Supporting Shared Understanding in Asynchronous Communication Contexts

Lukas Nagel[1][0000−0001−5003−2340], Oliver Karras[2][0000−0001−5336−6899], Seyed Mahdi Amiri[1], and Kurt Schneider[1][0000−0002−7456−8323]

[1] Leibniz Universität Hannover, Software Engineering Group, Germany
{lukas.nagel@inf, kurt.schneider@inf, m.amiri@stud}.uni-hannover.de,
[2] TIB - Leibniz Information Centre for Science and Techology, Hannover, Germany
oliver.karras@tib.eu

Abstract. **[Context and motivation]** The success of software projects depends on developing a system that satisfies the stakeholders' wishes and needs according to their mental models of the intended system. However, stakeholders may have different or misaligned mental models of the same system, resulting in conflicting requirements. For this reason, aligned mental models and thus a shared understanding of the project vision is essential for the success of software projects. **[Question/problem]** While it is already challenging to achieve shared understanding in synchronous contexts, such as meetings, it is even more challenging when only asynchronous contexts, like messaging services, are possible. When multiple stakeholders are involved from different locations and time zones, primarily asynchronous communication occurs. Despite the frequent use of software tools, like *Confluence*, to support asynchronous contexts, their use for the development of a shared understanding has hardly been analyzed. **[Principal ideas/results]** In this paper, we propose five concepts to help stakeholders develop a shared understanding in asynchronous communication contexts. We assess the adaptability of three existing software tools to our concepts, adapt these software tools accordingly, and develop our own prototype that implements all five concepts. In an experiment with 30 participants, we evaluate these four software tools and compare them to a control group that had no support in developing a shared understanding. **[Contribution]** Our results show the suitability of our concepts, as the participants using our concepts were able to achieve a higher level of shared understanding compared to the control group.

Keywords: requirements engineering · shared understanding · asynchronous communication

1 Introduction

A shared understanding of the project vision is paramount to the success of software projects, as its absence can lead to conflicting requirements [31]. Achieving this shared understanding is one of the key challenges in requirements engineering [14]. For this purpose, stakeholders must disclose, discuss, and align their mental models of the intended system to achieve a shared understanding [3]. However, stakeholders are often spread across different locations and time zones [22]. In this case, primarily asynchronous communication occurs, as stakeholders can hardly meet for synchronous in-person or even virtual meetings [13]. One way to achieve a shared understanding in asynchronous communication contexts is to distribute a written specification using standards like ISO/IEC/IEEE 29148:2018 [16]. Nevertheless, reading a written specification can be time-consuming due to its low communication richness and effectiveness [1]. For project visions specifically, a richer and more effective way for achieving a shared understanding is the use of so-called vision videos [21].

Vision videos support the development of a shared understanding, as they provide visual reference points to stimulate active discussions among stakeholders to align their mental models [17]. They are primarily used to support the elicitation, documentation and validation of requirements [21]. Nagel et al. [25] have successfully used vision videos to find misaligned mental models in asynchronous settings. However, simply watching a vision video without the opportunity to discuss its contents complicates the resolving of misalignments [25]. For this reason, stakeholders need suitable support for their discussions to achieve a shared understanding in asynchronous communication contexts.

The goal of this paper, which is based on a master's thesis by Amiri [2], is to *develop suitable concepts to support stakeholders in achieving a shared understanding in asynchronous communication contexts.*

In this paper, we propose five corresponding concepts that are designed to solve issues with asynchronous communication extracted from literature. We combine these concepts with vision videos to investigate whether they support stakeholders in achieving a shared understanding. Three existing software tools for asynchronous communication are assessed regarding their adaptability to our concepts and adapted accordingly. We also develop a prototype that implements all five concepts. In an experiment with 30 participants, we evaluate the four software tools and establish a baseline. Our results show evidence for the suitability of our concepts. All software tools support the achievement of a shared understanding. In particular, participants supported by our adaptation of the messaging service Discord and our developed prototype presented a statistically significantly higher level of shared understanding compared to the control group.

This paper is structured as follows: Section 2 discusses related work. We present our concepts in Section 3 and describe their implementation on existing tools in Section 4. In Section 5, we provide details on our experiment, whose results are reported in Section 6. Section 7 shows threats to validity. Our results are discussed in Section 8 before the paper is concluded in Section 9.

2 Related Work

Several works address achieving a shared understanding among stakeholders in requirements engineering. Glinz and Fricker [14] discuss the role of shared understanding in software engineering and identify enablers and obstacles. They also introduce implicit and explicit shared understanding. One technique to support the achievement of a shared understanding is the use of vision videos presenting the project vision. The term *vision video* has been defined by Schneider et al. [28] as a video *of a software-based system typically showing a problem, an envisioned solution, and its impact, pretending the solution already exists.* Creighton et al. [9] introduced the use of videos to visualize scenarios by presenting workflows that are not yet implemented. Brill et al. [6] expanded on this idea by investigating potential uses of videos in various phases of requirements engineering. The potential use of vision videos on multimedia platforms like YouTube has already been discussed by Schneider and Bertolli [27]. Karras et al. [19] investigated the use of vision videos on social-media platforms for CrowdRE. The videos motivated crowd members to provide feedback.

