A Conceptual Method for Eliciting Trust-related Software Features for Computer-mediated Introduction

Angela Borchert, Nicolás Emilio Diáz Ferreyra and Maritta Heisel
Department of Software Engineering, University of Duisburg-Essen, Duisburg, Germany

Keywords: Trustworthiness, Computer-Mediated Introduction, Requirements Elicitation.

Abstract: Computer-Mediated Introduction (CMI) describes the process in which individuals with compatible intentions get to know each other through social media platforms to eventually meet afterwards in the physical world (i.e. sharing economy and online dating). This process involves risks such as data misuse, self-esteem damage, fraud or violence. Therefore, it is important to assess the trustworthiness of other users before interacting with or meeting them. In order to support users in that process and, thereby, reducing risks associated with CMI use, previous work has come up with the approach to develop CMI platforms, which consider users' trust concerns regarding other users by software features addressing those. In line with that approach, we have developed a conceptual method for requirements engineers to systematically elicit trust-related software features for a safer, user-centred CMI. The method not only considers trust concerns, but also workarounds, trustworthiness facets and trustworthiness goals to derive requirements as a basis for appropriate trust-related software features. In this way, the method facilitates the development of application-specific software, which we illustratively show in an example for the online dating app Plenty of Fish.

1 INTRODUCTION

Social media offers many possibilities as a media channel due to its large number of users who want to be connected with other people. Services like the ones using *Computer-Mediated Introduction* (CMI) support this wish for connectivity. CMI offers users a realm in which they can get to know and connect with unfamiliar individuals with compatible interests to potentially have offline encounters (Obada-Obieh and Somayaji, 2017). Compatible interests may involve human qualities or resources users possess to satisfy mutual needs. Examples for CMI are sharing economy and online dating. While sharing economy is based on monetary exchange to enable services like private lodging, car drives or dog sitting between users, online dating focuses on social exchange. Compared to other kinds of social media, CMI can further be characterized into the property that it has different stages concerning the interaction with users of interest. These stages are *before, during* and *after* CMI users are connected (Obada-Obieh and Somayaji, 2017). The before stage includes the search for an appropriate other user who fits a user's needs. The during stage denotes the establishment of contact, the online interaction as well as the offline encounter so that the during stage can be further subclassified in these steps. The after stage describes the disconnection of both users on the online platform.

Though the merit of CMI is that users get to know new people, this also bears risks such as fraud, damaged self-esteem or violence (Obada-Obieh et al., 2017). Cues that are usually available in face-to-face interactions and are important to get an impression of an other individual are partly missing, different than in the offline context or easy to manipulate for giving altered impressions (Walther et al., 2005). This complicates the trustworthiness assessment of users of interest in CMI. However, it can be assumed that the trustworthiness assessment is a decisive factor for the decision-making process whether to interact with or meet another person (Rotter, 1980). Especially in the context of offline encounters based on online introductions, users have stated concerns about safety (Couch et al., 2012).

In a previous work, we thus identified the need that CMI applications should better assist users in assessing the trustworthiness of other end-users (Borchert et al., 2020). Since a CMI system i) modulates the perception users have about each other, ii) mediates their interaction and iii) may trigger

Borchert, A., Ferreyra, N. and Heisel, M.
A Conceptual Method for Eliciting Trust-related Software Features for Computer-mediated Introduction.
DOI: 10.5220/0009328102690280
In *Proceedings of the 15th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2020)*, pages 269-280
ISBN: 978-989-758-421-3

offline encounters, it can impact peoples' well-being to a large extent. Seeing this as a responsibility the system should take, this work introduces a requirements engineering method for eliciting software features that shall support users in their trustworthiness assessment. The objective of the method is to build CMI systems whose previously described risks are reduced so that CMI use is safer. This shall be accomplished by offering a user-centred software solution that respects users' trust concerns. Trust concerns can be regarded as the expression of doubts in the trustworthiness of other CMI users or specific interaction situations that differ in each context (cf. Kipnis, 1996). The proposed method is, thus, issue- and application-specific.

2 RELATED WORK

To the best of our knowledge, little effort has been put into methods for incorporating trustworthiness in the development process of information systems to elicit software features. Concerning the context of CMI, Obada-Obieh and Somayaji (2017) have identified trust mechanisms for the three stages of online dating applications, which are helpful for users to better assess the trustworthiness of other users and evaluate their own safety. Such trust mechanisms are, for example, safety guidelines shared by the CMI service provider that give users safety advice on how to behave in social interactions with other online daters. Currently not all CMI services offer these guidelines or those are not well presented so that online dating users may not notice and benefit from them. Another proposed trust mechanism for the during stage is to provide evolving communication steps for a better verification of CMI users. Those may involve first text-based and then voice-based communication, which might end up in time-limited video conversations. However, these proposals are not based on a structured development method or connected to further details valuable for software engineering. Therefore, they may give first creative impulses, but not support requirements engineers in developing relevant trust mechanisms in a structured way on their own.

Regarding such development methods, main work is done by Mohammadi et al. (2015), who introduced the term "trustworthiness-by-design" and proposed general mechanisms for social-technical systems. These mechanisms serve as an extension of existing software development methods by including procedures for systematically achieving trustworthy software. Striving for the same objective, Di Cerbo et al., (2015) suggest considering so-called trustworthiness certificates in order to measure and document trustworthiness-related properties of software during its development. By these trustworthiness certificates, the relation of trustworthiness to the information system can be controlled in every phase of the software life-cycle process.

However, both works aim to build trustworthy software in the sense that the system really performs as it promises. For that reason, they relate their methodological proposals to the concept of trust. In contrast, our proposed software development method considers the concept of trust in order to build software that supports end-users in evaluating i) whether users will act as expected and ii) whether offline encounters are safe. Our method focuses on interpersonal trust that is mediated by the system. This kind of trust differs in its nature and accompanying issues compared to trust in a system. Our method is especially developed for CMI services and focuses on users' mutual trustworthiness assessment.

