features to satisfy excitement and curiosity in games, entertainment and educational applications, peer and self-esteem in social networks and e-communities. Costs are the effort required to learn and use the application and may include installing and configuring software for personal needs. In the pre-use phase, benefits are the perceived expectations of utility, ease of use and possibly aesthetics in certain domains such as design-oriented websites in e-commerce [46]. Costs, based on perceptions at this stage, are low. In the initial phase, costs climb rapidly when users have to expend the effort of learning to use a new product, and possibly configuring and customizing products. Costs increase as users encounter usability problems, while the rewards of achieving goals may only be partially realized. At this stage, continued use is maintained by the expectation of future rewards. In later use phases, rewards increase as users achieve their goals, while costs decline as usability problems are solved or avoided by work-arounds. Acceptance is assured so long as the reward curve exceeds costs. If costs are too high or the perceived and realized rewards are too low, then users will abandon a product. Understanding user rewards and costs is vital to ensure effective RE for successful products for two reasons: first for achieving the best fit between FRs and SRs for how they operate in the user interface of interactive products and users' ways of working; and secondly to provide support that will reduce users' costs of operation (e.g. training) as well as increasing perceived benefits (e.g. by guided tours, explanation, unfolding more advanced functions).

The contribution of SRs to the product adoption timeline is illustrated in Figure 4, which illustrates how one such requirement, customization, may influence product acceptance. The shaded zone depicts the increase if both costs and benefits arising from customization in the upper line from no customization in the lower line. Trust in the product brand and other value related SRs will set the baseline for both anticipated benefits and costs. Assuming customization is undertaken early in the initial use phase, costs increase sharply as the user has to learn to operate editors and user interfaces to achieve customization/personalization and experiment with different options. Benefits at this stage are modest since operation of the new customized system will not have been used to realize the improved fit between the users' requirements and system functions. Costs remain high since the new customized features may also impose a learning burden and usability errors. Eventually these costs will decrease and benefits will continue to accrue over the product lifetime as it delivers more efficient and effective operation. The initial imbalance between costs and benefits explains why many users never customize products. If the perceived and actual costs are too high and initial benefits are not realized, then the system may fall into the zone of abandonment, shown in Figure 3, scenarios 2 and 3. SRs such as training, user guides, tutorials and peer mentoring can all help to reduce learning costs as well as explaining the benefits of improving the task fit for users.

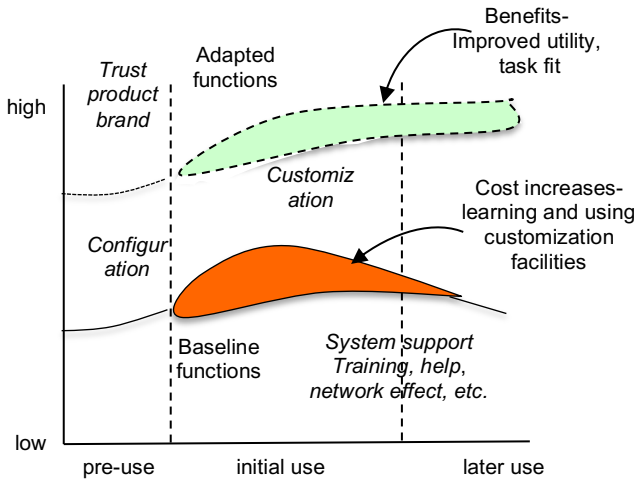

Fig. 4. Timeline of product acceptance showing phases, and the contribution of customization to costs and benefits [24,34, 48, 58]

Another example which fits within Figure 4 is the network effect SR in which social media apps, e-communities and other collaboration support systems increase in attractiveness as more users are recruited to join, thereby increasing the utility payback for each individual user. Costs in this case are forming relationships with other users, supported possibly by FRs for user profiles. SRs play important roles in product installation, configuration, training and user support during the pre- and initial use phases. Benefits depend on the match between requirements realized in an implementation, users' goals and social support, i.e. constructs which influence and contribute to the process of delivering system acceptance by users that are not direct user goals or FRs. The selection of SRs associated with each life cycle phase are illustrated in Table 2. Soft requirements dominate in the requirements phase where analysis of values, motivations and emotions may shape soft goals such as privacy, collaboration and innovations from values. User characteristics inform personal goals and personalization functions while motivations can suggest user interface design options, e.g. peer esteem by status indicators in social applications. At the group level, users' norms and cultures may need to be considered.