Another use case of videos in asynchronous settings is e-learning. Skylar [29] investigated the performance of students in synchronous and asynchronous online courses and found both to be effective. Furthermore, Clark [7] found one of the biggest advantages of asynchronous communication to be the opportunity for reflective thought processes in between messages. A work by Dowling and Lewis [11] discusses further disadvantages of both communication types. They mention the time pressure of synchronous meetings, which might lead to important contributions being missed. However, the temporal linearity of asynchronous communication is missing due to the distribution of comments on the same topic. A response could therefore be separated from the original comment, which hampers discussants following a discussion topic.

Braunschweig and Seaman [5] developed a technique to measure the shared understanding achieved by a group of stakeholders using *Pathfinder Networks* (PFNets). To use this technique, stakeholders fill out a spreadsheet with relatedness ratings of concept pairs. These ratings are then used to create graphs called PFNets as introduced by Dearholt and Schvaneveldt [10]. Shortest paths can be calculated by using the relatedness ratings as edge weights. The PFNets of a pair of stakeholders can be compared by determining the similarity of the neighborhoods of individual concept-nodes. Calculating the average of all concept similarities between two PFNets, a *Network Similarity* (NetSim) value for a stakeholder pair can be obtained.

3 Concepts for Supporting Shared Understanding

For the development of concepts supporting the achievement of a shared understanding among stakeholders communicating asynchronously, we collected common issues of asynchronous communication from existing literature. Based on these issues, we brainstormed concepts to minimize the impact of these issues. The concepts introduced in this paper are based on the master's thesis by Amiri [2]. Table 1 presents an overview of the identified issues and our concepts addressing them. In the following, we explain each concept in more detail.

Questions of Understanding: We adopt the concept of *Questions of Understanding* from the related work by Nagel et al. [24]. These questions ensure that

Table 1: Overview of our Concepts and the Issues they look to solve.

Concepts	**Issues**
Questions of Understanding	Differing Domain Knowledge [25,14] Misunderstandings [25,12,11]
Message Frames	Misunderstandings [25,12,11] Missing valuable Ideas [25,11] Sequential Ordering of Messages [11]
Req. Engineers as Facilitators	Missing valuable Ideas [25,11] Free-Riders [32]
Polls	Missing valuable Ideas [25,11] Reaching Final Conclusions [32]
Step-By-Step Design	Coordination of Steps [15,30]

all stakeholders understand the presented content of an artifact correctly and clarify domain-specific terminologies. Differing from prior research, we propose to force stakeholders to answer *Questions of Understanding* before being allowed to take part in a discussion. In this way, we can ensure that all discussion members have a basic understanding of the presented content.

Requirements Engineers as Facilitators: Synchronous meetings are often held under the guidance of a moderator who guides the participants [33,18]. A traditional moderator role cannot be present in asynchronous communication. However, the active and collaborative participation of all stakeholders, that can be motivated by a moderator [33], is still vital for achieving a shared understanding [4]. We therefore propose to have requirements engineers play a facilitating role in asynchronous communication. This can be done by providing some initial questions or reacting to comments made by stakeholders to motivate them to participate even more. However, requirements engineers should remain neutral in discussions so that stakeholders can reach final conclusions on their own.

Message Frames: A logical and sequential ordering of individual sentences is important to enable humans to reach conclusions from conversations [11]. Our concept of *Message Frames* looks to implement this idea on asynchronous communication, where such sequential orderings are hard to follow [11]. *Message Frames* are a filter for incoming information that structures outgoing messages. In asynchronous communication, the order of messages does not necessarily have to follow the order of discussion topics. Stakeholders can start a topic and return to the discussion after other stakeholders have commented with ideas on other topics. When messages regarding the same topic are located in widely different positions in the ordering of messages, it is hard for stakeholders to follow a discussion [11]. This issue is especially prevalent when the number of discussants and messages increase. *Message Frames* summarize comments dealing with the same topic in a logical order. For example, a requirements engineer could summarize all comments regarding the topic of “security” in one *Message Frame*. This makes it easier for stakeholders to finalize their thoughts on any given topic. *Message Frames* can thereby lead to more explicit shared understanding.

Polls: Polling is one possibility to reach definitive conclusions at the end of a discussion [32]. *Polls* can turn implicit shared understanding into explicit shared understanding [14]. We recommend using the *Paraphrasing Method* [14] to create the polling questions. By paraphrasing the comments made by the participants and asking for their feedback before enabling the polls, requirements engineers can ensure that there are no misunderstandings [14]. Additionally, we propose that stakeholders can suggest additional polling questions themselves. This allows them to directly ask their peers about unresolved uncertainties. A potential side benefit of the use of polls is that they can also be used to gather an initial indication of a group's level of shared understanding. Groups of stakeholders giving the same answer to a polling question are likely to have a higher level of shared understanding than other groups giving more diverse answers.