3 BACKGROUND

Our method presented in this work aims to give requirements engineers a step-wise guideline how to build CMI applications that respect trust in their design. Therefore, we first give an overview of trust and trustworthiness in Section 3.1. In Section 3.2, the trustworthiness framework for CMI (Borchert et al., 2020) places trust in the context of CMI services. Its elements are incorporated into our development method. Furthermore, our method extends the method for analysing and modelling trustworthiness requirements by Mohammadi and Heisel (2016a, 2016b), which is presented in Section 3.3. By referring to the trustworthiness framework for CMI in our method and building on the method of Mohammadi and Heisel, we provide a development approach that is tailored for CMI services. Moreover, we briefly present the online dating service *Plenty of Fish* (POF) in Section 3.4, since we refer to the application in our illustrative example.

3.1 Trust and Trustworthiness

Research has identified various characteristics as key elements of trust. On the one hand, trust can be described as a trustor's (subject that trusts) acceptance of and exposure to vulnerability due to certain risks and uncertainties linked to an interaction

process with a trustee (subject/object to be trusted) (Mayer et al., 1995). On the other hand, trust can be defined as positive expectations a trustor has in the trustee's intentions or behaviour (Möllering, 2005). Those expectations are related to the belief in the trustee being good and honest towards the trustor though having the ability to betray (Barber, 1983).

Trust comes into existence based on certain trustworthiness cues that the trustor perceives from the trustee. These cues vary depending on the context. Trustworthiness cues are then assessed by the trustor so that she can decide whether the trustee is trustworthy and whether the outcome of an interaction is fruitful (Beldad et al., 2010).

3.2 The Trustworthiness Framework for CMI

The trustworthiness framework for CMI (Borchert et al., 2020) places trust in the context of CMI. It represents the relation of trust, cues for assessing trustworthiness and the CMI information system. Therefore, the framework is considered within the method presented in this work as it is valuable for supporting CMI users in their trustworthiness assessment concerning other users.

The trustworthiness framework considers three types of trust that are involved with CMI use: i) system trust, ii) brand trust and iii) computer-mediated interpersonal trust. They originate from different disciplines, namely computer science, sociology, social psychology and business psychology. Taking the user-perspective in the context of the framework, the trustor is an individual CMI end-user, while the trustee differs regarding the type of trust. In the case of *system trust*, the trustee is an impersonal structure (Luhmann, 2018) as for example an information system (Keymolen, 2016), which is the CMI system here. In the case of *brand trust*, the service provider – meaning the organization that makes the information system available - can be regarded as trustee (cf. Ha and Perks, 2005; Thaichon et al., 2013). Finally, *computer-mediated interpersonal trust* describes interpersonal trust (Rotter, 1980) established via information systems. It denotes trust in the person of interest with whom the user interacts via the CMI system. Since the user assesses whether to trust or not to trust the other user based on certain cues presented by the information system, their trust relationship is mediated by the system itself. Borchert et al. (2020) assume that especially computer-mediated interpersonal trust develops during the stages of CMI. This is because user interactions on CMI portray the development of interpersonal relationships from the beginning to oftentimes the end. During interpersonal relationships, trust in each other is a dynamic variable that can strongly vary (Lewicki and Wiethoff, 2000). In comparison, system trust and brand trust are assumed to be relatively stable during the stages. They are more relevant for starting with and further using the CMI application (Borchert et al., 2020).

Another construct of the trustworthiness framework for CMI are the so-called *trustworthiness facets*. Trustworthiness facets represent cues for assessing trustworthiness originating from the disciplines mentioned before. Within these disciplines, trustworthiness cues are related to the different kinds of trustees and differ in their terminology.

In the field of computer science, these cues are called *trustworthiness attributes* and relate to system trust (Mohammadi et al., 2013). Examples for trustworthiness attributes are privacy, security, usability or data-related quality. Originating from social psychology, *factors of trustworthiness* are linked to interpersonal trust and, thus, considered for computer-mediated interpersonal trust. Factors of trustworthiness are benevolence, integrity, ability and predictability (Mayer et al., 1995). In the context of business psychology and sociology, cues like reputation, performance, benevolence or intentionality have been associated with brand trust (Sztompka, 1999; Büttner and Göritz, 2009). Originating from different works, they are not represented by a specific term like the other cues before. For distinction reasons, Borchert et al. (2020) have called them *trustworthiness characteristics*. Some facets appear in different disciplines (e.g. benevolence), but still have a similar meaning. Others have a different terminology but are highly related with each other regarding their definition (e.g. ability and performance). Therefore, it is conceivable that facets may relate to types of trust that they have not been considered for originally.

The trustworthiness framework for CMI proposes to address trustworthiness facets by CMI software features. As a conclusion, the user is supported in assessing the trustees. This is assumed to reduce risks associated with CMI use.

Figure 1 gives an overview of the trustworthiness framework for CMI by using the UML notation (OMG, 2003). Here, the relation of trust, trustworthiness facets and the CMI system become visually apparent. The framework says that a software feature, which is part of a CMI system, shall address trustworthiness facets which in turn affect trust. System trust, brand trust and computer-mediated

interpersonal trust are child classes of trust and, thus, specified as kinds of trust. The same is applicable for factors of trustworthiness, trustworthiness characteristics and trustworthiness attributes concerning trustworthiness facets.