Table 2. Soft requirements associated with each adoption life-cycle phase

Pre-use	Initial Use	Later Use
Promotions Explanation Demonstrations Brand reputation, trust External influences (reviews) Perceived utility	Utility (FRs) Usability Customisation Adaptation Training User support Social support Network effects	Utility (FRs) Efficiency Effectiveness Reliability Adaptability Evolution-flexibility Customer support

In the pre-use phase, SRs are a mix of marketing concerns (brand, reputation, promotions and demonstrations) and perceived product properties which users acquire from marketing or from exploring existing similar products. Prior to use, advertising, promotions, brand reputation and trust may increase perceived benefits and influence users' motivations to use a product. During initial use, utility and usability become key variables influencing users' attitudes and preferences. Installation support, configuration and customization facilities reduce learning and operating costs in the long term by improving the fit between the users' goals and product features. Customization and configuration may be refined to specify FRs for editors and facilities that enable users to choose options and alter the user interface to suit their needs. Further SRs in this phase are training and usage support (e.g. help desks), and social support where users may form self-help communities to solve usability problems with network effects for collaborative systems.

In the later use phase, traditional FRs that deliver utility for achieving user goals become increasingly important, although experience-sensitive user interface design can help users to gradually discover layers of complexity in a system, so that they are not swamped with too much functionality in the initial stages of learning to use a system. Over-complex systems might lead to rejection through a combination of learning costs and lack of perceived benefits. System functionality delivers efficiency through improved use while effectiveness reflects the impact of the whole socio-technical system realized as the integration of the software system and human operational processes that delivers more effective impact through learning and practice. Adaptability enables change and extension either automatically or by human-in-the-loop change, while evolution-flexibility is the capability of the system to monitor itself and either self-adapt [59] or be changed by human intervention. These SRs are related to evolutionary and awareness requirements [59] which need to be implemented before use in design of the software architecture, even though their benefits are only realized in longer-term use. Modelling user acceptance requirements using game paradigms [60] could be used to explore evolution of the impact of SRs over time

More formal modeling of SRs and their implications may be developed using the evolutionary intentions framework [61] to explore change in goal dependency models over time. However, the road to formalization of SRs may need to follow a more probabilistic rather than deterministic route since many SR are assumptions which may influence goals (or FRs) to a greater or lesser degree. Bayesian or other probabilistic models could be employed, although the effort of formalizing SRs which are weak influences may not justify the effort. We argue the main benefit of SRs is 'tools for thought' to be considered during the requirements and design process.

The framework is intended as an agenda of issues to guide requirements elicitation. The timeline trade-off model provides a management perspective for reviewing which SRs need to be considered before design and the role of support requirements in system design, as well as planning product release and support. Another perspective of the framework is consideration of the socio-technical system fit, extending RE into design of the human social system and operational procedures, as advocated by Callele et al. [54]. Soft requirements also focus

attention on how system acceptance may be determined not only by the software but also by changes that may be demanded in the social system, e.g. to working procedures, power relationships, responsibilities and authority. Many of these issues have been acknowledged in RE [3,4, 26], but methods and guidelines for addressing them are still limited.

We have used the term 'users' rather than 'stakeholders' in our review to direct attention towards operators of the system, because in internet applications users are the main stakeholders and owners. However, SRs are applicable to multi-stakeholder analysis which is beyond the remit of this initial framework. For example, whereas primary stakeholders (users) will experience benefits from use of system functions and support in training, secondary stakeholders (e.g. managers) who experience the system indirectly will have different SRs relating to the social system such as improving information, ability to control processes, workflows and manage people.

IV Research Road Map

Soft requirements pose several questions for future research. Essentially, there are two high-level problems: (i) analysis and planning SRs for technology adoption; and (ii) eliciting and specifying the implications for SRs in a wide range of applications where human choice, persuasion and decision making are major objectives. Discretionary-use applications in e-commerce, healthcare, education and entertainment pose these problems, since users first have to be persuaded to purchase and use them, and the system aim is to amuse, educate or influence the users' behaviour.

The product adoption timeline and taxonomy of soft requirements issues provide an agenda to guide requirements analysis, but SRs rely on expert interpretation from other domains, such as the psychology of motivation and persuasion. One research issue is how much of this knowledge can be codified and incorporated into practice for requirements engineers; alternatively, should this be solved by requirements management, e.g. building multi-expert teams?