Step-By-Step Design: Another drawback of asynchronous communication is the difficulty of coordinating the stakeholders [15]. Important steps could be performed in different orders, thereby creating a chasm between individual knowledge bases. Providing an explicit process is one way to counteract this phenomenon [15]. Therefore, we propose an enforcement of such a process. At first, our concepts only allow stakeholders to get familiar with the content of the presented artifact. Their next step is to answer *questions of understanding*, thereby ensuring that they have a common knowledge base. Stakeholders are only allowed to contribute to the discussion once they answer all *questions of understanding* correctly. Furthermore, our concepts also include fixed time frames for the existing steps. One task of moderators in synchronous meetings is to lead participants through the phases of the agenda within a given time [33]. We incorporate this aspect by providing fixed time frames for each step of the process. Stakeholders are thereby kept from delaying their participation. Simultaneously, the fixed time frames also provide requirements engineers with a concrete time at which feedback regarding the presented content will be available.

4 Implementation of Concepts

We developed a prototype that implements all five concepts to evaluate their suitability to our goal. We also assessed the adaptability of existing software tools for asynchronous communication, as preexisting familiarity with these tools could reduce the barrier of entry for stakeholders. An important factor in the choice of software tools was their capability to display a vision video. The video must be directly visible in the software tool so that stakeholders do not need to switch between applications to reduce their cognitive load [20]. We conducted a workshop with three participants to discuss different types of software tools and to choose individual representatives for our experiment. In this workshop, a total of 10 different software tools were discussed. Each participant was asked to identify advantages and disadvantages of the tools. Ultimately, we asked the workshop participants to pick three tools they considered to be best suited to the support of stakeholders in the asynchronous achievement of a shared understanding using vision videos. The multimedia platform *YouTube*, the wiki service *Confluence*, and the messaging service *Discord* were selected as the most suitable existing software tools. More information on the workshop can be found in the master thesis by Amiri [2] which this paper is based on. Table 2 presents an overview of the concepts and the manner in which they were implemented for each tool. The following paragraphs present the implementation of the concepts *Questions of Understanding*, *Polls* and *Step-By-Step Design*. Our concepts *Requirements Engineers as Facilitators* and *Message Frames* were not implemented as technical adaptations of the tools, but as tasks of the requirements engineer's role.

Table 2: Overview of the applicability of our concepts to each tool.
Applicabiltiy: ✓fully, ◯ partially, and ✋ only manually
* For YouTube, *Polls* had to be applied using a third party tool.

Concept	**YouTube**	**Confluence**	**Discord**	**Prototype**
Questions of Understanding	✓	✓	✓	✓
Requirements Engineers as Facilitators	✋	✋	✋	✋
Message Frames	✋	✋	✋	✋
Polls	◯*	◯	◯	✓
Step-By-Step Design	◯	◯	◯	✓

YouTube: *YouTube* provides built-in functionality for the presentation of video content. With over 2.1 billion worldwide users[3], most stakeholders should be familiar with the system. *YouTube* offers a comment system which provides functionality to answer previously made comments and to reference other users. *YouTube* also includes a description section in which more context can be given. We used this description section to provide the order of steps and the *Questions of Understanding*. However, there was no way to enforce the *Step-By-Step Design* or to hold *Polls*. While the *Like* and *Dislike* functionality of comments could be used, *YouTube* does not display the exact votes and would be limited to yes or no questions. Third party tools are required for other *Polls* and for the answering of *Questions of Understanding*.

Confluence: *Confluence* includes functionality to organize knowledge on pages and a comment system. Videos can be embedded directly on these pages. We created one page to view the video, one page to answer *Questions of Understanding*, one page for the comment section, and one final page for polling questions. In this way, we partially implement the *Step-By-Step Design*. However, the order of steps could not be enforced. There was also no built-in functionality for *Polls*. Instead, a suite of plugins is available within Atlassian's marketplace.

[3] https://www.statista.com/topics/2019/youtube/#dossierKeyfigures

Discord: *Discord* allows its users to create “Servers” for free. Servers consist of text and voice channels only available to invited users. Voice channels can be joined for conference calls. Text channels offer functionality to write messages, upload files, and embed images. Discord users can reference other messages or other users and pin messages to make them easier findable. Threads can be created that appear as a single message in the original chat history, but can be expanded into a new window with its own set of messages. This allows for a separation of especially important topics. We made use of these threads to implement our concept of *Questions of Understanding* by asking them in a separate thread. For the *Step-By-Step Design*, we pinned a message detailing the order of steps within the text channel. However, we could not enforce the compliance with this order. *Discord* also does not offer built-in functionality for polling. For this reason, we use free plugins that enabled our concept of *Polls*.

Prototype: The existing tools evaluated in this paper offer functionality suited to some of our concepts. However, none of them could be adapted to include all concepts to their full extent. For this reason, we developed a prototype that implements all five concepts. The prototype was implemented as a single page application, a screenshot of which can be found in Fig. 1. The prototype always displays the vision video at the top of the screen (1). Stakeholders can click through the pages of the prototype (2), which represent the *Step-By-Step Design*. Some pages only unlock after performing prior steps. The main area of the prototype displays the selected page's content (3). When providing new comments, stakeholders are required to give a headline to assist requirements engineers in the creation of *Message Frames*.