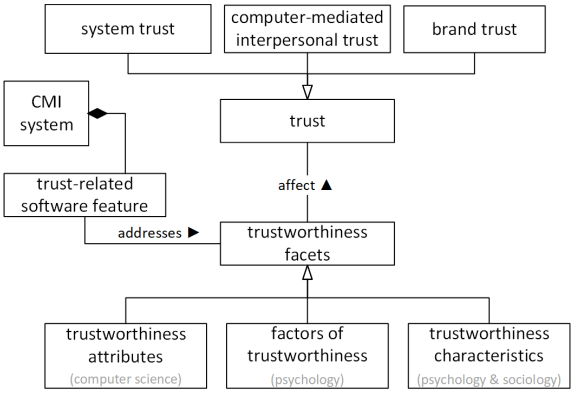

Figure 1: The trustworthiness framework for CMI (Borchert et al., 2020).

3.3 Method for Systematic Analysis of Trustworthiness Requirements by Mohammadi and Heisel

The method for systematic analysis of trustworthiness requirements by Mohammadi and Heisel (2016a, b) serves as basis for the method presented in this paper. It describes a top-down approach for requirements engineers whose objective is to achieve trustworthiness in information systems. An overview of the method is given by the grey boxes in Figure 2.

The first step of the method of Mohammadi and Heisel (2016a, b) is to obtain trust concerns of stakeholders that are involved with the software-to-be. Trust concerns describe their uncertainness of whether an outcome of a specific issue is as expected. Identifying trust concerns is valuable for gaining further understanding about the stakeholders themselves, their intentions and the context. Based on the identified trust concerns, the second step is to derive trustworthiness goals for the software. Trustworthiness goals describe the objectives stakeholders have for the given context and that are trust-related (Mohammadi et al., 2015). Those are then addressed by trustworthiness requirements which determine what capabilities or conditions need to be considered within the system (IEEE Standard Glossary of Software Engineering Terminology, 1990). The last step is to relate the requirements to trustworthiness properties, which realize the requirements in the business process for software development. Trustworthiness properties describe capabilities or qualities the system must meet to influence trust in a positive way (Mohammadi and Heisel, 2016a, b). All four steps mutually depend each other and can be seen as an iterative process.

The method of Mohammadi and Heisel (2016a, b) strongly relates to the i* goal modelling notation (Yu, 1997) and Business Process Model and Notation (BPMN) (Stroppi et al., 2011) on a fine granular model-based level. Goal models are used to map trustworthiness goals of organizations and other stakeholders to then relate them to trustworthiness requirements. They are tailored to the application context and valuable for obtaining rationales for the software development. Business process models visualize activities of business processes as well as their in- and output in a temporal order. In this context, they are useful for embedding trustworthiness requirements within the business process for developing software. For that reason, trustworthiness properties are included as elements within BPMN for directly addressing trustworthiness in the software development process. In addition to goal and business process modelling, Mohammadi and Heisel (2016b) propose pattern-based approaches for realizing the steps of their method.

3.4 The Online Dating Application "Plenty of Fish"

Plenty of Fish (POF) is an online dating application that has users in various countries like the US, Sweden or Germany. POF reported that it had over 4 million active daily users (datingsitesreviews.com, 2017).

In the stage *before* users are connected and interacting with each other, they can edit their profile by adding pictures or disclosing information like demographics, appearance, race, religion, interests or consumption behaviour regarding alcohol or drugs. In addition, users are able to view partner suggestions based on the matching of a POF personality questionnaire, generally browse through pictures and profiles of POF users or look out for other users based on the search criteria age, distance or online activity. In order to begin with the stage *during* connection and interaction, users can show their interest in a profile by signalizing that they would like to interact with the other user. Another option is to directly start exchanging messages. After some messages are exchanged, POF unlocks new communication features like exchanging pictures, voice messages or calls. The last stage, which is *after* a connection or interaction, can be reached by blocking and, thereby, ending the connection to another user so that an

interaction is not possible anymore or simply end the interaction by not exchanging messages anymore. In addition to the basic activities of POF, users can purchase an updated version, which allows them to get more insights about user profiles, whether a sent message has been read, who and when someone has viewed the own profile and to be more often proposed to others.

Examining trust mechanisms of online dating applications, Obada-Obieh and Somayaji (2017) classified POF as one of those services that do not check the authenticity of user identities. Therefore, previous traits or records that could jeopardize users' safety cannot be identified.

4 METHOD FOR ELICITING TRUST-RELATED SOFTWARE FEATURES FOR CMI

In this section, the method for eliciting trust-related software features for CMI is introduced. It follows a top-down approach that extends the method of Mohammadi and Heisel (2016a, b) by elements of the trustworthiness framework (see Figure 2). Unlike the method of Mohammadi and Heisel (2016a, b), its objective is not to build trustworthy software, but CMI software that supports trustworthiness assessments of its users regarding parties to interact with. Moreover, the method not only provides an approach which leverages existing CMI functionalities, but also provides guidelines for developing new software features that address users' safety. For that reason, the method mainly refers to computer-mediated interpersonal trust even though it is not precluded that system trust and brand trust might also be affected by resulting software features. Figure 2 gives an overview of the method for eliciting trust-related software features for CMI. It depicts the original method of Mohammadi and Heisel (2016a, b) by the grey boxes and shows the extensions via the green ones. The further one advances in the method, the more concrete the constructs of each step get for software development, which correlates with the dependency on implementation (x- and y-axis). Our method consists of the following steps that succeed each other, but may also affect former steps so that the approach is an iterative process:

1) Identifying trust concerns and workarounds.
2) Deriving trustworthiness goals and trustworthiness facets.
3) Determining trustworthiness requirements.
4) Inferring trust-related software features.
5) Establishing a collection of trust-related software features.

In the next subsections, each step will be further defined and explained. Examples for each step can be found in the subsections of Section 4.5. There, the conceptual method is exemplarily applied to the online dating application POF.