Schwartz values [49] and the VME taxonomy [14] provide guidance for value-based SRs but they need to be extended with social norms and culture [60] which may also motivate SRs for localization in global software markets. Further research on SRs and values taxonomies is necessary; however, rephrasing knowledge from psychology and sociology will be of limited use unless psychological constructs are linked to implications in the RE process and software architecture/design considerations. To improve the utility of SR knowledge for requirements engineers, research on the design implications of human-social-economic issues is necessary. Some views have been expressed on generic architecture implications of values [62], but more research is necessary to encapsulate expert practitioner experience, possibly as requirements architecture patterns. For example, the autonomy value implies self-control and freedom; however, this raises questions about software surveillance of human activity either practised overtly in image recognition/identity tracking security software or tacitly in social media. Awareness requirements [59] may therefore be connected implicitly with freedom and further SRs to safeguarding personal information. An example is GDPR legislation, where value-based analysis has been applied to investigating the potential implementation of privacy protection requirements [51].

The synthesis of RE and TAM perspectives raises the question of critical system success criteria which has received less attention in RE. The timeline and SR taxonomy provide simple tools for thought to guide analysis of issues that extend FRs and NFRs. However, the requirements phases and taxonomy need to be specialized for specific domains; for instance, there is considerable e-commerce research on persuasion to buy [58]. In contrast, persuasion in command and control domains to take a course of action recommended by the system will involve different psychological issues of situation awareness. The research road map is summarized in Table 3. The first goal, to extend the SR taxonomy, needs methodological and ontological investigation to define the concepts. Experimentation may help to ensure the terms are comprehensible and useful for analysis tasks, while case studies will help to understand SRs in current RE practice.

Table 3. Research issues and approaches

Research goal	Research approach	Related areas
1. Extend SR taxonomy	Literature reviews Case studies, Surveys Experiments	TAM, IS, values psychology, social sciences
2. Explain and document SRs	Methodology Surveys Case studies	Psychology, HCI, culture, socio-economics
3 Measure SRs	Surveys, experiments	Psychology, Social sciences, HCI
4. Specify implications	RE expertise capture Case studies	RE methods, scenarios, software architecture, requirements patterns
5. Tailor for domains	RE expertise capture Scenario analysis	Domain analysis, IS marketing, management studies
6. Apply and validate SRs	Tool support Case studies surveys	Systems engineering, domain studies

Information systems and management science literature may help ground this research for the second goal. Specifying implications to improve the utility of SRs for requirements engineers implies research to capture expertise from designers in HCI (user experience), software architects, practising requirements engineers and software developers. Encapsulating such knowledge as patterns linking SRs to implications and generic software architecture could realize the third goal. The fourth goal, for domain tailoring, may be best served by application, as in the fifth goal where experience in different domains will be necessary to refine implications and methods. Finally, application and validation need case studies of applications in real-world contexts as well as surveys to gather feedback on the effectiveness or otherwise of SRs in improving RE.

Elicitation methods may follow conventional RE techniques, such as scenario-based analysis in workshops and interviews. Previous values/SR-related research has suggested that scenarios, personae, and use cases in a technique combination approach [25,31,52] may be a suitable way forward. However, further research is needed to extend RE techniques with more active guidance for eliciting and analyzing the implications of SRs, e.g. improve taxonomies, patterns, support by hypertext tools to explain implications. Other possibilities are survey approaches based on taxonomies of values to capture high-level population views on acceptable trade-offs. These may be integrated with scenario-based testing of design options motivated by trade-offs for, and between, different stakeholders. We propose that informal SR maps may be useful support for reasoning about dependencies between SRs and FRs, as illustrated in Figure 5.

Figure 5 is based on a survey and interview study of the UK National Health Service (NHS) track and trace app [57], showing the SR influences on product/app acceptance. User SRs dominate choice as values (helpfulness, social order and equality), with emotion (fear of the disease) and trust in the provenance of the software and the socio-technical track and trace system.