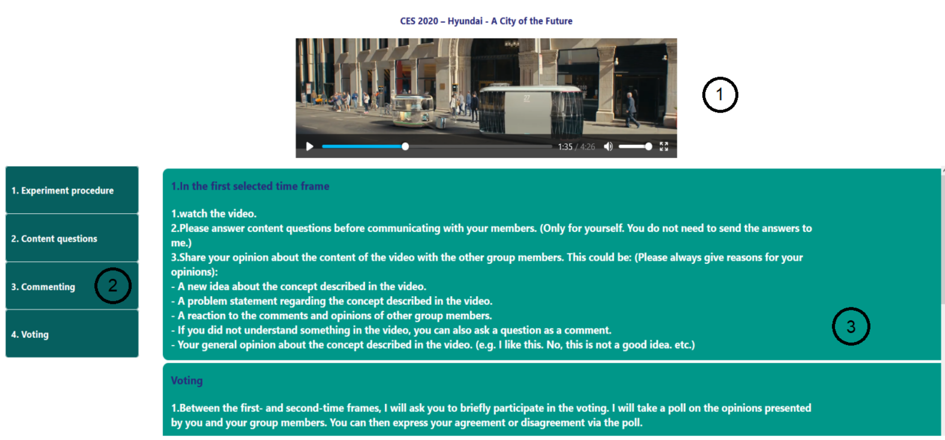


Fig. 1: Screenshot of the prototype presenting the *Experiment Procedure* page.

5 Experiment

A user study was conducted to evaluate our concepts for our research goal (cf. Section 1). Based on this goal, we defined the following two research questions.

RQ1: How suited are the six concepts to the support of stakeholders in achieving a shared understanding in asynchronous communication contexts?
RQ2: Which software tools are best suited to provide this support?

5.1 Experiment Design

Material: Our study utilizes a *vision video*[4] on future mobility produced by Hyundai and published on *YouTube* as the basis for all discussion topics. To enable the use of the four software tools mentioned in Section 4, members of the treatment groups were provided with new e-mail addresses and user accounts. We thereby preserved their privacy and lowered the barrier of entry.

Furthermore, each participant was provided with a link to a spreadsheet that was used to perform the PFNets method lined out in Section 2. Both the link to the spreadsheet and a second link to a questionnaire were distributed at the end of their participation. The questionnaire asked participants about their opinions on the suitability of asynchronous communication for the achievement of a shared understanding and their preference between synchronous and asynchronous communication methods. Another document guiding participants during their participation in the study was also provided. Since none of the participants had previously worked with the PFNets method, this guideline included information on the use of the spreadsheet and an example. The spreadsheet is available on Zenodo [23].

Participant Selection: We performed convenience sampling to recruit the participants for our study. Participation was not mandatory. A total of 30 participants took part in the study. All participants were active university students in Germany. Our only requirement for our participant selection was a functioning computer on which to watch the vision video, answer the questionnaire and fill out the PFNets spreadsheet. We were looking to include potential stakeholders for the topic of future mobility. Therefore, anyone participating in modern traffic is a viable participant.

Experiment Procedure: The study was conducted online over a total of five days, with each group participating on a single day. Participants were assigned to groups based on personal availability. Our only influence on these assignments was limited to the selection of time slots for participants whose availability was suited to multiple groups. The study was performed strictly online due to the Covid-19 pandemic. We performed an experiment session with a control group of 6 participants to establish a baseline. Members of this control group were asked to

[4] https://www.youtube.com/watch?v=J_OBgXalGFU

view the vision video on their own and had no support to discuss with any other group members. They were also explicitly asked to work on the spreadsheets on their own to ensure the validity of their answers. We designed the control group without any means of communication to measure the level of shared understanding that is created by simply watching the same vision video. To the best of our knowledge, no methodology for the achievement of a shared understanding in asynchronous communication contexts exists. Therefore, our study was designed to create a baseline of shared understanding when watching vision videos while also investigating the differences between supporting communication tools.

For members of the treatment groups, the study consisted of two distinct time windows. To ensure a strictly asynchronous setting, no participants were scheduled to take part at the same time. Participants were asked to perform the same set of steps during the two time windows. However, there were some differences in terms of the available functionality as outlined in Section 4.

In the first time window, participants were asked to watch the vision video for the first time before answering six *Questions of Understanding*. Participants were explicitly asked to answer these questions first before proceeding. However, this requirement could only be enforced in the prototype. Lastly, participants were allowed to leave comments and add to existing parts of the discussion. Between the two time windows, the experimenter scanned through the comments and created *Message Frames*. Polling questions were also determined.

The second time window started by providing the *Message Frames* before participants answered the polling questions. For the treatment group supported by YouTube, this was done via telephone. Next, each participant was asked to read the submitted comments and respond to them. After all participants had finished the second time window, they were asked to review the results of the *Polls* before answering the questionnaire and filling in the PFNets spreadsheet.