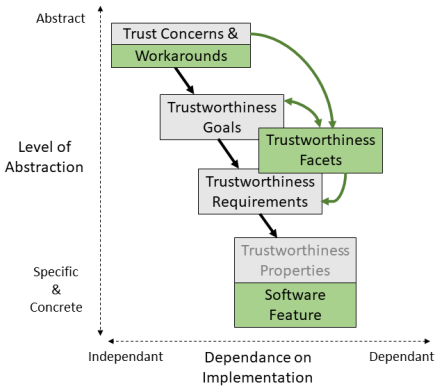

Figure 2: Overview of the method for eliciting trust-related software features in CMI. The grey boxes represent the method of Mohammadi and Heisel (2016a, b) (Section 3.3). The green boxes show the extension of the model of Mohammadi and Heisel.

4.1 Step 1: Identifying Trust Concerns and Workarounds

In order to design user-centred software, Marcelino-Jesus et al. (2014) recommend software engineers to consider knowledge, concerns and behaviour of the system's end-users. Since the method aims to reduce risks associated with CMI use by considering users' trustworthiness assessment of other users, we are especially interested in the trust concerns they have regarding those. Trust concerns are issues in a specific area of application that involve uncertainness whether the outcome of the issue is as expected (Kipnis, 1996). In addition, trust concerns convey a lack of trust in the situation or the trustee. Trust concerns are assumed to be crucial for the interaction decision with involved parties of CMI services (cf. Rotter, 1980). However, CMI platforms do not always meet the users' trust concerns and lead them to the application of alternative behavioural strategies (Obada-Obieh et al., 2017). For that reason, we additionally consider *workarounds* in our approach, because they are relevant for deriving software features that support users in addressing their trust concerns or performing their behavioural strategies directly within the application.

There are different possibilities how trust concerns and workarounds can be conducted.

Mohammadi and Heisel (2016b) introduced the pattern for identification of trust concerns - though it does not take workarounds into account. Other possibilities are to ask experts of the application area or the user target group. Regarding the user target group, three kinds of people can be asked:

1) Individuals, who are active in the application field offline, but are not using any related online service.
2) Individuals, who are active in the application field by using related online services from other service providers and not the one to be improved/developed.
3) Individuals, who already use an existing version of the system.

Engineers should choose the respondents depending on the status of the system to be developed and the objective they pursue. If only a concept of the service application exists, then individuals of type one or experts of the adequate offline activity might be relevant. Their offline experiences might give impulses to design the information system. If a software version already exists, then individuals of type two or three might be valuable to receive specific feedback.

4.2 Step 2: Deriving Trustworthiness Goals and Trustworthiness Facets

For the method presented in this work, we include both – the *trustworthiness goals* from the method of Mohammadi and Heisel (2016a, b) and the *trustworthiness facets* from the trustworthiness framework for CMI (Borchert et al., 2020). Trustworthiness goals correspond to trust-related objectives that the various stakeholders intend to achieve in the given context (Mohammadi et al, 2015). Like in the method of Mohammadi and Heisel (2016a, b), we intend to derive them from trust concerns. Trustworthiness goals should be pursued by the software to be developed to satisfy end-user's trust-related objectives, which in turn have an impact on the overall satisfaction of the application (Mohammadi and Heisel, 2016a, b).

In contrast to trustworthiness goals, trustworthiness facets describe cues that have been identified by literature as important for end-users to assess the trustee's trustworthiness (Borchert et al., 2020). Since facets are a basis for the emergence of trust, we see a relation to trust concerns and workarounds. Trust concerns and workarounds refer to a lack of trust that would not exist if the user had perceived facets of trustworthiness. Obtaining knowledge about the facets is important to later respect and include them in the system design. It is likely that trustworthiness facets differ regarding the diverse stages of CMI, since the requirements for each stage differ, too (Obada-Obieh and Somayaji, 2017). We assume that addressing as many trustworthiness facets of those that have been identified as important for end-user over the different stages of CMI increases the quality of users' trustworthiness assessment.

Both trustworthiness goals and facets can be derived from trust concerns and workarounds and provide an objective and a benchmark for how to overcome the concerns. Therefore, we assume a relation between trustworthiness goals and facets. It is conceivable that a goal may relate to several facets. Since goals and facets are still on an abstract level (see Figure 2), we conclude that the relation of trustworthiness goals and facets can be regarded detached from specific trust concerns and workarounds but rather as a general relation valid for the application area. Therefore, we propose to establish a *collection of trustworthiness goals and facets* for a specific application field, as for example online dating. Based on the collection, requirements engineers can infer what trustworthiness goals and facets a software feature needs to target.

4.3 Step 3: Determining Trustworthiness Requirements

Trustworthiness requirements are a subtype of software requirements. Like software requirements, they can be defined as a condition or capability that i) is "needed by a user to solve a problem or achieve an objective" or that ii) "must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents" (IEEE Standard Glossary of Software Engineering Terminology, 1990). However, trustworthiness requirements target the trust issue and are specifically characterized by addressing end-users' trust concerns (Mohammadi and Heisel, 2016c). Therefore, they are valuable for purposefully developing trust-related software features by determining concrete configurations for service-based systems.

In the method of Mohammadi and Heisel (2016b), trustworthiness requirements are a further refinement of the previously identified goals. Similar to their approach, we aim to determine trustworthiness requirements from the collection of trustworthiness goals and facets. By using the collection, trustworthiness requirements address a goal and also

consider the manner how the goal is achieved – namely by respecting the facets. This means that a trustworthiness goal is met by at least one trustworthiness requirement that in turn addresses at least one of the trustworthiness facets associated with the goal.