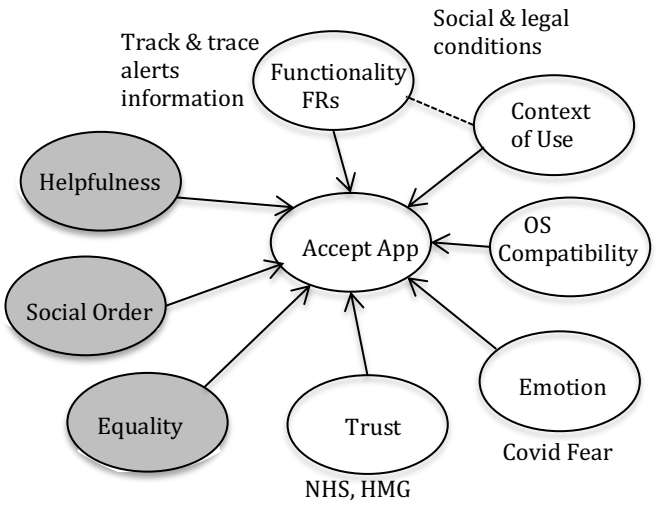

Fig. 5. Illustration of SRs for the NHS Track and Trace App, based on [57]. Values are shaded

Society values (not illustrated) interacted with user values with interpretations of helpfulness being explained as social responsibility to help research and prevent spread of the disease; similarly social order, with equality related to a feeling of 'all being in this together'. While most of the influences were positive, trust was divided among users between positive attitudes to the NHS (UK National Health Service), and negative attitudes towards the UK Government (HMG). OS compatibility referred to availability of the app on Android and Apple iOS operating systems. Functional requirements were present; however, these interacted with context of use SRs, as several users felt they did not need the app since they were self-isolating and hence unlikely to come into contact with others; others felt coerced into accepting the app since HMG mandated its use to access social and work venues. Acceptance was a complex picture, with FRs playing only a minor role compared to SRs. Indeed, the main system goal to provide track and trace functionality to alert users when they had been in proximity of an infected person was compromised by SRs of poor accuracy leading to many false alerts and leading to system rejection. Informal influence maps (as in figure 5) could be represented by dependencies in i* models [6], although different types and strengths of soft goal (SR) dependencies are another issue for further research.

Negotiation support could use a combination of informal maps, i* dependency diagrams and matrix representations to show interactions between SRs and FRs. Informal models combined with scenarios to express current and future systems could enable users and other stakeholders to discuss design options. Web-based scenario testing may allow selecting design options to be scaled up to the population level. A further issue is how to involve stakeholders themselves in trade-off decision making, particularly for persuasive applications. Here design rationale research might be revisited and combined with scenarios using video or animations to increase realism.

Tool support could start by adaptation of diagram editors and hypertext websites to link issues, and recommendations. A further tool research challenge lies in semi-automated analysis of SRs in text mining. Unfortunately, identifying SRs is difficult for the current generation of corpus text analysis tools, although advances in sentiment analysis point towards possible solutions.

VI Conclusions

Revisiting the three aims we set out for this perspective, we have produced a taxonomy of SRs which contribute towards research on a wider range of soft issues in RE. SRs need to inform future development of existing NFR taxonomies and standards [23,]. The second aim, to examine the convergence of technology acceptance, stimulated the time-line view of RE which points to further research in cost-benefit trade-off analysis as well as a new conception of requirements 'fit' in context of use. The third aim indicates several future research areas such as SR-inspired 'requirements patterns' to complement established SE patterns; and revisiting RE methods with new approaches to combine scenarios, argumentation and prototypes. Design thinking, creative cognition, and design fictions [64,65], which have been applied as new means of exploring requirements, could be guided by SRs. Soft requirements extend the unknowns conception of the RE problem while bringing together RE research with information systems/TAM and HCI perspectives of technology adoption. We believe there is further synergy at the boundary, for example in technology diffusion models and possibly automating adaptability for product-line RE.

SRs are relevant to a range of development contexts ranging from agile approaches to more structured development. SRs may fit into a management perspective of risks to consider during a project, whereas developers may use them to challenge assumptions in requirements, models and prototypes, as well as user stories in agile approaches. We have also explored

established research on a range of soft issues in RE, notably values, while pointing towards a wider perspective of social norms, culture and user motivations which have received less attention. The synergy between technology acceptance modelling in information systems in RE merits further attention; however, the SR quest extends to the social sciences and economics. A final reflection is that progress in soft RE is becoming more pressing with the changing nature of the problem as technology advances and with the current debate over the implications of values for AI. Acceptance of many new applications, indeed apps, is also influenced by trust and many SRs. Understanding how apps succeed, evolve and survive in the ecosystem of Facebook, Apple iOS, and Android is another challenge.