Data Analysis Procedures: To answer our research questions, we created two sets of hypotheses. Each set is designed to answer one research question. The first set of hypotheses aims at finding differences between each of the four treatment groups and the control group:

> **H1**$_{\text{i.0}}$: There is no difference in the shared understanding of participants between the control group and the treatment group supported by i.
> $$i \in \{YouTube, Confluence, Discord, Prototype\}$$

The second set deals with the differences between the different supporting tools. For example, we look to find a difference between the treatment group communicating via YouTube and the one being supported by the prototype:

> **H2**$_{\text{j.0}}$: There is no difference in the shared understanding of participants between a and b.
> $$j = (a, b) \textit{ with } a, b \in \{YouTube, Confluence, Discord, Prototype\}, a \neq b$$

To find data on which to base a potential rejection of these null hypotheses, we analyzed the PFNets spreadsheets filled out by our participants according to Braunschweig and Seaman [5]. Their technique resulted in network similarity (NetSim) values for all participant pairs. These were then used to calculate average NetSim values for each group and to calculate the statistical significance of differences in the achieved shared understanding between the groups. The statistical significance was determined by first testing for normal distribution using the Shapiro-Wilk test before applying the Mann-Whitney U test or the t-test, depending on the presence of a normal distribution. We also applied the Bonferroni-Holm correction. In addition, we extracted the results of the *Polls* and gathered answered questionnaires. For the *Polls*, we determined which choice was made by the majority of participants, before averaging the number of participants who were part of this majority for each poll performed in the respective treatment group. This resulted in the average size of the majority vote for each group. We analyzed the answers to the questionnaires descriptively.

6 Results

Our study focuses on measurements for the shared understanding within each group of the experiment. Furthermore, we also obtained information on participants' thoughts on the suitability of asynchronous communication contexts and their general opinion on the software tool they were supported by.

NetSim: We measured the shared understanding within the groups of our experiment using the aforementioned PFNets method. The results are available on Zenodo [23]. As our results were normally distributed for all groups, we used the t-test. The results of the Shapiro-Wilk test can be found in Table 3.

Table 3: Results of Shapiro-Wilk tests. Note that the sample size for a group of 6 participants is 15 as we obtained similarity values for each participant pair.

Tool	**W(15)**	**p**	**Normal Distribution?**
Control	0.889	0.067	**Yes**
YouTube	0.785	0.965	**Yes**
Confluence	0.969	0.841	**Yes**
Discord	0.969	0.842	**Yes**
Prototype	0.933	0.302	**Yes**

To test the set Hypotheses H1, we compared the values calculated for the control group with the values measured for each other software tool. We found statistically significant differences between the control group and the treatment groups supported by Discord and our prototype (cf. Table 4).
Hypotheses H2 were tested by determining the statistical significance of differences between the treatment groups. Such differences were found between the group supported by the prototype and all other treatment groups (cf. Table 5).

Table 4: Results for Hypotheses H1. The column *Corrected p* presents the p-values resulting from the Bonferroni-Holm correction.

$\mathbf{H1_{i.0}}$	**Tool**	**NetSim**			**p**	**Corrected p**	**Reject $\mathbf{H1_{i.0}}$?**
		Min	Max	Avg			
N/A	Control	0.118	0.533	0.250	N/A	N/A	N/A
$H1_{1.0}$	YouTube	0.105	0.476	0.297	0.12205	0.12205	No
$H1_{2.0}$	Confluence	0.160	0.467	0.307	0.05776	0.11552	No
$H1_{3.0}$	Discord	0.211	0.556	0.360	0.00401	0.01203	**Yes**
$H1_{4.0}$	Prototype	0.357	0.538	0.458	<0.00001	<0.001	**Yes**

Table 5: Results for Hypotheses H2. The column *Corrected p* presents the p-values resulting from the Bonferroni-Holm correction.

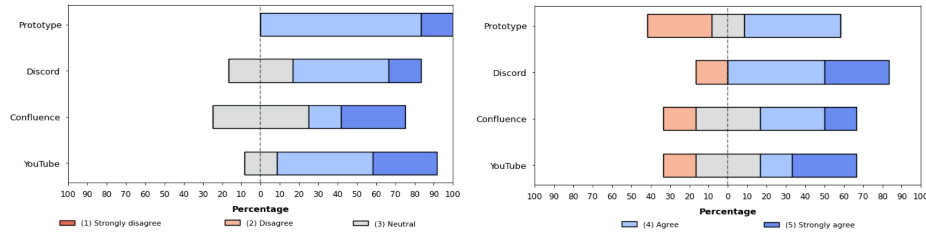
$\mathbf{H2_j}$	**Tool A**	**Tool B**	**p**	**Corrected p**	**Reject $\mathbf{H2_{j.0}}$?**
$H2_1$	YouTube	Confluence	0.39055	0.39055	No
$H2_2$	YouTube	Discord	0.05919	0.17757	No
$H2_3$	YouTube	Prototype	0.00002	<0.001	**Yes**
$H2_4$	Confluence	Discord	0.06814	0.17757	No
$H2_5$	Confluence	Prototype	<0.00001	<0.001	**Yes**
$H2_6$	Discord	Prototype	0.00211	0.00844	**Yes**

To gain a better understanding of the magnitude of the differences between the examined groups, we calculated the effect sizes for all comparisons that were positively tested for statistical significance. The results of these calculations can be found in Table 6.

Table 6: Effect sizes for statistically significant differences between groups. We interpret the calculated values according to Cohen [8] and Sawilowsky [26].