4.4 Step 4: Inferring Trust-related Software Features

In the last step of this method, trust-related software features are inferred from trustworthiness requirements from step three. Software features are a very abstract concept, for which a multitude of definitions exist (Berger et al., 2015). Common definitions describe features as "a logical unit of behaviour specified by a set of functional and [non-functional] requirements" (Bosch, 2000, p.194) or "a feature is also a distinguishable characteristic of a concept (e.g. system, component, etc.) that is relevant to some stakeholder of the concept" (Robak et al., 2002, p.288). They can be seen as reusable solutions within a software for a specific problem corresponding to, for example, user-interface requirements, certain application logics or tasks on an infrastructural level (Berger et al., 2015). In the case of our method, we speak of *trust-related* software features, because we set them in the context of trust concerns end-users have. Trust-related software features are particularly valuable to CMI services because they relate to trustworthiness facets that are important to help users assess the trustworthiness of other end users and the safety of interaction.

For deriving and developing trust-related software features, identified workarounds from step one can serve as creative support for the practitioner of this method. By keeping workarounds in mind, one can make sure to include software-features in the system that are not yet available but required by the users. It is up to the practitioner's expertise or creativity how trustworthiness requirements can be realized by trust-related software features. Another option is to consult experts or take a look at existing solutions to adapt them to CMI services.

With the trust-related software features, step four deviates from the method of Mohammadi and Heisel (2016b), because they replace trustworthiness properties (Figure 2, greyed out). Both trust-related software features and trustworthiness properties are concrete and implementable (see Figure 2, axis). However, trustworthiness properties have a very close connection to BPMN, which is included in this method. Moreover, they do not address trustworthiness facets, which is crucial for the development of CMI services.

4.5 Step 5: Establishing a Collection of Trust-related Software Features

In order to support reusable solutions for specific CMI applications, we propose to establish a *collection of trust-related software features* that contains a solution portfolio of implementable trust-related software features. It builds upon the collection of trustworthiness goals and facets mentioned in Section 4.2. The collection serves as an overview of identified constructs of the whole method, namely trust concerns, workarounds, trustworthiness goals, facets, requirements, software features and the CMI stage the features are relevant for. A feature is linked to a trustworthiness requirement and the associated trustworthiness goal and facets. It does not need to address all of the associated facets, but at least one. During the development process and by building the collection, additional facets that do not yet refer to the discussed trust concern, can by identified as relevant for the features. This shows the iterative process of the method and support enhancing system design.

The collection serves as a documentation of the method and facilitates a structured detection of i) trustworthiness facets that are not yet included in the system, ii) the identification of requirements and iii) the appropriate software features. The objective is to collect a multitude of software features over time so that in the end, every trust concern and facet is covered. We assume that this maximises the support that can be provided for the user's trustworthiness assessment.

4.6 Example: Applying the Method for Eliciting Trust-related Software Features to the Online Dating Application *Plenty of Fish*

POF is an online dating app, where end-users are mainly responsible for their own security and safety (Quiroz, 2013). By applying the here presented method, POF users could be more supported by offering trust-related software features. This example illustrates the method step-by-step concerning a specific use case. In the case of another instance, the explicit constructs of the method can be completely different. In the end, Section 4.6.6 shows an exemplary collection of trust-related software features for POF that documents the results of the method.

4.6.1 Example: Trust Concerns and Workarounds

The first step of the method is to identify trust concerns and workarounds of POF. As far as we know, no explicit research has been done in this direction for POF. For this example, we therefore rely on general trust concerns concerning online dating.

Online dating users have stated that they are worried whether profiles are fake or not. In order to check the authenticity of profiles, they employ the workaround of looking for the person concerned on other social network sites (Obada-Obieh et al., 2017). We assume that this especially occurs in the stages *before* and *during* a connection/match of two end-users, when end-user decides to start or continue an interaction.

4.6.2 Example: Trustworthiness Goals

Based on trust concerns and workarounds, trustworthiness goals can be determined. The goal of end-users in this context is to check the *authenticity* of other users. Authenticity means that a presented profile corresponds to a true identity. A true identity is not conform with the misrepresentation of identifying personal information like name, age, ethnicity, gender, marital status or job, for example (Leppänen et al., 2015). Authenticity precludes identity theft or social bots (Douceur, 2002; Jin et al., 2011). Currently, POF does not have any mechanisms for verifying user authenticity (Obada-Obieh and Somayaji, 2017).

4.6.3 Example: Trustworthiness Facets

In order to check the authenticity of another user, end-users need cues like trustworthiness facets for assessment. *Before* interacting with someone, users tend to examine online dating profiles for further information (Obada-Obieh et al., 2017). The more detailed information is provided in a profile, the better the trustworthiness assessment. Therefore, users may look out for *data-related quality*, which is a facet that describes the way information is provided (Mohammadi et al., 2013). *During* the interaction, facets like *honesty* and *performance* could be relevant for checking whether a profile is fake or not. While honesty means that users say the truth (Xia, 2013), performance displays the actual behaviour presented by the interaction partner (Sztompka, 1999).

The trustworthiness facets data-related quality, honesty and performance can be linked to the trustworthiness goal of checking authenticity. These relations are detached from the POF example. This illustrates to the description of the *collection of trustworthiness goals and facets* (Section 4.2) as a general overview of the relationship of goals and facets, which is valid for the application area online dating applications.

4.6.4 Example: Trustworthiness Requirements

Trustworthiness requirements (TR) describe what condition or capability POF needs to include, which must correspond to the trustworthiness goal and relate to at least one facet.

In order to satisfy data-related quality, POF needs to provide information about users that are deemed interesting or useful. Therefore, a requirement is to obtain such information, which can be done by asking users for self-disclosure (TR1).

Moreover, it is valuable for users to know whether self-disclosed information of a profile corresponds to the truth and represents a user's identity. For that, POF requires to prove and notify users about this circumstance (TR2, TR3). In doing so, POF addresses the facet honesty. In the case of honesty and performance, both facets can be addressed, if POF proves and notifies, or enables users to prove, whether disclosed information of a user matches the behaviour she shows (TR4, TR5, TR6).