This research was partially supported by EPSRC project (EP/T017627/1) Twenty-20 Insight.

References

[1] S. Robertson and J. Robertson, Mastering the requirements process. Harlow: Addison Wesley, 1999.

[2] A.C.W. Finkelstein and J. Dowell, "A Comedy of Errors: The London Ambulance Service case study", Proceedings 8th International Workshop on Software Specification and Design, 1996, pp. 2-4.

[3] M. Bergman, J.L. King and K. Lyytinen, "Large-scale requirements analysis revisited: The need for understanding the political ecology of requirements engineering", Requirements Engineering, vol 7 pp. 152-171. 2002

[4] I. Ramos, D.M. Berry and J. Carvalho, "The role of emotion, values and beliefs in the construction of innovative work realities", Proceedings Soft-Ware 2002: Computing in an Imperfect World. Berlin: Springer, 2002, pp. 300-314.

[5] M. Glinz, "On non-functional requirements", Proceedings 15th IEEE International Requirements Engineering Conference, pp 21-26, 2007.

[6] E.S.K. Yu, P. Giorgini, N.A.M. Maiden and J. Mylopoulos, "Social modeling for requirements engineering: an introduction" in E.S.K. Yu, P. Giorgini, N.A.M. Maiden and J. Mylopoulos (eds) Social Modeling for Requirements Engineering. Cambridge, MA: MIT Press, 2011, pp. 3-10.

[7] D. Mougouei, H. Perera, W. Hussain, R. Shams and J. Whittle, "Operationalizing human values in software: A research roadmap", Proceedings of 26th ACM Joint Meeting on European Software Engineering Conference, 2002, pp. 780-784.

[8] J. Whittle, M.A. Ferrario, W. Simm and W. Hussain, "A case for human values in software engineering", IEEE Software, vol 38 (1), pp 106-113, 2019.

[9] A. Paiva, I. Leite, H. Boukricha and I. Wachsmuth, "Empathy in virtual agents and robots: A survey", TiiS, 2017, vol 7,3 pp. 1-40.

[10] R.K.E. Bellamy, K. Dey, M. Hind, S.C. Hoffman et al., X Think your artificial intelligence software is fair? Think again", IEEE Software, 2017, pp. 76-80.

[11] I.T. Sommerville, P. Rodden, P. Sawyer, R. Bentley and M. Twidale, "Integrating ethnography into the requirements engineering process" Proceedings IEEE Symposium on Requirements Engineering, 1993.

[12] M. Jirotka, R. Procter et al., "Collaboration and trust in healthcare innovation: The eDiaMoND case study". Journal of Computer-Supported Cooperative Work, 2005,vol 14, 4. pp. 369-398.

[13] B. Friedman, "Value-sensitive design" in D. Schular (ed.) Liberating voices: A pattern language for communication revolution. Cambridge MA: MIT Press, 2008, pp. 366-368.

[14] S. Thew and A.G. Sutcliffe, "Value Based Requirements Engineering: Method and experience", Requirements Engineering, 2018, vol 23,4, pp. 443-464.

[15] H. Perera, W. Hussain, D. Mougouei, R.A. Shams, A. Nurwidyantoro and J. Whittle, "Towards integrating human values into software: Mapping principles and rights of GDPR to values in 2019", IEEE 27th International Requirements Engineering Conference, 2019, pp. 404-409.

[16] W. Hussain, H. Perera, J. Whittle, A. Nurwidyantoro, R. Hoda, R.A. Shams and G. Oliver, "Human values in software engineering: Contrasting case studies of practice", IEEE Transactions on Software Engineering, 2020.

[17] J. Zdravkovic, E.O. Svee and C. Giannoulis, "Capturing consumer preferences as requirements for software product lines", Requirements Engineering, 2015, vol 20,1. pp. 71-90.

[18] V. Venkatesh and H. Bala, "Technology Acceptance Model 3 and a research agenda on interventions". Decision Sciences, 2008, vol 39,2, pp. 273-315.

[19] N. Marangunić,and A. Granić, "Technology Acceptance Model: A literature review from 1986 to 2013", Universal Access in the Information Society, 2015, pp. 81-95.

[20] C. Ncube and N. Maiden, "Guiding parallel requirements acquisition and COTS software selection", Proceedings IEEE International Symposium on Requirements Engineering, 1999, pp. 133-140.