Hypothesis	**Group A**	**Group B**	**Cohen's *d***	**Interpretation**
$H1_3$	Control	Discord	1.047	Large
$H1_4$	Control	Prototype	2.354	Huge
$H2_3$	YouTube	Prototype	1.789	Very large
$H2_5$	Confluence	Prototype	1.976	Very large
$H2_6$	Discord	Prototype	1.135	Large

Polls: Polls were created based on the discussion of each group. The groups supported by *YouTube*, *Confluence* and the prototype were asked eight polling questions each, while the group supported by *Discord* answered seven. We found average majority sizes of 72.6% for *YouTube*, 78.8% for *Confluence*, 71.1% for *Discord* and 76.8% for the group supported by the prototype.



(a) (b)

Fig. 2: Answers to the questionnaire regarding the suitability of asynchronous communication (a) and participants' preference between asynchronous and synchronous communication (b).

Questionnaire: The questionnaire consisted of questions regarding the general suitability of asynchronous communication contexts for discussing an artifact. The first question asked participants how suitable they thought asynchronous communication was for the discussion of a vision video's content. No statistically significant differences could be found between the groups. Out of 24 participants, 6 answered neutrally. All other 18 participants indicated that they agreed or strongly agreed that asynchronous communication is suitable. An overview of these results can be found in Fig. 2a.A second question addressed the preference between asynchronous and synchronous communication. Once again, no statistically significant differences could be found. The answers were diverse for all treatment groups. In total, no participant strongly preferred synchronous communication, while 5 participants indicated that they preferred synchronous communication and 5 participants answered neutrally. A total of 9 participants preferred asynchronous communication, with an additional 5 participants strongly preferring asynchronous communication. A visual representation of these results can be found in Fig. 2b. In addition to questions answered on Likert scales, we also asked open questions regarding positive and negative aspects of asynchronous communication. The most often mentioned positives were *having enough time to think*, *developing ideas* and the *temporal flexibility*. Negative aspects included *delayed answers* and *missed comments*, as well as the longer time required for final conclusions. The final question asked for opinions on a statement, indicating *Questions of Understanding* as valuable. Once again, no statistically significant differences could be found between the treatment groups. Only a single participant strongly disagreed, while 2 other participants gave neutral answers. 12 participants agreed with the statement, and a further 9 participants agreed strongly.

7 Threats to Validity

We report the threats to validity of our results according to Wohlin et al. [34].

The *conclusion validity* of our results is threatened by the small sample size. Having only six participants per treatment group increases the risk of statistical noise impacting the results. However, we chose to include three existing software tools in our evaluation rather than increase the sample size for only one or two, as we obtained three clear favorites in the workshop. Another threat to the conclusion validity is the fact that we asked participants who had only discussed the vision asynchronously about their preference between synchronous and asynchronous communication. Nevertheless, it is easy for participants to imagine synchronous discussions and the answers to the open questions of the questionnaire gave concrete reasons for this preference.

One threat to the *internal validity* of our study is the potential of exhausted participants giving incomplete answers. Participants of our study were asked to work in two time windows and asked to fill in multiple documents over the course of a day. We chose this type of study to reliably simulate an asynchronous setting and also gave participants a lengthy break between the time windows. Furthermore, participants could in theory have interacted with one another outside of the asynchronous communication tools. We minimized this threat by creating new accounts without any identifying information for all participants on all software tools used in the study.

A threat to the *construct validity* is the mono-method bias. We chose not to include further metrics to avoid an even higher potential for participant exhaustion. Another threat is that participants might understand the same term differently when filling in the PFNets spreadsheet. We only included terms that were short and clearly visible in the vision video to minimize this threat. Additionally, we only simulated the presence of different time zones by assigning distinct time frames to all participants. An experiment including multiple time zones would have been preferable, but was not feasible.

The *external validity* of our results is that participants knew that they were taking part in an experiment. A study with practitioners in a real-world use case would have been preferable. Another threat is the potential that we might have missed a suitable existing tool. However, we tried to minimize this threat by conducting the workshop and discussing the results with multiple researchers. Furthermore, the experiment was conducted over the course of a single day while a real-world application would likely be performed over the course of multiple days. We accepted this threat as the threat of participant exhaustion might have been increased further, had we conducted a multi-day study.

8 Discussion

The results of our study show clear differences between the achieved level of shared understanding among the participants of the five groups. In particular, we found that all treatment groups supported by one of the four software tools (*YouTube*, *Confluence*, *Discord*, and the prototype) achieved a higher average level of shared understanding than the control group. This finding is indicated by the higher average NetSim values, as a higher NetSim value indicates a higher level of shared understanding [5]. When comparing the results of the treatment groups, we found that the group supported by the prototype achieved a statistically significantly higher level of shared understanding than every other treatment group (cf. $H2_{3.1}$, $H2_{5.1}$ and $H2_{6.1}$).