4.6.5 Example: Trust-related Software Features

Having a look at the trustworthiness requirements, trust-related software features need to formulate how these can be realized in a concrete way. They shall address related trustworthiness facets and be assigned to a CMI stage. An overview of the identified trust-related software features is given in Table 1.

Requirement TR1 can be put into practice by offering users empty text input fields for information that they can include in their profile to motivate self-disclosure (SF1). This feature is linked to data-related quality and is relevant for the before stage, when users create their profile.

Another possibility to realize TR1 is to trigger self-disclosure behaviour by unlocking online dating functionalities (e.g. providing access to more information of other users or allowing to exchange messages with other users), if the profile is mostly completed (SF2). This software feature is provided by the German online dating website Parship, for example. Again, this feature is relevant for the before stage and refers to data-related quality.

Table 1: Example for the collection of trust-related software features concerning the trust concern whether online dating profiles are fake or not. Trust concern, workaround and trustworthiness goals are omitted in this table due to space constraints, but briefly summarized in Section 4.6.6.

Trustworthiness Requirement (TR)	Trust-Related Software Feature	Trustworthiness facets	CMI stage
TR 1: Asking users to disclose more information.	• SF1: Including empty text input fields to motivate self-disclosure	-Data-related quality	before
	• SF2: Unlock online dating functionalities (e.g. accessing more profile information of others) for completed profiles.	-Data-related quality	before
Requirement 2: Proving whether profiles represent a true identity.	• SF3: Asking users to upload a photograph of their ID, which is then manually checked by the service.	-Honesty	before
Requirement 3: Notifying users about verification of profiles.	• SF4: Graphical icon that classifies a profile as being verified. (relates to SF3)	-Data-related quality, Honesty	
Requirement 4: Proving whether disclosed information matches shown behaviour.	• SF5: Algorithm that compares disclosed information of profile and within communication.	-Honesty, Performance	during
	• SF6: Algorithm to prove whether user does not comply to "terms of use agreement" (e.g. identifying strong language as an indicator for bullying)	-Honesty, Performance	during
Requirement 5: Notifying users about mismatch of disclosed information and shown behaviour.	• SF7: Warning message when there is a mismatch about information disclosed during communication and within profile (relates to SF5).	-Data-related quality, Honesty, Performance	during
	• SF8: Warning message that informs users about own misbehaviour and possible consequences. (relates to SF6)	-Performance	during
	• SF9: Message to inform users affected by another user's misbehaviour. Comforting him/her and showing coping strategies (e.g. blocking user, contact for finding help) (relates to SF6)	-Benevolence, Performance	during
Requirement 6: Enabling users to check users' authenticity.	• SF10: Option to link online dating profile with other social media accounts (e.g. Instagram, Spotify) so that other users have access to it	-Honesty, Performance	before, during

Requirement TR2 can be realized by proving the user's ID card. POF could ask users to photograph and upload it (SF3). After the ID is manually checked by POF employees, profiles could receive a graphical icon notifying users that the profile is verified (SF4, referring to TR3). This feature is used by sharing economy platforms like Airbnb. Online dating users also have stated interest in this feature (Obada-Obieh et al., 2017). It is relevant before the interaction starts. By agreeing on this feature, users can prove their honesty. In addition, the graphical icon for verification is also linked to data-related quality, because it provides users with the additional information about verification.

For realizing TR4, POF could check whether information disclosed in messages during communication correspond to those that have been disclosed in the profile using an algorithm (SF5). If the algorithm finds a mismatch, POF could display a warning message directly after the behaviour has been shown (SF7 referring to TR5). This relates to the facets honesty and performance. Moreover, the notification is again an indication for data-related quality by the system. This feature is relevant for the stage during interaction.

A similar comparison of honesty and performance can be done by pointing out user behaviour that is incongruent to the "terms of use agreement", which every POF user has to accept before usage (SF6

referring to TR4). There, for example, users have agreed to not "harass, bully, stalk, intimidate, assault, defame, harm or otherwise mistreat any person". If POF detects strong language within messages that refers to such a behaviour, the person showing this behaviour can be admonished by referring to possible consequences if the behaviour is shown again (e.g. banned from community) (SF8 referring to TR5). This feature would refer to the users' performance and is relevant for the stage during interaction. Moreover, the affected user could receive a message of POF, which comforts him/her and offers possibilities how to cope with it (e.g. blocking or reporting) (SF9 referring to TR5). Such a feature could help user's in their well-being and safety. This feature gives feedback to the performance of other users. In addition, it relates to the facet benevolence shown by the service provider or system towards its users. At this point, requirements engineers should think of including benevolence as a facet for this use case. This triggers the development process in looking for more features how this facet can be satisfied, which improves system design. Features SF6, SF8 and SF9 are relevant for the stage during interaction.

Requirement TR6 is about users checking the match of disclosed information and shown behaviour on their own. This can be accomplished by realizing the identified workaround within the POF application. Therefore, POF could enable users to link their profile with other social media accounts, such as Instagram or Spotify, so that connected users have access to it. By this feature, users can represent their honesty and performance. Currently, POF is one of the few online dating applications that does not offer this feature (Obada-Obieh and Somayaji, 2017). This feature is relevant for the before and during stages.

4.6.6 Example: Collection of Trust-related Software Features

The *collection of trust-related software features* summarizes the results of the whole method. Table 1 shows how the collection can be built. In order to avoid a too large table, trust concerns, workarounds and trustworthiness goals are omitted here. The trust concern for this exemplary collection describes that online daters fear fake profile that do not represent a true identity. Workarounds imply the check of other users on additional social network sites. The trustworthiness goal for POF is to check the authenticity of user profiles.