[21] M. Morisio, C.B. Seaman et al., "COTS-based software development: Processes and open issues". Journal of Systems and Software 2002, vol 61,3, pp. 189-199.

[22] A.G. Sutcliffe and P. Sawyer, "Requirements elicitation: Towards the unknown unknowns", Proceedings 21st IEEE International Symposium on Requirements Engineering, 2013, pp. 92-104.

[23] L. Chung, B. Nixon, E.Yu and J. Mylopoulos, Non-functional requirements in software engineering. Amsterdam: Kluwer Academic Publishing, 2000.

[24] B. Hui, S. Laiskos and J. Mylopoulos, "Requirements analysis for customisable software: A goals skills preferences framework", Proceedings IEEE Joint International Conference on Requirements Engineering, 2003, pp. 117-126.

[25] A.G. Sutcliffe, "Analysing the effectiveness of human activity systems with i*" in P. Georgiini and E. Yu (eds), Social Modelling for Requirements Engineering. Cambridge MA: MIT Press, 2010, pp. 1139-1192.

[26] A. Milne and N. Maiden, "Power and politics in requirements engineering: A proposed research agenda", Proceedings 19th IEEE International Requirements Engineering Conference, 2011, pp. 187-196.

[27] R. Fuentes-Fernández, J.J. Gómez-Sanz and J. Pavón, "Understanding the human context in requirements elicitation", Requirements Engineering, 2010, vol 15,3, pp. 267-283.

[28] J. Gordjin, E. Yu and B. Van Der Raadt, "e-service design using i* and e3value modelling", IEEE Software, 2006, pp. 26-33.

[29] A.G. Sutcliffe, S. Fickas and M.M. Sohlberg, "PC-RE: A method for personal and contextual requirements engineering with some experience", Requirements Engineering, 2006, vol 11, pp. 157-163.

[30] S. Garde and P. Knaup, "Requirements engineering in health care: The example of chemotherapy planning in paediatric oncology". Requirements Engineering, 2006, pp 265-278.

[31] L.M. Cysneiros, "Requirements engineering in the health care domain", IEE International Requirements Engineering Conference, 2002, pp. 404-409.,