These results substantiate the suitability of our concepts to support stakeholders in achieving a shared understanding in an asynchronous communication context. First, all software tools, even adapted with only a partial implementation of our concepts, result in a higher level of shared understanding than the control group. In accordance with the results of Nagel et al. [25], our results show the importance of enabling discussions between stakeholders in asynchronous settings. Even partial concepts already help to achieve a better understanding, as they improve stakeholders' capabilities to communicate with each other. Second, implementing all concepts to their full extent (as in the prototype) provides a solid basis for achieving a higher level of shared understanding. In all four software tools, we tried to implement each concept as fully as possible. However, for the three existing tools, we had no access to their source code and thus had to make compromises, such as using plugins, to enable the concept as intended. In contrast, the prototype allowed us to implement and combine the concepts to reach their full potential. For this reason, the main difference between the prototype and the adapted software tools is the degree to which the concepts could be implemented. While the results show that even the partial implementations lead to a higher shared understanding than the control group, the prototype achieved the best results overall with effect sizes ranging from large to huge [8,26]. We assume that the main reason for these results is the concept *Step-By-Step Design*. This concept provides a structured framework for all other concepts. For example, the prototype enforces the answering of *Questions of Understanding* before participants can access the comment section due to the *Step-By-Step Design*. In this way, the full implementation of the *Step-By-Step Design* emphasized the importance of these questions and ensured that the participants are familiar with the video content before writing any comment. As a consequence, the concepts were better integrated and combined, resulting in a higher level of shared understanding of stakeholders. Based on these insights, we provide the following answers to our research questions:

Answer to RQ1: The concepts presented in this paper are suited to the support of stakeholders in achieving a shared understanding in an asynchronous communication context. Our participants indicated that *Questions of Understanding* and the *Step-By-Step Design* were especially meaningful.

Answer to RQ2: We found *Discord* to be the most suited existing tool for being adapted to our concepts. However, the group supported by our prototype achieved an even higher level of shared understanding that is statistically significantly different from all other treatment groups. Further development of the prototype to achieve shared understanding in asynchronous communication contexts is a promising endeavor for future research.

Besides the analysis of the shared understanding among the stakeholders in the respective groups, we also investigated the participants' attitude towards the idea of being supported in achieving a shared understanding in asynchronous communication contexts. According to our results, most of them preferred the use of asynchronous communication contexts over synchronous ones. They justified this decision with a higher flexibility to take their time to think about the presented vision and for the development of questions, answers, and ideas for the discussion with the other stakeholders. This finding is in line with the advantages of asynchronous communication contexts found by Dowling and Lewis [11].

However, the generalizability of our results is limited. The groups of participants supported by each software tool are probably smaller than in a real-world setting. In addition, the participants had no real value in understanding the presented vision due to the fictitious experimental context. Nevertheless, our concepts are a promising starting point for future research. On the one hand, future work needs to investigate how each concept individually contributes to a shared understanding, as we only investigated all concepts together. On the other hand, we observed difficulties in the experiment such as language barriers and terminology issues for which we only have the partial solutions of a *Step-By-Step Design* combined with mandatory *Questions of Understanding*.

In summary, our results reveal the value of asynchronous communication contexts. Stakeholders are able to disclose, discuss and align their mental models within an asynchronous context to achieve a shared understanding. An even higher level of shared understanding can be accomplished when using the full extent of our concepts. We conclude that the concepts described in this paper fulfill our goal. In this way, we *developed suitable concepts to support stakeholders in achieving a shared understanding in asynchronous communication contexts.*

9 Conclusion

A shared understanding between stakeholders is vital for successful software projects. The discussion of vision videos present one possible way to achieve such a shared understanding, even in asynchronous settings. However, these discussions depend on asynchronous communication methods. In this paper, we presented concepts to support achieving a shared understanding between stakeholders in asynchronous communication contexts. We adapted existing software tools and developed a prototype according to our concepts and conducted a user study. This study substantiates the suitability of our concepts for supporting shard understanding in asynchronous communication contexts.

In future research, we plan to increase the sample size of our study to obtain more reliable results. We also plan on evaluating our concepts in isolation and to compare our results to the shared understanding created in a synchronous meeting. For the concepts *Requirements Engineers as Facilitators* and *Message Frames* we seek to investigate how requirements engineers can be supported while performing the associated tasks. Furthermore, the PFNets spreadsheet could be extended with terms relating to the topics discussed by the groups.The findings of this paper indicate the potential of our concepts. Further research efforts might lead to a definitive tool supporting the achievement of a shared understanding among stakeholders in asynchronous settings.

Acknowledgement

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under Grant No.: 289386339, project ViViUse.