5 DISCUSSION AND FUTURE WORK

This work introduces a conceptual method for the elicitation of trust-related software features in CMI services. Our method aims at supporting requirements engineers in both, the development of new features and the improvement of existing CMI services. The objective is to reduce risks associated with CMI use by supporting end-users in assessing the trustworthiness of other users and their safety during an interaction. Therefore, the method considers end-users' trust concerns, their workarounds, trustworthiness goals and trustworthiness facets to establish trustworthiness requirements, which are the basis for developing trust-related software features.

Properties of CMI services are the introduction and interaction with unfamiliar users online that might lead to offline encounters. Based on that, CMI usage can be divided into the stages *before*, *during* and *after* a connection/match of two end-users. Obada-Obieh and Somayaji (2017) have detected different requirements concerning the stages of CMI services (in particular online dating) and a need for trust mechanisms in online dating applications. Therefore, they propose ideas of how trust mechanisms can look like for each stage. Our method complies to their findings and provides a structural approach for developing such trust mechanisms, which we call trust-related software features. By applying the method, requirements engineers are encouraged to be diligent while formulating requirements and linked facets for the CMI service. This leads to a concrete description of software features and an enhanced software design.

Having a detailed look at trust and trustworthiness, trustworthiness facets are key elements in our method for reducing risks in CMI. They are the basis for deciding whether to trust someone and whether an interaction is safe. Since trust concerns for online dating are assumed to differ concerning the various stages of CMI (cf. Gibbs et al., 2011), this also applies for trustworthiness facets. With the help of our method, software features can be tailored to the relevant facets for the individual stages leading to improved CMI use.

Overall, resulting trustworthiness requirements and software features follow one specific goal and meets one specific trust concern. However, they may get into conflict with other trustworthiness goals across the system. For instance, TR1 of our example could jeopardize a trustworthiness goal such as privacy because it demands the disclosure of personal information from the users. Therefore, future research

should investigate ways to resolve conflicts between requirements/features while maximizing the users' benefit.

Moreover, this work describes a conceptual method whose stepwise realization is not further determined, yet. Mohammadi and Heisel (2016a) have specified patterns for the identification of trust concerns and for the specification of trustworthiness requirements, in order to provide requirements engineers with clear guidelines within their method. Other possibilities are qualitative approaches like interviews of user target groups or experts (Hubbard et al., 2000). In future work, we will further define how the method can exactly be applied step by step.

A limitation of this work is that it is based on former research and theoretical conclusions. However, it is important to evaluate the method by using it for a concrete development of a CMI application. In addition, relationships proposed by the trustworthiness framework for CMI (Borchert et al., 2020) can also be tested. It is important to prove, whether the proposed software features really have an effect on system trust, brand trust or computer-mediated interpersonal trust.

Moreover, it is a challenge to evaluate whether software developed with this method really reduces risks and supports safe offline encounters. Future research should survey end-users and developers about their perception in this point or conduct long-term studies to observe whether the rate of unwanted incidents is reduced.

6 CONCLUSION

This work proposes an approach for requirements engineers to build CMI services that support end-users in their mutual trustworthiness assessment in the CMI stages *before*, *during* and *after* they are connected with each other. The objective is to reduce risks associated with CMI use to increase the rate of safe offline encounters. In order to accomplish this, the method considers end-users' trust concerns and counter strategies to infer trustworthiness goals and trustworthiness facets. By considering those, requirements for CMI services can be obtained that most likely have an impact on computer-mediated interpersonal trust-relationships. Based on that, trust-related software features can be derived that support a safer use in each CMI stage and increases user satisfaction with the CMI service. Since this work presents a conceptual method for eliciting trust-related software features for CMI, future work tackles the refinement of the method by detailed procedures for each step.

ACKNOWLEDGEMENTS

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group "User-Centred Social Media".

REFERENCES

Barber, B. 1983. *The logic and limits of trust* (Vol. 96). New Brunswick, NJ: Rutgers University Press.

Beldad, A., De Jong, M., Steehouder, M. 2010. How shall I trust the faceless and the intangible? A literature review on the antecedents of online trust. *Computers in human behavior*, *26*(5), 857-869.

Berger, T., Lettner, D., Rubin, J., Grünbacher, P., Silva, A., Becker, M., Czarnecki, K. 2015. What is a feature?: a qualitative study of features in industrial software product lines. In *Proceedings of the 19th International Conference on Software Product Line*, 16-25. ACM.

Borchert, A., Diáz Ferreyra, N. E., Heisel, M. 2020. Submitted for publication.

Bosch, J. 2000. *Design and use of software architectures: adopting and evolving a product-line approach*. Pearson Education.

Büttner, O. B., Göritz, A. S. 2008. Perceived trustworthiness of online shops. *Journal of Consumer Behaviour: An International Research Review*, *7*(1), 35-50.

Couch, D., Liamputtong, P., Pitts, M. 2012. What are the real and perceived risks and dangers of online dating? Perspectives from online daters: Health risks in the media. *Health, Risk & Society*, *14*(7-8), 697-714.

Datingsitesreviews.com, 2017. Plenty of Fish debuts new conversation feature and redesigned app. Retrieved December 2019, from https://www.datingsitesreviews.com/users.php?mode=profile&uid=4448.

Di Cerbo, F., Mohammadi, N. G., Paulus, S. 2015. Evidence-based trustworthiness of internet-based services through controlled software development. In *Cyber Security and Privacy Forum*, 91-102, Springer, Cham.

Douceur, J. R. 2002. The sybil attack. In *International workshop on peer-to-peer systems*, 251-260. Springer, Berlin, Heidelberg.