[32] A.G. Sutcliffe, S. Thew et al., "User engagement by user-centred

design in e-health". Philosophical Transactions of the Royal Society, 2010, pp. 4209-4224.
[33] M.M. Magni, S. Taylor and V. Venkatesh, "To play or not to play? A cross temporal investigation using hedonic and instrumental perspectives to explain user intentions to explore a technology", International Journal of Human-Computer Studies, 2010, vol 68, pp. 572-588.
[34] D. Cyr, M. Head and A. Ivanov, "Perceived interactivity leading to e-loyalty: Development of a model for cognitive-affective user responses". International Journal of Human-Computer Studies, 2009, pp. 850-869.
[35] H.-H. Teo, L.-B. Oh, L.-B., C. Liu and K.-K. Wei, 2003. "An empirical study of the effects of interactivity on web user attitude", International Journal of Human-Computer Studies, 2003, vol 58, 3, pp. 281-305.
[36] J.H. Wu, W.S. Shen et al., "Testing the technology acceptance model for evaluating healthcare professionals' intention to use an adverse event reporting system". International Journal for Quality in Health Care, 2008, vol 20, 2, pp. 123-129.
[37] M. Turner, B. Kitchenham et al., 2010. "Does the technology acceptance model predict actual use? A systematic literature review". Information and Software Technology, 2010, vol 52,5, pp. 463-479.
[38] R.J. Holden and B.T. Karsh, "The technology acceptance model: Its past and its future in health care", Journal of Biomedical Informatics, 2010, vol 43,1, pp. 159-172.
[39] G. Cockton, S. Kujala, P. Nurkka and T. Hölttä, Supporting worth mapping with sentence completion. Berlin: Springer, 2009.
[40] T. Lavie T. and N. Tractinsky, "Assessing dimensions of perceived visual aesthetics of web sites", International Journal of Human Computer Interaction, 2004, vol 60, pp. 269-298.
[41] M. Hassenzahl, "The interplay of beauty, goodness and usability in interactive product", Human-Computer-Interaction, 2004, vol 19, pp. 319-349.
[42] S. Diefenback and M. Hassenzahl, "The beauty dilemma: Beauty is valued but discounted in product choice", Proceedings CHI 2009, 2009, pp. 1419-1426.
[43] J. Hartmann, A.G. Sutcliffe and A. De Angeli, "Towards a theory of user judgment of aesthetics and user interface quality", TOCHI, 2009, vol 15, pp. 1-30.
[44] S. Kujala, V. Roto et al., "UX Curve: A method for evaluating long-term user experience", Interacting with Computers, 2011, vol 25, 3, pp. 473-483.
[45] V. Mendoza, D.G. Novick and E. Paso, "Usability over time", Proceedings SIG DOC '05, 2005, pp. 151-158.
[46] T. Porat and N. Tractinsky, "It's a pleasure buying here: The effects of web-store design on consumers' emotions and attitudes", Human-Computer Interaction, 2012, vol 27, 3, pp. 235-276.
[47] A.G. Sutcliffe, "Designing for user engagement: Aesthetic and attractive user interfaces", in J.M. Carroll (ed.) Morgan and Claypool, 2010.
[48] J. Hart and A.G. Sutcliffe, "Is it all about the apps or the device? User experience and technology acceptance among iPad users", International Journal of Human-Computer Studies, 2019, vol 130, pp. 93-112
[49] S.H. Schwartz, "A theory of cultural values and some implications for work", Applied Psychology, 1999, vol 48, 1pp. 23-47.
[50] W. Hussain, M. Shahin et al., "How can human values be addressed in agile methods? A case study on SAFe", IEEE Transactions on Software Engineering, 2022.
[51] H. Perera, W. Hussain et al., "Towards integrating human values into software: Mapping principles and rights of GDPR to values in 2019", IEEE 27th International Requirements Engineering Conference, 2019, pp. 404-409.
[52] W. Hussain, H. Perera et al., "Human values in software engineering: Contrasting case studies of practice", IEEE Transactions on Software Engineering, 2020.
[53] H. Obie, M. Shahin et al., "Does domain change the opinion of individuals on human values? A preliminary investigation on eHealth apps end-users", arXiv preprint arXiv:211001832, 2021.
[54] D. Callele, W. Krzysztof and B. Penzenstadler, "New frontiers for Requirements Engineering", IEEE 25th International Requirements Engineering Conference, 2017, pp. 184-193.
[55] D.G. Gordon and T.D. Breaux, "Assessing regulatory change through legal requirements coverage modeling", 21st IEEE International Requirements Engineering Conference, 2013, pp. 145-154.
[56] A.H. Maslow, R. Frager, C. McReynolds, R. Cox and J. Fadiman, Motivation and personality. New York: Addison Wesley-Longman, 1987.
[57] A.G. Sutcliffe, P. Sawyer, W. Liu and N. Bencomo, "Investigating the potential impact of values on requirements and software engineering", IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Society, 2021, pp. 39-47.
[58] B.J. Fogg,"A behavior model for persuasive design", Proceedings of the 4th International Conference on Persuasive Technology, 2009, pp. 1-7.
[59] V.E. Silva Souza, A. Lapouchnian, W.N. Robinson and J. Mylopoulos, "Awareness requirements for adaptive systems", Proceedings SEAMS-11, 2011, pp. 60-69.
[60] L. Piras, P. Giorgini, and J. Mylopoulos, "Acceptance requirements and their gamification solutions," 24th IEEE Int. Requirements Engineering Conference, 2016. Pp 365-370
[61] A.M. Grubb, & M. Chechik. . "Formal reasoning for analyzing goal models that evolve over time". Requirements Engineering, 2012, vol 26, 3, 423-457.
[62] M. Krumbholz, N. Maiden, B. Wangler and L. Bergman, "How culture might impact on the implementation of Enterprise Resource Planning packages", Berlin: Springer, 2000, pp. 279-293.
[63] A.G. Sutcliffe and S. Thew, "The design implications of users' values for software and system architecture" in I. Mistrík et al. (eds) Economics-Driven Software Architecture EDSA, Amsterdam: Elsevier, 2014.
[64] A. Darby, E. Tsekleves and P. Sawyer, "Speculative requirements: Design fiction and RE", IEEE 26th International Requirements Engineering Conference, 2018, pp. 388-393.
[65] J. Horkoff, N. Maiden N. and D. Asboth, "Creative goal modeling for innovative requirements", Information and Software Technology, 2019, pp. 85-100.