References

1. Ambler, S.: Agile modeling: effective practices for extreme programming and the unified process. John Wiley & Sons (2002)
2. Amiri, S.M.: Konzeptionierung eines Tools zur Herstellung gemeinsamen Verständnisses durch asynchrone Betrachtung von Vision Videos. Master thesis, Leibniz Universität Hannover (2022)
3. Aranda, J.: A theory of shared understanding for software organizations. University of Toronto (2010)
4. Bittner, E.A.C., Leimeister, J.M.: Why shared understanding matters–Engineering a collaboration process for shared understanding to improve collaboration effectiveness in heterogeneous teams. In: 46th Hawaii International Conference on System Sciences. IEEE (2013)
5. Braunschweig, B., Seaman, C.: Measuring shared understanding in software project teams using pathfinder networks. In: 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (2014)
6. Brill, O., Schneider, K., Knauss, E.: Videos vs. use cases: Can videos capture more requirements under time pressure? In: International Working Conference on Requirements Engineering: Foundation for Software Quality. Springer (2010)
7. Clark, C., Strudler, N., Grove, K.: Comparing asynchronous and synchronous video vs. text based discussions in an online teacher education course. Online Learning **19**(3) (2015)
8. Cohen, J.: edition 2. statistical power analysis for the behavioral sciences (1988)
9. Creighton, O., Ott, M., Bruegge, B.: Software cinema – video-based requirements engineering. In: 14th IEEE International Requirements Engineering Conference (2006)
10. Dearholt, D.W., Schvaneveldt, R.W.: Properties of Pathfinder Networks. Ablex Publishing Corp., USA (1990)
11. Dowling, K.L., Louis, R.D.S.: Asynchronous implementation of the nominal group technique: is it effective? Decision Support Systems **29**(3) (2000)
12. Garrison, D.R.: E-learning in the 21st century: A community of inquiry framework for research and practice. Routledge (2016)
13. Girgensohn, A., Marlow, J., Shipman, F., Wilcox, L.: HyperMeeting: Supporting asynchronous meetings with hypervideo. In: 23rd ACM international conference on Multimedia (2015)
14. Glinz, M., Fricker, S.A.: On shared understanding in software engineering: an essay. Computer Science – Research and Development **30**(3) (2015)
15. Hiltz, S.R., Dufner, D., Holmes, M., Poole, S.: Distributed group support systems: Social dynamics and design dilemmas. Journal of Organizational Computing and Electronic Commerce **1**(2) (1991)

16. ISO Central Secretary: Systems and software engineering - Life cycle processes - Requirements engineering. Standard ISO/IEC/IEEE 29148:2018 (2018)
17. Karras, O.: Supporting Requirements Communication for Shared Understanding by Applying Vision Videos in Requirements Engineering. Logos (2021)
18. Karras, O., Kiesling, S., Schneider, K.: Supporting requirements elicitation by tool-supported video analysis. In: 24th International Requirements Engineering Conference. IEEE (2016)
19. Karras, O., Kristo, E., Klünder, J.: The Potential of Using Vision Videos for CrowdRE: Video Comments as a Source of Feedback. In: 29th International Requirements Engineering Conference Workshops. IEEE (2021)
20. Karras, O., Risch, A., Schneider, K.: Interrelating Use Cases and Associated Requirements by Links: An Eye tracking study on the impact of different linking variants on the reading behavior. In: 22nd International Conference on Evaluation and Assessment in Software Engineering (2018)
21. Karras, O., Schneider, K., Fricker, S.A.: Representing software project vision by means of video: a quality model for vision videos. Journal of Systems and Software **162** (2020)
22. Lloyd, W.J., Rosson, M.B., Arthur, J.D.: Effectiveness of elicitation techniques in distributed requirements engineering. In: IEEE Joint International Conference on Requirements Engineering. IEEE (2002)
23. Nagel, L., Amiri, S.M.: Supplementary Material - Supporting Shared Understanding in Asynchronous Communication Contexts (Feb 2023). `https://doi.org/10.5281/zenodo.7649336`
24. Nagel, L., Karras, O.: Keep Your Stakeholders Engaged: Interactive Vision Videos in Requirements Engineering. In: IEEE 29th International Requirements Engineering Conference Workshops. IEEE (2021)
25. Nagel, L., Shi, J., Busch, M.: Viewing Vision Videos Online: Opportunities for Distributed Stakeholders. In: IEEE 29th International Requirements Engineering Conference Workshops (2021)
26. Sawilowsky, S.S.: New effect size rules of thumb. Journal of modern applied statistical methods **8**(2), 26 (2009)
27. Schneider, K., Bertolli, L.M.: Video variants for CrowdRE: how to create linear videos, vision videos, and interactive videos. In: IEEE 27th International Requirements Engineering Conference Workshops. IEEE (2019)
28. Schneider, K., Busch, M., Karras, O., Schrapel, M., Rohs, M.: Refining vision videos. In: 25th International Working Conference on Requirements Engineering: Foundation for Software Quality. Springer (2019)
29. Skylar, A.A.: A comparison of asynchronous online text-based lectures and synchronous interactive web conferencing lectures. Issues in Teacher Edu. **18**(2) (2009)
30. Smith, J.Y., Vanecek, M.T.: Dispersed group decision making using nonsimultaneous computer conferencing: A report of research. Journal of Management Information Systems **7**(2) (1990)
31. Van Lamsweerde, A.: Requirements engineering in the year 00: A research perspective. In: 22nd International conference on Software engineering (2000)
32. Warkentin, M., Beranek, P.M.: Training to improve virtual team communication. Information systems journal **9**(4), 271–289 (1999)
33. Wheeler, B.C., Valacich, J.S.: Facilitation, GSS, and training as sources of process restrictiveness and guidance for structured group decision making: An empirical assessment. Information Systems Research **7**(4) (1996)
34. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in software engineering. Springer Science & Business Media (2012)