Gibbs, J. L., Ellison, N. B., Lai, C. H. 2011. First comes love, then comes Google: An investigation of uncertainty reduction strategies and self-disclosure in online dating. *Communication Research*, *38*(1), 70-100.

Ha, H. Y., Perks, H. 2005. Effects of consumer perceptions of brand experience on the web: Brand familiarity, satisfaction and brand trust. *Journal of Consumer Behaviour: An International Research Review*, *4*(6), 438-452.

Hubbard, R., Schroeder, C. N., Mead, N. R. 2000. An assessment of the relative efficiency of a facilitator-driven requirements collection process with respect to the conventional interview method. In *Proceedings Fourth International Conference on Requirements Engineering. ICRE 2000*. 178-186. IEEE.

IEEE Standards Coordinating Committee. 1990. IEEE Standard Glossary of Software Engineering Terminology (IEEE Std 610.12-1990). Los Alamitos. *CA: IEEE Computer Society*, *169*.

Jin, L., Takabi, H., Joshi, J. B. 2011. Towards active detection of identity clone attacks on online social networks. In *Proceedings of the first ACM conference on Data and application security and privacy*, 27-38. ACM.

Keymolen, E. 2016. Trust on the line: a philosophycal exploration of trust in the networked era.

Kipnis, D. 1996. Trust and technology. *Trust in organizations: Frontiers of theory and research*, *39*, 50.

Leppänen, S., Møller, J. S., Nørreby, T. R., Stæhrc, A., Kytölä, S. 2015. Authenticity, normativity and social media. *Discourse, Context and Media*, *8*.

Lewicki, R. J., Wiethoff, C. 2000. Trust, trust development, and trust repair. *The handbook of conflict resolution: Theory and practice*, *1*(1), 86-107.

Luhmann, N. 2018. *Trust and power*. John Wiley & Sons.

Marcelino-Jesus, E., Sarraipa, J., Agostinho, C., Jardim-Goncalves, R. 2014. A Requirements Engineering Methodology for Technological Innovations Assessment. In *ISPE CE*, 577-586.

Mayer, R. C., Davis, J. H., Schoorman, F. D. 1995. An integrative model of organizational trust. *Academy of management review*, *20*(3), 709-734.

Mohammadi, N. G., Paulus, S., Bishr, M., Metzger, A., Koennecke, H., Hartenstein, S., Pohl, K. 2013. An Analysis of Software Quality Attributes and Their Contribution to Trustworthiness. In *CLOSER*, 542-552.

Mohammadi, N. G., Bandyszak, T., Paulus, S., Meland, P. H., Weyer, T., Pohl, K. 2015. Extending Software Development Methodologies to Support Trustworthiness-by-Design. In *CAiSE Forum*, 213-220.

Mohammadi, N. G., Heisel, M. 2016a. Patterns for identification of trust concerns and specification of trustworthiness requirements. In *Proceedings of the 21st European Conference on Pattern Languages of Programs,* 31. ACM.

Mohammadi, N. G., Heisel, M. 2016b. A framework for systematic analysis and modeling of trustworthiness requirements using i* and BPMN. In *International Conference on Trust and Privacy in Digital Business*, 3-18. Springer, Cham.

Mohammadi, N. G., Heisel, M. 2016c. Enhancing business process models with trustworthiness requirements. In *IFIP International Conference on Trust Management*, 33-51. Springer, Cham.

Möllering, G. 2005. The trust/control duality: An integrative perspective on positive expectations of others. *International sociology*, *20*(3), 283-305.

Obada-Obieh, B., Somayaji, A., 2017. Can I believe you?: Establishing Trust in Computer Mediated Introductions. In *Proceedings of the 2017 New Security Paradigms Workshop*, 94-106. ACM.

Obada-Obieh, B., Chiasson, S., Somayaji, A. 2017. "Don't Break My Heart!": User Security Strategies for Online Dating. In *Workshop on Usable Security (USEC)*.

Object Management Group, 2003. UML 2.0 Infrastructure – Final adopted specification. http://www.omg.org/docs/ad/03-09-15.pdf

Quiroz, P. A. (2013). From Finding the Perfect Love Online to Satellite Dating and 'Loving-the-One-You're Near' A Look at Grindr, Skout, Plenty of Fish, Meet Moi, Zoosk and Assisted Serendipity. *Humanity & Society*, *37*(2), 181-185.

Robak, S., Franczyk, B., Politowicz, K. 2002. Extending the UML for modelling variability for system families. *International Journal of Applied Mathematics and Computer Science*, *12*, 285-298.

Rotter, J. B. 1980. Interpersonal trust, trustworthiness, and gullibility. *American psychologist*, *35*(1), 1.

Stroppi, L. J. R., Chiotti, O., Villarreal, P. D. 2011. Extending BPMN 2.0: method and tool support. In *International Workshop on Business Process Modeling Notation*, 59-73. Springer, Berlin, Heidelberg.

Sztompka, P. 1999. *Trust: A sociological theory*. Cambridge University Press.

Thaichon, P., Quach, T. N., Lobo, A. 2013. Marketing communications: Factors influencing brand loyalty of internet service provider. In *meeting of Australian and New Zealand Marketing Academy Conference. Auckland, New Zealand*.

Walther, J. B., Loh, T., Granka, L. 2005. Let me count the ways: The interchange of verbal and nonverbal cues in computer-mediated and face-to-face affinity. *Journal of language and social psychology*, *24*(1), 36-65.

Xia, L. 2013. Effects of companies' responses to consumer criticism in social media. *International Journal of Electronic Commerce*, *17*(4), 73-100.

Yu, E. S. K. 1997. Towards modelling and reasoning support for early-phase requirements engineering. In *Proceedings of ISRE'97: 3rd IEEE International Symposium on Requirements Engineering*, 226-235. IEEE.